

CM Bulk Import Export and CM REST Northbound Interface

Interwork Description

Copyright

© Ericsson AB 2017 - 2020. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Trademark List

All trademarks mentioned herein are the property of their respective owners. These are shown in the document [Trademark Information](#).



Contents

1	CM Bulk Import Export and CM REST Northbound Interface	1
1.1	Application Programming Interface Documentation	1
1.2	Configuration Management Services Use Cases	2
1.3	File Import Use Cases	3
1.4	File Export Use Cases	4
1.5	Service Specific Documentation	6
1.6	Service Specific Resource Additional Attributes and Schema	7
1.7	Import Services	8
1.8	Export Services	8
2	Prerequisites	10
2.1	Authentication	10
3	Resources Details	11
3.1	Configuration	11
3.2	Job	12
3.3	Retry	15
3.4	Report	15
3.5	Filter	16
4	Bulk Export Additional Attributes	18
4.1	GET Endpoints for EXPORT	18
4.2	POST Endpoints for EXPORT	23
4.3	DELETE Endpoints for EXPORT	33
5	Bulk Import Additional Attributes	35
5.1	GET Endpoint for IMPORT	35
5.2	POST Endpoint for IMPORT	40
6	Configuration Additional Attributes	45
6.1	GET Endpoint for Configuration	45
6.2	POST Endpoint for Configuration	46
7	Exporting and Importing PLM Links	48
8	Details of Errors Management	55
8.1	JBOSS Error	55



8.2	REST Error	56
8.3	Error Response Code for Configuration	56
8.4	Error Response Code for Export	57
8.5	Error Response Codes for Import	59
9	Appendix A - CM REST API Documentation	61
10	Appendix B - CM REST Service Specific API Documentation	62
11	Appendix C - Bulk Import Validation Phases	63
	Reference List	66



1 CM Bulk Import Export and CM REST Northbound Interface

The CM REST Northbound Interface (NBI) is an OSS feature deployed in the ENM Domain Management to supply external systems.

These systems include Network Resource Management System, and read and write access to all CM data and metadata using a Machine-to-Machine interface.

This document describes the following two features:

- CM Configuration
- CM Bulk

Note: The CM Bulk Import use cases in this document are now deprecated and all customers are advised to migrate to the [CM Bulk Import REST NBI](#). The Export use cases in this document are **not** deprecated.

1.1 Application Programming Interface Documentation

The Application Programming Interface (API) documentation contains the schema of the resources defined for this interface. These include: job, configuration, and report.

An "additionalProperties" keyword is also available. This adds service-specific additional attributes (refer to Service Specific Documentation section).

The API documentation also contains:

- Resources and endpoints
- REST operations (HTTP methods) available for each endpoint
- URI parameters and query parameters related to each operation
- Schema to be used in request and response body for each operation
- Examples

The documentation is available in CPI library under CM Bulk Import Export and CM REST Northbound Interface API documentation in *Supplementary Documentation for CM Bulk Import Export and CM REST Northbound Interface, 190 89-LZN 708 0882*.



Resources

Data presented by the CM NBI interface are arranged in the following resources:

1. Configuration
2. Job
3. Retry
4. Report
5. Filter

For more details, refer to [Resources Details](#) on page 11.

Hypertext Application Language (HAL)

HAL is included in the CM Rest NBI interface: a "_links" keyword is available where it is possible to specify links.

In accordance to the HAL specifications, a "self" link is mandatory:

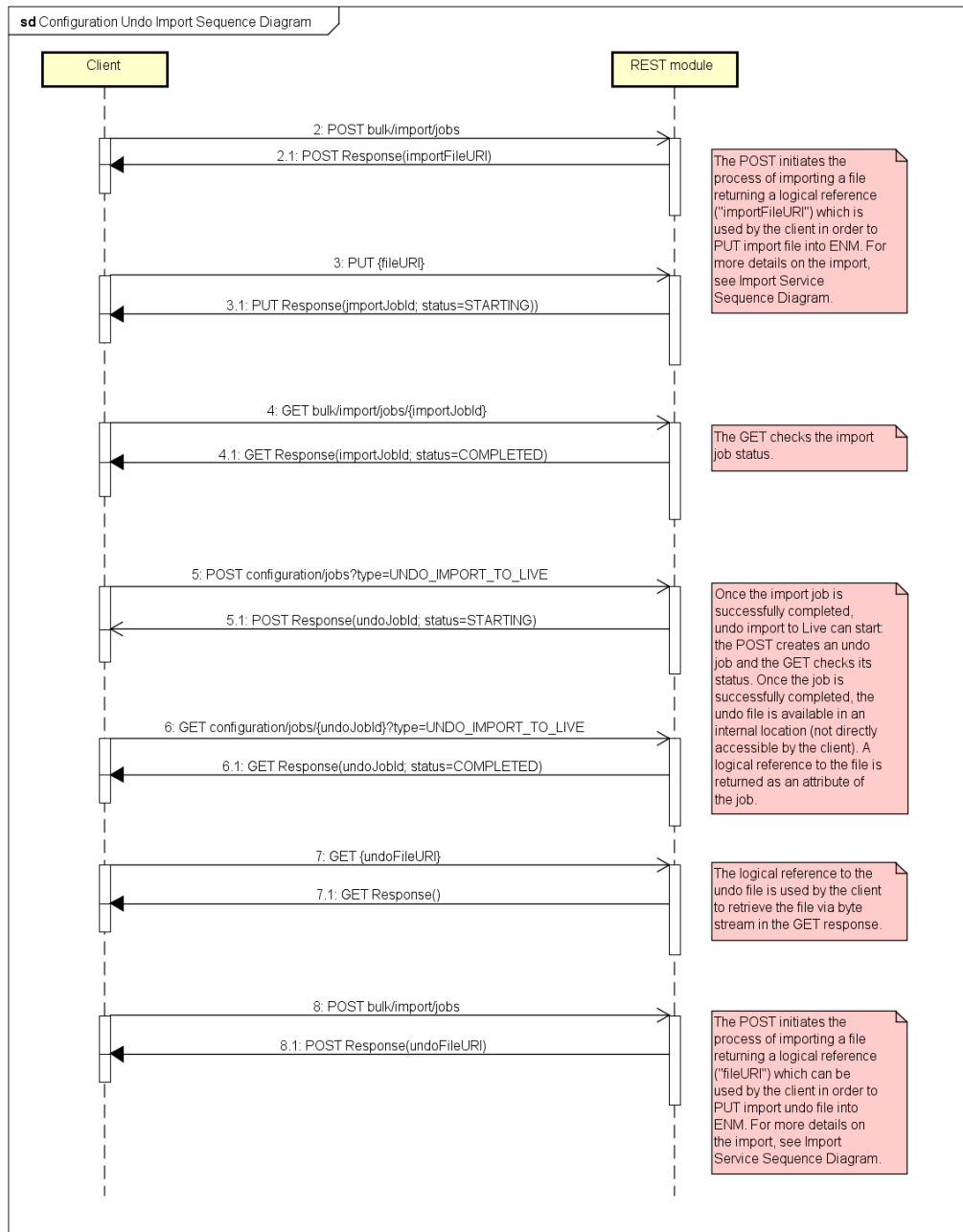
- In the context of an endpoint related to a resource, HAL describes other resources having an association with it (for example: reports having an association with jobs).
- In the context of root endpoints, HAL describes the endpoints for the supported services (for example: /bulk points to import and export services) without a specific reference to a resource.

1.2 Configuration Management Services Use Cases

Configuration Undo Import to Live

The Configuration Service Undo Import to Live Sequence Diagram displays the following Use cases:

- Import the configuration to Live.
- Get the job status of the previously created import job.
- Undo the configuration import using 3GPP/dynamic file format.
- Get the job status of the previously created undo job.
- Undo configuration file download.
- Import the undo configuration file to Live Configuration.



powered by Astah

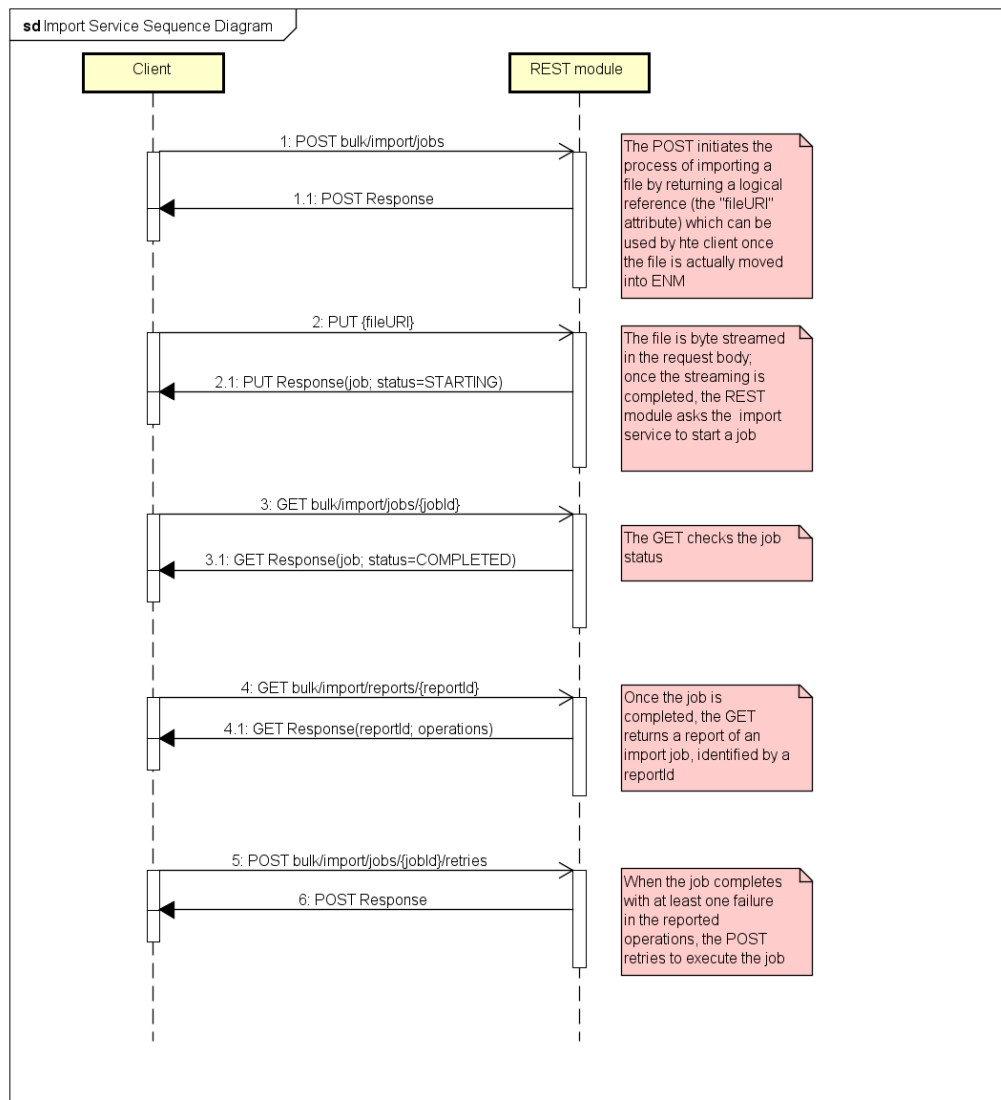
1.3 File Import Use Cases

Import

The Import Service Sequence Diagram shows the Use cases related to the import of a configuration in the ENM. The Use cases include the following:



- Import of a 3GPP/dynamic file in a configuration.
- Get the status of the import job previously created.
- Get the reports of the import job (identified by a reportId) previously created.
- If the job completes but some operation failed, retry to execute the job.



powered by Astah

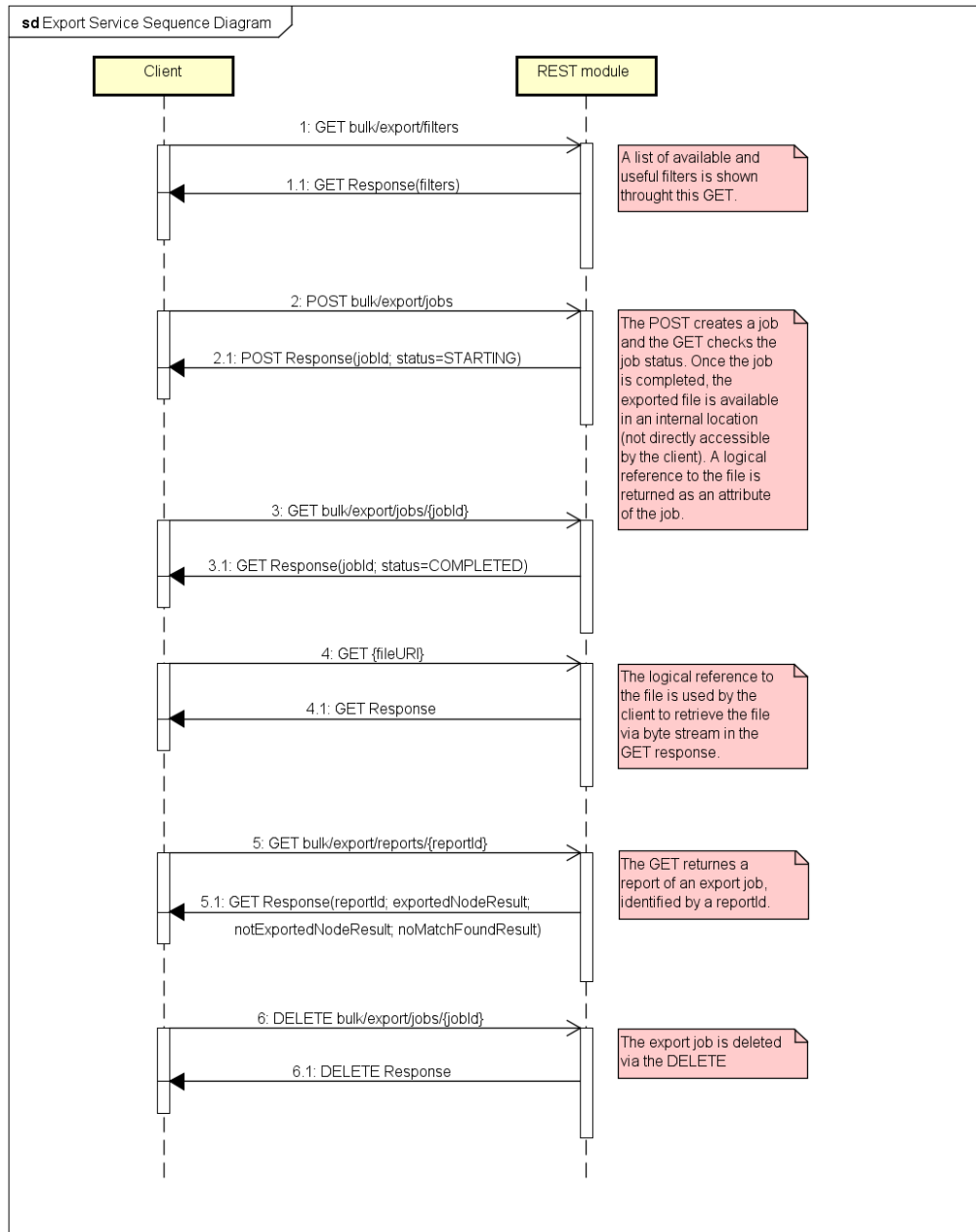
1.4 File Export Use Cases

Export

The Export Service Sequence Diagram displays the Use cases related to the export of a configuration from the ENM. The Use cases include the following:



- Get the list of possible filters to use for the export.
- Export of a configuration in a 3GPP/dynamic file.
- Get the status of the export job previously created.
- Download of the file previously created.
- Get the reports of a job (identified by a reportId) previously created.
- Delete the export job.



powered by Astah

1.5 Service Specific Documentation

Service Specific Endpoints

The endpoints representing the logical path of the file to transfer are used by the Bulk services. These are not exposed on the interface.

URI is an internal convention, and is reported by the REST module to the client to trigger the file transfer.



The documentation is available in the CPI library under CM Bulk Import Export and CM REST Northbound Interface API documentation in *Supplementary Documentation for CM Bulk Import Export and CM REST Northbound Interface*, 190 89-LZN 708 0882.

1.6 Service Specific Resource Additional Attributes and Schema

This section details the resource additional attributes and specific schema for every service, for the endpoints and HTTP commands where applied.

The complex structures are detailed in separate boxes for clarity.

The `additionalProperties` key in the schema is used to add additional attributes:

- `additionalProperties = true`, when additional attributes are available, this is the default so this notation can be omitted.
- `additionalProperties = false` is mandatory when no additional attributes are included in the JSON.

Configuration Services

Note: The field type must be equal to the following:

UNDO_IMPORT_TO_LIVE

HTTP	Endpoint	Resource	Additional Attributes
POST			
POST request (for undo import)	/configuration/jobs	job	fileFormat: string
GET			
GET response (for undo import)	For all jobs: /configuration/jobs? type=UNDO_IMPORT_TO_LIVE For a specific jobId: /configuration/jobs/ {jobId}? type=UNDO_IMPORT_TO_LIVE	job	fileURI: string (this attribute is the same for all jobs or for a specific job)
GET request	/fileURI		No additional attributes needed



1.7 Import Services

This section contains details on resource, additional attributes, and specific schema for Import service, for the endpoints and HTTP commands where applied.

Note: The field type must be equal to IMPORT.

HTTP	Endpoint	Resource	Additional Attributes
POST			
POST request	/bulk/import/jobs	job	fileFormat: string fileName: string continueOnErr: string validate: string
POST response	/bulk/import/jobs	job	fileUri: string
PUT request	/fileUri	job	No additional attributes needed
POST request	/bulk/import/jobs/{jobId}/retries	retry	No additional attributes needed
POST response	/bulk/import/jobs/{jobId}/retries	retry	retriedJobId: string
GET			
GET response	For all jobs: /bulk/import/jobs For a specific job: /bulk/import/jobs/{jobId}	job	configName: string fileFormat: string fileName: string totalManagedObjectCreated: integer totalManagedObjectUpdated: integer totalManagedObjectDeleted: integer nodesCopied: integer nodesNotCopied: integer (These attributes are the same for all jobs or for a specific job)
GET response	/bulk/import/reports/{reportId}	report	operations: array of operation_details.json

1.8 Export Services

This section details resource, additional attributes, and specific schema for Export service, for the endpoints and HTTP commands where applied.

Note: The field type must be equal to EXPORT.

HTTP	Endpoint	Resource	Additional Attributes
POST			
POST request	/bulk/export/jobs	job	fileFormat: string dataCategories: dataCategories.json exportFilter: exportFilter.json nodeSearchCriteria: nodeSearchCriteria.json



HTTP	Endpoint	Resource	Additional Attributes
			compressionType: string enumTranslate: boolean jobName: string userFilter: userFilter.json syncState: string prettyFormat: boolean
GET			
GET response	For all jobs: /bulk/export/jobs For a specific job: /bulk/export/jobs/{jobId}	job	jobName: string expectedNodesExported: integer nodesExported: integer nodesNotExported: integer nodesNoMatchFound: integer numberOfMosExported: integer userId: string fileURI: string (These attributes are the same for all jobs or for a specific job)
GET request	/fileURI		No additional attributes needed.
GET response	/bulk/export/reports/{reportId}	report	exportedNodesResult: array of ExportNodeResult.json notExportedNodesResult: array of ExportNodeResult.json noMatchFoundResult: array of ExportNodeResult.json fileName: string
GET response	/bulk/export/filters	filter	No additional attributes needed. For this response, the list of filters is present.
DELETE			
DELETE response	/bulk/export/jobs/{jobId}	job	No additional attributes needed.
DELETE response	/bulk/export/jobs?jobName={jobName}	job	jobName: string



2 Prerequisites

Authorization

Users must have CM REST NBI capabilities.

Two predefined roles with the mentioned capabilities are available:

- CM_REST_Administrator
- CM_REST_Operator

The CM_REST_Administrator role can perform HTTPS POST, PUT, DELETE, and GET operations. The CM_REST_Operator role can perform HTTPS GET only.

2.1 Authentication

Prerequisites

- A valid Username and Password, with a CM REST NBI role to access the ENM System.
- A fully deployed ENM System.
- CM Bulk Import operations are subject to Target Based Access Control (TBAC) checks. Therefore all nodes in the job must be available in the target group of the executing user to avoid authorization errors.

Authentication through NBI

Establish a session with the ENM system.

For details, refer to:

- *Export ENM PKI Root CA Certificate* section in the *ENM Public Key Infrastructure System Administrator Guide, 2/1543-AOM 901 151-3*.
- *Establish a User Session over REST* and *cURL examples for Identity and Access Management* sections in the *ENM Identity and Access Management Programmers Guide, 19817-CNA 403 3016*.



3 Resources Details

This section is about the data shown from the CM NBI.

The data are presented with the following resource format:

- Configuration
- Job
- Retry
- Report
- Filter

For each resource, it is possible to find the schema that defines the common attributes. These attributes are listed in the properties key:

Configuration	An object able to manage the ENM configuration.
Job	An object managing asynchronously the operation of Undo over the configurations, in addition to Import and Export functionalities.
Retry	An object managing asynchronously an Import command on Live Configuration in particular cases of a failed Import operation or operations.
Report	Shows specific and detailed results for many operations where a job is generated.
Filter	Returns the list of filters available by querying different entities.

3.1 Configuration

The following schema defines Configuration:

Configuration Schema

```
{
  "type": "object",
  "description": "A collection of configurations",
  "properties": {
    "configurations": {
      "type": "array",
      "items": {
```



```
    "type": "object",
    "title": "configuration",
    "properties": {
      "id": {
        "type": "integer"
      },
      "name": {
        "type": "string"
      }
    },
    "additionalProperties": false
  },
  "links": {
    "type": "object",
    "properties": {
      "self": {
        "type": "object",
        "properties": {
          "href": {
            "type": "string"
          }
        }
      },
      "required": [
        "href"
      ]
    }
  },
  "required": [
    "self"
  ]
}
}
```

3.1.1 Configuration Common Attributes

This section describes the common attribute for the resource Configuration.

Configuration Common Attributes	Description	Values Example
id	An integer, representing the identifier of the configuration.	

3.2 Job

Every Job has the following common attributes:

- id
- type
- status
- statusReason
- creationTime
- startTime



- endTime
- lastUpdateTime

The following schema defines the Job:

Job Schema

```
{
  "type": "object",
  "title": "job",
  "properties": {
    "id": {
      "type": "integer"
    },
    "status": {
      "type": "string"
    },
    "statusReason": {
      "type": "string"
    },
    "creationTime": {
      "type": "string"
    },
    "startTime": {
      "type": "string"
    },
    "endTime": {
      "type": "string"
    },
    "lastUpdateTime": {
      "type": "string"
    },
    "type": {
      "type": "string"
    },
    "_links": {
      "type": "object",
      "properties": {
        "self": {
          "type": "object",
          "properties": {
            "href": {
              "type": "string"
            }
          },
          "required": [
            "href"
          ]
        }
      },
      "required": [
        "self"
      ]
    }
  },
  "required": [
    "self"
  ]
}
```

Note: The endTime field in the response is present only when the job status is COMPLETED or FAILED.



3.2.1 Job Common Attributes

This section describes the common attributes for the resource Job.

Job Common Attributes	Description	Example Values
id	An integer, indicating the number of the job.	Ex: 1, 2....
status	A string, indicating the status of the job.	<ul style="list-style-type: none"> — STARTED — IN PROGRESS — COMPLETED — FAILED — STOPPED — STOPPING — UPGRADE INTERRUPTED — N/A
statusReason	A string, indicating further details about the job state. For the IMPORT job type, it is present if the 'status' is FAILED. Otherwise, it is always present.	Ex: "Error ...Child of type EUtranCellTDD with name ...already exists...."
creationTime	A time stamp indicating the time of the creation of the job.	Ex: 2016-05-06T13:46:25.912
startTime	A time stamp indicating the start time of the job.	Ex: 2016-05-06T13:46:25.921
endTime	A time stamp indicating the end time of the job. ^[1]	Ex: 2016-05-06T13:46:26.053
lastUpdateTime	A time stamp indicating the last update time applied to the job.	Ex: 2016-05-06T13:46:26.053
type	The type of the job.	<ul style="list-style-type: none"> — UNDO_IMPORT_TO_LIVE — IMPORT — EXPORT

3.2.2 Job Status Details

This section contains the Job Status description for each Job Type.

Job Status	Description	Job Type
STARTED	Job starts to be in progress.	EXPORT, UNDO_IMPORT_TO_LIVE
IN_PROGRESS	Import job is in progress.	IMPORT
COMPLETED	Job completed.	IMPORT, EXPORT, UNDO_IMPORT_TO_LIVE
FAILED	Job failure.	IMPORT, EXPORT, UNDO_IMPORT_TO_LIVE
STOPPED	Identified that an upgrade is in progress.	EXPORT, UNDO_IMPORT_TO_LIVE



Job Status	Description	Job Type
STOPPING	Transactional state used only during upgrade, for example when the container is shutting down or resources are not available.	EXPORT, UNDO_IMPORT_TO_LIVE

3.3 Retry

The following schema defines Retry:

Retry Schema

```
{
  "type": "object",
  "title": "retry",
  "properties": {
    "type": {
      "type": "string"
    },
    "continueOnErr": {
      "type": "string"
    }
  }
}
```

3.3.1 Retry Common Attributes

This section describes the common attributes for the resource Retry.

Retry Common Attributes	Description	Example Values
type	A string: it defines the type of the job to be retried.	— IMPORT
continueOnErr	A string that ensures the import continues and attempts to import all file data even when a non fatal error is encountered. (When it is not present, the value is the one present in the Import job to be retried.)	— CONTINUE/NODE — OPERATION — STOP

3.4 Report

The following schema defines Report:

Report Schema

```
{
  "type": "object",
  "title": "report",
  "properties": {
    "id": {
      "type": "integer"
    },
    "type": {
      "type": "string"
    }
  }
}
```



```

    "_links": {
      "type": "object",
      "properties": {
        "self": {
          "type": "object",
          "properties": {
            "href": {
              "type": "string"
            }
          }
        },
        "required": [
          "href"
        ]
      }
    },
    "required": [
      "self"
    ]
  }
}
}
}

```

3.4.1 Report Common Attributes

This section describes the common attributes for the resource Report.

Report Common Attributes	Description	Example Values
id	An integer: it represents the identifier of the report.	
type	A string: it defines the type of the report.	<ul style="list-style-type: none"> — IMPORT — EXPORT

3.5 Filter

The following schema defines Filter:

Filters Schema

```

{
  "type": "object",
  "description": "A collection of filters",
  "properties": {
    "filters": {
      "type": "array",
      "items": {
        "type": "object",
        "title": "exportFilter",
        "properties": {
          "namespace": {
            "type": "string"
          },
          "name": {
            "type": "string"
          },
          "version": {
            "type": "string"
          },
          "description": {
            "type": "string"
          }
        }
      }
    },
    "additionalProperties": false
  }
}

```



```

    }
  },
  "_links": {
    "type": "object",
    "properties": {
      "self": {
        "type": "object",
        "properties": {
          "href": {
            "type": "string"
          }
        }
      },
      "required": [
        "href"
      ]
    }
  },
  "required": [
    "self"
  ]
}
}
}
}

```

3.5.1 ExportFilter Common Attributes

This section describes the common attributes for the resource `exportFilter`.

ExportFilter Common Attributes	Description	Example Values
<code>nameSpace</code>	A string: it is the Namespace of the filter.	<code>PredefinedExportFilter</code>
<code>name</code>	A string: it is the name of the filter.	<ul style="list-style-type: none"> — ERBS — Inventory Data — SON — Subnetwork — 3GPP_Topology — Dynamic_Topology
<code>version</code>	A string: it is the version of the filter.	<code>1.0.0</code>
<code>description</code>	The description of the filter.	Any string



4 Bulk Export Additional Attributes

This section lists the supported endpoints with additional EXPORT attributes.

These are shown in the following table:

Context	HTTP Supported Methods	
/bulk/export/jobs	GET	POST
/bulk/export/jobs/{jobId}	GET	DELETE
/bulk/export/jobs?jobName={jobName}	DELETE	
/bulk/export/reports/{reportId}	GET	
/bulk/export/filters	GET	

4.1 GET Endpoints for EXPORT

This section lists the GET Endpoints having additional attributes for EXPORT.

4.1.1 GET /bulk/export/jobs

HTTP Supported Method	Method Information
GET	Returns the list of the available export jobs

Table 1 GET Response Additional Attributes

Additional Attributes	Description	Optional
expectedNodesExported	The number of expected exported nodes	no
nodesExported	The number of exported nodes	no
nodesNotExported	The number of nodes not exported	no
nodesNoMatchFound	The number of nodes not matching	no
numberOfMosExported	The number of MOs exported	no
userId	The name of the user	no
fileURI	The URI of the export file	no
jobName	The name of the job	no

When starting an export job, it is possible to optionally specify a job name. The specified job name cannot already exist. It must start with a letter, numeral or underscore. It can contain any other combination of letters, numerals,



underscores, or hyphens. Example: "[a-zA-Z0-9_]([a-zA-Z0-9_] | '-')". If a job name is specified, the generated export file is named after the job name.

If a job name is not supplied, the system will generate a default file name in the format "export_<filetype>_<timestamp>_<unique_id>". The timestamp format is "yyyy-MM-dd'T'HH-mm-ss-SSS", for example: "2019-10-30T17-53-59-177". The unique ID is a random series of hexDigit values in the format "<8*hexDigit>-<4*hexDigit>-<4*hexDigit>-<4*hexDigit>-<12*hexDigit>", for example: "553a3fa1-676c-4a38-b0d7-2b3c214301c8".

The fileURI is generated in the format "bulk/export/<encrypted_text>HDA<job_name>.<suffix>".

Note: If ENIQ management is used, the fileURI attribute does not represent a real URI (for example: "fileUri": "ENIQ_TOPOLOGY_EXPORT"), so the fileURI attribute cannot be used to download the export file.

In case of CM Export of the entire TCIM database or selected TCIM MOs, the expectedNodesExported value is always 1.

Example 1

```
{
  "jobs": [
    {
      "id": 1,
      "type": "EXPORT",
      "status": "COMPLETED",
      "statusReason": "COMPLETED",
      "creationTime": "2015-09-24T20:24:52.112",
      "startTime": "2015-09-24T20:24:52.131",
      "endTime": "2015-09-24T20:24:52.131",
      "lastUpdatedTime": "2015-09-24T20:24:52.131",
      "jobName": "Job_15",
      "expectedNodesExported": "10",
      "nodesExported": "10",
      "nodesNotExported": "0",
      "nodesNoMatchFound": "0",
      "numberOfMosExported": "156",
      "userId": "administrator",
      "fileURI": {
        "href": "/bulk/export/PZcVUIcLCYUEdfGZUWTRBPZaNFUaPZcYVFIKaaFwDAAKKTATZ
RKKQVVUHDAexport_3GPP_2017-03-30T15-14-18-117_4866266b-fb1c-435f-91cd-e4e8cd4da0
8d.xml" →
      },
      "_links": {
        "self": {
          "href": "/bulk/export/jobs/1"
        }
      },
      "reports": {
        "href": "/bulk/export/reports/1"
      }
    }
  ],
  {
    "id": 2,
    "type": "EXPORT",
    "status": "COMPLETED",
    "statusReason": "COMPLETED",
    "creationTime": "2015-09-24T22:32:04.521",
    "startTime": "2015-09-24T22:32:05.123",
    "endTime": "2015-09-24T22:43:14.522",
    "lastUpdatedTime": "2015-09-24T22:32:05.123",
    "jobName": "Job_25",
    "expectedNodesExported": "7",
    "nodesExported": "7",
    "nodesNotExported": "0",
    "nodesNoMatchFound": "0",
  }
}
```



```

      "numberOfMosExported": "128",
      "userId": "administrator",
      "fileURI": {
        "href": "/bulk/export/PZcVUIcLCYUEdfGZUWTRBPZaNFUaPZcYVFIKaaftVSJZUTEI
cQJcYVFIKaaHDAexport_dynamic_2017-07-04T10-28-33-061_63faf57d-d232-4f1d-bc7e-3cc
6c8102585.zip"
      },
      "_links": {
        "self": {
          "href": "/bulk/export/jobs/2"
        },
        "reports": {
          "href": "/bulk/export/reports/2"
        }
      }
    },
    "_links": {
      "self": {
        "href": "/bulk/export/jobs"
      }
    }
  }
}

```

4.1.2 GET /bulk/export/jobs/{jobId}

HTTP Supported Method	Method Information
GET	Returns the status of a specific previously created export job.

GET Response Additional Attributes

The additional attributes for the GET /bulk/export/jobs/{jobId} are the same additional attributes as for the GET /bulk/export/jobs.

4.1.3 GET /bulk/export/reports/{reportId}

HTTP Supported Method	Method Information
GET	Returns the details about the exported nodes involved in a specific job.

If the request is done when the job status is COMPLETED or FAILED, the response contains details of the final status of the job.

If the request is done while the job status is still IN PROGRESS, the response contains details of the status of the job at the time the request is raised.

GET Report Additional Attributes

The additional attributes for the GET /bulk/export/reports/{reportId} are the following:

Additional Attributes exportResponse.json	Description	Optional	Value
exportedNodesResult	The expected exported nodes.	no	Array of the ExportNodeResult.json



Additional Attributes exportResponse.json	Description	Optional	Value
notExportedNodesResult	The nodes not exported.	no	Array of the ExportNodeResult.json
noMatchFoundResult	The nodes not matching.	no	Array of the ExportNodeResult.json
fileUri	The URI for the export file.	no	String

Example 2

```

{
  "id": 70,
  "type": "EXPORT",
  "exportedNodesResult": [
    {
      "nodeName": "ERBS5",
      "fdn": "SubNetwork=Ireland,SubNetwork=Athlone,MeContext=ERBS5",
      "statusMessage": "",
      "fileName": "",
      "exportStatus": "EXPORTED"
    }
  ],
  "noMatchFoundResult": [],
  "notExportedNodesResult": [],
  "fileUri": "/bulk/export/PZcVUIcLCYUEdfGZUWTRBPZaNFUaPZcYVFIKaaFwDAKKTATZRKKQV
VUHDAexport_3GPP_2017-03-30T15-14-18-117_4866266b-fb1c-435f-91cd-e4e8cd4da08d.xml"
  "_links": {
    "self": {
      "href": "/bulk/export/reports/70"
    }
  }
}

```

4.1.4 exportNodeResult.json

The exportNodeResult.json is the schema for the export node result.

Additional Attributes exportNodeResult.json	Description	Optional
nodeName	The name of the node	no
fdn	The fdn of the node	no
statusMessage	The message that indicates the status of the node	no
fileName	The name of the file that contains the information on the exported node The value reported is ""	no
exportStatus	The result of the export operation for the node	no

exportNodeResult.json

```

{
  "type": "object",
  "title": "ExportNodeResult",
  "properties": {
    "nodeName": {
      "type": "string"
    },
    "fdn": {

```



```

    "type": "string"
  },
  "statusMessage": {
    "type": "string"
  },
  "fileName": {
    "type": "string"
  },
  "exportStatus": {
    "type": "string"
  }
}
}
}

```

4.1.5 GET /bulk/export/filters

HTTP Supported Method	Method Information
GET	Returns the available filters.

GET Response

This endpoint does not have additional attributes but a particular field: the list of filters.

Filter

The structure of a single filter is the following:

Filter	Description	Optional
namespace	The name of the Namespace	no
name	The name of the filter	no
version	The version of the filter	no
description	The description about the filter	no

Example 3

```

{
  "_links":{
    "self":{
      "href":"/bulk/export/filters/"
    }
  },
  "filters":[
    {
      "name":"3GPP_Topology",
      "description":"This is the predefined export filter model for Network R →
esource Topology data.",
      "version":"1.0.0",
      "namespace":"PredefinedExportFilter"
    },
    {
      "name":"AxeInventoryData",
      "description":"This is the predefined export filter model for AXE based →
nodes Inventory Data.",
      "version":"1.0.0",
      "namespace":"PredefinedExportFilter"
    },
    {
      "name":"Dynamic_Topology",
      "description":"This filter exports ENM Network Element Topology data.",
      "version":"1.0.0",
      "namespace":"PredefinedExportFilter"
    }
  ]
}

```



```

    },
    {
      "name": "ERBS",
      "description": "This is the predefined export filter model for ERBS node
s.",
      "version": "1.0.0",
      "namespace": "PredefinedExportFilter"
    },
    {
      "name": "In_Service_Performance_Data",
      "description": "This is the predefined export filter model for Performan
ce specific MOs of Base band Radio nodes and DU Radio nodes.",
      "version": "1.0.0",
      "namespace": "PredefinedExportFilter"
    },
    {
      "name": "InventoryData",
      "description": "This is the predefined export filter model for CPP based
nodes Inventory Data.",
      "version": "1.0.0",
      "namespace": "PredefinedExportFilter"
    },
    {
      "name": "LTE_MPC_OTDOA",
      "description": "This is the predefined export filter model for filtering
MOs for LTE MPC OTDOA.",
      "version": "1.0.0",
      "namespace": "PredefinedExportFilter"
    },
    {
      "name": "SON",
      "description": "This is the predefined export filter model for ERBS node
s SON Data.",
      "version": "1.0.0",
      "namespace": "PredefinedExportFilter"
    },
    {
      "name": "SON_EDNET",
      "description": "This is the predefined export filter model for ERBS node
s SON EDNET Data.",
      "version": "1.0.0",
      "namespace": "PredefinedExportFilter"
    },
    {
      "name": "Subnetwork",
      "description": "This is the predefined export filter model for filtering
MOs under Subnetwork for ERBS nodes.",
      "version": "1.0.0",
      "namespace": "PredefinedExportFilter"
    }
  ]
}

```

4.2 POST Endpoints for EXPORT

This section lists the POST Endpoints having additional attributes for EXPORT.

4.2.1 POST /bulk/export/jobs

HTTP Supported Method	Method Information
POST	Starts an export job. It returns the job id and the status of the related job.

POST Request Attributes

The additional attributes for the POST /bulk/export/jobs are:



- fileFormat
- dataCategories
- nodeSearchCriteria
- exportFilter
- compressionType
- enumTranslate
- jobName
- userFilter
- syncState
- prettyFormat

In JSON, the order of the fields is not relevant.

Additional Attributes	Description	Optional	Default Value	Values Examples
fileFormat	The file format.	no		<ul style="list-style-type: none"> — 3GPP — dynamic
dataCategories	<p>The list of export data categories which determine the data models that are exported. The allowed values of the list elements are:</p> <ul style="list-style-type: none"> — NETWORK_RESOURCE_DATA Export file includes SubNetwork, MeContext, managed element, and their subtree data. — ENM_DATA Export file includes Network Element and its subtree data. ENM_DATA must be specified for dataCategories to export a never synced node, that is, a node without an associated managed element or MeContext. 	yes	<p>The default depends on the export type as follows:</p> <ul style="list-style-type: none"> — 3GPP ["NETWORK_RESOURCE_DATA"] or ["NETWORK_RESOURCE_DATA", "ENM_DATA"] Depending on system configuration parameter. — dynamic ["NETWORK_RESOURCE_DATA", "ENM_DATA"] 	<pre>["NETWORK_RESOURCE_DATA", "ENM_DATA"] ["TCIM_DATA"]</pre>



Additional Attributes	Description	Optional	Default Value	Values Examples
	<p>— TCIM_DATA Export file includes Network and its subtree data. TCIM_DATA is only supported for the FDN scope Network=1. It cannot be specified with any other data category or scope.</p> <p>The order of elements in the list determines the order of the corresponding data in the export file. However, duplicate elements are ignored.</p> <p>If exporting data for multiple nodes, the output is grouped by data categories rather than by node.</p>			
exportFilter	exportFilter.json ⁽¹⁾	yes	<pre>"exportFilter": { "namespace": "NO_FILTER", "name": "NO_FILTER", "version": "NO_FILTER" }</pre>	
nodeSearchCriteria	nodeSearchCriteria.json ⁽²⁾	yes	<pre>"nodeSearchCriteria": { "nodeSearchScopes": [{ "scopeType": "UNSPECIFIED", "matchCondition": "NO_MATCH_REQUIRED" }], "collections": [], "savedSearches": [] }</pre>	
compressionType	The type of file compression.	yes	ZIP	<p>— ZIP</p> <p>— GZIP</p> <p>— NONE</p>
enumTranslate	Flag to determine how to represent enumeration attributes: by literal (true) or by integer (false).	yes	true	<p>— true</p> <p>— false</p>
jobName	The name to be assigned to the starting job.	yes	-----	<p>Must start with a letter, numeral, or underscore. It can contain any other combination of letters, numerals, underscores, or hyphens.</p> <p>Example: "[a-zA-Z0-9_]([a-zA-Z0-9_] -)*"</p>
userFilter	userFilter.json ⁽³⁾	yes		"userFilter": {



Additional Attributes	Description	Optional	Default Value	Values Examples
				<pre>"value": "ENodeBFunction. (eNBId, eNodeBPlmnId); EUTranCellFDD.*" }</pre>
syncState	<p>Flag to determine which nodes are exported depending on the synchronization status (the value of CmFunction.syncStatus) :</p> <ul style="list-style-type: none"> — SYNC: Only synchronized nodes are exported. Synchronized nodes are nodes with CmFunction.syncStatus equal to SYNCHRONIZED. — ALL: Nodes with any synchronization state are included in the export file. <p>Non-synchronized nodes are added to the 'nodesNotExported' count in the GET response for the export job endpoint "/bulk/export/jobs/{jobId}". The GET response of the job report endpoint "/bulk/export/reports/{reportId}" includes the non-synchronized nodes in the "notExportedNodesResult".</p>	yes	If syncState is not specified, the default behavior configured by the system administrator is used.	<ul style="list-style-type: none"> — SYNC — ALL
prettyFormat	<p>Flag to determine if the export file needs to be pretty formatted: pretty format (true) or non-pretty format (false). Pretty formatting generates export files with Line Feeds and indentations (white spaces). If the parameter is not included, the default behavior configured by the system administrator is used.</p> <p>Note:</p>	yes	System Default	<ul style="list-style-type: none"> — true — false



Additional Attributes	Description	Optional	Default Value	Values Examples
	This parameter applies only to 3GPP export files. For dynamic export files, this parameter does not apply and is ignored.			

- (1) exportFilter is mutually exclusive with userFilter and the opposite is also true.
- (2) The maximum number of node search scopes that are supported is 700.
- (3) If userFilter contains non-persistent attributes, the time taken for export depends on external factors such as network latency and node response time. This delay in response is also applicable for pre-defined filters, if they are known to contain non-persistent attributes. An example of such pre-defined filter is In_Service_Performance_Data.

Note: *TCIM (Transport Common Information Model) Model Description* details information on the managed objects, attributes, and relationships. See [TCIM Model Description](#).

4.2.2 exportFilter.json

The `exportFilter.json` represents the JSON schema for the filter.

Additional Attributes exportFilter.json	Description	Optional	Default Values	Example Values
nameSpace	The name of the nameSpace	no		<ul style="list-style-type: none"> — PredefinedExportFilter — NO_FILTER —
name	The name of the filter	no		<ul style="list-style-type: none"> — ERBS — SON — InventoryData — Subnetwork — NO_FILTER —
version	The version of the filter	yes		<ul style="list-style-type: none"> — 1.0.0 — NO_FILTER —

exportFilter.json

```
{
  "type": "object",
  "title": "exportFilter",
  "properties": {
    "namespace": {
      "type": "string"
    }
  }
}
```



```

    "name": {
      "type": "string"
    },
    "version": {
      "type": "string"
    }
  }
}

```

4.2.3 nodeSearchCriteria.json

The nodeSearchCriteria.json represents the JSON schema for the node search criteria

Additional Attribute nodeSearchCriteria.json	Description	Optional	Default Value	Example Values
nodeSearchScopes	nodeSearchScopes.json (2)	Yes	If no node scope is specified, the default behavior is to export all nodes. This is the equivalent to the following: <pre> "nodeSearchScopes": [{ "scopeType": "UNSPECIFIED", "matchCondition": "NO_MATCH_REQUIRED" }] </pre>	
collections	Can be used to specify a list of Collections indicating nodes to export	Yes	[]	Specify single collection: "collections": ["MyCollection"] Specify multiple collections: "collections": ["MyCollection1", "MyCollection2"]
savedSearches	Can be used to specify a list Saved Searches indicating nodes to export	Yes	[]	Specify single saved search: "savedSearches": ["MySavedSearch"] Specify multiple saved searches: "savedSearches": ["MySavedSearch1", "MySavedSearch2"]

The nodeSearchScope.json represents the JSON schema for the node search scope.

Additional Attribute nodeSearchScope.json	Description	Default Value	Example Values
scopeType	The parameter for search (by FDN or by nodeName).	If nodeSearchScope is not present, then the default value is: UNSPECIFIED.	The possible values are: — FDN — NODE_NAME



Additional Attribute nodeSearchScope.json	Description	Default Value	Example Values
			<ul style="list-style-type: none"> — UNSPECIFIED (In this case, value = null. For scopeType, null is not acceptable)
matchCondition		<p>If nodeSearchScope is not present, then the default value is: NO_MATCH_REQUIRED.</p>	<p>If scopeType = FDN, the field is ignored. If scopeType = NODE_NAME, the possible values are:</p> <ul style="list-style-type: none"> — EQUALS — STARTS_WITH — ENDS_WITH — CONTAINS — NO_MATCH_REQUIRED (In this case, value = null. For matchCondition, null is not acceptable)
value	<p>The value of the node name, partial node name, or FDN. As for the FDN, the resulting search scope is the node that the specified FDN belongs to. Both Network Resource Model and ENM Data Model FDNs are supported.</p>		<p>Possible values are:</p> <ul style="list-style-type: none"> — Node name (for example, ERBS5) — Managed Object FDN (for example, MeContext=ERBS005, ManagedElement=1, ENodeBFunction=1, EUTranCellFDD=Cell1 or NetworkElement=ERBS002) The node that the specified FDN belongs to is included. Any FDN from the Network Resource Data Model or ENM Data Model can be used. — SubNetwork FDN (for example SubNetwork=ERBS003) All nodes within the specified SubNetwork are included. — null
neType	<p>The NE type of the NodeSearchScope used for filtering. Applicable when the scopeType is NODE_NAME or UNSPECIFIED.</p>		<p>Possible values are:</p> <ul style="list-style-type: none"> — ERBS5 — SGSN-MME — RadioNode — router-6672 <p>and so on.</p>



Example 4 nodeSearchCriteria.json Schema

```

{
  "type": "object",
  "title": "nodeSearchCriteria",
  "properties": {
    "nodeSearchScopes": {
      "type": "array",
      "items": {
        "$ref": "#/definitions/nodeSearchScope"
      }
    },
    "collections": {
      "type": "array",
      "items": {
        "type": "string"
      }
    },
    "savedSearches": {
      "type": "array",
      "items": {
        "type": "string"
      }
    }
  },
  "definitions": {
    "nodeSearchScope": {
      "type": "object",
      "properties": {
        "scopeType": {
          "type": "string"
        },
        "matchCondition": {
          "type": "string"
        },
        "value": {
          "type": "string"
        },
        "neType": {
          "type": "string"
        }
      }
    }
  }
}

```

4.2.4 userFilter.json

The userFilter.json represents the JSON schema for the user-defined filter.

Additional Attributes userFilter.json	Description	Optional	Default Values	Example Values
value	The content of the user-defined filter	No		INCLUDE FILTER EXAMPLES — "EUtranCellFDD.tac" — "EUtranCellFDD.*" — "ENodeBFunction+.*" — "EUtranCellFDD.(*)"



Additional Attributes userFilter.json	Description	Optional	Default Values	Example Values
				<ul style="list-style-type: none"> — "EUTranCellFDD. (TAC, altitude, cellId)". — "EUTranCellFDD. (*, pmErabBrMax, pmErabBrSum)" — "ManagedElement.userLabel;ENodeBFunction.randomUpdateInterval;EUTranCellFDD.cellId". — "EUTranCellFDD.<m>" — "ENodeBFunction+.<m>" — "EUTranCellFDD.<m>" — "ENodeBFunction. (eNBId,eNodeBPlmnId);EUTranCellFDD.<m>" — "EUTranCellFDD. (<m>,pmErabBrMax,pmErabBrSum)" — "EUTranCellFDD.<w>" — "ENodeBFunction+.<w>" — "EUTranCellFDD.<w>" — "ENodeBFunction. (eNBId,eNodeBPlmnId);EUTranCellFDD.<w>" — "EUTranCellFDD. (<w>,pmErabBrMax,pmErabBrSum)" <p>EXCLUDE FILTER EXAMPLES</p> <ul style="list-style-type: none"> — "!ENodeBFunction" — "!ENodeBFunction+" <p>The syntax supported for user-defined filters is described in the <i>User-Defined Filter File</i> section of the <i>CLI Help Center</i>, which is available at the following URL: <a href="https://<apache_host_name>/#help/app/cliapp/topic/tutorials_ccredit/export/UserDefinedFilterFile">https://<apache_host_name>/#help/app/cliapp/topic/tutorials_ccredit/export/UserDefinedFilterFile. The <apache_host_name> is the httpd hostname. Only a single line of filter content is supported as the filter content value. To use the content of a multi-line filter file, remove the line breaks to obtain a single line. For the INCLUDE MO Class, EXCLUDE MO Class, and EXCLUDE with subtree cases, the recommended maximum number of filter clauses is 300. Numbers higher than this can affect the performance of the export. For the INCLUDE with subtree case, the recommended maximum number of filter clauses is 10. Higher numbers can affect the performance of the export.</p>

Example 5 userFilter.json

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "title": "userFilter",
  "description": "Specifies the content of a user defined filter",
  "type": "object",
  "properties": {
    "value": {
      "type": "string"
    }
  }
},
```



```
  "required": ["value"]  
}
```

4.2.5 exportTCIM.json

The exportTCIM.json represents the JSON (fixed schema) to export TCIM data.

```
{  
  "nodeSearchCriteria" : {  
    "nodeSearchScopes" : [  
      {  
        "scopeType" : "FDN",  
        "matchCondition" : "EQUALS",  
        "value" : "Network=1"  
      }  
    ]  
  },  
  "dataCategories" : [ "TCIM_DATA" ],  
  "fileFormat" : "3GPP",  
  "configName" : "Live",  
  "type" : "EXPORT"  
}
```



4.3 DELETE Endpoints for EXPORT

This section lists the DELETE Endpoints having additional attributes for EXPORT.

4.3.1 DELETE /bulk/export/jobs/{jobId}

HTTP Supported Method	Method Information
DELETE	Deletes the export job with the specified jobId.

DELETE Response

Example 6

```
{
  "_links": {
    "self": {
      "href": "/bulk/export/jobs/173"
    },
    "export_jobs": {
      "href": "/bulk/export/jobs/"
    }
  },
  "type": "EXPORT",
  "id": 173
}
```

4.3.2 DELETE /bulk/export/jobs?jobName={jobName}

HTTP Supported Method	Method Information
DELETE	Deletes the export job with the specified jobName.

DELETE Response Additional Attribute

The additional attribute for the DELETE /bulk/export/jobs?jobName={jobName} is jobName.

Additional Attribute	Description	Optional
jobName	The name of the deleted export job.	no

Example 7

```
{
  "_links": {
    "self": {
      "href": "/bulk/export/jobs?jobName=export_dynamic_2017-04-12T08-39-11-080_02e48948-87e7-482f-91a9-6347bce1768c" →
    },
    "export_jobs": {
      "href": "/bulk/export/jobs/"
    }
  },
  "type": "EXPORT",
  "jobName": "export_dynamic_2017-04-12T08-39-11-080_02e48948-87e7-482f-91a9-634 →
}
```



```
7bce1768c"  
}
```



5 Bulk Import Additional Attributes

This section lists the supported endpoints with additional IMPORT attributes.

Note: The CM Bulk Import use cases in this document are now deprecated and all customers are advised to migrate to the [CM Bulk Import REST NBI](#).

Context	HTTP Supported Methods	
/bulk/import/jobs	GET	POST
/bulk/import/jobs/{jobId}	GET	
/bulk/import/reports/{reportId}	GET	

5.1 GET Endpoint for IMPORT

This section lists the GET Endpoints having additional attributes for IMPORT.

5.1.1 GET /bulk/import/jobs without Query Parameters

HTTP Supported Methods	Method Information
GET	Returns the list of the available import jobs. Note: The Response displays the most recent jobs only and shows a maximum of 10,000 entries.

GET Response Additional Attributes:

The additional attributes for the GET /bulk/import/jobs are:

- configName
- fileName
- fileFormat
- totalManagedObjectCreated
- totalManagedObjectUpdated
- totalManagedObjectDeleted
- nodesCopied



— nodesNotCopied

Additional Attributes	Description	Optional
configName	The name of the configuration.	no
fileName	The name of the file.	no
fileFormat	The format of the file.	no
totalManagedObjectCreated	The number of MOS created.	no
totalManagedObjectUpdated	The number of MOS updated.	no
totalManagedObjectDeleted	The number of MOS deleted.	no
nodesCopied	The number of copied nodes.	no
nodesNotCopied	The number of nodes not copied.	no

Example 8

```
{
  "jobs": [
    {
      "id": 1,
      "type": "IMPORT",
      "status": "COMPLETED",
      "startTime": "2016-02-12T13:46:25.912",
      "endTime": "2016-02-12T13:45:16.126",
      "configName": "Live",
      "totalManagedObjectDeleted": 0,
      "fileName": "import.xml",
      "totalManagedObjectUpdated": 3,
      "totalManagedObjectCreated": 2,
      "fileFormat": "3GPP",
      "nodesCopied": 2,
      "nodeNotCopied": 0,
      "userId": "Administrator",
      "_links": {
        "self": {
          "href": "/bulk/import/jobs/1"
        }
      }
    },
    {
      "id": 2,
      "type": "IMPORT",
      "status": "COMPLETED",
      "startTime": "2016-02-12T115:15:78.169",
      "endTime": "2016-02-12T115:15:81.365",
      "configName": "Live",
      "totalManagedObjectDeleted": 0,
      "fileName": "import.xml",
      "totalManagedObjectUpdated": 2,
      "totalManagedObjectCreated": 1,
      "fileFormat": "3GPP",
      "nodesCopied": 1,
      "nodeNotCopied": 1,
      "userId": "Administrator",
      "_links": {
        "self": {
          "href": "/bulk/import/jobs/2"
        }
      }
    }
  ],
  "_links": {
    "self": {
      "href": "/bulk/import/jobs"
    }
  }
}
```



5.1.2 GET /bulk/import/jobs?begin={beginDate}&end={endDate} with Query Parameters

HTTP Supported Methods	Method Information
GET	Returns the list of the available import jobs having start time after begin query parameter value and before end query parameter value. Note: The Response displays the most recent jobs only and shows a maximum of 10,000 entries.

It is possible to call a request GET /bulk/import/jobs with query parameters begin (time in string form) and end (time in string form), where the end time is after begin time. The response includes only those import jobs having start time after begin query parameter value and before end query parameter value.

Supported Time formats for beginDate and endDate are the following:

1. Date Time Format - "yyyy-MM-dd'T'HH:mm:ss"
2. Time Format - "HH:mm:ss"
3. Date Format - "yyyy-MM-dd"

GET Response Additional Attributes

The additional attributes for the GET /bulk/import/jobs?begin={beginDate}&end={endDate} are the same additional attributes as for the GET /bulk/import/jobs.

5.1.3 GET /bulk/import/jobs/{jobId}

HTTP Supported Method	Method Information
GET	Returns the status of a specific previously created import job.

GET Response Additional Attributes

The additional attributes for the GET /bulk/import/jobs/{jobId} are the same additional attributes as for the GET /bulk/import/jobs.

5.1.4 GET /bulk/import/reports/{reportId}

HTTP Supported Method	Method Information
GET	Returns the details about the imported nodes involved in a specific job.

If the request is done when the job status is COMPLETED or FAILED, the response contains details of the final status of the job.



If the request is done while the job status is still IN PROGRESS, the response contains details of the status of the job at the time the request is raised.

GET Response Additional Attributes

The additional attribute for the GET `/bulk/import/reports/{reportId}` is:

Additional Attribute	Description	Optional	Value
operations	The job details	no	Array of <code>operation_details.json</code>

Example 9

```

{
  "_links": {
    "self": {
      "href": "/bulk/import/reports/1"
    }
  },
  "id": 1,
  "type": "IMPORT",
  "operations": [
    {
      "importStatus": "SUCCESS",
      "operationType": "create",
      "updateTime": "2016-01-06T13:48:13.423",
      "fdn": "MeContext=THEAVENGERS_1,ManagedElement=1,ENodeBFunction=1",
      "lineNumber": 136
    },
    {
      "importStatus": "SUCCESS",
      "operationType": "create",
      "updateTime": "2016-01-06T13:48:25.126",
      "fdn": "MeContext=THEAVENGERS_1,ManagedElement=1,ENodeBFunction=1,EUtranCe
11TDD=THEAVENGERS1_EUC1",
      "lineNumber": 142
    },
    {
      "importStatus": "SUCCESS",
      "operationType": "create",
      "updateTime": "2016-01-06T14:25:25.475",
      "fdn": "MeContext=THEAVENGERS_1,ManagedElement=1,ENodeBFunction=1,EUtranCe
11TDD=THEAVENGERS1_EUC2",
      "lineNumber": 149
    },
    {
      "importStatus": "FAILED",
      "operationType": "create",
      "updateTime": "2016-01-06T15:15:46.515",
      "fdn": "MeContext=LTE02ERBS00001,ManagedElement=1,ENodeBFunction=1,EUtranC
e11FDD=LTE20ERBS00001-13"
      "failureInformation": "Child of type EUtranCellFDD with name LTE20ERBS0000
1-13 already exists under the parent MO MeContext=LTE02ERBS00001,ManagedElement
=1,ENodeBFunction=1",
      "lineNumber": 156
    }
  ]
}
-----
{
  "_links": {
    "self": {
      "href": "/bulk/import/reports/134"
    },
    "retries": {
      "href": "/bulk/import/jobs/134/retries"
    }
  },
  "id": 134,
  "type": "IMPORT",
  "operations": [

```



```

    {
      "failureInformation": "Error 7021 : An error occurred mediating a UP
DATE operation for the persistence object: Managed Object::   fdn:MeContext=LTE
05ERBS00003,ManagedElement=1, type:ManagedElement, namespace:ERBS_NODE_MODEL, ve
rsion:10.1.40PM, root cause: Node ID: svc-1-mscm. Exception occurred: Error retr
ieving moci connection: Error occurred creating ORB for node: ConnectionConfig [
ipAddress=192.168.100.147, secure=false], isIPv6: false, error msg [Client is u
nable to establish a connection to the NE with IP address 192.168.100.147 during
the timeout period with ]",
      "importStatus": "FAILED",
      "updateTime": "2017-06-27T08:03:16.573",
      "operationType": "update",
      "fdn": "MeContext=LTE05ERBS00003,ManagedElement=1",
      "lineNumber": 1
    },
    {
      "importStatus": "SUCCESS",
      "updateTime": "2017-06-27T08:03:17.075",
      "operationType": "UPDATE",
      "fdn": "MeContext=LTE05ERBS00004,ManagedElement=1",
      "lineNumber": 6
    }
  ]
}

```

5.1.5 operation_details.json

The operation_details.json is the schema for the operation details.

Additional Attribute operation_details.json	Description	Optional
importStatus	The status of the single bulk operation.	no
operationType	The type of the single bulk operation.	no
updateTime	The time of the single bulk operation.	no
fdn	The fully Distinguished Name of the MO involved in the operation.	no
failureInformation	The failure information on the operation that caused the error.	yes Present only if an error occurs.
lineNumber	The line number in the import file related to the single bulk operation.	no

Example 10

```

{
  "type": "object",
  "title": "operation_details",
  "properties": {
    "importStatusMessage": {
      "type": "string"
    },
    "operationType": {
      "type": "string"
    },
    "updateTime": {
      "type": "string"
    },
    "fdn": {
      "type": "string"
    },
    "failureInformation": {

```



```
    "type": "string"
  },
  "lineNumber": {
    "type": "integer"
  }
}
```

5.2 POST Endpoint for IMPORT

This section lists the POST endpoints with additional attributes.

Note: There is no Bulk CM Import support for Managed Objects below BscM on the BSC node via this REST interface. CM Bulk Import to Live for BscM Managed Objects on the BSC node is supported by the Import to Live interface documented in *CM Bulk Import REST Northbound Interface* [Interwork Description](#).

5.2.1 POST /bulk/import/jobs

HTTP Supported Method	Method Information
POST	Returns the file URI to be used to stream the file to the server.

POST Request Attributes

The additional attributes for the POST /bulk/import/jobs are the following:

- fileFormat
- FileName
- continueOnErr
- validate

Note: In JSON, the order of the fields is not relevant.

Additional Attributes	Description	Optional	Default Value	Example Values
fileFormat	The file format.	no		— 3GPP — dynamic
fileName	The name of the file.	no		
continueOnErr	Without this flag, the default behavior is immediate import failure on the first error encountered. A string that can assume one of these values: CONTINUE/NODE, OPERATION, or STOP. This field ensures that	yes	STOP	— CONTINUE/ NODE — OPERATION — STOP



Additional Attributes	Description	Optional	Default Value	Example Values
	import continues and attempts to import all file data even when a non-fatal error is encountered; that is, invalid attribute or missing mandatory attribute. Fatal errors such as an incorrectly formatted import file still cause immediate import failure.			
validate	<p>This field facilitates inclusion or exclusion of one or more selected validation phases during an import.</p> <p>Supported values are 'noinstance' and 'default'. When the 'noinstance' value is supplied, the import service skips instance validation rules.</p> <p>With 'default' or no validation flag, the import service behavior is to proceed executing the operations. This only happens if all the validation phases (Model Validation and Instance Validation) are successful.</p> <p>For more details on the validation phases, refer to Appendix C - Bulk Import Validation Phases on page 63</p>	yes	Default	<ul style="list-style-type: none"> — noinstance — default

Stop on Error:

A string that can assume one of these values: CONTINUE/NODE, An operation failure for a single node: it stops the execution of operations pertaining to the same or successive nodes.

Continue on Error on Node Level:

An operation failure for a single node: it affects the execution of operations pertaining to the same nodes and continues with the execution of successive nodes. All the operations before the failed operations are ended and all the operations after the failed operation are skipped. Any subsequent operations for the same node are not processed.

Continue on Error on Operation Level:

Verify that an operation failure for a single node does not affect the execution of operations pertaining to the same or successive nodes. All the operations before and after the failed operations are executed successfully. Any subsequent operations for the same node are processed.



Example 11

```

{
  "type": "IMPORT",
  "fileFormat": "3GPP",
  "fileName": "Import_File_2.xml",
  "continueOnErr": "STOP"
}
-----
{
  "type": "IMPORT",
  "fileFormat": "dynamic",
  "fileName": "import_Dyn.txt",
  "continueOnErr": "CONTINUE"
}
-----
{
  "type": "IMPORT",
  "fileFormat": "3GPP",
  "fileName": "Import_File_3.xml",
  "continueOnErr": "NODE"
}
-----
{
  "type": "IMPORT",
  "fileFormat": "dynamic",
  "fileName": "Import_File_4.txt",
  "continueOnErr": "OPERATION"
}
-----
{
  "type": "IMPORT",
  "fileFormat": "3GPP",
  "fileName": "Import_File_5.xml",
  "validate": "noinstance"
}
-----
{
  "type": "IMPORT",
  "fileFormat": "3GPP",
  "fileName": "Import_File_6.xml",
  "validate": "default"
}

```

POST Response Additional Attribute:

Additional attribute for the POST /bulk/import/jobs:

- fileUri

Additional Attribute	Description	Optional
fileUri	The name of the import file	no

Example 12

```

{
  "_links": {
    "self": {
      "href": "/bulk/import/jobs"
    }
  },
  "type": "IMPORT",
  "fileUri": "/bulk/import/jobs/Live/3GPP/Import_File_2.xml/true/STOP/default"
}
-----
{
  "_links": {
    "self": {
      "href": "/bulk/import/jobs"
    }
  }
}

```



```

    },
    "type": "IMPORT",
    "fileUri": "/bulk/import/jobs/Live/dynamic/import_Dyn.txt/false/CONTINUE/default" →
  }
}
-----
{
  "_links": {
    "self": {
      "href": "/bulk/import/jobs"
    }
  },
  "type": "IMPORT",
  "fileUri": "/bulk/import/jobs/Live/3GPP/Import_File_3.xml/false/NODE/default"
}
-----
{
  "_links": {
    "self": {
      "href": "/bulk/import/jobs"
    }
  },
  "type": "IMPORT",
  "fileUri": "/bulk/import/jobs/Live/dynamic/Import_File_4.txt/false/OPERATION/default" →
}
-----
{
  "_links": {
    "self": {
      "href": "/bulk/import/jobs"
    }
  },
  "type": "IMPORT",
  "fileUri": "/bulk/import/jobs/Live/dynamic/import_Dyn_live.csv/true/CONTINUE/default" →
}
-----
{
  "_links": {
    "self": {
      "href": "/bulk/import/jobs"
    }
  },
  "type": "IMPORT",
  "fileUri": "/bulk/import/jobs/Live/3GPP/Import_File_5.xml/true/STOP/noinstance" →
}
-----
{
  "_links": {
    "self": {
      "href": "/bulk/import/jobs"
    }
  },
  "type": "IMPORT",
  "fileUri": "/bulk/import/jobs/Live/3GPP/Import_File_6.xml/true/STOP/default"
}
}

```

Note: To import links, the file must contain relevant TCIM link data under the Network =1 datacontainer of datatype VsDataContainer and the involved interfaces must already be configured. Nodes must be in a sync state.

TCIM (Transport Common Information Model) Model Description details information on the managed objects, attributes and relationships. See [TCIM Model Description](#).



5.2.2 POST /bulk/import/jobs/{jobId}/retries

HTTP Supported Method	Method Information
POST	Asks for the retry of a completed import to Live job which failed during the execution phase.

POST Response Additional Attribute:

The additional attribute for the POST /bulk/import/jobs/{jobId}/retries is:

— `retriedJobId`

Additional Attribute	Description	Optional
<code>retriedJobId</code>	The number of the job to be retried	yes

Note: In json the order of the fields is not relevant.

Example 13

```
{
  "type": "IMPORT",
  "retriedJobId": 133,
  "continueOnErr": "NODE"
}
```



6 Configuration Additional Attributes

The supported endpoints having additional attributes for CONFIGURATION are shown in the following table:

Context	HTTP Supported Methods	
/configuration/jobs	GET	POST
/configuration/jobs/{jobId}	GET	
/configuration/reports/{reportId}	GET	

6.1 GET Endpoint for Configuration

The following list of GET Endpoints have additional attributes for Configuration:

6.1.1 GET /configuration/jobs?type=UNDO_IMPORT_TO_LIVE

HTTP Supported Method	Method Information
GET	Returns the list of the available jobs in accordance to the selected type.

Get Response Additional Attributes

The additional attribute is:

- fileURI

Additional Attribute	Description	Optional
fileURI	The name of the file generated for undo	no

Example 14

```
{
  "jobs": [
    {
      "_links": {
        "self": {
          "href": "/configuration/jobs/309?type=UNDO_IMPORT_TO_LIVE"
        }
      },
      "id": 309,
      "status": "COMPLETED",
      "statusReason": "COMPLETED",
      "type": "UNDO_IMPORT_TO_LIVE",
      "creationTime": "2016-10-07T14:36:53.997",
      "startTime": "2016-10-07T14:36:53.997",
      "endTime": "2016-10-07T14:36:54.682",
      "lastUpdateTime": "2016-10-07T14:36:54.682",
      "userId": "administrator",
      "jobId": "289",
      "undoOperations": 3,
      "totalOperations": 3,
      "fileUri": "/configuration/PZcVUIcLCYUedfGZUWTRBPacTTVIHDAundo_2016-11-17T14-36-53_309.txt"
    }
  ],
}
```



```

{
  "_links": {
    "self": {
      "href": "/configuration/jobs/300?type=UNDO_IMPORT_TO_LIVE"
    }
  },
  "id": 300,
  "status": "COMPLETED",
  "statusReason": "COMPLETED",
  "type": "UNDO_IMPORT_TO_LIVE",
  "creationTime": "2016-10-07T13:38:41.664",
  "startTime": "2016-10-07T13:38:41.664",
  "endTime": "2016-10-07T13:38:42.172",
  "lastUpdateTime": "2016-10-07T13:38:42.172",
  "userId": "administrator",
  "jobId": "198",
  "undoOperations": 3,
  "totalOperations": 3,
  "fileUri": "/configuration/PZcVUIcLCYUEdfGZUWTRBPacTTVIHDAundo_2016-11-17T
13-38-41_300.txt"
},
  "_links": {
    "self": {
      "href": "/configuration/jobs?type=UNDO_IMPORT_TO_LIVE"
    }
  }
}

```

Note: The endTime and lastUpdatedTime fields in the response are present only when the job status is COMPLETED or FAILED.

6.1.2 GET /configuration/jobs/{jobId}?type=UNDO_IMPORT_TO_LIVE

HTTP Supported Method	Method Information
GET	Returns the status of a specific job in accordance to the selected type.

Get Response Additional Attributes

The additional attributes are the same in case of "all jobs" request.

6.2 POST Endpoint for Configuration

Following the list of POST Endpoints having additional properties for Configuration:

6.2.1 POST /configuration/jobs (UNDO IMPORT)

HTTP Supported Method	Method Information
POST	A new job to undo a previous import job is started. The job id and the status of the job are returned.

Post Request Attributes

The additional attribute is:



— fileFormat

Additional Attribute	Description	Optional	Default Value	Values Example
fileFormat	The format requested for the undo file	yes	<ul style="list-style-type: none">— dynamic, if the Dynamic Value Package License is installed— 3GPP, otherwise	The possible values are: <ul style="list-style-type: none">— 3GPP— dynamic

Example 15

```
{
  "type": "UNDO_IMPORT_TO_LIVE",
  "id": "369",
  "fileFormat": "dynamic"
}
```



7 Exporting and Importing PLM Links

In some cases it is useful to save and then regenerate the Links configuration.

Exporting Links

To export the links use the specified filter with the following command:

```
POST https://enmapache.atthtem.eei.ericsson.se/bulk/export/jobs
```

```
JSON body
```

```
{
  "nodeSearchCriteria" : {
    "nodeSearchScopes" : [
      {
        "scopeType" : "FDN",
        "matchCondition" : "EQUALS",
        "value" : "Network=1",
        "neType" : null
      }
    ]
  },
  "userFilter": {
    "value" : "Link.*"
  },
  "fileFormat" : "3GPP",
  "configName" : "Live",
  "type" : "EXPORT",
  "dataCategories" : [
    "TCIM_DATA"
  ]
}
```



Exported File Example (Extract) :

```

<?xml version="1.0" encoding="UTF-8"?>
<bulkCmConfigDataFile xmlns="configData.xsd" xmlns:gn="geranNrm.xsd" xmlns:xn="g
genericNrm.xsd" xmlns:un="utranNrm.xsd" xmlns:es="EricssonSpecificAttributes.xsd" →
>
  <fileHeader vendorName="Ericsson" fileFormatVersion="32.615 V4.5"/>
  <configData dnPrefix="Undefined">
    <xn:VsDataContainer id="1">
      <xn:attributes>
        <xn:vsDataType>vsDataNetwork</xn:vsDataType>
        <xn:vsDataFormatVersion>EricssonSpecificAttributes</xn:vsDataFor →
matVersion>
        <es:vsDataNetwork/>
      </xn:attributes>
      <xn:VsDataContainer id="Id-1574697969625-SPFRER60001_1/1-SPFRER60002 →
_1/1">
        <xn:attributes>
          <xn:vsDataType>vsDataLink</xn:vsDataType>
          <xn:vsDataFormatVersion>EricssonSpecificAttributes</xn:vsDat →
aFormatVersion>
          <es:vsDataLink>
            <es:link-state>DEFINED</es:link-state>
            <es:sdn-applicable/>
            <es:link-id>Id-1574697969625-SPFRER60001_1/1-SPFRER60002 →
_1/1</es:link-id>
            <es:supporting-links-layer-rate/>
            <es:description>ETHERNET_1Gb</es:description>
            <es:active>true</es:active>
            <es:sdn-populated/>
            <es:type>NOT_AVAILABLE</es:type>
            <es:source-tp>vsDataNetwork=1,vsDataNode=SPFRER60001,vsD →
ataTerminationPoint=SPFRER60001_1_1/1</es:source-tp>
            <es:supporting-links/>
            <es:alarm-status>UNKNOWN</es:alarm-status>
            <es:link-name>SPFRER60001_1/1-SPFRER60002_1/1</es:link-n →
ame>
            <es:dest-tp>vsDataNetwork=1,vsDataNode=SPFRER60002,vsDat →
aTerminationPoint=SPFRER60002_1_1/1</es:dest-tp>
            <es:supported-links/>
            <es:discovery-state>NOT_DISCOVERED</es:discovery-state>
            <es:supporting-links-number/>
            <es:layer-rate>ETHERNET_1Gb</es:layer-rate>
            <es:creation-date>Mon Nov 25 16:06:09 GMT 2019</es:creat →
ion-date>

```



```
        <es:created-by>administrator</es:created-by>
      </es:vsDataLink>
    </xn:attributes>
  </xn:VsDataContainer>
  <xn:VsDataContainer id="Id-1574698014125-SPFRER60001_1/3-SPFRER60002_
1/3">
    <xn:attributes>
      <xn:vsDataType>vsDataLink</xn:vsDataType>
      <xn:vsDataFormatVersion>EricssonSpecificAttributes</xn:vsDa
taFormatVersion>
      <es:vsDataLink>
        <es:link-state>DEFINED</es:link-state>
        <es:sdn-applicable/>
        <es:link-id>Id-1574698014125-SPFRER60001_1/3-SPFRER60002
_1/3</es:link-id>
        <es:supporting-links-layer-rate/>
        <es:description>ETHERNET__1Gb</es:description>
        <es:active>true</es:active>
        <es:sdn-populated/>
        <es:type>NOT_AVAILABLE</es:type>
        <es:source-tp>vsDataNetwork=1,vsDataNode=SPFRER60001,vsD
ataTerminationPoint=SPFRER60001_1_1/3</es:source-tp>
        <es:supporting-links/>
        <es:alarm-status>UNKNOWN</es:alarm-status>
        <es:link-name>SPFRER60001_1/3-SPFRER60002_1/3</es:link-n
ame>
        <es:dest-tp>vsDataNetwork=1,vsDataNode=SPFRER60002,vsDat
aTerminationPoint=SPFRER60002_1_1/3</es:dest-tp>
        <es:supported-links/>
        <es:discovery-state>NOT_DISCOVERED</es:discovery-state>
        <es:supporting-links-number/>
        <es:layer-rate>ETHERNET_1Gb</es:layer-rate>
        <es:creation-date>Mon Nov 25 16:06:54 GMT 2019</es:creat
ion-date>
        <es:created-by>administrator</es:created-by>
      </es:vsDataLink>
    </xn:attributes>
  </xn:VsDataContainer>
</xn:VsDataContainer>
</configData>
<fileFooter dateTime="2019-11-25T16:07:46.088Z"/>
</bulkCmConfigDataFile>
```



Adapting the File to Import

To adapt the file to Import, modify the exported file according to the following criteria:

- Remove all information not related to links.
- Substitute all “-<” characters with “<”.
- For each link, search for “<xn:VsDataContainer id=“Id-IDnumber-linkname”>” and update it to “<xn:VsDataContainer modifier=“create” id=“Id-IDnumber-linkname”>”.
- Delete all empty attributes (most common ones are: <es:supporting-links-layer-rate/>, <es:supporting-links-number/>, <sdn-populated/>, <es:sdn-applicable/>).
- Alter the non-empty attributes as needed.
- Add the string modifier=“create” in the xn:VsDataContainer related to each link.

Example Content of the Modified File

```
<?xml version="1.0" encoding="UTF-8"?>

<bulkCmConfigDataFile xmlns="configData.xsd" xmlns:gn="geranNrm.xsd" xmlns:xn="genericNrm.xsd" xmlns:un="utranNrm.xsd" xmlns:es="EricssonSpecificAttributes.xsd" →
<fileHeader vendorName="Ericsson" fileFormatVersion="32.615 V4.5"/> →
<configData dnPrefix="Undefined">
  <xn:VsDataContainer id="1">
    <xn:attributes>
      <xn:vsDataType>vsDataNetwork</xn:vsDataType>
      <xn:vsDataFormatVersion>EricssonSpecificAttr →
      ibutes</xn:vsDataFormatVersion>
      <es:vsDataNetwork/>
    </xn:attributes>
    <xn:VsDataContainer modifier="create" id="Id-1574 →
    675951414-SPFRER60001_1/1-SPFRER60002_1/1">
```



```

      <xn:attributes>
        <xn:vsDataType>vsDataLink</xn:vsDataT →
type>
        <xn:vsDataFormatVersion>EricssonSpeci →
ficAttributes</xn:vsDataFormatVersion>
        <es:vsDataLink>
          <es:link-state>DEFINED</es:lin →
k-state>
          <es:link-id>Id-1574675951414-S →
PFRER60001_1/1-SPFRER60002_1/1</es:link-id>
          <es:description>ETHERNET_1Gb</ →
es:description>
          <es:active>>true</es:active>
          <es:type>NOT_AVAILABLE</es:typ →
e>
          <es:source-tp>vsDataNetwork=1 →
,vsDataNode=SPFRER60001,vsDataTerminationPoint=SPFRER60001_1_1/1 →
</es:source-tp>
          <es:supporting-links/>
          <es:alarm-status>UNKNOWN</es: →
alarm-status>
          <es:link-name>SPFRER60001_1/1 →
-SPFRER60002_1/1</es:link-name>
          <es:dest-tp>vsDataNetwork=1,v →
sDataNode=SPFRER60002,vsDataTerminationPoint=SPFRER60002_1_1/1</ →
es:dest-tp>
          <es:supported-links/>
          <es:discovery-state>NOT_DISCO →
VERED</es:discovery-state>
          <es:layer-rate>ETHERNET_1Gb</ →
es:layer-rate>
          <es:created-by>administrator< →
/es:created-by>
        </es:vsDataLink>

```



```

        </xn:attributes>
    </xn:VsDataContainer>
    <xn:VsDataContainer modifier="create" id="Id-1574 675992256-SPFRER60001_1/3-SPFRER60002_1/3">
        <xn:attributes>
            <xn:vsDataType>vsDataLink</xn:vsD
ataType>
            <xn:vsDataFormatVersion>EricssonS
pecificAttributes</xn:vsDataFormatVersion>
            <es:vsDataLink>
                <es:link-state>DEFINED</es:lin
k-state>
                <es:link-id>Id-1574675992256-S
PFRRER60001_1/3-SPFRER60002_1/3</es:link-id>
                <es:description>ETHERNET__1Gb<
/es:description>
                <es:active>>true</es:active>
                <es:type>NOT_AVAILABLE</es:typ
e>
                <es:source-tp>vsDataNetwork=1,
vsDataNode=SPFRER60001,vsDataTerminationPoint=SPFRER60001_1_1/3<
/es:source-tp>
                <es:supporting-links/>
                <es:alarm-status>UNKNOWN</es:a
larm-status>
                <es:link-name>SPFRER60001_1/3-
SPFRER60002_1/3</es:link-name>
                <es:dest-tp>vsDataNetwork=1,vs
DataNode=SPFRER60002,vsDataTerminationPoint=SPFRER60002_1_1/3</e
s:dest-tp>
                <es:supported-links/>
                <es:discovery-state>NOT_DISCOV
ERED</es:discovery-state>

```



```
s:layer-rate>                                <es:layer-rate>ETHERNET_1Gb</e  →
es:created-by>                                <es:created-by>administrator</  →
                                                </es:vsDataLink>
                                                </xn:attributes>
                                                </xn:VsDataContainer>
</xn:VsDataContainer>
</configData>
<fileFooter dateTime="2019-11-25T10:15:09.544Z"/>
</bulkCmConfigDataFile>
```

Perform the import with the usual NBI import RESTs to apply the links to the live configuration.



8 Details of Errors Management

Two types of errors can occur:

- JBOSS error and exception
- REST error (in accordance with CDS Rest convention Error Handling)

8.1 JBOSS Error

JBOSS errors can occur as a result of one of the following:

- an incorrect endpoint
- errors that relate to JSON
- a value that does not belong to the declared Enum instance names

These errors have the following fields:

- http status code
- type
- message
- description

The following example displays the error that occurs when a user inserts a bad endpoint:

Example 16

```
<html>
  <head>
    <title>JBoss Web/7.3.2.Final-redhat-1 - JBWEB000064: Error report</title>
  </head>
  .....
  <body>
    <h1>JBWEB000065: HTTP Status 404 - Could not find resource for relative : →
    /configurations1111 of full path: https://enmapache.athtem.eei.ericsson.se/conf →
    igation/configurations1111</h1>
    <HR size="1" noshade="noshade">
      <b>JBWEB000309: type</b> JBWEB000067: Status report
      <b>JBWEB000068: message</b>
      <u>Could not find resource for relative : /configurations1111 of full →
      path: https://enmapache.athtem.eei.ericsson.se/configuration/configurations1111< →
      /u>
      <b>JBWEB000069: description</b>
      <u>JBWEB000124: The requested resource is not available.</u>
      <HR size="1" noshade="noshade">
        <h3>JBoss Web/7.3.2.Final-redhat-1</h3>
      </body>
    </html>
```



8.2 REST Error

A REST error occurs if a mistake is inserted in input fields (for example, if a non existing configuration name in the request is used), and an error message is reported.

The error message is shown as:

```
{
  "userMessage": " Error message to the user",
  "statusCode": 400,
  "internalErrorCode": 10203,
  "developerMessage": "Specific message to help developers solve the issue",
  "time": 2015-09-24T20:24:52.131
}
```

The message properties include:

- userMessage:** the message to be displayed to the end user
- statusCode:** the proper status code returned
- internalErrorCode:** the internal error code
- developerMessage:** a descriptive message including as much information as possible to clarify the problem
- time:** the timestamp corresponding to the time in which the error occurred

8.3 Error Response Code for Configuration

8.3.1 REST Error Response Codes for Configuration

The following is a minimum set of response codes:

HTTP Status Code	Description	Internal error code	Solution/Action required
HTTP 400 BAD REQUEST	Invalid job ID: {0}	9009	Specify a valid job ID.
HTTP 400 BAD REQUEST	Insufficient permissions to run command	9010	If it is necessary for the operation to be performed by the current user, the user profile must be updated by the system administrator.
HTTP 400 BAD REQUEST	The Job Type is invalid for the Configuration Service	-3	Provide the right Type for the Configuration Service.



HTTP Status Code	Description	Internal error code	Solution/Action required
HTTP 400 BAD REQUEST	Insufficient access rights to perform the operation	5001	If it is necessary for the operation to be performed by the current user, the user profile must be updated by the system administrator.
HTTP 500 INTERNAL SERVER ERROR	CM Config internal error	9000	Request Ericsson product support.

Note: {0} indicates a variable part of the message that is substituted by relevant information from the request; for example, configuration name, job ID, attribute name.

8.3.2 JBOSS Error Response Codes for Configuration

The following is a minimum set of codes:

HTTP status	Type	Message	Description
JBWEB000065: HTTP Status 400 org.codehaus.jackson.JsonParseException: Unexpected character ...	JBWEB000309: type JBWEB000067: Status report	JBWEB000068: message org.codehaus.jackson.JsonParseException: Unexpected character	JBWEB000069: description JBWEB000120: The request sent by the client was syntactically incorrect
JBWEB000065: HTTP Status 400 org.codehaus.jackson.map.JsonMappingException: Can not construct instance of ...	JBWEB000309: type JBWEB000067: Status report	JBWEB000068: message org.codehaus.jackson.map.JsonMappingException: Can not construct instance ofvalue not one of undeclared Enum instance names....	JBWEB000069: description JBWEB000120: The request sent by the client was syntactically incorrect
JBWEB000065: HTTP Status 404 Could not find resource for relative ...	JBWEB000309: type JBWEB000067: Status report	JBWEB000068: message Could not find resource for relative ...	JBWEB000069: description JBWEB000124: The requested resource is not available

8.4 Error Response Code for Export

8.4.1 REST Error Response Codes for Export

The following is a minimum set of response codes:

HTTP Status Code	Description	Internal error code	Solution/Action required
HTTP 400 BAD REQUEST	Invalid file path: {0}	-2	Provide the right file path for the download export operation.
HTTP 400 BAD REQUEST	The following Type IMPORT is invalid for the Export Service	-3	Provide the right Type for the Export Service.
HTTP 400 BAD REQUEST	File format Missing	1018	Provide the mandatory argument(s): {0}
HTTP 400 BAD REQUEST	User defined filter and predefined filter cannot be used together	8004	Specify either one of user defined filter or predefined filter.
HTTP 400 BAD REQUEST	User-defined filter contains both include and exclude filter clauses.	8009	Ensure user-defined filter content does not contain both



HTTP Status Code	Description	Internal error code	Solution/Action required
			include and exclude filter clauses.
HTTP 400 BAD REQUEST	User-defined filter {0} contains an invalid MO class definition: {1}	8009	Provide MO class and attribute definitions containing lowercase and uppercase letters, numbers, _- characters.
HTTP 400 BAD REQUEST	User-defined filter {0} contains an invalid MO attribute definition:{1}	8009	Provide MO class and attribute definitions containing lowercase and uppercase letters, numbers, _- characters.
HTTP 400 BAD REQUEST	User-defined filter {0} contains an invalid element inside attribute list: {1}	8009	Provide MO class and attribute definitions containing lowercase and uppercase letters, numbers, _- characters.
HTTP 400 BAD REQUEST	User-defined filter {0} contains unsupported attribute	8009	Remove attribute specification from filter.
HTTP 400 BAD REQUEST	User-defined filter {0} does not contain matching brackets for attribute list: {1}	8009	Provide matching brackets for attribute list.
HTTP 400 BAD REQUEST	User-defined filter {0} does not contain period character.	8009	Provide period between MO class and attributes definition.
HTTP 400 BAD REQUEST	User-defined filter {0} contains too many period characters.	8009	Provide period between MO class and attributes definition.
HTTP 400 BAD REQUEST	User-defined filter {0} contains blank characters.	8009	Remove blank characters from the given filter.
HTTP 400 BAD REQUEST	No nodes specified	8010	Specify nodes to export.
HTTP 400 BAD REQUEST	Invalid scope specified	8011	Specify a valid scope.
HTTP 400 BAD REQUEST	Invalid search criteria specified	8012	Specify a valid search criteria.
HTTP 404 NOT FOUND	Search criteria did not match any nodes	8013	Specify a valid search criteria.
HTTP 400 BAD REQUEST	Transformer not found for the export type {0}	8014	Specify a valid export type.
HTTP 400 BAD REQUEST	Invalid job ID {0}	8015	
HTTP 401 UNAUTHORIZED	Insufficient permissions to run command	8017	Log in as a user with permissions to run command.
HTTP 400 BAD REQUEST	Invalid input specified for managed objects filter	8018	Specify valid managed object names to filter.
HTTP 400 BAD REQUEST	Invalid input specified for export filter	8020	Specify a valid export filter.
HTTP 400 BAD REQUEST	Job with name {0} not found	8025	Specify a valid job name.
HTTP 400 BAD REQUEST	Job name {0} already exists	8026	Specify another job name.
HTTP 500 INTERNAL SERVER CODE	Exception occurred during export process.	8000	Request Ericsson product support.
HTTP 500 INTERNAL SERVER CODE	The server is currently busy due to a high number of exports	8027	Try again later.
	Insufficient disk space available for export job on {0}	8028	Refer to <i>Reduce System Usage for Shared File System for Bulk Export</i> in the <i>ENM System Administrator Guide</i> .
HTTP 500 INTERNAL SERVER CODE	No license active for {0} export type.	8029	Contact the system administrator to activate the appropriate license.
HTTP 400 BAD REQUEST	Invalid neType {0}	8031	Specify a valid neType.
HTTP 400 BAD REQUEST	Specified Collection {0}, does not exist or you have insufficient access rights. Exception message: {1}	8032	Specify a valid Collection.



HTTP Status Code	Description	Internal error code	Solution/Action required
HTTP 400 BAD REQUEST	Specified Saved Search {0}, does not exist or you have insufficient access rights. Exception message: {1}	8033	Specify a valid Saved Search.
HTTP 503 SERVICE UNAVAILABLE	Topology Search Service is not available. Exception message: {0}	8034	Try again later.
HTTP 400 BAD REQUEST	Unsupported operation error: {0}	8035	Check the online documentation.
HTTP 500 INTERNAL SERVER CODE	N/A	9999	This is an unhandled system error. Please check the error log for more details.

Note: {0}, {1}, and the following, indicate variable parts of the message that are substituted by relevant information from the request, such as FDN, MO type, attribute name, and so on. {2} indicates a variable part of the message that will be substituted by relevant information on user-defined filter error. {3} indicates a variable suggested solution/action that must be performed to correct user-defined filter.

8.5 Error Response Codes for Import

8.5.1 REST Error Response Codes for Import

The following is a minimum set of response codes:

HTTP Status Code	Description	Internal error code	Solution/Action required
HTTP 400 BAD REQUEST	The supplied file format {0} is not supported for import.	7006	specify a supported file format.
HTTP 400 BAD REQUEST	Unable to retrieve job details for job id: {0,number,#}.	7008	check the job id.
HTTP 400 BAD REQUEST	The following mandatory argument(s) are required: {0}	1018	Provide the mandatory argument(s): {0}
HTTP 400 BAD REQUEST	The following Type EXPORT is invalid for the Import Service	-3	Provide the right Type for the Import Service.
HTTP 500 INTERNAL SERVER ERROR	N/A	-2	This is an unhandled system error, check the error log for more details.
HTTP 500 INTERNAL SERVER ERROR	The server was unable to process the request due to ongoing upgrade activity.	7009	issue the import command again.
HTTP 400 BAD REQUEST	The supplied modifier \"{0}\" is not a valid import operation.	7010	specify 'create','update' or 'delete' as the modifier.
HTTP 500 INTERNAL SERVER ERROR	The model version for the managed object \"{0}\" has changed during the import operation.	7011	issue the import command again.
HTTP 400 BAD REQUEST	Incorrect attribute \"{0}\" supplied for the specified managed object \"{1}\"	7012	check the model for allowed attributes
HTTP 400 BAD REQUEST	{0}	7014	fix the issue and re-execute the import command.



HTTP Status Code	Description	Internal error code	Solution/Action required
HTTP 500 INTERNAL SERVER ERROR	N/A	9999	This is an unhandled system error, check the error log for more details.
HTTP 400 BAD REQUEST	There is an xml formatting error in the import file.	7001	consult with online help to ensure the file is well formed.
HTTP 400 BAD REQUEST	The specified file does not consist of data and is invalid.	7002	ensure the specified file name is correct.
HTTP 400 BAD REQUEST	The specified file {0} was not found.	7003	ensure the specified file name is correct.
HTTP 400 BAD REQUEST	Error trying to read zip file.	7004	ensure the zip file name specified is correct, and that the zip file contains relevant import data.
HTTP 400 BAD REQUEST	The specified file is of an unsupported 3GPP file format version.	7005	Check the 3GPP file format version in the file header.
HTTP 500 INTERNAL SERVER ERROR	{0}	7014	Fix the issue and re-execute the import command.
HTTP 400 BAD REQUEST	The supplied FDN \"{0}\" does not exist in the configuration with name \"{1}\".	1001	Specify an existing FDN. It may be necessary to first create the desired FDN.
HTTP 400 BAD REQUEST	Type {0} not found.	1012	Check the model.
HTTP 400 BAD REQUEST	The configuration {0} does not exist.	1015	Create configuration {0} first, or correct the name.
HTTP 400 BAD REQUEST	Cannot update value of read-only attribute \"{0}\" for MO type \"{1}\". /	1021	Check the model for attribute write behaviour.
HTTP 403 FORBIDDEN	Insufficient access rights to perform the operation	5001	If it is necessary that the operation is performed by the current user, the user profile must be updated by the system administrator.

Note: {0}, {1}, and the following, indicate variable parts of the message that are substituted by relevant information from the request, such as FDN, MO type, and attribute name. If error 7014 occurs, {0} is substituted by an error message that is sent from the data persistence service and caught in the system.



9 Appendix A - CM REST API Documentation

Refer to CM Bulk Import Export and CM REST Northbound Interface API documentation in Supplementary Documentation for CM Bulk Import Export and CM REST Northbound Interface, 190 89-LZN 708 0882.



10 Appendix B - CM REST Service Specific API Documentation

Refer to CM Bulk Import Export and CM REST Northbound Interface API documentation in Supplementary Documentation for CM Bulk Import Export and CM REST Northbound Interface, 190 89-LZN 708 0882.



11 Appendix C - Bulk Import Validation Phases

The model validation phase will perform the following checks:

Checks at Managed Object Class (MOC) level:

- The MOC exists in the model
- For create or delete operations, the MOC is not system created
- For deletion of a child MOC, the relationship between parent and child is not marked 'not deletable'

Checks at attribute level for create or modify operations:

- The attribute exists in the model (for the specified MOC)
- The attribute is read only or restricted
- The attribute is of the correct type
- For numeric and boolean data types, the attribute is within the allowed value range or set of values
- For union and bits data types, the attribute is within the allowed set of values
- For strings, the attribute is within the allowed length range
- For enumerations, the attribute has a valid value from a predefined allowed set of values (This can be either a string or the corresponding integer value)
- For create operation, all the mandatory attributes are provided
- Members of complex attributes (structs and lists) are also subject to the same checks as simple attributes (types, ranges etc.)

Checks at Managed Object Class (MOC) level for action operations:

- The action exists in the model
- The action has a void return type
- All mandatory arguments are provided
- Provided arguments are supported and have valid data type

In some cases it is not possible to perform the complete set of model validation checks on a MOC or attribute. For example, if the MOC does not exist in the model, no further checking of the MOC or its attributes is possible. If an attribute has an incorrect data type it is not possible to check its value range etc.



If enabled, the instance validation phase will perform additional checks to ensure that import operations are valid against target configuration. It also checks import operation validity against previous import operations in the import file. This validation helps in reducing the risk of undesired updates being applied to the network. Instance validation is done for the below operation types.

CREATE:

The following checks are performed to verify the create operation.

- MO was neither created by the import file nor existent in the target configuration
- Any Parent MO was not deleted by the import file and exists in target configuration
- Cumulative number of MOs of same type under the Parent MO in DB and Import file should not exceed maximum cardinality constraint (if applicable)

DELETE:

The following checks are performed to verify the delete operation.

- MO was either created by the import file or existent in the target configuration
- MO was not deleted by the import file
- Any Parent MO was not deleted by the import file
- Cumulative number of MOs of same type under the Parent MO in DB and Import file should not fall below minimum cardinality constraint (if applicable)

UPDATE:

The following checks are performed to verify the update operation.

- MO was either created by the import file or existent in the target configuration
- MO was not deleted by the import file
- Any Parent MO was not deleted by the import file

ACTION:

The following checks are performed to verify the action operation.

- MO was either created by the import file or existent in the target configuration
- MO was not deleted by the import file



- Any Parent MO was not deleted by the import file

Note: Instance validation checks are not exhaustive and import execution failures are possible.



Reference List

- [1] *CM Bulk Import REST Northbound Interface*, 2/155 19-CNA 403 3141
- [2] *TCIM Model Description*, 3/15519-CNA 403 3141