

# Transforming Common resources Fields with the API

Last Modified on 08/30/2021 1:03 pm EDT

A transformation is the result of the process of mapping fields in your connector instance resources to existing fields and objects in a common resource (common resource). After you create a common resource, you will select a connector instance, choose the resource containing the objects that you want to map to the common resource, and map fields to the common resource. The end result is a transformation of the selected objects in the connector instance.

You can test the APIs described in this section using our interactive documentation. Open SAP Open Connectors, and then click **API Docs** in the header.

On this page

## Retrieve a List of Common resources

You can retrieve a list of the common resources in your account. The list includes all fields defined in the common resource.

To retrieve a list of common resources:

1. Call the following:

```
GET /accounts/objects/definitions
```

2. Continue to the next step: [map fields to create a transformation](#).

## cURL Example

```
curl -X GET \  
  https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/accounts/objects/definitions \  
  -H 'authorization: User {USER_SECRET}, Organization {ORGANIZATION_SECRET}' \  
  -H 'content-type: application/json'
```

## Map Fields to Create a Default Transformation

SAP Open Connectors provides several APIs to map resources. This section describes mapping fields for an account-level default transformation. The result is a default transformation for all instances of a specific connector.

To map fields:

1. Construct a JSON body as shown below. For descriptions of each parameter, see [Transformation JSON Parameters](#).

```

{
  "level": "account",
  "vendorName": "",
  "fields": [
    {
      "path": "",
      "type": "",
      "vendorPath": "",
      "vendorType": ""
    }
  ]
}


```

2. Call the following, including the JSON body from the previous step:

```
POST /accounts/elements/{keyOrId}/transformations/{objectName}
```

**Note:** Replace {keyOrId} with the connector key or id and replace {objectName} with the name of the common resource.

## Transformation JSON Parameters

Parameter	Description	Required (Y/N)
level	The access level of the transformation, either <code>account</code> or <code>instance</code> .	N. Default depends on endpoint.
vendorName	The name of the resource that contains the fields that you want to map to the common resource.	N
fields	An object containing the field names and data types of the common resource and the vendor resource. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> <p> <b>Tip:</b> To get a list of fields in a resource, call <code>GET hubs/{hub}/objects/{RESOURCE}/metadata</code> .</p> </div>	N
path	The name of the field in the common resource.	Y
vendorPath	The name of the field in the vendor resource.	Y
type	The data type of the field in the common resource. Data types can be <code>boolean</code> , <code>string</code> , <code>date</code> , and <code>number</code> .	N
vendorType	The data type of the field at the vendor. Unless the format is <code>date</code> , you do not need to include this parameter. If the format is <code>date</code> , also include a mask.	N

## cURL Example

```

curl -X POST \
  https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/accounts/elements/sfdc/transformati
nsformations/myContacts \
  -H 'authorization: User {USER_SECRET}, Organization {ORGANIZATION_SECRET}' \
  -H 'content-type: application/json' \
  -d '
{
  "level": "account",
  "vendorName": "Contact",
  "fields": [
    {
      "type": "string",
      "path": "FirstName",
      "vendorPath": "FirstName"
    },
    {
      "type": "string",
      "path": "id",
      "vendorPath": "Id"
    },
    {
      "type": "string",
      "path": "LastName",
      "vendorPath": "LastName"
    },
    {
      "type": "date",
      "path": "birthdate",
      "vendorPath": "Birthdate",
      "vendorType": "date",
      "configuration": [
        ]
      }
    ]
  }
]'

```

## Map Fields at the Instance Level

Using the SAP Open Connectors `instances` endpoint, you can map fields at the instance level. Mapping fields is a two-step process that includes creating an instance level resource, and then mapping fields to it.

To create an instance-level common resource and map fields to it:

1. Construct a JSON body for the instance level common resource as shown below (see [New VDR JSON Parameters](#)):

```

{
  "fields": [
    {
      "type": "",
      "path": ""
    }
  ]
}

```

2. Create the common resource. Make the following API call with the JSON body from the previous step, replacing `{id}` with the instance id, and replacing `{objectName}` with the name of the common resource:

```
POST /instances/{id}/objects/{objectName}/definitions
```

3. Construct a JSON body to map fields to the new common resource as shown below. For descriptions of each parameter, see [Transformation JSON Parameters](#).

```
{
  "level": "instance",
  "vendorName": "",
  "fields": [
    {
      "path": "",
      "type": "",
      "vendorPath": "",
      "vendorType": ""
    }
  ]
}
```

4. Map fields to the common resource. Call the following, including the JSON body from the previous step:

```
POST /instances/{id}/transformations/{objectName}
```

**Note:** Replace `{id}` with the instance id and replace `{objectName}` with the name of the common resource.

## cURL Example

### Step 1: Create the common resource

```
curl -X POST \
  https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/instances/{id}/objects/{objectName}/definitions \
  -H 'authorization: User {USER_SECRET}, Organization {ORGANIZATION_SECRET}' \
  -H 'content-type: application/json' \
  -d '{
    "fields": [
      {
        "type": "string",
        "path": "title"
      }
    ]
  }'
```

### Step 2: Map fields to the common resource

```

curl -X POST \
  https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/instances/{id}/transformations/{objectName} \
  -H 'authorization: User {USER_SECRET}, Organization {ORGANIZATION_SECRET}' \
  -H 'content-type: application/json' \
  -d '
{
  "vendorName": "Contact",
  "level": "instance",
  "fields": [
    {
      "path": "title",
      "type": "string",
      "vendorPath": "Title"
    }
  ]
}'

```

## Map Fields at the Account Level

Using the SAP Open Connectors `accounts` endpoint, you can map fields at the account level. Mapping fields is a two-step process that includes creating an account level resource, and then mapping fields to it.

To create an account-level common resource and map fields to it:

1. Construct a JSON body for the account level common resource as shown below (see [New VDR JSON Parameters](#)):

```

{
  "fields": [
    {
      "type": "",
      "path": ""
    }
  ]
}

```

2. Create the common resource. Make one of the following API calls with the JSON body from the previous step, replacing `{id}` with the account id, and replacing `{objectName}` with the name of the common resource:

```
POST /accounts/objects/{objectName}/definitions
```

**Note:** Use this API call to create a common resource at the default account level.

```
POST /accounts/{id}/objects/{objectName}/definitions
```

**Note:** Use this API call to specify an account by id.

3. Construct a JSON body to map fields to the new common resource as shown below. For descriptions of each parameter, see [Transformation JSON Parameters](#).

```

{
  "level": "instance",
  "vendorName": "",
  "fields": [
    {
      "path": "",
      "type": "",
      "vendorPath": "",
      "vendorType": ""
    }
  ]
}

```

4. Map fields to the common resource. Call the following, including the JSON body from the previous step:

```
POST /accounts/{id}/elements/{keyOrId}/transformations/{objectName}
```

**Note:** Replace `{id}` with the instance id, replace `{keyOrId}` with the connector key or id, and replace `{objectName}` with the name of the common resource.

## cURL Example

### Step 1: Create the common resource (default account)

```

curl -X POST \
  https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/accounts/objects/{objectName}/definitions \
  -H 'authorization: User , Organization ' \
  -H 'content-type: application/json' \
  -d '
{
  "fields": [
    {
      "type": "string",
      "path": "title"
    }
  ]
}'

```

### Step 1: Create the common resource (specific account)

```

curl -X POST \
  https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/accounts/{id}/objects/{objectName}/definitions \
  -H 'authorization: User , Organization ' \
  -H 'content-type: application/json' \
  -d '
{
  "fields": [
    {
      "type": "string",
      "path": "title"
    }
  ]
}'

```

### Step 2: Map fields to the common resource

```

curl -X POST \
  https://api.openconnectors.us2.ext.hana.ondemand.com/elements/api-v2/accounts/{id}/elements/{keyO
rIdtransformations}/{objectName} \
  -H 'authorization: User , Organization ' \
  -H 'content-type: application/json' \
  -d '{
    "vendorName": "Contact",
    "level": "account",
    "fields": [
      {
        "path": "title",
        "type": "string",
        "vendorPath": "Title"
      }
    ]
  }'

```

## Map Complex Objects

You can use regular expressions as values for the JSON body parameters.

Examples:

- Get the value of the name field from the Products array where id = 4.

```

{
  "fields": [
    {
      "path": "AppleIpadName",
      "vendorPath": "Products[id=4].name"
    }
  ]
}

```

- Get the value of the name field from the Products array where id = 2.

```

{
  "fields": [
    {
      "path": "AppleNewProductName",
      "vendorPath": "details.Products[?(@.id==2)].name"
    }
  ]
}

```

- Get the touchId value from the Products array with id=2 which is inside the features object.

```

{
  "fields": [
    {
      "path": "AppleTouchProductId",
      "vendorPath": "Products[?(@.features.touchId==true)].id"
    }
  ]
}

```

- Get the Id value of the Products array at index 0.

```
{
  "fields": [
    {
      "path": "AppleProductId",
      "vendorPath": "Products[0].id"
    }
  ]
}
```

- Get a count of the Products array.

```
{
  "fields": [
    {
      "path": "AppleProductsCount",
      "vendorPath": "Products[*].size()"
    }
  ]
}
```

---