

API for Product Modeling v2

(Administration)

Overview

An API to post, put, get and delete [product modeling](#) data.

API General Info

Response Content Type

It is possible to switch between `json` and `xml` response types by setting an `Accept` header to either `application/json` or `application/xml` respectively.

Successful Responses

Response Codes

- **200** - The request was executed successfully and the server returned some data. The return content type is specified inside a response header called `content-type`. Example content types are `application/xml`, `application/json`, `application/zip` and `text/plain`.
- **204** - The request was executed successfully and the server returned no data. If the `Location` header was present then the server responded with a redirect.

Error Responses

If the API call resulted in an error then the error code will be returned along with the error object in the response body. The error object contains the attributes listed below:

- **type** - type of the error
- **cause** - the root cause of the error
- **message** - message containing details of the related error
- **uuid** - a unique identifier of the error which also may appear in the exception logs
- **timestamp** - the time of the error

Note: For chunked responses, the socket will be closed prematurely if an error occurs after the initial chunk containing the status code has been sent.

Response Codes

- **400** - The request contains invalid parameters or there are invalid settings. Consult the error message and/or check Administration for invalid settings.
- **401** - The user is not authenticated.
- **403** - The user is not authorized access to the specified endpoint.
- **404** - The specified endpoint does not exist or a request parameter is invalid. Consult the error message.
- **405** - The specified endpoint does not support the requested operation. Consult the error message.
- **409** - The specified endpoint is in a state incompatible with the requested operation. Consult the error message.
- **500** - Unknown error occurred while processing the request. Consult the CPQ logs for more information.
- **504** - The specified concurrent calls limit was exceeded and the server was not able to serve the response in a specified timeout limit. That error is always served with the xml body containing the description.

Endpoints by Category

Assembly

GET /api/product-modeling/v2/[ticketId]/assembly/list

Retrieve all assemblies defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
Result	GetAssemblyListOutput	A list of assembly resources

GET /api/product-modeling/v2/[ticketId]/assembly/[id]

Retrieve an assembly defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the assembly
Result	GetAssemblyOutput	An assembly resource

GET /api/product-modeling/v2/[ticketId]/assembly?name=[name]

Retrieve an assembly defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the assembly
<i>Result</i>	GetAssemblyOutput	An assembly resource

POST /api/product-modeling/v2/[ticketId]/assembly?input=[input]

Create or replace an existing assembly in a ticket. Note that when creating combination tables, you need to do 2 calls. The first call creates the columns so you can use the column id in the second call defining the rows.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostAssemblyInput	The assembly to create or replace the existing assembly with
<i>Result</i>	PostAssemblyOutput	A named reference to the assembly

POST /api/product-modeling/v2/[ticketId]/assembly/list?input=[input]

Create, replace or delete assemblies defined in a ticket. Posted assemblies not already present will be created, posted existing assemblies will be replaced keeping their original ids and not posted existing assemblies will optionally be deleted.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostAssemblyListInput	The list of assemblies to replace existing assemblies with
<i>Result</i>	PostAssemblyListOutput	A list of named assembly references

PUT /api/product-modeling/v2/[ticketId]/assembly/[id]?input=[input]

Replace an existing assembly defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the target assembly resource
input	PutAssemblyInput	The assembly to replace the existing assembly with
<i>Result</i>	PutAssemblyOutput	A named reference to the assembly

DELETE /api/product-modeling/v2/[ticketId]/assembly/list?input=[input]

Delete assembly defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	DeleteAssemblyListInput	A list of named assembly references
<i>Result</i>	DeleteAssemblyListOutput	A list of named assembly references

DELETE /api/product-modeling/v2/[ticketId]/assembly/[id]

Delete an assembly defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the assembly
<i>Result</i>	DeleteAssemblyOutput	A named assembly reference

DELETE /api/product-modeling/v2/[ticketId]/assembly?name=[name]

Delete an assembly defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket

Parameter	Type	Description
name	String	The name of the assembly
<i>Result</i>	DeleteAssemblyOutput	A named assembly reference

Attribute Category

GET /api/product-modeling/v2/[ticketId]/attribute-category/list

Retrieve all attribute categories defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
<i>Result</i>	GetAttributeCategoryListOutput	A list of attribute category resources

GET /api/product-modeling/v2/[ticketId]/attribute-category/[id]

Retrieve an attribute category defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the attribute category
<i>Result</i>	GetAttributeCategoryOutput	An attribute category resource

GET /api/product-modeling/v2/[ticketId]/attribute-category?name=[name]

Retrieve an attribute category defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the attribute category

Parameter	Type	Description
<i>Result</i>	GetAttributeCategoryOutput	An attribute category resource

POST /api/product-modeling/v2/[ticketId]/attribute-category?input=[input]

Create or replace an existing attribute category in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostAttributeCategoryInput	The attributeCategory to create or replace the existing attribute category with
<i>Result</i>	PostAttributeCategoryOutput	A named reference to the attribute category

POST /api/product-modeling/v2/[ticketId]/attribute-category/list?input=[input]

Create, replace or delete attribute categories defined in a ticket. Posted attribute categories not already present will be created, posted existing attribute categories will be replaced keeping their original ids and not posted existing attribute categories will optionally be deleted.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostAttributeCategoryListInput	The list of attribute categories to replace existing attribute categories with
<i>Result</i>	PostAttributeCategoryListOutput	A list of named attribute category references

PUT /api/product-modeling/v2/[ticketId]/attribute-category/[id]?input=[input]

Replace an existing attribute category defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the target attributeCategory resource
input	PutAttributeCategoryInput	The attributeCategory to replace the existing attribute category with
<i>Result</i>	PutAttributeCategoryOutput	A named reference to the attribute category

DELETE /api/product-modeling/v2/[ticketId]/attribute-category/list?input=[input]

Delete attribute categories defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	DeleteAttributeCategoryListInput	A list of named attribute category references
<i>Result</i>	DeleteAttributeCategoryListOutput	A list of named attribute category references

DELETE /api/product-modeling/v2/[ticketId]/attribute-category/[id]

Delete an attribute category defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the attribute category

Parameter	Type	Description
<i>Result</i>	DeleteAttributeCategoryOutput	A named attribute category reference

DELETE /api/product-modeling/v2/[ticketId]/attribute-category?name=[name]

Delete an attribute category defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the attribute category
<i>Result</i>	DeleteAttributeCategoryOutput	A named attribute category reference

BoM Meta Data

GET /api/product-modeling/v2/[ticketId]/bom-meta-data/list

Retrieve all BoM Meta Data properties defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
<i>Result</i>	GetBomMetaDataPropertyListOutput	A list of property resources

GET /api/product-modeling/v2/[ticketId]/bom-meta-data/[id]

Retrieve a BoM Meta Data property defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket

Parameter	Type	Description
id	String	The identity of the property
<i>Result</i>	GetBomMetaDataPropertyOutput	A property resource

GET /api/product-modeling/v2/[ticketId]/bom-meta-data?name=[name]

Retrieve a BoM Meta Data property defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the property
<i>Result</i>	GetBomMetaDataPropertyOutput	A property resource

POST /api/product-modeling/v2/[ticketId]/bom-meta-data?input=[input]

Create or replace an existing BoM Meta Data property in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostBomMetaDataPropertyInput	The property to create or replace the existing property with
<i>Result</i>	PostBomMetaDataPropertyOutput	A named reference to the property

POST /api/product-modeling/v2/[ticketId]/bom-meta-data/list?input=[input]

Create, replace or delete BoM Meta Data properties defined in a ticket. Posted properties not already present will be created, posted existing properties will be replaced keeping their original IDs and not posted existing properties will optionally be deleted.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostBomMetaDataPropertyListInput	The list of properties to replace existing properties with
<i>Result</i>	PostBomMetaDataPropertyListOutput	A list of named property references

PUT /api/product-modeling/v2/[ticketId]/bom-meta-data/[id]?input=[input]

Replace an existing BoM Meta Data property defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the target property resource
input	PutBomMetaDataPropertyInput	The property to replace the existing property with
<i>Result</i>	PutBomMetaDataPropertyOutput	A named reference to the property

DELETE /api/product-modeling/v2/[ticketId]/bom-meta-data/list?input=[input]

Delete BoM Meta Data properties defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket

Parameter	Type	Description
input	DeleteBomMetaDataPropertyListInput	A list of named property references
<i>Result</i>	DeleteBomMetaDataPropertyListOutput	A list of named property references

DELETE /api/product-modeling/v2/[ticketId]/bom-meta-data/[id]

Delete a BoM Meta Data property defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the property
<i>Result</i>	DeleteBomMetaDataPropertyOutput	A named property reference

DELETE /api/product-modeling/v2/[ticketId]/bom-meta-data?name=[name]

Delete a BoM Meta Data property defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the property
<i>Result</i>	DeleteBomMetaDataPropertyOutput	A named property reference

Domain

GET /api/product-modeling/v2/[ticketId]/domain/list

Retrieve all domains defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
<i>Result</i>	GetDomainListOutput	A list of domain resources

GET /api/product-modeling/v2/[ticketId]/domain/[id]

Retrieve a domain defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the domain
<i>Result</i>	GetDomainOutput	A domain resource

GET /api/product-modeling/v2/[ticketId]/domain?name=[name]

Retrieve a domain defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the domain
<i>Result</i>	GetDomainOutput	A domain resource

POST /api/product-modeling/v2/[ticketId]/domain?input=[input]

Create or replace an existing domain in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostDomainInput	The domain to create or replace the existing domain with
<i>Result</i>	PostDomainOutput	A named reference to the domain

POST /api/product-modeling/v2/[ticketId]/domain/list?input=[input]

Create, replace or delete domains defined in a ticket. Posted domains not already present will be created, posted existing domains will be replaced keeping their original ids and not posted existing domains will optionally be deleted.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostDomainListInput	The list of domains to replace existing domains with
<i>Result</i>	PostDomainListOutput	A list of named domain references

PUT /api/product-modeling/v2/[ticketId]/domain/[id]?input=[input]

Replace an existing domain defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the target domain resource
input	PutDomainInput	The domain to replace the existing domain with
<i>Result</i>	PutDomainOutput	A named reference to the domain

DELETE /api/product-modeling/v2/[ticketId]/domain/list?input=[input]

Delete domains defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	DeleteDomainListInput	A list of named domain references
<i>Result</i>	DeleteDomainListOutput	A list of named domain references

DELETE /api/product-modeling/v2/[ticketId]/domain/[id]

Delete a domain defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the domain
<i>Result</i>	DeleteDomainOutput	A named domain reference

DELETE /api/product-modeling/v2/[ticketId]/domain?name=[name]

Delete a domain defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the domain
<i>Result</i>	DeleteDomainOutput	A named domain reference

Global Feature

GET /api/product-modeling/v2/[ticketId]/global-feature/list

Retrieve all global features defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
<i>Result</i>	GetGlobalFeatureListOutput	A list of global feature resources

GET /api/product-modeling/v2/[ticketId]/global-feature/[id]

Retrieve a global feature defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the feature
<i>Result</i>	GetGlobalFeatureOutput	A feature resource

GET /api/product-modeling/v2/[ticketId]/global-feature?name=[name]

Retrieve a global feature defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the feature
<i>Result</i>	GetGlobalFeatureOutput	A feature resource

POST /api/product-modeling/v2/[ticketId]/global-feature?input=[input]

Create or replace an existing global feature in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostGlobalFeatureInput	The feature to create or replace the existing feature with
<i>Result</i>	PostGlobalFeatureOutput	A named reference to the feature

POST /api/product-modeling/v2/[ticketId]/global-feature/list?input=[input]

Create, replace or delete global features defined in a ticket. Posted global features not already present will be created, posted existing global features will be replaced keeping their original ids and not posted existing global features will optionally be deleted.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostGlobalFeatureListInput	The list of global features to replace existing global features with
<i>Result</i>	PostGlobalFeatureListOutput	A list of named global feature references

PUT /api/product-modeling/v2/[ticketId]/global-feature/[id]?input=[input]

Replace an existing global feature defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the target feature resource
input	PutGlobalFeatureInput	The feature to replace the existing feature with
<i>Result</i>	PutGlobalFeatureOutput	A named reference to the feature

DELETE /api/product-modeling/v2/[ticketId]/global-feature/list?input=[input]

Delete global features defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	DeleteGlobalFeatureListInput	A list of named feature references
<i>Result</i>	DeleteGlobalFeatureListOutput	A list of named feature references

DELETE /api/product-modeling/v2/[ticketId]/global-feature/[id]

Delete a global feature defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the feature
<i>Result</i>	DeleteGlobalFeatureOutput	A named feature reference

DELETE /api/product-modeling/v2/[ticketId]/global-feature?name=[name]

Delete a global feature defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the feature
<i>Result</i>	DeleteGlobalFeatureOutput	A named feature reference

Module

Variables in BoM Meta Data

BoM meta data of type *Rich Text* support inserting [variables](#) into the meta data property value.

Variable is inserted to the body of the property value using a dedicated tag `<variable>`. The full syntax is:

```
<variable name="featureName">featureName</variable>
```

where `featureName` is a name of a module feature (either global or in scope of the given module).

Note

The text placed between the opening and closing tag will effectively be replaced with the feature name when processing the input.

The compact syntax (empty-element tag) is also supported:

```
<variable name="featureName"/>
```

Note

The compact representation of the variable will effectively be replaced with the full syntax when processing the input.

Example

Sample value of BoM meta data property value that uses a variable referring to the `arm_length` feature:

```
<p>The length of the arm is <variable name="arm_length">arm_length</
```

Note

If `arm_length` is not a valid module feature name (either existing one or declared within the same input), the input will not be accepted.

Images in BoM Meta Data

BoM Meta Data of type *Rich Text* support also inserting images into the Meta Data property value. The image must have been previously stored in another Meta Data property.

It can be inserted to the body of the property value like a variable.

Image uses a dedicated tag `<metaimg>`, which has attributes:

metaimg attribute	Mandatory	Description
data-metaimg-prop-id	yes	The identifier of the Meta Data property
data-metaimg-prop-display-id	yes	The text displayed on the image placeholder
data-metaimg-align	no	The horizontal alignment of the image. Can be "center" (default), "right" or "left".
data-metaimg-size-type	no	The method used to describe the size of the image. Can be "default", "scale" or "custom".
data-metaimg-	no	The parametrisation of image size. Can be:

metaimg attribute	Mandatory	Description
size-value		- an integer between 0 and 100 in the case of "scale" type, - a string containing a json in the case of "custom" type. The json must have "width" and "height" attributes, for example: <code>{"width": 150, "height": 100}</code> .

GET /api/product-modeling/v2/[ticketId]/module/list

Retrieve all modules defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
<i>Result</i>	GetModuleListOutput	A list of module resources

GET /api/product-modeling/v2/[ticketId]/module/[id]

Retrieve a module defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the module
<i>Result</i>	GetModuleOutput	A module resource

GET /api/product-modeling/v2/[ticketId]/module?name=[name]

Retrieve a module defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the module
<i>Result</i>	GetModuleOutput	A module resource

POST /api/product-modeling/v2/[ticketId]/module?input=[input]

Create or replace an existing module in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostModuleInput	The module to create or replace the existing module with
<i>Result</i>	PostModuleOutput	A named reference to the module

POST /api/product-modeling/v2/[ticketId]/module/list?input=[input]

Create, replace or delete modules defined in a ticket. Posted modules not already present will be created, posted existing modules will be replaced keeping their original ids and not posted existing modules will optionally be deleted.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostModuleListInput	The list of modules to replace existing modules with
<i>Result</i>	PostModuleListOutput	A list of named module references

PUT /api/product-modeling/v2/[ticketId]/module/[id]?input=[input]

Replace an existing module defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the target module resource
input	PutModuleInput	The module to replace the existing module with
<i>Result</i>	PutModuleOutput	A named reference to the module

DELETE /api/product-modeling/v2/[ticketId]/module/list?input=[input]

Delete modules defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	DeleteModuleListInput	A list of named module references
<i>Result</i>	DeleteModuleListOutput	A list of named module references

DELETE /api/product-modeling/v2/[ticketId]/module/[id]

Delete a module defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the module
<i>Result</i>	DeleteModuleOutput	A named module reference

DELETE /api/product-modeling/v2/[ticketId]/module?name=[name]

Delete a module defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the module
<i>Result</i>	DeleteModuleOutput	A named module reference

Module Variant Images

GET /api/product-modeling/v2/[ticketId]/module/[moduleId]/variant/[variantId]/image/list

List BoM Meta Data properties with links to downloadable images

Parameter	Type	Description
ticketId	String	The identity of the ticket
moduleId	String	The identity of the module
variantId	String	The identity of the variant
<i>Result</i>	GetVariantImagePropertyListOutput	List of variant image URLs, either relative (for uploaded images) or absolute (for URL-based images)

POST /api/product-modeling/v2/[ticketId]/module/[moduleId]/variant/[variantId]/image/[propertyId]?file=[file]&locale=[locale]

Upload an image to BoM Meta Data property

Parameter	Type	Description
ticketId	String	The identity of the ticket
moduleId	String	The identity of the module
variantId	String	The identity of the variant
propertyId	String	The identity of the Meta Data property
file	Binary	The file
locale	String (Optional)	Optional
<i>Result</i>	String	The relative URL to the image

GET /api/product-modeling/v2/[ticketId]/module/[moduleId]/variant/[variantId]/image/[propertyId]?locale=[locale]

Download an image from BoM Meta Data property

Parameter	Type	Description
ticketId	String	The identity of the ticket
moduleId	String	The identity of the module
variantId	String	The identity of the variant
propertyId	String	The identity of the Meta Data property
locale	String (Optional)	Optional
<i>Result</i>	Binary	The image

DELETE /api/product-modeling/v2/[ticketId]/module/[moduleId]/variant/[variantId]/image/[propertyId]?locale=[locale]

Delete an image from BoM Meta Data property

Parameter	Type	Description
ticketId	String	The identity of the ticket
moduleId	String	The identity of the module
variantId	String	The identity of the variant
propertyId	String	The identity of the Meta Data property
locale	String (Optional)	Optional
<i>Result</i>	Empty	

Part

GET /api/product-modeling/v2/[ticketId]/part/list

Retrieve all parts defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
<i>Result</i>	GetPartListOutput	A list of part resources

GET /api/product-modeling/v2/[ticketId]/part/[id]

Retrieve a part defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the feature
<i>Result</i>	GetPartOutput	A feature resource

GET /api/product-modeling/v2/[ticketId]/part?name=[name]

Retrieve a part defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the feature
<i>Result</i>	GetPartOutput	A feature resource

POST /api/product-modeling/v2/[ticketId]/part?input=[input]

Create or replace an existing part in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostPartInput	The feature to create or replace the existing feature with
<i>Result</i>	PostPartOutput	A named reference to the feature

POST /api/product-modeling/v2/[ticketId]/part/list?input=[input]

Create, replace or delete parts defined in a ticket. Posted parts not already present will be created, posted existing parts will be replaced keeping their original ids and not posted existing parts will optionally be deleted.

Parameter	Type	Description
ticketId	String	The identity of the ticket

Parameter	Type	Description
input	PostPartListInput	The list of parts to replace existing parts with
<i>Result</i>	PostPartListOutput	A list of named part references

PUT /api/product-modeling/v2/[ticketId]/part/[id]?input=[input]

Replace an existing part defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the target feature resource
input	PutPartInput	The feature to replace the existing feature with
<i>Result</i>	PutPartOutput	A named reference to the feature

DELETE /api/product-modeling/v2/[ticketId]/part/list?input=[input]

Delete parts defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	DeletePartListInput	A list of named feature references
<i>Result</i>	DeletePartListOutput	A list of named feature references

DELETE /api/product-modeling/v2/[ticketId]/part/[id]

Delete a part defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the feature

Parameter	Type	Description
<i>Result</i>	DeletePartOutput	A named feature reference

DELETE /api/product-modeling/v2/[ticketId]/part?name=[name]

Delete a part defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the feature
<i>Result</i>	DeletePartOutput	A named feature reference

Part Feature

GET /api/product-modeling/v2/[ticketId]/part-feature/list

Retrieve all part features defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
<i>Result</i>	GetPartFeatureListOutput	A list of part feature resources

GET /api/product-modeling/v2/[ticketId]/part-feature/[id]

Retrieve a part feature defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the feature
<i>Result</i>	GetPartFeatureOutput	A feature resource

GET /api/product-modeling/v2/[ticketId]/part-feature?name=[name]

Retrieve a part feature defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
name	String	The name of the feature
<i>Result</i>	GetPartFeatureOutput	A feature resource

POST /api/product-modeling/v2/[ticketId]/part-feature?input=[input]

Create or replace an existing part feature in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostPartFeatureInput	The feature to create or replace the existing feature with
<i>Result</i>	PostPartFeatureOutput	A named reference to the feature

POST /api/product-modeling/v2/[ticketId]/part-feature/list?input=[input]

Create, replace or delete part features defined in a ticket. Posted part features not already present will be created, posted existing part features will be replaced keeping their original ids and not posted existing part features will optionally be deleted.

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	PostPartFeatureListInput	The list of part features to replace existing part features with
<i>Result</i>	PostPartFeatureListOutput	A list of named part feature references

PUT /api/product-modeling/v2/[ticketId]/part-feature/[id]?input=[input]

Replace an existing part feature defined in a ticket.

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the target feature resource
input	PutPartFeatureInput	The feature to replace the existing feature with
<i>Result</i>	PutPartFeatureOutput	A named reference to the feature

DELETE /api/product-modeling/v2/[ticketId]/part-feature/list?input=[input]

Delete part features defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
input	DeletePartFeatureListInput	A list of named feature references
<i>Result</i>	DeletePartFeatureListOutput	A list of named feature references

DELETE /api/product-modeling/v2/[ticketId]/part-feature/[id]

Delete a part feature defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket
id	String	The identity of the feature
<i>Result</i>	DeletePartFeatureOutput	A named feature reference

DELETE /api/product-modeling/v2/[ticketId]/part-feature?name=[name]

Delete a part feature defined in a ticket

Parameter	Type	Description
ticketId	String	The identity of the ticket

Parameter	Type	Description
name	String	The name of the feature
<i>Result</i>	DeletePartFeatureOutput	A named feature reference

Data Types by Category

Assembly

DeleteAssemblyListInput

A delete assembly list input.

```
JSON XML
{
  "assemblyNamedReferenceList" : [
    {
      "id" : "", // The assembly resource id.
      "name" : "" // The assembly name.
    },
    ...
  ] // The assembly named reference list.
}
```

DeleteAssemblyListOutput

A delete assembly output.

```
JSON XML
{
  "assemblyNamedReferenceList" : [
    {
      "id" : "", // The assembly resource id.
      "name" : "" // The assembly name.
    },
    ...
  ] // The assembly named reference list.
}
```

DeleteAssemblyOutput

A delete assembly output.

JSON XML

```
{
  "assemblyNamedReference" : {
    "id" : "", // The assembly resource id.
    "name" : "" // The assembly name.
  } // The assembly named reference.
}
```

GetAssemblyListOutput

A get assembly resource list output. Note that each of the related items is listed separately with a parent reference pointing to the assembly.

```

{
  "assemblyResourceList" : [
    {
      "assemblyReference" : {
        "id" : "" // The assembly resource id.
      }, // The assembly reference.
      "assembly" : {
        "name" : "", // The assembly name.
        "description" : "", // (Optional) The assembly description.
        "descriptionTranslations" : {
          "key" : ""
        }, // (Optional) The assembly description translations by language.
        "consistencyCheckStrategy" : "", // (Optional) The assembly consistency check strategy.
        "variantEnabled" : false, // (Optional) Whether assembly variant is enabled.
        "virtualVariant" : {
          "name" : "",
          "description" : "",
          "image" : "",
          "document" : "",
          "longDescription" : "",
          "variantValueList" : [
            {
              "attributeNamedReference" : {
                "id" : "", // The attribute resource id.
                "name" : "" // The attribute name.
              }, // The variant assembly attribute named reference.
              "featureNamedReference" : {
                "id" : "", // The feature resource id.
                "name" : "" // The feature name.
              }, // The variant feature named reference. Required for assembly variant.
              "value" : "" // The variant value. Multiple values must be provided for assembly variant.
            },
            ...
          ]
        } // (Optional) Values of the assembly virtual variant. When default value of field 'name' is set and other fields are left unassigned.
      } // The assembly.
    },
    ...
  ], // The assembly resource list.
  "attributeResourceList" : [
    {
      "attributeReference" : {
        "id" : "" // The attribute resource id.
      }, // The attribute reference.
      "attribute" : {

```

```

"parentAssemblyNamedReference" : {
  "id" : "", // The parent assembly resource id.
  "name" : "" // The parent assembly name.
}, // The parent assembly.
"attributeCategoryNamedReference" : {
  "id" : "", // The attribute category resource id.
  "name" : "" // The attribute category name.
}, // The attribute category.
"domainNamedReference" : {
  "id" : "", // The domain resource id.
  "name" : "" // The domain name.
}, // The attribute domain.
"name" : "", // The attribute name.
"description" : "", // (Optional) The attribute description.
"descriptionTranslations" : {
  "key" : ""
}, // (Optional) The attribute description translations by l
"io" : false, // (Optional) Whether it is an I/O attribute.
"defaultView" : false, // (Optional) Whether the attribute i
"aggregationStrategy" : "", // (Optional) The aggregation fu
"aggregateList" : [
  {
    "attributeNamedReference" : {
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    }, // The attribute named reference.
    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The feature named reference.
    "positionNamedReference" : {
      "id" : "", // The position resource id.
      "name" : "" // The position name.
    } // The position named reference.
  },
  ...
] // (Optional) The list of aggregated attributes and featur
} // The attribute.
},
...
], // The assembly attribute resource list for listed assemblies.
"positionResourceList" : [
  {
    "positionReference" : {
      "id" : "" // The position resource id.
    }, // The position reference.
    "position" : {
      "parentAssemblyNamedReference" : {

```

```

        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
    }, // The parent assembly.
    "assemblyNamedReference" : {
        "id" : "", // The assembly resource id.
        "name" : "" // The assembly name.
    }, // (Optional) The assembly at the position.
    "moduleNamedReference" : {
        "id" : "", // The module resource id.
        "name" : "" // The module name.
    }, // (Optional) The module at the position.
    "name" : "", // The position name.
    "type" : "", // The position type: "Assembly", "Module", "Op
    "description" : "", // (Optional) The position description.
    "descriptionTranslations" : {
        "key" : ""
    }, // (Optional) The position description translations by la
    "defaultView" : false, // (Optional) Whether the position is
    "dynamic" : false, // (Optional) Whether the position is dyn
If "false", the number of instances is "Static".
If "true", the number of instances can be either "Sizeable" or "Conf
    "numberOfInstancesAttributeReference" : {
        "id" : "", // The attribute resource id.
        "name" : "" // The attribute name.
    }, // (Optional) The attribute that defines number of instan
If this is set, and dynamic is "true", then number of instances is "
    "staticNumberOfInstances" : 1, // (Optional) Number of insta
If not specified, the default value is 1.
It is undefined for dynamic positions.
    "enabled" : false, // (Optional) Whether the position is ena
    "qtyType" : "", // The position qty type. Allows values: "St
    "qty" : 1, // (Optional) The position qty. Used when qtyType
    "qtyMin" : 1, // (Optional) The position minimum qty. Used w
    "qtyMax" : 1 // (Optional) The position maximum qty. Used wh
    } // The position.
    },
    ...
], // The assembly position resource list for listed assemblies.
"variantResourceList" : [
    {
        "variantReference" : {
            "id" : "" // The variant resource id.
        }, // The variant reference.
        "variant" : {
            "parentAssemblyNamedReference" : {
                "id" : "", // The parent assembly resource id.
                "name" : "" // The parent assembly name.
            }, // (Optional) The parent assembly. Required for assembly

```

```

"parentModuleNamedReference" : {
  "id" : "", // The parent module resource id.
  "name" : "" // The parent module name.
}, // (Optional) The parent module. Required for module vari
"name" : "", // The variant name.
"description" : "", // (Optional) The variant description.
"descriptionTranslations" : {
  "key" : ""
}, // (Optional) The variant description translations by lan
"status" : "", // The variant status.
"longDescription" : "", // (Optional) The variant long descr
"longDescriptionTranslations" : {
  "key" : ""
}, // (Optional) The variant long description translations b
"image" : "", // (Optional) The variant image.
"document" : "", // (Optional) The variant document.
"variantValueList" : [
  {
    "attributeNamedReference" : {
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    }, // The variant assembly attribute named reference. Re
    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The variant feature named reference. Required for
    "value" : "" // The variant value. Multiple values must
  },
  ...
], // (Optional) The variant value list. The list should con
"variantPropertyValueList" : [
  {
    "propertyNamedReference" : {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    }, // The property reference
    "value" : "", // The property value.
    "valueTranslations" : {
      "key" : ""
    } // (Optional) The property value translations by langu
  },
  ...
], // (Optional) The variant BoM Meta Data property values 1
"variantPartList" : [
  {
    "partNamedReference" : {
      "id" : "", // The part resource id. Note: id and name
      "name" : "" // The part name.
    }
  }
]

```

```

        }, // The part named reference.
        "qty" : 1.0 // The part quantity.
    },
    ...
    ] // (Optional) The variant part list.
} // The variant.
},
...
], // The variant resource list for the listed assemblies.
"ruleResourceList" : [
{
    "rule" : {
        "assemblyNamedReference" : {
            "id" : "", // The assembly resource id.
            "name" : "" // The assembly name.
        }, // The assembly for the rule.
        "id" : "", // The rule id.
        "type" : "", // The rule type. Allows values: "Constraint" c
        "ruleGroup" : "", // (Optional) Category of rule. Default va
        "enabled" : false, // (Optional) If the rule is enabled or n
        "combinationRuleColumnList" : [
            {
                "id" : "", // The combination rule column id.
                "value" : "", // The combination rule column value.
                "type" : "", // (Optional) The type of the combination r
                "attributeNamedReference" : {
                    "id" : "", // The attribute resource id.
                    "name" : "" // The attribute name.
                }, // The combination rule column attribute named refere
                "featureNamedReference" : {
                    "id" : "", // The feature resource id.
                    "name" : "" // The feature name.
                }, // The combination rule column feature named referenc
                "moduleNamedReference" : {
                    "id" : "", // The module resource id.
                    "name" : "" // The module name.
                }, // The combination rule column module named reference
                "assemblyNamedReference" : {
                    "id" : "", // The assembly resource id.
                    "name" : "" // The assembly name.
                }, // The combination rule column assembly named referen
                "positionNamedReference" : {
                    "id" : "", // The position resource id.
                    "name" : "" // The position name.
                }, // The combination rule column position named referen
                "variantNamedReference" : {
                    "id" : "", // The variant resource id.
                    "name" : "" // The variant name.
                }
            }
        ]
    }
}
]

```

```

        } // The combination rule column variant reference. This
    },
    ...
], // (Optional) Columns for type "Combination".
"combinationRuleRowList" : [
    {
        "id" : "", // The combination rule row id.
        "combinationRuleCellList" : [
            {
                "combinationRuleColumnReference" : {
                    "id" : "" // The combination rule column id.
                }, // The combination rule column reference.
                "value" : "" // The combination rule column cell val
            },
            ...
        ] // The combination rule row cells.
    },
    ...
], // (Optional) Rows for type "Combination".
"constraint" : "" // (Optional) The constraint expression for
} // The rule.
},
...
] // The rule resource list for the listed assemblies.
}

```

GetAssemblyOutput

A get assembly resource output.

```

{
  "assemblyResource" : {
    "assemblyReference" : {
      "id" : "" // The assembly resource id.
    }, // The assembly reference.
    "assembly" : {
      "name" : "", // The assembly name.
      "description" : "", // (Optional) The assembly description.
      "descriptionTranslations" : {
        "key" : ""
      }, // (Optional) The assembly description translations by lang
      "consistencyCheckStrategy" : "", // (Optional) The assembly cc
Supported values: Standard, Weaker. If not specified, then Standard
      "variantEnabled" : false, // (Optional) Whether assembly varia
      "virtualVariant" : {
        "name" : "",
        "description" : "",
        "image" : "",
        "document" : "",
        "longDescription" : "",
        "variantValueList" : [
          {
            "attributeNamedReference" : {
              "id" : "", // The attribute resource id.
              "name" : "" // The attribute name.
            }, // The variant assembly attribute named reference. Re
            "featureNamedReference" : {
              "id" : "", // The feature resource id.
              "name" : "" // The feature name.
            }, // The variant feature named reference. Required for
            "value" : "" // The variant value. Multiple values must
          },
          ...
        ]
      } // (Optional) Values of the assembly virtual variant. When n
default value of field 'name' is set and other fields are left unass
    } // The assembly.
  }, // The assembly resource.
  "positionResourceList" : [
    {
      "positionReference" : {
        "id" : "" // The position resource id.
      }, // The position reference.
      "position" : {
        "parentAssemblyNamedReference" : {
          "id" : "", // The parent assembly resource id.
          "name" : "" // The parent assembly name.
        }
      }
    }
  ]
}

```

```

    }, // The parent assembly.
    "assemblyNamedReference" : {
        "id" : "", // The assembly resource id.
        "name" : "" // The assembly name.
    }, // (Optional) The assembly at the position.
    "moduleNamedReference" : {
        "id" : "", // The module resource id.
        "name" : "" // The module name.
    }, // (Optional) The module at the position.
    "name" : "", // The position name.
    "type" : "", // The position type: "Assembly", "Module", "Op
    "description" : "", // (Optional) The position description.
    "descriptionTranslations" : {
        "key" : ""
    }, // (Optional) The position description translations by la
    "defaultView" : false, // (Optional) Whether the position is
    "dynamic" : false, // (Optional) Whether the position is dyn
If "false", the number of instances is "Static".
If "true", the number of instances can be either "Sizeable" or "Conf
    "numberOfInstancesAttributeReference" : {
        "id" : "", // The attribute resource id.
        "name" : "" // The attribute name.
    }, // (Optional) The attribute that defines number of instan
If this is set, and dynamic is "true", then number of instances is "
    "staticNumberOfInstances" : 1, // (Optional) Number of insta
If not specified, the default value is 1.
It is undefined for dynamic positions.
    "enabled" : false, // (Optional) Whether the position is ena
    "qtyType" : "", // The position qty type. Allows values: "St
    "qty" : 1, // (Optional) The position qty. Used when qtyType
    "qtyMin" : 1, // (Optional) The position minimum qty. Used w
    "qtyMax" : 1 // (Optional) The position maximum qty. Used wh
    } // The position.
    },
    ...
], // The position resource list.
"attributeResourceList" : [
    {
        "attributeReference" : {
            "id" : "" // The attribute resource id.
        }, // The attribute reference.
        "attribute" : {
            "parentAssemblyNamedReference" : {
                "id" : "", // The parent assembly resource id.
                "name" : "" // The parent assembly name.
            }, // The parent assembly.
            "attributeCategoryNamedReference" : {
                "id" : "", // The attribute category resource id.

```

```

    "name" : "" // The attribute category name.
  }, // The attribute category.
  "domainNamedReference" : {
    "id" : "", // The domain resource id.
    "name" : "" // The domain name.
  }, // The attribute domain.
  "name" : "", // The attribute name.
  "description" : "", // (Optional) The attribute description.
  "descriptionTranslations" : {
    "key" : ""
  }, // (Optional) The attribute description translations by l
  "io" : false, // (Optional) Whether it is an I/O attribute.
  "defaultView" : false, // (Optional) Whether the attribute i
  "aggregationStrategy" : "", // (Optional) The aggregation fu
  "aggregateList" : [
    {
      "attributeNamedReference" : {
        "id" : "", // The attribute resource id.
        "name" : "" // The attribute name.
      }, // The attribute named reference.
      "featureNamedReference" : {
        "id" : "", // The feature resource id.
        "name" : "" // The feature name.
      }, // The feature named reference.
      "positionNamedReference" : {
        "id" : "", // The position resource id.
        "name" : "" // The position name.
      } // The position named reference.
    },
    ...
  ] // (Optional) The list of aggregated attributes and featur
} // The attribute.
},
...
], // The attribute resource list.
"variantResourceList" : [
  {
    "variantReference" : {
      "id" : "" // The variant resource id.
    }, // The variant reference.
    "variant" : {
      "parentAssemblyNamedReference" : {
        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
      }, // (Optional) The parent assembly. Required for assembly
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }
    }
  }
]

```

```

}, // (Optional) The parent module. Required for module vari
"name" : "", // The variant name.
"description" : "", // (Optional) The variant description.
"descriptionTranslations" : {
  "key" : ""
}, // (Optional) The variant description translations by lan
"status" : "", // The variant status.
"longDescription" : "", // (Optional) The variant long descr
"longDescriptionTranslations" : {
  "key" : ""
}, // (Optional) The variant long description translations b
"image" : "", // (Optional) The variant image.
"document" : "", // (Optional) The variant document.
"variantValueList" : [
  {
    "attributeNamedReference" : {
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    }, // The variant assembly attribute named reference. Re
    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The variant feature named reference. Required for
    "value" : "" // The variant value. Multiple values must
  },
  ...
], // (Optional) The variant value list. The list should con
"variantPropertyValueList" : [
  {
    "propertyNamedReference" : {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    }, // The property reference
    "value" : "", // The property value.
    "valueTranslations" : {
      "key" : ""
    } // (Optional) The property value translations by langu
  },
  ...
], // (Optional) The variant BoM Meta Data property values 1
"variantPartList" : [
  {
    "partNamedReference" : {
      "id" : "", // The part resource id. Note: id and name
      "name" : "" // The part name.
    }, // The part named reference.
    "qty" : 1.0 // The part quantity.
  },

```

```

        ...
        ] // (Optional) The variant part list.
    } // The variant.
},
...
], // The variant resource list.
"ruleResourceList" : [
    {
        "rule" : {
            "assemblyNamedReference" : {
                "id" : "", // The assembly resource id.
                "name" : "" // The assembly name.
            }, // The assembly for the rule.
            "id" : "", // The rule id.
            "type" : "", // The rule type. Allows values: "Constraint" c
            "ruleGroup" : "", // (Optional) Category of rule. Default va
            "enabled" : false, // (Optional) If the rule is enabled or n
            "combinationRuleColumnList" : [
                {
                    "id" : "", // The combination rule column id.
                    "value" : "", // The combination rule column value.
                    "type" : "", // (Optional) The type of the combination r
                    "attributeNamedReference" : {
                        "id" : "", // The attribute resource id.
                        "name" : "" // The attribute name.
                    }, // The combination rule column attribute named refere
                    "featureNamedReference" : {
                        "id" : "", // The feature resource id.
                        "name" : "" // The feature name.
                    }, // The combination rule column feature named referenc
                    "moduleNameReference" : {
                        "id" : "", // The module resource id.
                        "name" : "" // The module name.
                    }, // The combination rule column module named reference
                    "assemblyNamedReference" : {
                        "id" : "", // The assembly resource id.
                        "name" : "" // The assembly name.
                    }, // The combination rule column assembly named referen
                    "positionNamedReference" : {
                        "id" : "", // The position resource id.
                        "name" : "" // The position name.
                    }, // The combination rule column position named referen
                    "variantNamedReference" : {
                        "id" : "", // The variant resource id.
                        "name" : "" // The variant name.
                    } // The combination rule column variant reference. This
                },
                ...
            ]
        }
    }
]

```

```

], // (Optional) Columns for type "Combination".
"combinationRuleRowList" : [
  {
    "id" : "", // The combination rule row id.
    "combinationRuleCellList" : [
      {
        "combinationRuleColumnReference" : {
          "id" : "" // The combination rule column id.
        }, // The combination rule column reference.
        "value" : "" // The combination rule column cell val
      },
      ...
    ] // The combination rule row cells.
  },
  ...
], // (Optional) Rows for type "Combination".
"constraint" : "" // (Optional) The constraint expression for
} // The rule.
},
...
] // The rule resource list.
}

```

PostAssemblyInput

A post assembly input.

```

{
  "assembly" : {
    "name" : "", // The assembly name.
    "description" : "", // (Optional) The assembly description.
    "descriptionTranslations" : {
      "key" : ""
    }, // (Optional) The assembly description translations by language
    "consistencyCheckStrategy" : "", // (Optional) The assembly consistency check strategy.
    Supported values: Standard, Weaker. If not specified, then Standard
    "variantEnabled" : false, // (Optional) Whether assembly variant is enabled.
    "virtualVariant" : {
      "name" : "",
      "description" : "",
      "image" : "",
      "document" : "",
      "longDescription" : "",
      "variantValueList" : [
        {
          "attributeNamedReference" : {
            "id" : "", // The attribute resource id.
            "name" : "" // The attribute name.
          }, // The variant assembly attribute named reference. Required for feature named reference.
          "featureNamedReference" : {
            "id" : "", // The feature resource id.
            "name" : "" // The feature name.
          }, // The variant feature named reference. Required for feature named reference.
          "value" : "" // The variant value. Multiple values must be supported.
        },
        ...
      ]
    }, // (Optional) Values of the assembly virtual variant. When not specified, the default value of field 'name' is set and other fields are left unspecified.
  }, // The assembly.
  "attributeList" : [
    {
      "parentAssemblyNamedReference" : {
        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
      }, // The parent assembly.
      "attributeCategoryNamedReference" : {
        "id" : "", // The attribute category resource id.
        "name" : "" // The attribute category name.
      }, // The attribute category.
      "domainNamedReference" : {
        "id" : "", // The domain resource id.
        "name" : "" // The domain name.
      }, // The attribute domain.
    }
  ]
}

```

```

"name" : "", // The attribute name.
"description" : "", // (Optional) The attribute description.
"descriptionTranslations" : {
  "key" : ""
}, // (Optional) The attribute description translations by lan
"io" : false, // (Optional) Whether it is an I/O attribute.
"defaultView" : false, // (Optional) Whether the attribute is
"aggregationStrategy" : "", // (Optional) The aggregation func
"aggregateList" : [
  {
    "attributeNamedReference" : {
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    }, // The attribute named reference.
    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The feature named reference.
    "positionNamedReference" : {
      "id" : "", // The position resource id.
      "name" : "" // The position name.
    } // The position named reference.
  },
  ...
] // (Optional) The list of aggregated attributes and features
},
...
], // The attribute list for assembly.
"positionList" : [
  {
    "parentAssemblyNamedReference" : {
      "id" : "", // The parent assembly resource id.
      "name" : "" // The parent assembly name.
    }, // The parent assembly.
    "assemblyNamedReference" : {
      "id" : "", // The assembly resource id.
      "name" : "" // The assembly name.
    }, // (Optional) The assembly at the position.
    "moduleNamedReference" : {
      "id" : "", // The module resource id.
      "name" : "" // The module name.
    }, // (Optional) The module at the position.
    "name" : "", // The position name.
    "type" : "", // The position type: "Assembly", "Module", "Opti
    "description" : "", // (Optional) The position description.
    "descriptionTranslations" : {
      "key" : ""
    }, // (Optional) The position description translations by lang

```

```

        "defaultView" : false, // (Optional) Whether the position is v
        "dynamic" : false, // (Optional) Whether the position is dynar
If "false", the number of instances is "Static".
If "true", the number of instances can be either "Sizeable" or "Conf
        "numberOfInstancesAttributeReference" : {
            "id" : "", // The attribute resource id.
            "name" : "" // The attribute name.
        }, // (Optional) The attribute that defines number of instance
If this is set, and dynamic is "true", then number of instances is "
        "staticNumberOfInstances" : 1, // (Optional) Number of instanc
If not specified, the default value is 1.
It is undefined for dynamic positions.
        "enabled" : false, // (Optional) Whether the position is enabl
        "qtyType" : "", // The position qty type. Allows values: "Stat
        "qty" : 1, // (Optional) The position qty. Used when qtyType i
        "qtyMin" : 1, // (Optional) The position minimum qty. Used whe
        "qtyMax" : 1 // (Optional) The position maximum qty. Used when
    },
    ...
], // The position list for assembly.
"variantList" : [
    {
        "parentAssemblyNamedReference" : {
            "id" : "", // The parent assembly resource id.
            "name" : "" // The parent assembly name.
        }, // (Optional) The parent assembly. Required for assembly va
        "parentModuleNamedReference" : {
            "id" : "", // The parent module resource id.
            "name" : "" // The parent module name.
        }, // (Optional) The parent module. Required for module varian
        "name" : "", // The variant name.
        "description" : "", // (Optional) The variant description.
        "descriptionTranslations" : {
            "key" : ""
        }, // (Optional) The variant description translations by langu
        "status" : "", // The variant status.
        "longDescription" : "", // (Optional) The variant long descrip
        "longDescriptionTranslations" : {
            "key" : ""
        }, // (Optional) The variant long description translations by
        "image" : "", // (Optional) The variant image.
        "document" : "", // (Optional) The variant document.
        "variantValueList" : [
            {
                "attributeNamedReference" : {
                    "id" : "", // The attribute resource id.
                    "name" : "" // The attribute name.
                }, // The variant assembly attribute named reference. Requ

```

```

    "featureNamedReference" : {
        "id" : "", // The feature resource id.
        "name" : "" // The feature name.
    }, // The variant feature named reference. Required for fe
    "value" : "" // The variant value. Multiple values must be
    },
    ...
], // (Optional) The variant value list. The list should conta
"variantPropertyValueList" : [
    {
        "propertyNamedReference" : {
            "id" : "", // The property resource ID.
            "name" : "" // The property name.
        }, // The property reference
        "value" : "", // The property value.
        "valueTranslations" : {
            "key" : ""
        } // (Optional) The property value translations by languag
    },
    ...
], // (Optional) The variant BoM Meta Data property values lis
"variantPartList" : [
    {
        "partNamedReference" : {
            "id" : "", // The part resource id. Note: id and name ca
            "name" : "" // The part name.
        }, // The part named reference.
        "qty" : 1.0 // The part quantity.
    },
    ...
] // (Optional) The variant part list.
},
...
], // (Optional) The variant list for assembly.
"ruleList" : [
    {
        "assemblyNamedReference" : {
            "id" : "", // The assembly resource id.
            "name" : "" // The assembly name.
        }, // The assembly for the rule.
        "id" : "", // The rule id.
        "type" : "", // The rule type. Allows values: "Constraint" or
        "ruleGroup" : "", // (Optional) Category of rule. Default valu
        "enabled" : false, // (Optional) If the rule is enabled or not
        "combinationRuleColumnList" : [
            {
                "id" : "", // The combination rule column id.
                "value" : "", // The combination rule column value.
            }
        ]
    }
]

```

```

"type" : "", // (Optional) The type of the combination rule
"attributeNamedReference" : {
  "id" : "", // The attribute resource id.
  "name" : "" // The attribute name.
}, // The combination rule column attribute named reference
"featureNamedReference" : {
  "id" : "", // The feature resource id.
  "name" : "" // The feature name.
}, // The combination rule column feature named reference.
"moduleNameReference" : {
  "id" : "", // The module resource id.
  "name" : "" // The module name.
}, // The combination rule column module named reference.
"assemblyNamedReference" : {
  "id" : "", // The assembly resource id.
  "name" : "" // The assembly name.
}, // The combination rule column assembly named reference
"positionNamedReference" : {
  "id" : "", // The position resource id.
  "name" : "" // The position name.
}, // The combination rule column position named reference
"variantNamedReference" : {
  "id" : "", // The variant resource id.
  "name" : "" // The variant name.
} // The combination rule column variant reference. This is
},
...
], // (Optional) Columns for type "Combination".
"combinationRuleRowList" : [
{
  "id" : "", // The combination rule row id.
  "combinationRuleCellList" : [
    {
      "combinationRuleColumnReference" : {
        "id" : "" // The combination rule column id.
      }, // The combination rule column reference.
      "value" : "" // The combination rule column cell value
    },
    ...
  ] // The combination rule row cells.
},
...
], // (Optional) Rows for type "Combination".
"constraint" : "" // (Optional) The constraint expression for
},
...

```

```
] // (Optional) The rule list for assembly.  
}
```

PostAssemblyListInput

A post assembly list input. Note that each of the related items are listed separately with a parent reference pointing to the assembly.

```

{
  "assemblyList" : [
    {
      "name" : "", // The assembly name.
      "description" : "", // (Optional) The assembly description.
      "descriptionTranslations" : {
        "key" : ""
      }, // (Optional) The assembly description translations by lang
      "consistencyCheckStrategy" : "", // (Optional) The assembly cc
      Supported values: Standard, Weaker. If not specified, then Standard
      "variantEnabled" : false, // (Optional) Whether assembly varia
      "virtualVariant" : {
        "name" : "",
        "description" : "",
        "image" : "",
        "document" : "",
        "longDescription" : "",
        "variantValueList" : [
          {
            "attributeNamedReference" : {
              "id" : "", // The attribute resource id.
              "name" : "" // The attribute name.
            }, // The variant assembly attribute named reference. Re
            "featureNamedReference" : {
              "id" : "", // The feature resource id.
              "name" : "" // The feature name.
            }, // The variant feature named reference. Required for
            "value" : "" // The variant value. Multiple values must
          },
          ...
        ]
      } // (Optional) Values of the assembly virtual variant. When n
      default value of field 'name' is set and other fields are left unass
    },
    ...
  ], // The assembly list.
  "attributeList" : [
    {
      "parentAssemblyNamedReference" : {
        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
      }, // The parent assembly.
      "attributeCategoryNamedReference" : {
        "id" : "", // The attribute category resource id.
        "name" : "" // The attribute category name.
      }, // The attribute category.
      "domainNamedReference" : {

```

```

    "id" : "", // The domain resource id.
    "name" : "" // The domain name.
  }, // The attribute domain.
  "name" : "", // The attribute name.
  "description" : "", // (Optional) The attribute description.
  "descriptionTranslations" : {
    "key" : ""
  }, // (Optional) The attribute description translations by lan
  "io" : false, // (Optional) Whether it is an I/O attribute.
  "defaultView" : false, // (Optional) Whether the attribute is
  "aggregationStrategy" : "", // (Optional) The aggregation func
  "aggregateList" : [
    {
      "attributeNamedReference" : {
        "id" : "", // The attribute resource id.
        "name" : "" // The attribute name.
      }, // The attribute named reference.
      "featureNamedReference" : {
        "id" : "", // The feature resource id.
        "name" : "" // The feature name.
      }, // The feature named reference.
      "positionNamedReference" : {
        "id" : "", // The position resource id.
        "name" : "" // The position name.
      } // The position named reference.
    },
    ...
  ] // (Optional) The list of aggregated attributes and features
},
...
], // The assembly attribute list for listed assemblies.
"positionList" : [
  {
    "parentAssemblyNamedReference" : {
      "id" : "", // The parent assembly resource id.
      "name" : "" // The parent assembly name.
    }, // The parent assembly.
    "assemblyNamedReference" : {
      "id" : "", // The assembly resource id.
      "name" : "" // The assembly name.
    }, // (Optional) The assembly at the position.
    "moduleNamedReference" : {
      "id" : "", // The module resource id.
      "name" : "" // The module name.
    }, // (Optional) The module at the position.
    "name" : "", // The position name.
    "type" : "", // The position type: "Assembly", "Module", "Opti
    "description" : "", // (Optional) The position description.

```

```

    "descriptionTranslations" : {
      "key" : ""
    }, // (Optional) The position description translations by lang
    "defaultView" : false, // (Optional) Whether the position is v
    "dynamic" : false, // (Optional) Whether the position is dynar
    If "false", the number of instances is "Static".
    If "true", the number of instances can be either "Sizeable" or "Conf
      "numberOfInstancesAttributeReference" : {
        "id" : "", // The attribute resource id.
        "name" : "" // The attribute name.
      }, // (Optional) The attribute that defines number of instance
    If this is set, and dynamic is "true", then number of instances is "
      "staticNumberOfInstances" : 1, // (Optional) Number of instanc
    If not specified, the default value is 1.
    It is undefined for dynamic positions.
    "enabled" : false, // (Optional) Whether the position is enabl
    "qtyType" : "", // The position qty type. Allows values: "Stat
    "qty" : 1, // (Optional) The position qty. Used when qtyType i
    "qtyMin" : 1, // (Optional) The position minimum qty. Used whe
    "qtyMax" : 1 // (Optional) The position maximum qty. Used when
  },
  ...
], // The assembly position list for listed assemblies.
"variantList" : [
  {
    "parentAssemblyNamedReference" : {
      "id" : "", // The parent assembly resource id.
      "name" : "" // The parent assembly name.
    }, // (Optional) The parent assembly. Required for assembly va
    "parentModuleNamedReference" : {
      "id" : "", // The parent module resource id.
      "name" : "" // The parent module name.
    }, // (Optional) The parent module. Required for module varian
    "name" : "", // The variant name.
    "description" : "", // (Optional) The variant description.
    "descriptionTranslations" : {
      "key" : ""
    }, // (Optional) The variant description translations by langu
    "status" : "", // The variant status.
    "longDescription" : "", // (Optional) The variant long descrip
    "longDescriptionTranslations" : {
      "key" : ""
    }, // (Optional) The variant long description translations by
    "image" : "", // (Optional) The variant image.
    "document" : "", // (Optional) The variant document.
    "variantValueList" : [
      {
        "attributeNamedReference" : {

```

```

        "id" : "", // The attribute resource id.
        "name" : "" // The attribute name.
    }, // The variant assembly attribute named reference. Requ
    "featureNamedReference" : {
        "id" : "", // The feature resource id.
        "name" : "" // The feature name.
    }, // The variant feature named reference. Required for fe
    "value" : "" // The variant value. Multiple values must be
    },
    ...
], // (Optional) The variant value list. The list should conta
"variantPropertyValueList" : [
    {
        "propertyNamedReference" : {
            "id" : "", // The property resource ID.
            "name" : "" // The property name.
        }, // The property reference
        "value" : "", // The property value.
        "valueTranslations" : {
            "key" : ""
        } // (Optional) The property value translations by languag
    },
    ...
], // (Optional) The variant BoM Meta Data property values lis
"variantPartList" : [
    {
        "partNamedReference" : {
            "id" : "", // The part resource id. Note: id and name ca
            "name" : "" // The part name.
        }, // The part named reference.
        "qty" : 1.0 // The part quantity.
    },
    ...
] // (Optional) The variant part list.
},
...
], // (Optional) The assembly variant list for listed assemblies.
"ruleList" : [
    {
        "assemblyNamedReference" : {
            "id" : "", // The assembly resource id.
            "name" : "" // The assembly name.
        }, // The assembly for the rule.
        "id" : "", // The rule id.
        "type" : "", // The rule type. Allows values: "Constraint" or
        "ruleGroup" : "", // (Optional) Category of rule. Default valu
        "enabled" : false, // (Optional) If the rule is enabled or not
        "combinationRuleColumnList" : [

```

```

{
  "id" : "", // The combination rule column id.
  "value" : "", // The combination rule column value.
  "type" : "", // (Optional) The type of the combination rul
  "attributeNamedReference" : {
    "id" : "", // The attribute resource id.
    "name" : "" // The attribute name.
  }, // The combination rule column attribute named referenc
  "featureNamedReference" : {
    "id" : "", // The feature resource id.
    "name" : "" // The feature name.
  }, // The combination rule column feature named reference.
  "moduleNamedReference" : {
    "id" : "", // The module resource id.
    "name" : "" // The module name.
  }, // The combination rule column module named reference.
  "assemblyNamedReference" : {
    "id" : "", // The assembly resource id.
    "name" : "" // The assembly name.
  }, // The combination rule column assembly named reference
  "positionNamedReference" : {
    "id" : "", // The position resource id.
    "name" : "" // The position name.
  }, // The combination rule column position named reference
  "variantNamedReference" : {
    "id" : "", // The variant resource id.
    "name" : "" // The variant name.
  } // The combination rule column variant reference. This i
},
...
], // (Optional) Columns for type "Combination".
"combinationRuleRowList" : [
{
  "id" : "", // The combination rule row id.
  "combinationRuleCellList" : [
    {
      "combinationRuleColumnReference" : {
        "id" : "" // The combination rule column id.
      }, // The combination rule column reference.
      "value" : "" // The combination rule column cell value
    },
    ...
  ] // The combination rule row cells.
},
...
], // (Optional) Rows for type "Combination".
"constraint" : "" // (Optional) The constraint expression for
},

```

```
...
], // (Optional) The assembly rule list for listed assemblies.
"removeNotReplaced" : false // (Optional) Set to true to remove ex
}
```

PostAssemblyListOutput

A post assembly list output. Note that each of the related items are listed separately with a parent reference pointing to the assembly.

```
{
  "assemblyNamedReferenceList" : [
    {
      "id" : "", // The assembly resource id.
      "name" : "" // The assembly name.
    },
    ...
  ], // The assembly named reference list.
  "assemblyAttributeNamedReferenceList" : [
    {
      "parentAssemblyNamedReference" : {
        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
      }, // The parent assembly named reference.
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    },
    ...
  ], // The assembly attribute named reference list.
  "assemblyPositionNamedReferenceList" : [
    {
      "parentAssemblyNamedReference" : {
        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
      }, // The parent assembly named reference.
      "id" : "", // The position resource id.
      "name" : "" // The position name.
    },
    ...
  ], // The assembly position named reference list.
  "assemblyVariantNamedReferenceList" : [
    {
      "parentAssemblyNamedReference" : {
        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
      }, // The parent assembly named reference.
      "id" : "", // The variant resource id.
      "name" : "" // The variant name.
    },
    ...
  ], // The assembly variant named reference list.
  "ruleReferenceList" : [
    {
      "id" : "" // The rule resource id.
    },
    ...
  ]
}
```

```
] // The assembly rule reference list.  
}
```

PostAssemblyOutput

A post assembly output.

```
JSON XML  
  
{  
  "assemblyNamedReference" : {  
    "id" : "", // The assembly resource id.  
    "name" : "" // The assembly name.  
  }, // The assembly named reference.  
  "attributeNamedReferenceList" : [  
    {  
      "id" : "", // The attribute resource id.  
      "name" : "" // The attribute name.  
    },  
    ...  
  ], // The attribute named reference list.  
  "positionNamedReferenceList" : [  
    {  
      "id" : "", // The position resource id.  
      "name" : "" // The position name.  
    },  
    ...  
  ], // The position named reference list.  
  "variantNamedReferenceList" : [  
    {  
      "id" : "", // The variant resource id.  
      "name" : "" // The variant name.  
    },  
    ...  
  ], // The variant named reference list.  
  "ruleReferenceList" : [  
    {  
      "id" : "" // The rule resource id.  
    },  
    ...  
  ] // The rule reference list.  
}
```

PutAssemblyInput

A put assembly input.

```

{
  "assembly" : {
    "name" : "", // The assembly name.
    "description" : "", // (Optional) The assembly description.
    "descriptionTranslations" : {
      "key" : ""
    }, // (Optional) The assembly description translations by language
    "consistencyCheckStrategy" : "", // (Optional) The assembly consistency check strategy.
    Supported values: Standard, Weaker. If not specified, then Standard
    "variantEnabled" : false, // (Optional) Whether assembly variant is enabled.
    "virtualVariant" : {
      "name" : "",
      "description" : "",
      "image" : "",
      "document" : "",
      "longDescription" : "",
      "variantValueList" : [
        {
          "attributeNamedReference" : {
            "id" : "", // The attribute resource id.
            "name" : "" // The attribute name.
          }, // The variant assembly attribute named reference. Required for feature named reference.
          "featureNamedReference" : {
            "id" : "", // The feature resource id.
            "name" : "" // The feature name.
          }, // The variant feature named reference. Required for feature named reference.
          "value" : "" // The variant value. Multiple values must be supported.
        },
        ...
      ]
    }, // (Optional) Values of the assembly virtual variant. When not specified, the default value of field 'name' is set and other fields are left unspecified.
  }, // The assembly.
  "attributeList" : [
    {
      "parentAssemblyNamedReference" : {
        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
      }, // The parent assembly.
      "attributeCategoryNamedReference" : {
        "id" : "", // The attribute category resource id.
        "name" : "" // The attribute category name.
      }, // The attribute category.
      "domainNamedReference" : {
        "id" : "", // The domain resource id.
        "name" : "" // The domain name.
      }, // The attribute domain.
    }
  ]
}

```

```

"name" : "", // The attribute name.
"description" : "", // (Optional) The attribute description.
"descriptionTranslations" : {
  "key" : ""
}, // (Optional) The attribute description translations by lan
"io" : false, // (Optional) Whether it is an I/O attribute.
"defaultView" : false, // (Optional) Whether the attribute is
"aggregationStrategy" : "", // (Optional) The aggregation func
"aggregateList" : [
  {
    "attributeNamedReference" : {
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    }, // The attribute named reference.
    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The feature named reference.
    "positionNamedReference" : {
      "id" : "", // The position resource id.
      "name" : "" // The position name.
    } // The position named reference.
  },
  ...
] // (Optional) The list of aggregated attributes and features
},
...
], // The attribute list for assembly.
"positionList" : [
  {
    "parentAssemblyNamedReference" : {
      "id" : "", // The parent assembly resource id.
      "name" : "" // The parent assembly name.
    }, // The parent assembly.
    "assemblyNamedReference" : {
      "id" : "", // The assembly resource id.
      "name" : "" // The assembly name.
    }, // (Optional) The assembly at the position.
    "moduleNamedReference" : {
      "id" : "", // The module resource id.
      "name" : "" // The module name.
    }, // (Optional) The module at the position.
    "name" : "", // The position name.
    "type" : "", // The position type: "Assembly", "Module", "Opti
    "description" : "", // (Optional) The position description.
    "descriptionTranslations" : {
      "key" : ""
    }, // (Optional) The position description translations by lang

```

```

        "defaultView" : false, // (Optional) Whether the position is v
        "dynamic" : false, // (Optional) Whether the position is dynar
If "false", the number of instances is "Static".
If "true", the number of instances can be either "Sizeable" or "Conf
        "numberOfInstancesAttributeReference" : {
            "id" : "", // The attribute resource id.
            "name" : "" // The attribute name.
        }, // (Optional) The attribute that defines number of instance
If this is set, and dynamic is "true", then number of instances is "
        "staticNumberOfInstances" : 1, // (Optional) Number of instanc
If not specified, the default value is 1.
It is undefined for dynamic positions.
        "enabled" : false, // (Optional) Whether the position is enabl
        "qtyType" : "", // The position qty type. Allows values: "Stat
        "qty" : 1, // (Optional) The position qty. Used when qtyType i
        "qtyMin" : 1, // (Optional) The position minimum qty. Used whe
        "qtyMax" : 1 // (Optional) The position maximum qty. Used when
    },
    ...
], // The position list for assembly.
"variantList" : [
    {
        "parentAssemblyNamedReference" : {
            "id" : "", // The parent assembly resource id.
            "name" : "" // The parent assembly name.
        }, // (Optional) The parent assembly. Required for assembly va
        "parentModuleNamedReference" : {
            "id" : "", // The parent module resource id.
            "name" : "" // The parent module name.
        }, // (Optional) The parent module. Required for module varian
        "name" : "", // The variant name.
        "description" : "", // (Optional) The variant description.
        "descriptionTranslations" : {
            "key" : ""
        }, // (Optional) The variant description translations by langu
        "status" : "", // The variant status.
        "longDescription" : "", // (Optional) The variant long descrip
        "longDescriptionTranslations" : {
            "key" : ""
        }, // (Optional) The variant long description translations by
        "image" : "", // (Optional) The variant image.
        "document" : "", // (Optional) The variant document.
        "variantValueList" : [
            {
                "attributeNamedReference" : {
                    "id" : "", // The attribute resource id.
                    "name" : "" // The attribute name.
                }, // The variant assembly attribute named reference. Requ

```

```

    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The variant feature named reference. Required for fe
    "value" : "" // The variant value. Multiple values must be
  },
  ...
], // (Optional) The variant value list. The list should conta
"variantPropertyValueList" : [
  {
    "propertyNamedReference" : {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    }, // The property reference
    "value" : "", // The property value.
    "valueTranslations" : {
      "key" : ""
    } // (Optional) The property value translations by languag
  },
  ...
], // (Optional) The variant BoM Meta Data property values lis
"variantPartList" : [
  {
    "partNamedReference" : {
      "id" : "", // The part resource id. Note: id and name ca
      "name" : "" // The part name.
    }, // The part named reference.
    "qty" : 1.0 // The part quantity.
  },
  ...
] // (Optional) The variant part list.
},
...
], // (Optional) The variant list for assembly.
"ruleList" : [
  {
    "assemblyNamedReference" : {
      "id" : "", // The assembly resource id.
      "name" : "" // The assembly name.
    }, // The assembly for the rule.
    "id" : "", // The rule id.
    "type" : "", // The rule type. Allows values: "Constraint" or
    "ruleGroup" : "", // (Optional) Category of rule. Default valu
    "enabled" : false, // (Optional) If the rule is enabled or not
    "combinationRuleColumnList" : [
      {
        "id" : "", // The combination rule column id.
        "value" : "", // The combination rule column value.

```

```

"type" : "", // (Optional) The type of the combination rule
"attributeNamedReference" : {
  "id" : "", // The attribute resource id.
  "name" : "" // The attribute name.
}, // The combination rule column attribute named reference
"featureNamedReference" : {
  "id" : "", // The feature resource id.
  "name" : "" // The feature name.
}, // The combination rule column feature named reference.
"moduleNameReference" : {
  "id" : "", // The module resource id.
  "name" : "" // The module name.
}, // The combination rule column module named reference.
"assemblyNamedReference" : {
  "id" : "", // The assembly resource id.
  "name" : "" // The assembly name.
}, // The combination rule column assembly named reference
"positionNamedReference" : {
  "id" : "", // The position resource id.
  "name" : "" // The position name.
}, // The combination rule column position named reference
"variantNamedReference" : {
  "id" : "", // The variant resource id.
  "name" : "" // The variant name.
} // The combination rule column variant reference. This is
},
...
], // (Optional) Columns for type "Combination".
"combinationRuleRowList" : [
{
  "id" : "", // The combination rule row id.
  "combinationRuleCellList" : [
    {
      "combinationRuleColumnReference" : {
        "id" : "" // The combination rule column id.
      }, // The combination rule column reference.
      "value" : "" // The combination rule column cell value
    },
    ...
  ] // The combination rule row cells.
},
...
], // (Optional) Rows for type "Combination".
"constraint" : "" // (Optional) The constraint expression for
},
...

```

```
] // (Optional) The rule list for assembly.  
}
```

PutAssemblyOutput

A put assembly output.

```
JSON XML  
{  
  "assemblyNamedReference" : {  
    "id" : "", // The assembly resource id.  
    "name" : "" // The assembly name.  
  }, // The assembly named reference.  
  "attributeNamedReferenceList" : [  
    {  
      "id" : "", // The attribute resource id.  
      "name" : "" // The attribute name.  
    },  
    ...  
  ], // The attribute named reference list.  
  "positionNamedReferenceList" : [  
    {  
      "id" : "", // The position resource id.  
      "name" : "" // The position name.  
    },  
    ...  
  ], // The position named reference list.  
  "variantNamedReferenceList" : [  
    {  
      "id" : "", // The variant resource id.  
      "name" : "" // The variant name.  
    },  
    ...  
  ], // The variant named reference list.  
  "ruleReferenceList" : [  
    {  
      "id" : "" // The rule resource id.  
    },  
    ...  
  ] // The rule reference list.  
}
```

Attribute Category

DeleteAttributeCategoryListInput

A delete attribute category list input.

```
JSON XML
{
  "attributeCategoryNamedReferenceList" : [
    {
      "id" : "", // The attribute category resource id.
      "name" : "" // The attribute category name.
    },
    ...
  ] // The attribute category named reference list.
}
```

DeleteAttributeCategoryListOutput

A delete attribute category output.

```
JSON XML
{
  "attributeCategoryNamedReferenceList" : [
    {
      "id" : "", // The attribute category resource id.
      "name" : "" // The attribute category name.
    },
    ...
  ] // The attribute category named reference list.
}
```

DeleteAttributeCategoryOutput

A delete attribute category output.

```
JSON XML
{
  "attributeCategoryNamedReference" : {
    "id" : "", // The attribute category resource id.
    "name" : "" // The attribute category name.
  } // The attribute category named reference.
}
```

GetAttributeCategoryListOutput

A get attribute category resource list output.

```
{
  "attributeCategoryResourceList" : [
    {
      "attributeCategoryReference" : {
        "id" : "" // The attribute category resource id.
      }, // The attribute category reference.
      "attributeCategory" : {
        "parentAttributeCategoryNamedReference" : {
          "id" : "", // The parent attribute category resource id.
          "name" : "" // The parent attribute category name.
        }, // (Optional) The parent attribute category.
        "name" : "", // The attribute category name.
        "description" : "", // (Optional) The attribute category des
        "descriptionTranslations" : {
          "key" : ""
        } // (Optional) The attribute category description translati
      } // The attribute category.
    },
    ...
  ] // The attribute category resource list.
}
```

GetAttributeCategoryOutput

A get attribute category resource output.

JSON XML

```
{
  "attributeCategoryResource" : {
    "attributeCategoryReference" : {
      "id" : "" // The attribute category resource id.
    }, // The attribute category reference.
    "attributeCategory" : {
      "parentAttributeCategoryNamedReference" : {
        "id" : "", // The parent attribute category resource id.
        "name" : "" // The parent attribute category name.
      }, // (Optional) The parent attribute category.
      "name" : "", // The attribute category name.
      "description" : "", // (Optional) The attribute category descr
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The attribute category description translation
    } // The attribute category.
  } // The attribute category resource.
}
```

PostAttributeCategoryInput

A post attribute category input.

JSON XML

```
{
  "attributeCategory" : {
    "parentAttributeCategoryNamedReference" : {
      "id" : "", // The parent attribute category resource id.
      "name" : "" // The parent attribute category name.
    }, // (Optional) The parent attribute category.
    "name" : "", // The attribute category name.
    "description" : "", // (Optional) The attribute category descrip
    "descriptionTranslations" : {
      "key" : ""
    } // (Optional) The attribute category description translations
  } // The attribute category.
}
```

PostAttributeCategoryListInput

A post attribute category list input.

JSON XML

```
{
  "attributeCategoryList" : [
    {
      "parentAttributeCategoryNamedReference" : {
        "id" : "", // The parent attribute category resource id.
        "name" : "" // The parent attribute category name.
      }, // (Optional) The parent attribute category.
      "name" : "", // The attribute category name.
      "description" : "", // (Optional) The attribute category descr
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The attribute category description translation
    },
    ...
  ], // The attribute category list.
  "removeNotReplaced" : false // (Optional) Set to true to remove ex
}
```

PostAttributeCategoryListOutput

A post attribute category list output.

JSON XML

```
{
  "attributeCategoryNamedReferenceList" : [
    {
      "id" : "", // The attribute category resource id.
      "name" : "" // The attribute category name.
    },
    ...
  ] // The attribute category named reference list.
}
```

PostAttributeCategoryOutput

A post attribute category output.

JSON XML


```
{
  "attributeCategoryNamedReference" : {
    "id" : "", // The attribute category resource id.
    "name" : "" // The attribute category name.
  } // The attribute category named reference.
}
```

PutAttributeCategoryInput

A put attribute category input.

JSON XML

```
{
  "attributeCategory" : {
    "parentAttributeCategoryNamedReference" : {
      "id" : "", // The parent attribute category resource id.
      "name" : "" // The parent attribute category name.
    }, // (Optional) The parent attribute category.
    "name" : "", // The attribute category name.
    "description" : "", // (Optional) The attribute category descrip
    "descriptionTranslations" : {
      "key" : ""
    } // (Optional) The attribute category description translations
  } // The attribute category.
}
```



PutAttributeCategoryOutput

A put attribute category output.

JSON XML

```
{
  "attributeCategoryNamedReference" : {
    "id" : "", // The attribute category resource id.
    "name" : "" // The attribute category name.
  } // The attribute category named reference.
}
```

BoM Meta Data

DeleteBomMetaDataPropertyListInput

A delete BoM Meta Data property list input.

```
JSON XML
{
  "bomMetaDataPropertyNamedReferenceList" : [
    {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    },
    ...
  ] // The property named reference list.
}
```

DeleteBomMetaDataPropertyListOutput

A delete BoM Meta Data property list output.

```
JSON XML
{
  "bomMetaDataPropertyNamedReferenceList" : [
    {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    },
    ...
  ] // The property named reference list.
}
```

DeleteBomMetaDataPropertyOutput

A delete BoM Meta Data property output.

```
JSON XML
{
  "bomMetaDataPropertyNamedReference" : {
    "id" : "", // The property resource ID.
    "name" : "" // The property name.
  } // The property named reference.
}
```

GetBomMetaDataPropertyListOutput

A get BoM Meta Data property resource list output.

JSON XML

```
{
  "bomMetaDataPropertyResourceList" : [
    {
      "bomMetaDataPropertyReference" : {
        "id" : "" // The property resource ID.
      }, // The property reference.
      "bomMetaDataProperty" : {
        "name" : "", // The property name.
        "propertyId" : "", // The property ID.
        "description" : "", // (Optional) The property description.
        "descriptionTranslations" : {
          "key" : ""
        }, // (Optional) The property description translations by language.
        "type" : "" // The property type.
      } // The property.
    },
    ...
  ] // The property resource list.
}
```

GetBomMetaDataPropertyOutput

A get BoM Meta Data property resource output.

JSON XML

```
{
  "bomMetaDataPropertyResource" : {
    "bomMetaDataPropertyReference" : {
      "id" : "" // The property resource ID.
    }, // The property reference.
    "bomMetaDataProperty" : {
      "name" : "", // The property name.
      "propertyId" : "", // The property ID.
      "description" : "", // (Optional) The property description.
      "descriptionTranslations" : {
        "key" : ""
      }, // (Optional) The property description translations by language.
      "type" : "" // The property type.
    } // The property.
  } // The property resource.
}
```

PostBomMetaDataPropertyListInput

A post BoM Meta Data property list input.

```
JSON XML
{
  "bomMetaDataPropertyList" : [
    {
      "name" : "", // The property name.
      "propertyId" : "", // The property ID.
      "description" : "", // (Optional) The property description.
      "descriptionTranslations" : {
        "key" : ""
      }, // (Optional) The property description translations by lang
      "type" : "" // The property type.
    },
    ...
  ], // The property list.
  "removeNotReplaced" : false // (Optional) Set to true to remove ex
}
```

PostBomMetaDataPropertyListOutput

A post BoM Meta Data property list output.

```
JSON XML
{
  "bomMetaDataPropertyNamedReferenceList" : [
    {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    },
    ...
  ] // The property named reference list.
}
```

PutBomMetaDataPropertyInput

A put BoM Meta Data property input.

JSON XML

```
{
  "bomMetaDataProperty" : {
    "name" : "", // The property name.
    "propertyId" : "", // The property ID.
    "description" : "", // (Optional) The property description.
    "descriptionTranslations" : {
      "key" : ""
    }, // (Optional) The property description translations by language
    "type" : "" // The property type.
  } // The property.
}
```

PutBomMetaDataPropertyOutput

A put BoM Meta Data property output.

JSON XML

```
{
  "bomMetaDataPropertyNamedReference" : {
    "id" : "", // The property resource ID.
    "name" : "" // The property name.
  } // The property named reference.
}
```

Domain

DeleteDomainListInput

A delete domain list input.

JSON XML

```
{
  "domainNamedReferenceList" : [
    {
      "id" : "", // The domain resource id.
      "name" : "" // The domain name.
    },
    ...
  ] // The domain named reference list.
}
```

DeleteDomainListOutput

A delete domain output.

```
JSON XML
{
  "domainNamedReferenceList" : [
    {
      "id" : "", // The domain resource id.
      "name" : "" // The domain name.
    },
    ...
  ] // The domain named reference list.
}
```

DeleteDomainOutput

A delete domain output.

```
JSON XML
{
  "domainNamedReference" : {
    "id" : "", // The domain resource id.
    "name" : "" // The domain name.
  } // The domain named reference.
}
```

GetDomainListOutput

A get domain resource list output.

```

{
  "domainResourceList" : [
    {
      "domainReference" : {
        "id" : "" // The domain resource id.
      }, // The domain reference.
      "domain" : {
        "name" : "", // The domain name.
        "description" : "", // (Optional) The domain description.
        "type" : "", // The domain type.
        "enumElementList" : [
          {
            "name" : "", // The domain element name.
            "description" : "", // (Optional) The domain element des
            "descriptionTranslations" : {
              "key" : ""
            }, // (Optional) The domain element description translat
            "longDescription" : "", // (Optional) The domain element
            "longDescriptionTranslations" : {
              "key" : ""
            }, // (Optional) The domain element long description tra
            "uomTypeElementId" : "", // (Optional) Unit Type Element
            "value" : "" // (Optional) The Domain Element Value is s
          },
          ...
        ], // (Optional) The elements of the domain. Only applicable
        "booleanYes" : {
          "name" : "", // The name of the boolean Yes.
          "nameTranslations" : {
            "key" : ""
          } // (Optional) The name translations by language of the b
        }, // (Optional) The boolean Yes element of the domain. Only
        "booleanNo" : {
          "name" : "", // The name of the boolean No.
          "nameTranslations" : {
            "key" : ""
          } // (Optional) The name translations by language of the b
        }, // (Optional) The boolean No element of the domain. Only
        "floatRange" : {
          "min" : 1.0, // The minimum float value.
          "max" : 1.0 // The maximum float value.
        }, // (Optional) The range of the float domain. Only applica
        "integerRange" : {
          "min" : 1, // The minimum integer value.
          "max" : 1 // The maximum integer value.
        }, // (Optional) The range of the integer domain. Only appli
        "uomItem" : "", // (Optional) The unit of measure id. Option

```

```
    "uomDomain" : "", // (Optional) The unit domain id. Optional
    "appendUnitSymbol" : false, // (Optional) If set to true the
In this case the domain elements do not contain the symbol. Default:
    "valueType" : "", // (Optional) Defines if the domain elemen
Empty by default (domain elements do not have connected values). Ava
Float = floating point decimal values, Fixed = fixed point decimal v
    "decimals" : 1 // (Optional) The number of decimals when val
Not defined if valueType is not Fixed.
    } // The domain.
  },
  ...
] // The domain resource list.
}
```

GetDomainOutput

A get domain resource output.

```

{
  "domainResource" : {
    "domainReference" : {
      "id" : "" // The domain resource id.
    }, // The domain reference.
    "domain" : {
      "name" : "", // The domain name.
      "description" : "", // (Optional) The domain description.
      "type" : "", // The domain type.
      "enumElementList" : [
        {
          "name" : "", // The domain element name.
          "description" : "", // (Optional) The domain element descr
          "descriptionTranslations" : {
            "key" : ""
          }, // (Optional) The domain element description translatio
          "longDescription" : "", // (Optional) The domain element l
          "longDescriptionTranslations" : {
            "key" : ""
          }, // (Optional) The domain element long description trans
          "uomTypeElementId" : "", // (Optional) Unit Type Element i
          "value" : "" // (Optional) The Domain Element Value is spe
        },
        ...
      ], // (Optional) The elements of the domain. Only applicabl
      "booleanYes" : {
        "name" : "", // The name of the boolean Yes.
        "nameTranslations" : {
          "key" : ""
        } // (Optional) The name translations by language of the boc
      }, // (Optional) The boolean Yes element of the domain. Only a
      "booleanNo" : {
        "name" : "", // The name of the boolean No.
        "nameTranslations" : {
          "key" : ""
        } // (Optional) The name translations by language of the boc
      }, // (Optional) The boolean No element of the domain. Only ap
      "floatRange" : {
        "min" : 1.0, // The minimum float value.
        "max" : 1.0 // The maximum float value.
      }, // (Optional) The range of the float domain. Only applicabl
      "integerRange" : {
        "min" : 1, // The minimum integer value.
        "max" : 1 // The maximum integer value.
      }, // (Optional) The range of the integer domain. Only applica
      "uomItem" : "", // (Optional) The unit of measure id. Optional
      "uomDomain" : "", // (Optional) The unit domain id. Optional,

```

```
    "appendUnitSymbol" : false, // (Optional) If set to true then
    In this case the domain elements do not contain the symbol. Default:
    "valueType" : "", // (Optional) Defines if the domain elements
    Empty by default (domain elements do not have connected values). Ava
    Float = floating point decimal values, Fixed = fixed point decimal v
    "decimals" : 1 // (Optional) The number of decimals when value
    Not defined if valueType is not Fixed.
    } // The domain.
  } // The domain resource.
}
```

PostDomainInput

A post domain input.

```

{
  "domain" : {
    "name" : "", // The domain name.
    "description" : "", // (Optional) The domain description.
    "type" : "", // The domain type.
    "enumElementList" : [
      {
        "name" : "", // The domain element name.
        "description" : "", // (Optional) The domain element descrip
        "descriptionTranslations" : {
          "key" : ""
        }, // (Optional) The domain element description translations
        "longDescription" : "", // (Optional) The domain element lon
        "longDescriptionTranslations" : {
          "key" : ""
        }, // (Optional) The domain element long description transla
        "uomTypeElementId" : "", // (Optional) Unit Type Element id.
        "value" : "" // (Optional) The Domain Element Value is speci
      },
      ...
    ], // (Optional) The elements of the domain. Only applicable to
    "booleanYes" : {
      "name" : "", // The name of the boolean Yes.
      "nameTranslations" : {
        "key" : ""
      } // (Optional) The name translations by language of the boole
    }, // (Optional) The boolean Yes element of the domain. Only app
    "booleanNo" : {
      "name" : "", // The name of the boolean No.
      "nameTranslations" : {
        "key" : ""
      } // (Optional) The name translations by language of the boole
    }, // (Optional) The boolean No element of the domain. Only appl
    "floatRange" : {
      "min" : 1.0, // The minimum float value.
      "max" : 1.0 // The maximum float value.
    }, // (Optional) The range of the float domain. Only applicable
    "integerRange" : {
      "min" : 1, // The minimum integer value.
      "max" : 1 // The maximum integer value.
    }, // (Optional) The range of the integer domain. Only applicabl
    "uomItem" : "", // (Optional) The unit of measure id. Optional,
    "uomDomain" : "", // (Optional) The unit domain id. Optional, on
    "appendUnitSymbol" : false, // (Optional) If set to true then th
    "valueType" : "", // (Optional) Defines if the domain elements h
  }
}

```

In this case the domain elements do not contain the symbol. Default: Empty by default (domain elements do not have connected values). Ava

```
Float = floating point decimal values, Fixed = fixed point decimal v
    "decimals" : 1 // (Optional) The number of decimals when valueTy
Not defined if valueType is not Fixed.
    } // The domain.
}
```

PostDomainListInput

A post domain list input.

```

{
  "domainList" : [
    {
      "name" : "", // The domain name.
      "description" : "", // (Optional) The domain description.
      "type" : "", // The domain type.
      "enumElementList" : [
        {
          "name" : "", // The domain element name.
          "description" : "", // (Optional) The domain element descr
          "descriptionTranslations" : {
            "key" : ""
          }, // (Optional) The domain element description translati
          "longDescription" : "", // (Optional) The domain element l
          "longDescriptionTranslations" : {
            "key" : ""
          }, // (Optional) The domain element long description trans
          "uomTypeElementId" : "", // (Optional) Unit Type Element i
          "value" : "" // (Optional) The Domain Element Value is spe
        },
        ...
      ], // (Optional) The elements of the domain. Only applicable t
      "booleanYes" : {
        "name" : "", // The name of the boolean Yes.
        "nameTranslations" : {
          "key" : ""
        } // (Optional) The name translations by language of the boc
      }, // (Optional) The boolean Yes element of the domain. Only a
      "booleanNo" : {
        "name" : "", // The name of the boolean No.
        "nameTranslations" : {
          "key" : ""
        } // (Optional) The name translations by language of the boc
      }, // (Optional) The boolean No element of the domain. Only ap
      "floatRange" : {
        "min" : 1.0, // The minimum float value.
        "max" : 1.0 // The maximum float value.
      }, // (Optional) The range of the float domain. Only applicabl
      "integerRange" : {
        "min" : 1, // The minimum integer value.
        "max" : 1 // The maximum integer value.
      }, // (Optional) The range of the integer domain. Only applica
      "uomItem" : "", // (Optional) The unit of measure id. Optional
      "uomDomain" : "", // (Optional) The unit domain id. Optional,
      "appendUnitSymbol" : false, // (Optional) If set to true then
      // In this case the domain elements do not contain the symbol. Default:
      "valueType" : "", // (Optional) Defines if the domain elements

```

Empty by default (domain elements do not have connected values). Available = floating point decimal values, Fixed = fixed point decimal values
"decimals" : 1 // (Optional) The number of decimals when value is not defined if valueType is not Fixed.

```
    },  
    ...  
  ], // The domain list.  
  "removeNotReplaced" : false // (Optional) Set to true to remove elements  
}
```

PostDomainListOutput

A post domain list output.

```
JSON XML  
{  
  "domainNamedReferenceList" : [  
    {  
      "id" : "", // The domain resource id.  
      "name" : "" // The domain name.  
    },  
    ...  
  ] // The domain named reference list.  
}
```

PostDomainOutput

A post domain output.

```
JSON XML  
{  
  "domainNamedReference" : {  
    "id" : "", // The domain resource id.  
    "name" : "" // The domain name.  
  } // The domain named reference.  
}
```

PutDomainInput

A put domain input.

```

{
  "domain" : {
    "name" : "", // The domain name.
    "description" : "", // (Optional) The domain description.
    "type" : "", // The domain type.
    "enumElementList" : [
      {
        "name" : "", // The domain element name.
        "description" : "", // (Optional) The domain element descrip
        "descriptionTranslations" : {
          "key" : ""
        }, // (Optional) The domain element description translations
        "longDescription" : "", // (Optional) The domain element lon
        "longDescriptionTranslations" : {
          "key" : ""
        }, // (Optional) The domain element long description transla
        "uomTypeElementId" : "", // (Optional) Unit Type Element id.
        "value" : "" // (Optional) The Domain Element Value is speci
      },
      ...
    ], // (Optional) The elements of the domain. Only applicable to
    "booleanYes" : {
      "name" : "", // The name of the boolean Yes.
      "nameTranslations" : {
        "key" : ""
      } // (Optional) The name translations by language of the boole
    }, // (Optional) The boolean Yes element of the domain. Only app
    "booleanNo" : {
      "name" : "", // The name of the boolean No.
      "nameTranslations" : {
        "key" : ""
      } // (Optional) The name translations by language of the boole
    }, // (Optional) The boolean No element of the domain. Only appl
    "floatRange" : {
      "min" : 1.0, // The minimum float value.
      "max" : 1.0 // The maximum float value.
    }, // (Optional) The range of the float domain. Only applicable
    "integerRange" : {
      "min" : 1, // The minimum integer value.
      "max" : 1 // The maximum integer value.
    }, // (Optional) The range of the integer domain. Only applicabl
    "uomItem" : "", // (Optional) The unit of measure id. Optional,
    "uomDomain" : "", // (Optional) The unit domain id. Optional, on
    "appendUnitSymbol" : false, // (Optional) If set to true then th
    In this case the domain elements do not contain the symbol. Default:
    "valueType" : "", // (Optional) Defines if the domain elements h
    Empty by default (domain elements do not have connected values). Ava

```

```
Float = floating point decimal values, Fixed = fixed point decimal v
    "decimals" : 1 // (Optional) The number of decimals when valueTy
Not defined if valueType is not Fixed.
    } // The domain.
}
```

PutDomainOutput

A put domain output.

```
JSON XML
{
  "domainNamedReference" : {
    "id" : "", // The domain resource id.
    "name" : "" // The domain name.
  } // The domain named reference.
}
```

Global Feature

DeleteGlobalFeatureListInput

A delete global feature list input.

```
JSON XML
{
  "featureNamedReferenceList" : [
    {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    },
    ...
  ] // The feature named reference list.
}
```

DeleteGlobalFeatureListOutput

A delete global feature output.

JSON XML

```
{
  "featureNamedReferenceList" : [
    {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    },
    ...
  ] // The feature named reference list.
}
```

DeleteGlobalFeatureOutput

A delete global feature output.

JSON XML

```
{
  "featureNamedReference" : {
    "id" : "", // The feature resource id.
    "name" : "" // The feature name.
  } // The feature named reference.
}
```

GetGlobalFeatureListOutput

A get feature resource list output.

```
{
  "featureResourceList" : [
    {
      "featureReference" : {
        "id" : "" // The feature resource id.
      }, // The feature reference.
      "feature" : {
        "parentModuleNamedReference" : {
          "id" : "", // The parent module resource id.
          "name" : "" // The parent module name.
        }, // (Optional) The parent module. Not applicable for global
        "domainNamedReference" : {
          "id" : "", // The domain resource id.
          "name" : "" // The domain name.
        }, // The feature domain.
        "name" : "", // The feature name.
        "description" : "", // (Optional) The feature description.
        "initialValue" : "", // (Optional) The feature initial value
        "descriptionTranslations" : {
          "key" : ""
        } // (Optional) The feature description translations by lang
      } // The feature.
    },
    ...
  ] // The feature resource list.
}
```

GetGlobalFeatureOutput

A get global feature resource output.

JSON XML

```
{
  "featureResource" : {
    "featureReference" : {
      "id" : "" // The feature resource id.
    }, // The feature reference.
    "feature" : {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // (Optional) The parent module. Not applicable for global
      "domainNamedReference" : {
        "id" : "", // The domain resource id.
        "name" : "" // The domain name.
      }, // The feature domain.
      "name" : "", // The feature name.
      "description" : "", // (Optional) The feature description.
      "initialValue" : "", // (Optional) The feature initial value.
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The feature description translations by language
    } // The feature.
  } // The feature resource.
}
```

PostBomMetaDataPropertyInput

A post BoM Meta Data property input.

JSON XML

```
{
  "bomMetaDataProperty" : {
    "name" : "", // The property name.
    "propertyId" : "", // The property ID.
    "description" : "", // (Optional) The property description.
    "descriptionTranslations" : {
      "key" : ""
    }, // (Optional) The property description translations by language
    "type" : "" // The property type.
  } // The feature.
}
```

PostBomMetaDataPropertyOutput

A post BoM Meta Data property output.

JSON XML

```
{
  "bomMetaDataPropertyNamedReference" : {
    "id" : "", // The property resource ID.
    "name" : "" // The property name.
  } // The property named reference.
}
```

PostGlobalFeatureInput

A post global feature input.

JSON XML

```
{
  "feature" : {
    "parentModuleNamedReference" : {
      "id" : "", // The parent module resource id.
      "name" : "" // The parent module name.
    }, // (Optional) The parent module. Not applicable for global fe
    "domainNamedReference" : {
      "id" : "", // The domain resource id.
      "name" : "" // The domain name.
    }, // The feature domain.
    "name" : "", // The feature name.
    "description" : "", // (Optional) The feature description.
    "initialValue" : "", // (Optional) The feature initial value.
    "descriptionTranslations" : {
      "key" : ""
    } // (Optional) The feature description translations by language
  } // The feature.
}
```

PostGlobalFeatureListInput

A post global feature list input.

JSON XML

```
{
  "featureList" : [
    {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // (Optional) The parent module. Not applicable for global
      "domainNamedReference" : {
        "id" : "", // The domain resource id.
        "name" : "" // The domain name.
      }, // The feature domain.
      "name" : "", // The feature name.
      "description" : "", // (Optional) The feature description.
      "initialValue" : "", // (Optional) The feature initial value.
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The feature description translations by language
    },
    ...
  ], // The global feature list.
  "removeNotReplaced" : false // (Optional) Set to true to remove ex
}
```

PostGlobalFeatureListOutput

A post global feature list output.

JSON XML

```
{
  "featureNamedReferenceList" : [
    {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    },
    ...
  ] // The global feature named reference list.
}
```

PostGlobalFeatureOutput

A post global feature output.

JSON XML


```
{
  "featureNamedReference" : {
    "id" : "", // The feature resource id.
    "name" : "" // The feature name.
  } // The feature named reference.
}
```

PutGlobalFeatureInput

A put global feature input.

JSON XML

```
{
  "feature" : {
    "parentModuleNamedReference" : {
      "id" : "", // The parent module resource id.
      "name" : "" // The parent module name.
    }, // (Optional) The parent module. Not applicable for global fe
    "domainNamedReference" : {
      "id" : "", // The domain resource id.
      "name" : "" // The domain name.
    }, // The feature domain.
    "name" : "", // The feature name.
    "description" : "", // (Optional) The feature description.
    "initialValue" : "", // (Optional) The feature initial value.
    "descriptionTranslations" : {
      "key" : ""
    } // (Optional) The feature description translations by language
  } // The feature.
}
```



PutGlobalFeatureOutput

A put global feature output.

JSON XML

```
{
  "featureNamedReference" : {
    "id" : "", // The feature resource id.
    "name" : "" // The feature name.
  } // The feature named reference.
}
```

Module

DeleteModuleListInput

A delete module list input.

```
JSON XML
{
  "moduleNamedReferenceList" : [
    {
      "id" : "", // The module resource id.
      "name" : "" // The module name.
    },
    ...
  ] // The module named reference list.
}
```

DeleteModuleListOutput

A delete module output.

```
JSON XML
{
  "moduleNamedReferenceList" : [
    {
      "id" : "", // The module resource id.
      "name" : "" // The module name.
    },
    ...
  ] // The module named reference list.
}
```

DeleteModuleOutput

A delete module output.

```
JSON XML
{
  "moduleNamedReference" : {
    "id" : "", // The module resource id.
    "name" : "" // The module name.
  } // The module named reference.
}
```

GetModuleListOutput

A get module resource list output. Note that each of the related items is listed separately with a parent reference pointing to the module.

```

{
  "moduleResourceList" : [
    {
      "moduleReference" : {
        "id" : "" // The module resource id.
      }, // The module reference.
      "module" : {
        "name" : "", // The module name.
        "description" : "", // (Optional) The module description.
        "serviceModule" : false, // (Optional) The flag designating
        "descriptionTranslations" : {
          "key" : ""
        } // (Optional) The module description translations by langu
      } // The module.
    },
    ...
  ], // The module resource list.
  "featureResourceList" : [
    {
      "featureReference" : {
        "id" : "" // The feature resource id.
      }, // The feature reference.
      "feature" : {
        "parentModuleNamedReference" : {
          "id" : "", // The parent module resource id.
          "name" : "" // The parent module name.
        }, // (Optional) The parent module. Not applicable for globa
        "domainNamedReference" : {
          "id" : "", // The domain resource id.
          "name" : "" // The domain name.
        }, // The feature domain.
        "name" : "", // The feature name.
        "description" : "", // (Optional) The feature description.
        "initialValue" : "", // (Optional) The feature initial value
        "descriptionTranslations" : {
          "key" : ""
        } // (Optional) The feature description translations by lang
      } // The feature.
    },
    ...
  ], // The module feature resource list for listed modules.
  "variantResourceList" : [
    {
      "variantReference" : {
        "id" : "" // The variant resource id.
      }, // The variant reference.
      "variant" : {

```

```

"parentAssemblyNamedReference" : {
  "id" : "", // The parent assembly resource id.
  "name" : "" // The parent assembly name.
}, // (Optional) The parent assembly. Required for assembly
"parentModuleNamedReference" : {
  "id" : "", // The parent module resource id.
  "name" : "" // The parent module name.
}, // (Optional) The parent module. Required for module vari
"name" : "", // The variant name.
"description" : "", // (Optional) The variant description.
"descriptionTranslations" : {
  "key" : ""
}, // (Optional) The variant description translations by lan
"status" : "", // The variant status.
"longDescription" : "", // (Optional) The variant long descr
"longDescriptionTranslations" : {
  "key" : ""
}, // (Optional) The variant long description translations b
"image" : "", // (Optional) The variant image.
"document" : "", // (Optional) The variant document.
"variantValueList" : [
  {
    "attributeNamedReference" : {
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    }, // The variant assembly attribute named reference. Re
    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The variant feature named reference. Required for
    "value" : "" // The variant value. Multiple values must
  },
  ...
], // (Optional) The variant value list. The list should con
"variantPropertyValueList" : [
  {
    "propertyNamedReference" : {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    }, // The property reference
    "value" : "", // The property value.
    "valueTranslations" : {
      "key" : ""
    } // (Optional) The property value translations by langu
  },
  ...
], // (Optional) The variant BoM Meta Data property values l
"variantPartList" : [

```

```
    {
      "partNamedReference" : {
        "id" : "", // The part resource id. Note: id and name
        "name" : "" // The part name.
      }, // The part named reference.
      "qty" : 1.0 // The part quantity.
    },
    ...
  ] // (Optional) The variant part list.
} // The variant.
},
...
] // The variant resource list for the listed modules.
}
```

GetModuleOutput

A get module resource output.

```

{
  "moduleResource" : {
    "moduleReference" : {
      "id" : "" // The module resource id.
    }, // The module reference.
    "module" : {
      "name" : "", // The module name.
      "description" : "", // (Optional) The module description.
      "serviceModule" : false, // (Optional) The flag designating th
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The module description translations by languag
    } // The module.
  }, // The module resource.
  "featureResourceList" : [
    {
      "featureReference" : {
        "id" : "" // The feature resource id.
      }, // The feature reference.
      "feature" : {
        "parentModuleNamedReference" : {
          "id" : "", // The parent module resource id.
          "name" : "" // The parent module name.
        }, // (Optional) The parent module. Not applicable for globa
        "domainNamedReference" : {
          "id" : "", // The domain resource id.
          "name" : "" // The domain name.
        }, // The feature domain.
        "name" : "", // The feature name.
        "description" : "", // (Optional) The feature description.
        "initialValue" : "", // (Optional) The feature initial value
        "descriptionTranslations" : {
          "key" : ""
        } // (Optional) The feature description translations by lang
      } // The feature.
    },
    ...
  ], // The feature resource list for the module resource.
  "variantResourceList" : [
    {
      "variantReference" : {
        "id" : "" // The variant resource id.
      }, // The variant reference.
      "variant" : {
        "parentAssemblyNamedReference" : {
          "id" : "", // The parent assembly resource id.
          "name" : "" // The parent assembly name.
        }
      }
    }
  ]
}

```

```

}, // (Optional) The parent assembly. Required for assembly
"parentModuleNamedReference" : {
  "id" : "", // The parent module resource id.
  "name" : "" // The parent module name.
}, // (Optional) The parent module. Required for module vari
"name" : "", // The variant name.
"description" : "", // (Optional) The variant description.
"descriptionTranslations" : {
  "key" : ""
}, // (Optional) The variant description translations by lan
"status" : "", // The variant status.
"longDescription" : "", // (Optional) The variant long descr
"longDescriptionTranslations" : {
  "key" : ""
}, // (Optional) The variant long description translations b
"image" : "", // (Optional) The variant image.
"document" : "", // (Optional) The variant document.
"variantValueList" : [
  {
    "attributeNamedReference" : {
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    }, // The variant assembly attribute named reference. Re
    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The variant feature named reference. Required for
    "value" : "" // The variant value. Multiple values must
  },
  ...
], // (Optional) The variant value list. The list should con
"variantPropertyValueList" : [
  {
    "propertyNamedReference" : {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    }, // The property reference
    "value" : "", // The property value.
    "valueTranslations" : {
      "key" : ""
    } // (Optional) The property value translations by langu
  },
  ...
], // (Optional) The variant BoM Meta Data property values l
"variantPartList" : [
  {
    "partNamedReference" : {
      "id" : "", // The part resource id. Note: id and name

```

```
        "name" : "" // The part name.
      }, // The part named reference.
      "qty" : 1.0 // The part quantity.
    },
    ...
  ] // (Optional) The variant part list.
} // The variant.
},
...
] // The variant resource list for the module resource.
}
```

PostModuleInput

A post module input.

```

{
  "module" : {
    "name" : "", // The module name.
    "description" : "", // (Optional) The module description.
    "serviceModule" : false, // (Optional) The flag designating the
    "descriptionTranslations" : {
      "key" : ""
    } // (Optional) The module description translations by language.
  }, // The module.
  "featureList" : [
    {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // (Optional) The parent module. Not applicable for global
      "domainNamedReference" : {
        "id" : "", // The domain resource id.
        "name" : "" // The domain name.
      }, // The feature domain.
      "name" : "", // The feature name.
      "description" : "", // (Optional) The feature description.
      "initialValue" : "", // (Optional) The feature initial value.
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The feature description translations by langua
    },
    ...
  ], // The featue list for module.
  "variantList" : [
    {
      "parentAssemblyNamedReference" : {
        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
      }, // (Optional) The parent assembly. Required for assembly va
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // (Optional) The parent module. Required for module varian
      "name" : "", // The variant name.
      "description" : "", // (Optional) The variant description.
      "descriptionTranslations" : {
        "key" : ""
      }, // (Optional) The variant description translations by langu
      "status" : "", // The variant status.
      "longDescription" : "", // (Optional) The variant long descrip
      "longDescriptionTranslations" : {
        "key" : ""
      }
    }
  ]
}

```

```

}, // (Optional) The variant long description translations by
"image" : "", // (Optional) The variant image.
"document" : "", // (Optional) The variant document.
"variantValueList" : [
  {
    "attributeNamedReference" : {
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    }, // The variant assembly attribute named reference. Requ
    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The variant feature named reference. Required for fe
    "value" : "" // The variant value. Multiple values must be
  },
  ...
], // (Optional) The variant value list. The list should conta
"variantPropertyValueList" : [
  {
    "propertyNamedReference" : {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    }, // The property reference
    "value" : "", // The property value.
    "valueTranslations" : {
      "key" : ""
    } // (Optional) The property value translations by languag
  },
  ...
], // (Optional) The variant BoM Meta Data property values lis
"variantPartList" : [
  {
    "partNamedReference" : {
      "id" : "", // The part resource id. Note: id and name ca
      "name" : "" // The part name.
    }, // The part named reference.
    "qty" : 1.0 // The part quantity.
  },
  ...
] // (Optional) The variant part list.
},
...
] // The variant list for module.
}

```

PostModuleListInput

A post module list input. Note that each of the related items is listed separately with a parent reference pointing to the module.

```
{
  "moduleList" : [
    {
      "name" : "", // The module name.
      "description" : "", // (Optional) The module description.
      "serviceModule" : false, // (Optional) The flag designating th
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The module description translations by languag
    },
    ...
  ], // The module list.
  "featureList" : [
    {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // (Optional) The parent module. Not applicable for global
      "domainNamedReference" : {
        "id" : "", // The domain resource id.
        "name" : "" // The domain name.
      }, // The feature domain.
      "name" : "", // The feature name.
      "description" : "", // (Optional) The feature description.
      "initialValue" : "", // (Optional) The feature initial value.
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The feature description translations by langua
    },
    ...
  ], // The featue list for listed modules.
  "variantList" : [
    {
      "parentAssemblyNamedReference" : {
        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
      }, // (Optional) The parent assembly. Required for assembly va
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // (Optional) The parent module. Required for module varian
      "name" : "", // The variant name.
      "description" : "", // (Optional) The variant description.
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The variant description translations by langu
      "status" : "", // The variant status.
    }
  ]
}
```

```

"longDescription" : "", // (Optional) The variant long descrip
"longDescriptionTranslations" : {
  "key" : ""
}, // (Optional) The variant long description translations by
"image" : "", // (Optional) The variant image.
"document" : "", // (Optional) The variant document.
"variantValueList" : [
  {
    "attributeNamedReference" : {
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    }, // The variant assembly attribute named reference. Requ
    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The variant feature named reference. Required for fe
    "value" : "" // The variant value. Multiple values must be
  },
  ...
], // (Optional) The variant value list. The list should conta
"variantPropertyValueList" : [
  {
    "propertyNamedReference" : {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    }, // The property reference
    "value" : "", // The property value.
    "valueTranslations" : {
      "key" : ""
    } // (Optional) The property value translations by languag
  },
  ...
], // (Optional) The variant BoM Meta Data property values lis
"variantPartList" : [
  {
    "partNamedReference" : {
      "id" : "", // The part resource id. Note: id and name ca
      "name" : "" // The part name.
    }, // The part named reference.
    "qty" : 1.0 // The part quantity.
  },
  ...
] // (Optional) The variant part list.
},
...
], // The variant list for listed modules.

```

```
"removeNotReplaced" : false // (Optional) Set to true to remove ex
}
```

PostModuleListOutput

A post module list output. Note that each of the related items is listed separately with a parent reference pointing to the module.

```
JSON XML
{
  "moduleNamedReferenceList" : [
    {
      "id" : "", // The module resource id.
      "name" : "" // The module name.
    },
    ...
  ], // The module named reference list.
  "moduleFeatureNamedReferenceList" : [
    {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // The parent module named reference.
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    },
    ...
  ], // The module feature named reference list.
  "moduleVariantNamedReferenceList" : [
    {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // The parent module named reference.
      "id" : "", // The variant resource id.
      "name" : "" // The variant name.
    },
    ...
  ] // The module variant named reference list.
}
```

PostModuleOutput

A post module output.

```
{
  "moduleNameReference" : {
    "id" : "", // The module resource id.
    "name" : "" // The module name.
  }, // The module named reference.
  "moduleFeatureNamedReferenceList" : [
    {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // The parent module named reference.
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    },
    ...
  ], // The module feature named reference list.
  "moduleVariantNamedReferenceList" : [
    {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // The parent module named reference.
      "id" : "", // The variant resource id.
      "name" : "" // The variant name.
    },
    ...
  ] // The module variant named reference list.
}
```

PutModuleInput

A put module input.

```

{
  "module" : {
    "name" : "", // The module name.
    "description" : "", // (Optional) The module description.
    "serviceModule" : false, // (Optional) The flag designating the
    "descriptionTranslations" : {
      "key" : ""
    } // (Optional) The module description translations by language.
  }, // The module.
  "featureList" : [
    {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // (Optional) The parent module. Not applicable for global
      "domainNamedReference" : {
        "id" : "", // The domain resource id.
        "name" : "" // The domain name.
      }, // The feature domain.
      "name" : "", // The feature name.
      "description" : "", // (Optional) The feature description.
      "initialValue" : "", // (Optional) The feature initial value.
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The feature description translations by langua
    },
    ...
  ], // The featue list for module.
  "variantList" : [
    {
      "parentAssemblyNamedReference" : {
        "id" : "", // The parent assembly resource id.
        "name" : "" // The parent assembly name.
      }, // (Optional) The parent assembly. Required for assembly va
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // (Optional) The parent module. Required for module varian
      "name" : "", // The variant name.
      "description" : "", // (Optional) The variant description.
      "descriptionTranslations" : {
        "key" : ""
      }, // (Optional) The variant description translations by langu
      "status" : "", // The variant status.
      "longDescription" : "", // (Optional) The variant long descrip
      "longDescriptionTranslations" : {
        "key" : ""
      }
    }
  ]
}

```

```

}, // (Optional) The variant long description translations by
"image" : "", // (Optional) The variant image.
"document" : "", // (Optional) The variant document.
"variantValueList" : [
  {
    "attributeNamedReference" : {
      "id" : "", // The attribute resource id.
      "name" : "" // The attribute name.
    }, // The variant assembly attribute named reference. Requ
    "featureNamedReference" : {
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    }, // The variant feature named reference. Required for fe
    "value" : "" // The variant value. Multiple values must be
  },
  ...
], // (Optional) The variant value list. The list should conta
"variantPropertyValueList" : [
  {
    "propertyNamedReference" : {
      "id" : "", // The property resource ID.
      "name" : "" // The property name.
    }, // The property reference
    "value" : "", // The property value.
    "valueTranslations" : {
      "key" : ""
    } // (Optional) The property value translations by languag
  },
  ...
], // (Optional) The variant BoM Meta Data property values lis
"variantPartList" : [
  {
    "partNamedReference" : {
      "id" : "", // The part resource id. Note: id and name ca
      "name" : "" // The part name.
    }, // The part named reference.
    "qty" : 1.0 // The part quantity.
  },
  ...
] // (Optional) The variant part list.
},
...
] // The variant list for module.
}

```

PutModuleOutput

A put module output.

JSON XML

```
{
  "moduleNameReference" : {
    "id" : "", // The module resource id.
    "name" : "" // The module name.
  }, // The module named reference.
  "moduleFeatureNamedReferenceList" : [
    {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // The parent module named reference.
      "id" : "", // The feature resource id.
      "name" : "" // The feature name.
    },
    ...
  ], // The module feature named reference list.
  "moduleVariantNamedReferenceList" : [
    {
      "parentModuleNamedReference" : {
        "id" : "", // The parent module resource id.
        "name" : "" // The parent module name.
      }, // The parent module named reference.
      "id" : "", // The variant resource id.
      "name" : "" // The variant name.
    },
    ...
  ] // The module variant named reference list.
}
```

Module Variant Images

GetVariantImagePropertyListOutput

A list BoM Meta Data image property list output.

JSON XML

```
{
  "variantPropertyValueList" : [
    {
      "propertyNamedReference" : {
        "id" : "", // The property resource ID.
        "name" : "" // The property name.
      }, // The property reference
      "value" : "", // The property value.
      "valueTranslations" : {
        "key" : ""
      } // (Optional) The property value translations by language.
    },
    ...
  ] // variant property value list
}
```

Part

DeletePartListInput

A delete part list input.

JSON XML

```
{
  "partNamedReferenceList" : [
    {
      "id" : "", // The part resource id. Note: id and name cannot b
      "name" : "" // The part name.
    },
    ...
  ] // The part named reference list.
}
```

DeletePartListOutput

A delete part output.

```
JSON XML
{
  "partNamedReferenceList" : [
    {
      "id" : "", // The part resource id. Note: id and name cannot b
      "name" : "" // The part name.
    },
    ...
  ] // The part named reference list.
}
```

DeletePartOutput

A delete part output.

```
JSON XML
{
  "partNamedReference" : {
    "id" : "", // The part resource id. Note: id and name cannot be
    "name" : "" // The part name.
  } // The part named reference.
}
```

GetPartListOutput

A get part resource list output.

```
{
  "partResourceList" : [
    {
      "partReference" : {
        "id" : "" // The part resource id. Note: id and name cannot
      }, // The part reference.
      "part" : {
        "name" : "", // The part name.
        "description" : "", // (Optional) The part description.
        "descriptionTranslations" : {
          "key" : ""
        }, // (Optional) The part description translations by language
        "status" : "", // The part status.
        "image" : "", // (Optional) The part image.
        "partValueList" : [
          {
            "partFeatureNamedReference" : {
              "id" : "", // The partFeature resource id.
              "name" : "" // The part feature name.
            }, // The part feature named reference.
            "value" : "" // The part value.
          },
          ...
        ] // The part value list.
      } // The part.
    },
    ...
  ] // The part resource list.
}
```

GetPartOutput

A get part resource output.

```
{
  "partResource" : {
    "partReference" : {
      "id" : "" // The part resource id. Note: id and name cannot be
    }, // The part reference.
    "part" : {
      "name" : "", // The part name.
      "description" : "", // (Optional) The part description.
      "descriptionTranslations" : {
        "key" : ""
      }, // (Optional) The part description translations by language
      "status" : "", // The part status.
      "image" : "", // (Optional) The part image.
      "partValueList" : [
        {
          "partFeatureNamedReference" : {
            "id" : "", // The partFeature resource id.
            "name" : "" // The part feature name.
          }, // The part feature named reference.
          "value" : "" // The part value.
        },
        ...
      ] // The part value list.
    } // The part.
  } // The part resource.
}
```

PostPartInput

A post part input.

```
{
  "part" : {
    "name" : "", // The part name.
    "description" : "", // (Optional) The part description.
    "descriptionTranslations" : {
      "key" : ""
    }, // (Optional) The part description translations by language.
    "status" : "", // The part status.
    "image" : "", // (Optional) The part image.
    "partValueList" : [
      {
        "partFeatureNamedReference" : {
          "id" : "", // The partFeature resource id.
          "name" : "" // The part feature name.
        }, // The part feature named reference.
        "value" : "" // The part value.
      },
      ...
    ] // The part value list.
  } // The part.
}
```

PostPartListInput

A post part list input. Note that each of the related items is listed separately with a parent reference pointing to the part.

JSON XML

```
{
  "partList" : [
    {
      "name" : "", // The part name.
      "description" : "", // (Optional) The part description.
      "descriptionTranslations" : {
        "key" : ""
      }, // (Optional) The part description translations by language
      "status" : "", // The part status.
      "image" : "", // (Optional) The part image.
      "partValueList" : [
        {
          "partFeatureNamedReference" : {
            "id" : "", // The partFeature resource id.
            "name" : "" // The part feature name.
          }, // The part feature named reference.
          "value" : "" // The part value.
        },
        ...
      ] // The part value list.
    },
    ...
  ], // The part list.
  "removeNotReplaced" : false // (Optional) Set to true to remove ex
}
```

PostPartListOutput

A post part list output.

JSON XML

```
{
  "partNamedReferenceList" : [
    {
      "id" : "", // The part resource id. Note: id and name cannot b
      "name" : "" // The part name.
    },
    ...
  ] // The part named reference list.
}
```

PostPartOutput

A post part output.

```
JSON XML
{
  "partNamedReference" : {
    "id" : "", // The part resource id. Note: id and name cannot be
    "name" : "" // The part name.
  } // The part named reference.
}
```

PutPartInput

A put part input.

```
JSON XML
{
  "part" : {
    "name" : "", // The part name.
    "description" : "", // (Optional) The part description.
    "descriptionTranslations" : {
      "key" : ""
    }, // (Optional) The part description translations by language.
    "status" : "", // The part status.
    "image" : "", // (Optional) The part image.
    "partValueList" : [
      {
        "partFeatureNamedReference" : {
          "id" : "", // The partFeature resource id.
          "name" : "" // The part feature name.
        }, // The part feature named reference.
        "value" : "" // The part value.
      },
      ...
    ] // The part value list.
  } // The part.
}
```

PutPartOutput

A put part output.

JSON XML

```
{
  "partNamedReference" : {
    "id" : "", // The part resource id. Note: id and name cannot be
    "name" : "" // The part name.
  } // The part named reference.
}
```

Part Feature

DeletePartFeatureListInput

A delete part feature list input.

JSON XML

```
{
  "partFeatureNamedReferenceList" : [
    {
      "id" : "", // The partFeature resource id.
      "name" : "" // The part feature name.
    },
    ...
  ] // The part feature named reference list.
}
```

DeletePartFeatureListOutput

A delete part feature output.

JSON XML

```
{
  "partFeatureNamedReferenceList" : [
    {
      "id" : "", // The partFeature resource id.
      "name" : "" // The part feature name.
    },
    ...
  ] // The part feature named reference list.
}
```

DeletePartFeatureOutput

A delete part feature output.

JSON XML


```
{
  "partFeatureNamedReference" : {
    "id" : "", // The partFeature resource id.
    "name" : "" // The part feature name.
  } // The part feature named reference.
}
```

GetPartFeatureListOutput

A get part feature resource list output.

JSON XML

```
{
  "partFeatureResourceList" : [
    {
      "partFeatureReference" : {
        "id" : "" // The part feature resource id.
      }, // The part feature reference.
      "partFeature" : {
        "domainNamedReference" : {
          "id" : "", // The domain resource id.
          "name" : "" // The domain name.
        }, // The part feature domain.
        "name" : "", // The part feature name.
        "description" : "", // (Optional) The part feature descripti
        "descriptionTranslations" : {
          "key" : ""
        } // (Optional) The part feature description translations by
      } // The part feature.
    },
    ...
  ] // The part feature resource list.
}
```



GetPartFeatureOutput

A get part feature resource output.

JSON XML

```
{
  "partFeatureResource" : {
    "partFeatureReference" : {
      "id" : "" // The part feature resource id.
    }, // The part feature reference.
    "partFeature" : {
      "domainNamedReference" : {
        "id" : "", // The domain resource id.
        "name" : "" // The domain name.
      }, // The part feature domain.
      "name" : "", // The part feature name.
      "description" : "", // (Optional) The part feature description
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The part feature description translations by lan
    } // The part feature.
  } // The part feature resource.
}
```

PostPartFeatureInput

A post part feature input.

JSON XML

```
{
  "partFeature" : {
    "domainNamedReference" : {
      "id" : "", // The domain resource id.
      "name" : "" // The domain name.
    }, // The part feature domain.
    "name" : "", // The part feature name.
    "description" : "", // (Optional) The part feature description.
    "descriptionTranslations" : {
      "key" : ""
    } // (Optional) The part feature description translations by lan
  } // The part feature.
}
```

PostPartFeatureListInput

A post part feature list input.

JSON XML

```
{
  "partFeatureList" : [
    {
      "domainNamedReference" : {
        "id" : "", // The domain resource id.
        "name" : "" // The domain name.
      }, // The part feature domain.
      "name" : "", // The part feature name.
      "description" : "", // (Optional) The part feature description
      "descriptionTranslations" : {
        "key" : ""
      } // (Optional) The part feature description translations by l
    },
    ...
  ], // The part feature list.
  "removeNotReplaced" : false // (Optional) Set to true to remove ex
}
```

PostPartFeatureListOutput

A post part feature list output.

JSON XML

```
{
  "partFeatureNamedReferenceList" : [
    {
      "id" : "", // The partFeature resource id.
      "name" : "" // The part feature name.
    },
    ...
  ] // The part feature named reference list.
}
```

PostPartFeatureOutput

A post part feature output.

JSON XML


```
{
  "partFeatureNamedReference" : {
    "id" : "", // The partFeature resource id.
    "name" : "" // The part feature name.
  } // The part feature named reference.
}
```

PutPartFeatureInput

A put part feature input.

JSON XML

```
{
  "partFeature" : {
    "domainNamedReference" : {
      "id" : "", // The domain resource id.
      "name" : "" // The domain name.
    }, // The part feature domain.
    "name" : "", // The part feature name.
    "description" : "", // (Optional) The part feature description.
    "descriptionTranslations" : {
      "key" : ""
    } // (Optional) The part feature description translations by lan
  } // The part feature.
}
```



PutPartFeatureOutput

A put part feature output.

JSON XML

```
{
  "partFeatureNamedReference" : {
    "id" : "", // The partFeature resource id.
    "name" : "" // The part feature name.
  } // The part feature named reference.
}
```