



ELO Sync

API operations



Table of contents

approvaldetails	6
Get Endpoint - GetApprovalDetails	6
Get Endpoint - GetApprovalDetailsById	8
approvals	10
Get Endpoint - GetApprovals	10
Get Endpoint - GetApprovalById	12
Update Endpoint - UpdateApproval	14
Delete Endpoint - DeleteApprovalById	16
azureappnotification	17
Get Endpoint - AzureAdminNotificationCallback	17
configuration	18
Post Endpoint - ReloadConfig	18
eloidentities	20
Get Endpoint - QueryElIdentities	20
Get Endpoint - GetIdentityById	22
elosynctargets	24
Get Endpoint - GetEloSyncTargets	24
Get Endpoint - GetEloSyncTargetById	26
Update Endpoint - UpdateEloSyncTargetById	28
license	30
Get Endpoint - CheckEula	30
Get Endpoint - DownloadLocaleEulaAsPdf	31
Get Endpoint - GetLicense	32
login	33
Get Endpoint - LoginWithOpenIdConnect	33
Get Endpoint - OpenIdConnectCallback	34
Post Endpoint - Logout	35
Get Endpoint - IsAuthenticated	36
Get Endpoint - GetCurrentUserId	37
Get Endpoint - GetCurrentUserInfo	38
msalactivities	39
Get Endpoint - GetMsalActivities	39
Get Endpoint - GetMsalActivitiesById	41

onedrivesyncjobs	43
Post Endpoint - RunOneDriveSyncJob	43
Post Endpoint - CancelOneDriveSyncJob	44
onedrivesynctargets	45
Get Endpoint - GetOneDriveSyncTargets	45
Get Endpoint - GetOneDriveSyncTargetById	47
Update Endpoint - UpdateOneDriveSyncTargetById	49
programinfo	51
Get Endpoint - GetProgramInfo	51
Get Endpoint - GetBaseUrl	53
Get Endpoint - GetStatus	54
repositorieslist	55
Get Endpoint - GetRepositories	55
repositorycontent	56
Get Endpoint - GetFolderChildrenById	56
Get Endpoint - GetDocumentMasksForRepository	57
Get Endpoint - GetMaskNameById	58
Get Endpoint - GetPreviewUrlForSord	59
Get Endpoint - GetUserProfile	60
Get Endpoint - DownloadFileTypeIcon	62
Get Endpoint - DownloadSordTypeIcon	63
settings	64
Get Endpoint - GetUserSettingsById	64
Get Endpoint - GetGlobalSettings	65
Get Endpoint - GetJobSettingsById	66
Delete Endpoint - DeleteSettingById	67
Update Endpoint - UpdateSettingById	68
Post Endpoint - CreateSetting	70
sharepointsyncjobs	71
Post Endpoint - RunSharePointSyncJob	71
Post Endpoint - CancelSharepointSyncJob	72
sharepointsynctargets	73
Get Endpoint - GetSharePointSyncTargets	73
Get Endpoint - GetSharePointSyncTargetById	75
Update Endpoint - UpdateSharePointSyncTargetById	77

synccollections	79
Get Endpoint - GetSyncCollections	79
Get Endpoint - GetSyncCollectionById	80
syncconflictdetails	81
Get Endpoint - GetConflictDetails	81
Get Endpoint - GetConflictDetailsById	82
syncconflicts	83
Get Endpoint - GetConflicts	83
Get Endpoint - DeleteConflictById	85
Update Endpoint - UpdateConflictById	87
Delete Endpoint - GetConflictById	90
synchronizationactivities	92
Get Endpoint - GetActivities	92
Get Endpoint - GetActivityById	93
Get Endpoint - GetLatestActivity	94
synchronizationactivitydetails	95
Get Endpoint - GetActivityDetails	95
Get Endpoint - GetActivityDetailById	96
syncitems	97
Get Endpoint - GetSynchronizedItems	97
Get Endpoint - GetSynchronizedSystems	99
Get Endpoint - GetRelatedItemsById	101
syncjobs	102
Get Endpoint - GetSyncJobs	102
Post Endpoint - CreateSyncJob	104
Get Endpoint - GetSyncJobById	106
Update Endpoint - UpdateSyncJobById	108
Delete Endpoint - DeleteSyncJobById	110
Post Endpoint - EnableSyncJobById	112
Post Endpoint - DisableSyncJobById	114
Post Endpoint - RunSyncJob	116
Post Endpoint - CancelSyncJob	118
Get Endpoint - GetSharePointChildrenById	120
Post Endpoint - GetSharepointItemByUrl	122
Get Endpoint - GetSharePointPreviewUrlById	124
Get Endpoint - GetOneDrivePreviewUrlById	125
Get Endpoint - GetMyOneDrive	126
Get Endpoint - GetOneDriveFoldersById	127
Get Endpoint - GetGroupDrivesById	128

Get Endpoint - GetOneDriveUserGroups	129
Get Endpoint - GetDriveOfSharePointSiteById	130
syncjobsdocmasks	131
Get Endpoint - GetEloDocumentMasksForJobs	131
Get Endpoint - GetEloDocumentMaskById	132
Update Endpoint - UpdateEloDocumentMaskById	133
syncjobsfoldermasks	135
Get Endpoint - GetEloFolderMasksForJobs	135
Get Endpoint - GetEloFolderMaskById	136
Update Endpoint - UpdateEloFolderMaskById	137
syncjobsmasks	139
Get Endpoint - GetJobMasks	139
Get Endpoint - GetJobMaskById	141
Update Endpoint - UpdateJobMaskById	143
syncjobtriggerrelations	145
Get Endpoint - GetSyncJobTriggerRelations	145
Get Endpoint - GetSyncJobTriggerRelationById	146
timetrigger	147
Get Endpoint - GetTimeTriggers	147
Get Endpoint - GetTimeTriggerById	149
Update Endpoint - UpdateTimeTriggerById	150
trigger	152
Get Endpoint - GetTriggers	152
Get Endpoint - GetTriggerById	154

approvaldetails

Get Endpoint- GetApprovalDetails

Returns list of approval details.

HTTP Request

One of the following requests should be made:

```
GET /odata/approvaldetails
GET /odata/approvaldetails/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of approval details was successful.	array ApprovalDetailEntity
-----	---	----------------------------

400	If provided request data is invalid.
-----	--------------------------------------

500	When an server error occurs during the load of approval details.
-----	--

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/approvaldetails?repository=repol&$filter=approvalId eq 1234
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "value": [
    {
      "detailId": 123,
      "key": "key1",
      "value": "lorem ipsum"
    },
    {
      "detailId": 124,
      "key": "key2",
      "value": "dolor sit"
    }
  ],
  "@odata.nextLink": "..."
}
```

Get Endpoint - GetApprovalDetailsById

Returns approval detail with the provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/approvaldetails({key})
GET /odata/approvaldetails/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of synchronization detail that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of approval detail was successful.
 400 If provided request data is invalid.
 500 When an internal server occurs during the load of synchronization detail.

Type

array
 ApprovalDetailEntity

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/approvaldetails(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{  
  "detailId": 1,  
  "key": "SomeKey",  
  "value": "Lorem Ipsum Dolor Sit Amet"  
}
```

approvals

Get Endpoint - GetApprovals

Returns list of approvals.

HTTP Request

One of the following requests should be made:

```
GET /odata/approvals
GET /odata/approvals/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of approvals was successful.	array ApprovalEntity
-----	--	----------------------

400	If provided request data is invalid.
-----	--------------------------------------

500	When an server error occurs during the load of approvals.
-----	---

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/approvals?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "value": [
    {
      "approvalId": 123,
      "itemName": "Presentation.pptx",
      "itemId": "cc3ca239-730d-4f0f-8120-3dcfd0063931"
    },
    {
      "approvalId": 124,
      "itemName": "Important Document.pdf",
      "itemId": "e913acef-5c87-428b-8e3a-623c7964aa4e"
    }
  ],
  "@odata.nextLink": "...
}
```

Get Endpoint - GetApprovalById

Returns approval with the provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/approvals({key})
GET /odata/approvals/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of approval that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of approval was successful.

400 If provided request data is invalid.

500 When an internal server occurs during the load of approval.

Type

array ApprovalEntity

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/approvals(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "approvalId": 123,
  "itemName": "Important Document.pdf",
  "itemId": "b99a0cb6-f538-422d-a296-744882848f7d",
  "kind": "documentSizeExceeded"
}
```

Update Endpoint - UpdateApproval

Updates approval metadata with the provided data from HTTPS PUT call. Metadata must be passed in as json object in HTTP body. Not all approval property values can be changed by update, some of them are immutable. In that case bad request will be returned back.

HTTP Request

One of the following requests should be made:

```
PUT /odata/approvals({key})
PUT /odata/approvals/{key}
```

Path parameters

Name Description

key required. Id of approval that should be updated.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Content-Type required. application/json

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code	Description	Type
200	If update of approval was successful.	ApprovalEntity
400	If update contains data for properties which values are immutable or if no data for the update could be loaded from the request.	
401	Is access is denied.	
404	If no approval with the given id could be found.	
500	When an internal server occurs during the update of approval.	

Delete Endpoint - DeleteApprovalById

Deletes approval with the provided id.

HTTP Request

One of the following requests should be made:

```
DELETE /odata/approvals({key})
DELETE /odata/approvals/{key}
```

Path parameters

Name Description

key required. Id of approval that should be deleted.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 With value true if deletion was successful otherwise false.

401 Is access is denied.

404 If no approval with the given id could be found.

500 When an internal server occurs during the deletion of approval.

Type

boolean

azureappnotification

Get Endpoint - AzureAdminNotificationCallback

Handles events of changed app subscription data.

HTTP Request

One of the following requests should be made:

```
GET /elosync/azure/apps/changes
```

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	OK	
-----	----	--

configuration

Post Endpoint - ReloadConfig

Reloads the ELO Sync config to apply changes in the configuration.

HTTP Request

One of the following requests should be made:

```
POST /api/reloadconfig
```

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Responses

Code	Description	Type
------	-------------	------

200	'restartInProgress' if a restart is required after the config reload, otherwise 'noRestart'.	string
-----	--	--------

401	If access is not allowed with the given credentials/data.	
-----	---	--

500	While loading value an internal server error has occurred.	
-----	--	--

Example

Request

The following is an example of how to call the endpoint:

```
POST /api/reloadconfig
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

200 ok
noRestart

eloidentities

Get Endpoint - QueryElIdentities

Loads list of specified ELO identities(user/groups) from provided repository.

HTTP Request

One of the following requests should be made:

```
GET /odata/eloidentities
GET /odata/eloidentities/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
repository	optional
query	optional
includeUsers	optional
includeGroups	optional

Request headers

Name	Description
	Authorization required. Bearer <i>{token}</i> . See authorization

Responses

Code	Description	Type
200	Loading ELO identities was successfully.	array ElIdentityEntity
400	Provided request data is invalid.	
401	Loading ELO identity has failed. Caller has not necessary rights to load ELO identities.	
500	When an server error occurs during the load of identities.	

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/eloidentities?repository=repol&query=John&includeGroups=false
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "value": [
    {
      "id": "e61ff758-cfaf-4a0b-9bdc-1228ebc4edbc",
      "name": "JohnDoe",
      "isAdministrator": false
    },
    {
      "id": "73aeebb1-9e0b-4dfe-90bb-efcc674de2e3",
      "name": "JohnFoo",
      "isAdministrator": true
    }
  ],
  "@odata.nextLink": "...
}
```

Get Endpoint - GetIdentityById

Loads infos about specified ELO user from provided repository.

HTTP Request

One of the following requests should be made:

```
GET /odata/eloidentities/elosync.userorgroup(id={id},repository={repository})
GET /odata/eloidentities/userorgroup(id={id},repository={repository})
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name	Description
id	required
repository	required

Request headers

Name	Description
	Authorization required. Bearer <i>{token}</i> . See authorization

Responses

Code	Description	Type
200	Loading ELO identity was successfully.	array ElIdentityEntity
400	Provided request data is invalid.	
401	Loading ELO identity has failed. Caller has not necessary rights to load elo identities.	
500	When an server error occurs during the load of identities.	

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/eloidentities/userorgroup?repository=repo1&id=1234
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "value": [
    {
      "id": "36b2a0f8-3f0e-4c1c-8001-0d92b236fcf0",
      "idAsNumber": 1234,
      "name": "JohnDoe",
      "isAdministrator": false
    }
  ],
  "@odata.nextLink": "..."
}
```

elosynctargets

Get Endpoint - GetEloSyncTargets

Returns list of ELO sync targets.

HTTP Request

One of the following requests should be made:

```
GET /odata/elosynctargets
GET /odata/elosynctargets/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
------	-------------	------

200	Array of ELO sync targets.	array
-----	----------------------------	-------

400	If provided request data is invalid.	
-----	--------------------------------------	--

500	When an server error occurs during the load of elo sync targets.	EloSystemSyncTargetEntity
-----	--	---------------------------

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/elosynctargets?repository=rep01
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#EloSyncTargets",
  "value": [
    {
      "eloSystemSyncTargetId": 1,
      "folderName": "OneDrive",
      "folderId": "14513"
    },
    {
      "eloSystemSyncTargetId": 2,
      "folderName": "DriveMe",
      "folderId": "14535"
    }
  ],
  "@odata.nextLink": "Link for the next 100 items"
}
```

Get Endpoint - GetEloSyncTargetById

Returns ELO sync target with the provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/elosynctargets({key})
GET /odata/elosynctargets/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of ELO sync target that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 The requested ELO sync target.
400 If provided request data is invalid.
500 When an server error occurs during the load of elo sync target.

Type

array
EloSystemSyncTargetEntity

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/elosynctargets(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#SyncJobsFolderMasks",
  "value": [
    {
      "eloSystemSyncTargetId": 2,
      "folderName": "DriveMe",
      "folderId": "14535"
    }
  ]
}
```

Update Endpoint - UpdateEloSyncTargetById

Updates ELO sync target with the provided id.

HTTP Request

One of the following requests should be made:

```
PUT /odata/elosynctargets({key})  
PUT /odata/elosynctargets/{key}
```

Path parameters

Name Description

key required. Id of sync job elo sync target that should be updated.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Content-Type required. application/json

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code Description

200 Returns the updated sync job elo sync target.

Type

EloSystemSyncTargetEntity

Code	Description	Type
400	If update contains data for properties which values are immutable, if no data for the update could be loaded from the request or provided request data is invalid.	
500	When an server error occurs during the update of sync job elo sync target.	

Example

Request

The following is an example of how to call the endpoint:

```
PUT /odata/elosynctargets(1)?repository=rep01
Content-Type: application/json

{
  "eloSystemSyncTargetId": 2,
  "folderName": "DriveMe",
  "folderId": "14535"
}
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "eloSystemSyncTargetId": 2,
  "folderName": "DriveMe",
  "folderId": "14535"
}
```

license

Get Endpoint - CheckEula

Loads a value that indicates whether the user has agreed to the ELO Sync EULA or not.

HTTP Request

One of the following requests should be made:

```
GET /api/license/check-eula
```

Query parameters

Name	Description
------	-------------

repository optional.	Destination repository key.
----------------------	-----------------------------

Request headers

Name	Description
------	-------------

Authorization required.	Bearer <i>{token}</i> . See authorization
-------------------------	---

Responses

Code	Description	Type
200	True if user has agreed to ELO Sync EULA otherwise false.	boolean
404	No such repository has been found.	
500	While loading value an internal server error has occurred.	

Get Endpoint - DownloadLocaleEulaAsPdf

Loads ELO Sync EULA as pdf document.

HTTP Request

One of the following requests should be made:

```
GET /api/license/eula/{culture}
```

Path parameters

Name	Description
------	-------------

culture	required. Culture to use. For example: de-DE
---------	--

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
------	-------------	------

200	ELO Sync EULA as pdf document in provided culture.	
-----	--	--

400	No culture provided.	
-----	----------------------	--

404	No EULA file found.	
-----	---------------------	--

500	While loading EULA an internal server error has occurred.	
-----	---	--

Get Endpoint - GetLicense

Fetch the licensing information for a repository. With this endpoint the current licensing information can be read and then evaluated.

HTTP Request

One of the following requests should be made:

```
GET /api/license
```

Query parameters

Name	Description
------	-------------

repository	optional. The ELO repository for which the license information should be returned
------------	---

Request headers

Name	Description
------	-------------

Authorization	required Bearer <i>{token}</i> . See authorization
---------------	--

Responses

Code	Description	Type
200	The license information	LicenseInfo
404	No such repository has been found	
500	An unknown error occurred	

login

Get Endpoint - LoginWithOpenIdConnect

Starts login of ELO Sync user by OpenID Connect on ELO Sync.

HTTP Request

One of the following requests should be made:

```
GET /api/auth/login/openidconnect
```

Query parameters

Name	Description
culture	required. Culture to use. For example: de-DE
redirectUri	required. Path that should be called after successful login
repository	required. Destination repository key.

Request headers

Name	Description
Authorization	required Bearer <i>{token}</i> . See authorization

Responses

Code	Description	Type
200	Return a nex challenge containing the redirectUri	ChallengeResult
302	Redirected to the redirectUri.	
500	Issued on an internal server error.	

Get Endpoint - OpenIdConnectCallback

Handles forwarded logged-in users via OpenID connect on Microsoft.

HTTP Request

One of the following requests should be made:

```
GET /auth/login/signedin-oidconnect
```

Query parameters

Name	Description
------	-------------

culture	required. Culture to use for login on ix server.
redirectUri	required. Path that should be called after successful login
repository	required. Destination repository key.

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
200	If user is successfully logged in, it will be redirected to / path.	
400	Method call with invalid data.	
401	User/service calling method is not authenticated or given repository does not exists.	
500	Issued on an internal server error.	

Post Endpoint - Logout

logs out ELO Sync user from provided repository.

HTTP Request

One of the following requests should be made:

```
POST /api/auth/logout
```

Query parameters

Name	Description
------	-------------

repository	required. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
Content-Type	required. application/json

Responses

Code	Description	Type
200	Returns value indicating if logout is done successfully. Therefor returned value must be inspected.	boolean
401	User/service calling method is not authenticated or given repository could not be found.	
500	An internal server error has occurred while logout of ELO Sync delegated user.	

Get Endpoint - IsAuthenticated

Detects if user calling the endpoint is authenticated or not.

HTTP Request

One of the following requests should be made:

```
GET /api/auth/user/authenticated
```

Query parameters

Name	Description
------	-------------

	repository required. Destination repository key.
--	--

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	User/service ist authenticated.	
-----	---------------------------------	--

401	User/service is not authenticated.	
-----	------------------------------------	--

Get Endpoint - GetCurrentUserid

Retrieves the ELO Sync user ID of the REST API endpoint caller.

HTTP Request

One of the following requests should be made:

```
GET /api/auth/user/id
```

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
200	ELO Sync user id.	string
401	User/service is not authenticated.	
500	An internal server error has occurred while retrieving the user ID.	

Get Endpoint - GetCurrentUserInfo

Loads information about the ELO Sync user calling the REST API endpoint.

HTTP Request

One of the following requests should be made:

```
GET /api/auth/user/info
```

Query parameters

Name	Description
------	-------------

repository	required. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
200	Info about the logged in elo sync user.	LoggedInUserInfoDto
401	Loading user info by given credentials is not allowed.	
500	Loading info failed due to an internal server error.	

msalactivities

Get Endpoint - GetMsalActivities

Loads all MSAL activities.

HTTP Request

One of the following requests should be made:

```
GET /odata/msalactivities
GET /odata/msalactivities/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository optional.	Destination repository key.
----------------------	-----------------------------

Request headers

Name	Description
------	-------------

Authorization required.	Bearer <i>{token}</i> . See authorization
-------------------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of MSAL activities was successful.	array
-----	--	-------

400	If provided request data is invalid.	
-----	--------------------------------------	--

500	If while loading MSAL activities an error has occurred on the server side.	MsalAccountActivityEntity
-----	--	---------------------------

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/msalactivities?repository=rep01
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#MsalActivities",
  "value": [
    {
      "msalAccountActivityEntityId": 1,
      "accountIdentifier": true,
      "accountObjectId": false
      //....
    },
    //...
  ],
  "@odata.nextLink": "Link for the next 100 items"
}
```

Get Endpoint - GetMsalActivitiesById

Loads all MSAL activities.

HTTP Request

One of the following requests should be made:

```
GET /odata/msalactivities({key})
GET /odata/msalactivities/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of MSAL activities was successful.
 400 If provided request data is invalid.
 500 If while loading MSAL activities an error has occurred on the server side.

Type

array
 MsalAccountActivityEntity

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/msalactivities(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#MsalActivities",
  "value": [
    {
      "msalAccountActivityEntityId": 1,
      "accountIdentifier": true,
      "accountObjectId": false
      //....
    }
  ]
}
```

onedrivesyncjobs

Post Endpoint - RunOneDriveSyncJob

Starts a synchronization job with OneDrive

HTTP Request

One of the following requests should be made:

```
POST /api/onedrive/syncjob/run
```

Query parameters

Name	Description
------	-------------

syncJobId	required. ID of sync job which have to synchronized.
-----------	--

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Responses

Code	Description	Type
------	-------------	------

200	Starting onedrive sync job synchronization done successfully.	
-----	---	--

401	If access is not allowed with the given credentials/data.	
-----	---	--

500	Starting onedrive sync job synchronization failed due to an internal server error.	
-----	--	--

Post Endpoint - CancelOneDriveSyncJob

Cancels the current execution of a sync job. With this endpoint the execution of OneDrive sync jobs can be canceled. The cancellation will not take effect immediately as the remaining in-flight data is processed.

HTTP Request

One of the following requests should be made:

```
POST /api/onedrive/syncjob/cancel
```

Query parameters

Name	Description
------	-------------

syncJobId	required. ID of sync job which should be canceled.
-----------	--

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Responses

Code	Description	Type
202	The execution of the sync job has been canceled	
400	The format of the sync job id is invalid	
404	The sync job does not exist or is currently not running	
500	An error occurred while canceling a sync job.	

onedrivesynctargets

Get Endpoint - GetOneDriveSyncTargets

Returns list of OneDrive sync targets.

HTTP Request

One of the following requests should be made:

```
GET /odata/onedrivesynctargets
GET /odata/onedrivesynctargets/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required Bearer <i>{token}</i> . See authorization
---------------	--

Responses

Code	Description	Type
------	-------------	------

200	If the load of OneDrive sync targets was successful.	array
-----	--	-------

400	If provided request data is invalid.	
-----	--------------------------------------	--

500	When an server error occurs during the load of sync jobs onedrive sync targets.	OneDriveSyncTargetEntity
-----	---	--------------------------

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/onedrivesynctargets?repository=rep01
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#OneDriveSyncTargets",
  "value": [
    {
      "eloSystemSyncTargetId": 1,
      "folderName": "OneDrive",
      "folderId": "14513"
    },
    {
      "eloSystemSyncTargetId": 2,
      "folderName": "DriveMe",
      "folderId": "14535"
    },
    ...
  ],
  "@odata.nextLink": "Link for the next 100 items"
}
```

Get Endpoint - GetOneDriveSyncTargetById

Returns OneDrive sync target with provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/onedrivesynctargets({key})
GET /odata/onedrivesynctargets/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of sync job onedrive sync target that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of OneDrive sync target was successful.

400 If provided request data is invalid.

500 When an server error occurs during the load of sync job onedrive synctarget.

Type

array
OneDriveSyncTargetEntity

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/onedrivesynctargets(1)?repository=rep01
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#OneDriveSyncTargets",
  "value": [
    {
      "eloSystemSyncTargetId": 2,
      "folderName": "DriveMe",
      "folderId": "14535"
    }
  ]
}
```

Update Endpoint - UpdateOneDriveSyncTargetById

Updates OneDrive sync target with provided data.

HTTP Request

One of the following requests should be made:

```
PUT /odata/onedrivesynctargets({key})  
PUT /odata/onedrivesynctargets/{key}
```

Path parameters

Name Description

key required. Id of OneDrive sync target that should be updated.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Content-Type required. application/json

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code Description

200 If the update of OneDrive sync target was successful.

Type

OneDriveSyncTargetEntity

Code	Description	Type
400	If update contains data for properties which values are immutable or if no data for the update could be loaded from the request.	
500	When an server error occurs during the update of OneDrive sync target.	

Example

Request

The following is an example of how to call the endpoint:

```
PUT /odata/onedrivesynctargets(1)?repository=repo1
Content-Type: application/json

{
  "thirdSystemSyncTargetId": "Id of the object whose data is being updated.",
  "siteName": "Name of the SharePoint site you want to set.",
  "siteId": "ID of the SharePoint site you want to set.",
  //...
}
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "thirdSystemSyncTargetId": 2,
  "folderName": "DriveMe",
  "folderId": "14535"
}
```

programinfo

Get Endpoint - GetProgramInfo

Loads infos about the program.

HTTP Request

One of the following requests should be made:

```
GET /api/programinfo
```

Query parameters

Name	Description
------	-------------

repository optional.	Destination repository key.
----------------------	-----------------------------

Request headers

Name	Description
------	-------------

Authorization required.	Bearer <i>{token}</i> . See authorization
-------------------------	---

Responses

Code	Description	Type
------	-------------	------

200	Loading ELO Sync program info was successful.	ProgramInfoEntity
-----	---	-------------------

400	Request contains invalid data.	
-----	--------------------------------	--

401	Logged in user is not authenticated or program info can not be loaded by given data.	
-----	--	--

500	When an server error occurs during the load of program info.	
-----	--	--

Example

Request

The following is an example of how to call the endpoint:

```
GET /api/programinfo?repository=rep01
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  'version': '23.04.007.000',
  'ixversion': '23.04.000.000',
  'issuer': 'ELO Digital Office GmbH',
  'copyright': 'Copyright © 2024 ELO Digital Office GmbH'
}
```

Get Endpoint - GetBaseUrl

Loads ELO Sync Base URL.

HTTP Request

One of the following requests should be made:

```
GET /api/baseurl
```

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
200	Returns Base URL of ELO Sync.	string

Get Endpoint - GetStatus

Loads ELO Sync Status.

HTTP Request

One of the following requests should be made:

```
GET /api/status
```

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
200	Returns Status of ELO Sync.	string

repositorieslist

Get Endpoint - GetRepositories

Loads list of repositories specified in ELO Sync.

HTTP Request

One of the following requests should be made:

```
GET /api/repositories
```

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	Returns list of repositories supported by ELO Sync.	array
-----	---	-------

repositorycontent

Get Endpoint - GetFolderChildrenById

Loads child elements of folder with provided id.

HTTP Request

One of the following requests should be made:

```
GET /api/repository/folder/childs
```

Query parameters

Name	Description
repository	optional. ELO repository from where folders should be loaded.
folderId	optional. ID of repository folder which child elements should be loaded.
includeDocuments	optional. True if documents should be included otherwise false.

Request headers

Name	Description
	Authorization required. Bearer <i>{token}</i> . See authorization

Responses

Code	Description	Type
200	List of child elements.	array RepositorySordTreeItemDto
401	Logged in user is not authenticated or childs can not be loaded by given data.	
500	Loading child elements has failed due to an internal server error.	

Get Endpoint - GetDocumentMasksForRepository

Loads list of available doc masks from provided repository.

HTTP Request

One of the following requests should be made:

```
GET /api/repository/docmasks
```

Query parameters

Name	Description
repository	optional. Repository from which doc masks should be loaded.
includeDocumentMasks	optional. True if document doc masks should be loaded otherwise false.
includeFolderMasks	optional. True if folder doc masks should be loaded otherwise false.

Request headers

Name	Description
	Authorization required. Bearer <i>{token}</i> . See authorization

Responses

Code	Description	Type
200	List of loaded doc masks.	array DocMaskDto
401	Logged in user is not authenticated or doc masks can not be loaded by given data.	
500	Loading doc masks has failed due to an internal server error.	

Get Endpoint - GetMaskNameById

Retrieves mask name for the provided mask with the maskId from the provided repository.

HTTP Request

One of the following requests should be made:

```
GET /api/repository/maskname
```

Query parameters

Name	Description
------	-------------

repository	optional. Repository from which mask name should be loaded.
------------	---

maskId	optional. Id of mask where name should be loaded.
--------	---

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	Name of mask.	string
-----	---------------	--------

401	Logged in user is not authenticated or mask name can not be loaded by given data.	
-----	---	--

404	The Mask from the provided ID could not be retrieved.	
-----	---	--

500	Loading mask name has failed due to an internal server error.	
-----	---	--

Get Endpoint - GetPreviewUrlForSord

Load preview url for provided ELO item.

HTTP Request

One of the following requests should be made:

```
GET /api/repository/item/preview/url
```

Query parameters

Name	Description
------	-------------

repository	optional. ELO repository that contains item for which preview url have to be loaded.
------------	--

itemGuid	optional. Id of repository item for that preview url should be loaded.
----------	--

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
200	Preview url of provided ELO repository item.	string
400	Request contains invalid data.	
401	Logged in user is not authenticated.	
404	Preview url of provided ELO repository item could not be loaded because destination repository could not be found.	
500	Loading preview url for repository item failed due to an internal server error.	

Get Endpoint - GetUserProfile

Loads profile picture of authenticated user.

HTTP Request

One of the following requests should be made:

```
GET /api/user/profilepicture
```

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
200	Profile Picture.	string
400	Request contains invalid data.	
401	Logged in user is not authenticated.	
404	If no profile picture could be found.	
500	Loading profile picture has failed due to an internal server error.	

Example

Request

The following is an example of how to call the endpoint:

```
GET /api/user/profilepicture?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

The response will be the actual image-file.

Get Endpoint - DownloadFileTypeIcon

Loads and returns icon back in icon/png format from provided ELO repository.

HTTP Request

One of the following requests should be made:

```
GET /api/repository/docext/icon
```

Query parameters

Name	Description
------	-------------

documentExt	optional. Document extension for that icon should be loaded.
-------------	--

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	Icon for the provided document extension could be successfully loaded.	string
-----	--	--------

400	Given parameters are invalid.	
-----	-------------------------------	--

401	Logged in user is not authenticated.	
-----	--------------------------------------	--

404	Icon for the given document extension could not be loaded from the specified repository because destination repository could not be found or no ix connection could be loaded.	
-----	--	--

500	Loading document icon failed due to an internal server error.	
-----	---	--

Get Endpoint - DownloadSordTypeIcon

Loads and returns icon back in icon/png format from provided ELO repository.

HTTP Request

One of the following requests should be made:

```
GET /api/repository/sordtype/icon
```

Query parameters

Name	Description
------	-------------

typeld	optional. SordType ID for that icon should be loaded.
--------	---

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	Icon for the provided sordtype could be loaded successfully.	string
-----	--	--------

400	Given parameters are invalid.	
-----	-------------------------------	--

401	Logged in user is not authenticated.	
-----	--------------------------------------	--

404	Icon for the given sordtype could not be loaded from the specified repository because destination repository could not be found or no ix connection could be loaded.	
-----	--	--

500	Loading sordtype icon failed due to an internal server error.	
-----	---	--

settings

Get Endpoint - GetUserSettingsById

Loads settings for a provided ELO identity (user/group). If no identity id is provided than settings of the caller will be loaded.

HTTP Request

One of the following requests should be made:

```
GET /odata/settings/elosync.userorgroup(repository={repository},userorgroupid={userOrGroupId}
GET /odata/settings/userorgroup(repository={repository},userorgroupid={userOrGroupId},categ
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name	Description
repository	required. Destination repository.
userOrGroupId	required. ELO identity id of user or group.
category	required. Category of settings which should be loaded.
setting	required. Name of setting that should be loaded.

Request headers

Name	Description
Authorization	required. Bearer <i>{token}</i> . See authorization

Responses

Code	Description	Type
200	Settings loaded successfully.	array SettingsEntity
400	Data provided by request is invalid.	
500	Internal server error while loading settings.	

Get Endpoint - GetGlobalSettings

Loads global settings for a provided repository.

HTTP Request

One of the following requests should be made:

```
GET /odata/settings/elosync.global(repository={repository},setting={setting})
GET /odata/settings/global(repository={repository},setting={setting})
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name	Description
------	-------------

repository	required. Destination repository.
------------	-----------------------------------

setting	required. Name of setting that should be loaded.
---------	--

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
200	Settings loaded successfully.	array SettingsEntity
400	Data provided by request is invalid.	
500	Internal server error while loading settings.	

Get Endpoint - GetJobSettingsById

Loads settings for sync job with provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/settings/elosync.syncjob(repository={repository},syncjobid={syncJobId},category=
GET /odata/settings/syncjob(repository={repository},syncjobid={syncJobId},category={category})
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name	Description
repository	required. Destination repository.
syncJobId	required. Id of sync job which settings should be loaded.
category	required. Category settings which could be loaded.
setting	required. Setting that should be loaded.

Request headers

Name	Description
Authorization	required Bearer {token}. See authorization

Responses

Code	Description	Type
200	Settings loaded successfully.	array SettingsEntity
400	Data provided by request is invalid.	
500	Internal server error while loading settings.	

Delete Endpoint -DeleteSettingById

Deletes setting with the provided id.

HTTP Request

One of the following requests should be made:

```
DELETE /odata/settings({key})
DELETE /odata/settings/{key}
```

Path parameters

Name Description

key required. Id of setting that should be deleted.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code	Description	Type
200	Setting successfully deleted.	
400	Data provided by request is invalid.	
404	Setting with the provided key/id not found.	
500	Internal server error while deleting setting.	

Update Endpoint - UpdateSettingById

Updates data of setting with provided id.

HTTP Request

One of the following requests should be made:

```
PUT /odata/settings({key})
PUT /odata/settings/{key}
```

Path parameters

Name Description

key required. Id of setting that have to be updated.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Content-Type required. application/json

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code Description

200 Setting data updated successfully.

Type

SettingsEntity

Code	Description	Type
400	Data provided by request is invalid.	
404	Setting data with the provided key/id not found.	
500	Internal server error while updating setting.	

Post Endpoint - CreateSetting

Creates new setting.

HTTP Request

One of the following requests should be made:

```
POST /odata/settings
```

Query parameters

Name	Description
------	-------------

repository optional.	Destination repository key.
----------------------	-----------------------------

Request headers

Name	Description
------	-------------

Authorization required. Bearer <i>{token}</i> .	See authorization
---	-------------------

Content-Type required.	application/json
------------------------	------------------

Request body

SettingsEntity

Responses

Code	Description	Type
200	Setting created successfully.	SettingsEntity
400	Request contains invalid data.	
500	Internal server error while creating setting.	

sharepointsyncjobs

Post Endpoint - RunSharePointSyncJob

Starts a synchronization job with SharePoint

HTTP Request

One of the following requests should be made:

```
POST /api/sh/syncjob/run
```

Query parameters

Name	Description
------	-------------

syncJobId	required. ID of sync job which have to synchronized.
-----------	--

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Responses

Code	Description	Type
------	-------------	------

200	Starting sharepoint sync job synchronization done successfully.	
-----	---	--

401	If access is not allowed with the given credentials/data.	
-----	---	--

500	Starting sharepoint sync job synchronization failed due to an internal server error.	
-----	--	--

Post Endpoint - CancelSharepointSyncJob

Cancels the current execution of a sync job,With this endpoint the execution of SharePoint sync jobs can be canceled. The cancellation will not take effect immediately as the remaining in-flight data is processed.

HTTP Request

One of the following requests should be made:

```
POST /api/sh/syncjob/cancel
```

Query parameters

Name	Description
------	-------------

syncJobId	required. ID of sync job which should be cancelled.
-----------	---

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Responses

Code	Description	Type
202	The execution of the sync job has been canceled	
400	The format of the sync job id is invalid	
404	The sync job does not exist	
500	An error occurred while canceling a sync job.	

sharepointsynctargets

Get Endpoint - GetSharePointSyncTargets

Returns list of SharePoint sync targets.

HTTP Request

One of the following requests should be made:

```
GET /odata/sharepointsynctargets
GET /odata/sharepointsynctargets/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of SharePoint sync targets was successful.	array
-----	--	-------

400	If provided request data is invalid.	
-----	--------------------------------------	--

500	When an server error occurs during the load of SharePoint sync targets.	SharepointSyncTargetEntity
-----	---	----------------------------

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/sharepointsynctargets?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#SharepointSyncTargets",
  "value": [
    {
      "eloSystemSyncTargetId": 1,
      "folderName": "OneDrive",
      "folderId": "14513"
    },
    {
      "eloSystemSyncTargetId": 2,
      "folderName": "DriveMe",
      "folderId": "14535"
    },
    //...
  ],
  "@odata.nextLink": "Link for the next 100 items"
}
```

Get Endpoint - GetSharePointSyncTargetById

Loads data of SharePoint sync target with provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/sharepointsynctargets({key})
GET /odata/sharepointsynctargets/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of SharePoint sync target that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of SharePoint sync target was successful.
 400 If provided request data is invalid.
 500 When an server error occurs during the load of SharePoint sync target.

Type

array
 SharepointSyncTargetEntity

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/sharepointsynctargets(1)?repository=rep01
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#SharepointSyncTargets",
  "value": [
    {
      "eloSystemSyncTargetId": 2,
      "folderName": "DriveMe",
      "folderId": "14535"
    }
  ]
}
```

Update Endpoint - UpdateSharePointSyncTargetById

Updates SharePoint sync target with provided id.

HTTP Request

One of the following requests should be made:

```
PUT /odata/sharepointsynctargets({key})
PUT /odata/sharepointsynctargets/{key}
```

Path parameters

Name Description

key required. Id of sync job sharepoint sync target that should be updated.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Content-Type required. application/json

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code Description

200 If the update of SharePoint sync target was successful.

Type

SharepointSyncTargetEntity

Code	Description	Type
400	If provided request data is invalid.	
500	When an server error occurs during the update of SharePoint sync target.	

Example

Request

The following is an example of how to call the endpoint:

```
PUT /odata/sharepointsynctargets(1)?repository=rep1
Content-Type: application/json

{
  "thirdSystemSyncTargetId": "Id des Objekts dessen Daten aktualisiert werden sollen.",
  "siteName": "Name der Sharepointseite, der gesetzt werden soll.",
  "siteId": "ID der Sharepointseite, die gesetzt werden soll.",
  //...
}
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "thirdSystemSyncTargetId": 2,
  "folderName": "DriveMe",
  "folderId": "14535"
}
```

synccollections

Get Endpoint - GetSyncCollections

Enumerates all synchronized collections within this service.

HTTP Request

One of the following requests should be made:

```
GET /odata/synccollections
GET /odata/synccollections/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Request headers

Name	Description
------	-------------

	Authorization required Bearer <i>{token}</i> . See authorization
--	--

Responses

Code	Description	Type
200	Returned if the information about the collections could be retrieved successfully	array SyncCollectionEntity
500	Returned if an error occurred while fetching information about the synchronized collections.	

Get Endpoint - GetSyncCollectionById

Returns information about a specific synchronized collection.

HTTP Request

One of the following requests should be made:

```
GET /odata/synccollections({key})
GET /odata/synccollections/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name	Description
------	-------------

key	required
-----	----------

Request headers

Name	Description
------	-------------

Authorization required Bearer <i>{token}</i> . See authorization
--

Responses

Code	Description	Type
------	-------------	------

200	Returned if the information about the collection could be retrieved successfully	array SyncCollectionEntity
500	Returned if an error occurred while fetching information about a collection.	

syncconflictdetails

Get Endpoint - GetConflictDetails

Returns list of all synchronization details.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncconflictdetails
GET /odata/syncconflictdetails/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository optional.	Destination repository key.
----------------------	-----------------------------

Request headers

Name	Description
------	-------------

Authorization required.	Bearer <i>{token}</i> . See authorization
-------------------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of synchronization details was successful.	array
-----	--	-------

500	When an server error occurs during the load of synchronization details.	SyncConflictDetailEntity
-----	---	--------------------------

Get Endpoint - GetConflictDetailsById

Returns synchronization detail with the given id.

HTTP Request

One of the following requests should be made:

```
GET /odata/synconflictetails({key})
GET /odata/synconflictetails/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of synchronization detail that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of synchronization detail was successful.

500 When an internal server occurs during the load of synchronization detail.

Type

array
SyncConflictDetailEntity

syncconflicts

Get Endpoint - GetConflicts

Returns list of synchronization conflicts.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncconflicts
GET /odata/syncconflicts/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of synchronization conflicts was successful.	array
-----	--	-------

400	Request contains invalid data.	
-----	--------------------------------	--

500	When an server error occurs during the load of synchronization conflicts.	SyncConflictEntity
-----	---	--------------------

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/syncconflicts?repository=repo1&$expand=details
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "value": [
    {
      "syncConflictId": 123,
      "title": "foo",
      "firstId": "2f968f3d-fccf-49a9-9963-1b4b77170139",
      "secondId": "3eb1350c-a8a0-4cb1-9071-d9942c5b8ad9",
      "details": [
        {
          "key": "key1",
          "value": "Lorem ipsum"
        }
      ]
    },
    {
      "syncConflictId": 456,
      "title": "bar",
      "firstId": "a3038606-3d67-42f3-8d21-3560cd87f3cd",
      "secondId": "32dd61c2-7dfd-4426-9744-4f8534d10451",
      "details": [
        {
          "key": "key1",
          "value": "Dolor Sit"
        }
      ]
    }
  ],
  "@odata.nextLink": "..."
}
```

Get Endpoint - -DeleteConflictById

Returns synchronization conflict with provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/synconflicts({key})
GET /odata/synconflicts/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of sharepoint sync target that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200	If the load of sharepoint sync target was successful.
400	Request contains invalid data.
500	When an server error occurs during the load of sharepoint synctarget.

Type

array
SyncConflictEntity

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/syncconflicts(1)?repository=repo1&$expand=details
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "syncConflictId": 1,
  "title": "Example",
  "firstId": "3143bd06-f34d-47cc-ac1d-448846a3ce01",
  "secondId": "3eb1350c-a8a0-4cb1-9071-d9942c5b8ad9",
  "details": [
    {
      "key": "key1",
      "value": "Lorem ipsum"
    }
  ]
}
```

Update Endpoint - UpdateConflictById

Updates data of the synchronization conflict with provided id.

HTTP Request

One of the following requests should be made:

```
PUT /odata/synconflicts({key})
PUT /odata/synconflicts/{key}
```

Path parameters

Name Description

key required. Id of synchronization conflict that should be updated.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Content-Type required. application/json

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code Description

200 If the update of synchronisation conflict was successful.

Type

SyncConflictEntity

Code Description**Type**

400	Request contains invalid data.
404	If no synchronisation conflict with the given id could be found.
500	When an internal server occurs during the update of synchronisation conflict.

Example**Request**

The following is an example of how to call the endpoint:

```
PUT /odata/synconflicts(1)?repository=repo1
Content-Type: application/json

{"syncConflictId": 1,
 "title": "Example",
 "firstId": "3143bd06-f34d-47cc-ac1d-448846a3ce01",
 "secondId": "3eb1350c-a8a0-4cb1-9071-d9942c5b8ad9",
 "details": [
   {
     "key": "key1",
     "value": "Lorem ipsum"
   }
 ]
}
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "syncConflictId": 1,
  "title": "Example",
  "firstId": "3143bd06-f34d-47cc-ac1d-448846a3ce01",
  "secondId": "3eb1350c-a8a0-4cb1-9071-d9942c5b8ad9",
  "details": [
    {
      "key": "key1",
      "value": "Lorem ipsum"
    }
  ]
}
```

```
]
}
```

Delete Endpoint - GetConflictById

Deletes synchronization conflict with provided id.

HTTP Request

One of the following requests should be made:

```
DELETE /odata/synconflcts({key})
DELETE /odata/synconflcts/{key}
```

Path parameters

Name Description

key required. Id of synchronization conflict that should be deleted.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 True if deletion was successful otherwise false.

400 Request contains invalid data.

404 If no synchronization conflict with the provided id could be found.

500 When an internal server occurs during the deletion of synchronization conflict.

Type

boolean

Example

Request

The following is an example of how to call the endpoint:

```
DELETE /odata/synconflcts(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
HTTP/1.1 200 OK  
Content-Type: application/json  
  
true
```

synchronizationactivities

Get Endpoint - GetActivities

Returns list of synchronization protocols.

HTTP Request

One of the following requests should be made:

```
GET /odata/synchronizationactivities
GET /odata/synchronizationactivities/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
repository optional	Destination repository key.

Request headers

Name	Description
Authorization required.	Bearer <i>{token}</i> . See authorization

Responses

Code	Description	Type
200	If the load of protocol was successful.	array
400	If provided request data is invalid.	SynchronizationActivityEntity
500	When an server error occurs during the load of protocols.	

Get Endpoint - GetActivityById

Returns synchronization protocol entry with the provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/synchronizationactivities({key})
GET /odata/synchronizationactivities/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of synchronization protocol activity that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of the protocol entry was successful.
 400 If provided request data is invalid.
 500 When an server error occurs during the load of entry.

Type

array SynchronizationActivityEntity

Get Endpoint - GetLatestActivity

Returns a last entry from synchronization protocol list.

HTTP Request

One of the following requests should be made:

```
GET /odata/synchronizationactivities/elosync.latest(syncjobid={syncJobId})
GET /odata/synchronizationactivities/latest(syncjobid={syncJobId})
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name	Description
syncJobId	required

Query parameters

Name	Description
repository	optional. Destination repository key.

Request headers

Name	Description
Authorization	required. Bearer {token}. See authorization

Responses

Code	Description	Type
200	If the load of the protocol entry was successful.	array SynchronizationActivityEntity
400	If provided request data is invalid.	
500	When an server error occurs during the load of protocols.	

synchronizationactivitydetails

Get Endpoint - GetActivityDetails

Returns list of synchronization protocol details.

HTTP Request

One of the following requests should be made:

```
GET /odata/synchronizationactivitydetails
GET /odata/synchronizationactivitydetails/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository optional.	Destination repository key.
----------------------	-----------------------------

Request headers

Name	Description
------	-------------

Authorization required.	Bearer <i>{token}</i> . See authorization
-------------------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of protocol details was successful.	array
-----	---	-------

400	If provided request data is invalid.	SynchronizationActivityDetailEntity
-----	--------------------------------------	-------------------------------------

500	When an server error occurs during the load of protocol details.	
-----	--	--

Get Endpoint - GetActivityDetailById

Returns synchronization protocol detail with the provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/synchronizationactivitydetails({key})
GET /odata/synchronizationactivitydetails/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of synchronization protocol detail that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of protocol detail was successful.
400 If provided request data is invalid.
500 When an server error occurs during the load of protocol detail.

Type

array
SynchronizationActivityDetailEntity

syncitems

Get Endpoint - GetSynchronizedItems

Returns all items that are synchronized.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncitems
GET /odata/syncitems/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	Returned if the information about the items could be retrieved successfully	array SyncItemEntity
500	Returned if an error occurred while fetching information about the items.	

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/syncitems?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
"@odata.context": "https://elosyncurl/odata/$metadata#SyncItems",
  "value": [
    {
      "id": "...",
      "systemId": "...",
      "systemItemId": "...",
      // ... more
    },
    // ... more
  ],
  "@odata.nextLink": "link for next page of items"
}
```

Get Endpoint - GetSynchronizedSystems

Returns a synchronized item of a synchronization job.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncitems({key})
GET /odata/syncitems/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200	Returned if the information about the system could be retrieved successfully
500	Returned if an error occurred while fetching information about a system.

Type

array
SyncItemEntity

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/syncitems(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#SyncItems",
  "value": [
    {
      "id": "...",
      "systemId": "...",
      "systemItemId": "...",
      // ... more
    }
  ]
}
```

Get Endpoint - GetRelatedItemsById

Returns all related items of a synchronized item.

HTTP Request

One of the following requests should be made:

```
GET /api/syncitems/{{key}}/related
```

Path parameters

Name	Description
------	-------------

key	required. The id of the synchronized item
-----	---

Request headers

Name	Description
------	-------------

Authorization required Bearer <i>{token}</i> . See authorization
--

Responses

Code	Description	Type
------	-------------	------

200	Returned if the information about the related items could be retrieved successfully	array SyncItemEntity
500	Returned if an error occurred while fetching information about the related items.	

syncjobs

Get Endpoint - GetSyncJobs

Returns list of sync jobs. OData format is supported. [See documentation on](#)

HTTP Request

One of the following requests should be made:

```
GET /odata/syncjobs
GET /odata/syncjobs/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of syncjobs was successful.	array SyncJobEntity
-----	---	---------------------

400	Request contains invalid data.	BadRequestODataResult
-----	--------------------------------	-----------------------

401	If access is not allowed with the given credentials/data.	
-----	---	--

500	When an server error occurs during the load of sync jobs.	
-----	---	--

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/syncjobs?repository=rep01
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "value": [
    {
      "syncJobId": 1,
      "name": "Job1"
    },
    {
      "syncJobId": 2,
      "name": "Job2"
    }
  ],
  "@odata.nextLink": "...
}
```

Post Endpoint - CreateSyncJob

Creates a new new sync job with provided data.

HTTP Request

One of the following requests should be made:

```
POST /odata/syncjobs
```

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
Content-Type	required. application/json

Request body

SyncJobEntity

Responses

Code	Description	Type
------	-------------	------

200	New sync job.	SyncJobEntity
-----	---------------	---------------

400	Request contains invalid data.	
-----	--------------------------------	--

401	If access is not allowed with the given credentials/data.	
-----	---	--

500	Creating new sync job failed due to an internal server error.	
-----	---	--

Example

Request

The following is an example of how to call the endpoint:

```
POST /odata/syncjobs?repository=rep01
Content-Type: application/json
```

```
{
```

```
"name": "foo",  
"repositoryUrl": "http://elosever:9090/ix-Repo/ix",  
"syncTargetSystem": "SharePoint"  
}
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
HTTP/1.1 201 Created  
Content-Type: application/json  
  
{  
  "syncJobId": 123,  
  "name": "foo",  
  "repositoryUrl": "http://elosever:9090/ix-Repo/ix",  
  "syncTargetSystem": "SharePoint"  
}
```

Get Endpoint - GetSyncJobById

Loads sync job with the provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncjobs({key})
GET /odata/syncjobs/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of syncjob that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If loading of sync job was successful.
 400 Request contains invalid data.
 401 If access is not allowed with the given credentials/data.
 500 When an internal server occurs during the load of the sync job.

Type

SyncJobEntity

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/syncjobs(1)?$select=name,repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{  
  "name": "Job name"  
}
```

Update Endpoint - UpdateSyncJobById

Updates a sync job

HTTP Request

One of the following requests should be made:

```
PUT /odata/syncjobs({key})
PUT /odata/syncjobs/{key}
```

Path parameters

Name	Description
------	-------------

key	required
-----	----------

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code	Description	Type
------	-------------	------

200	If update of syncjob was successful.	SyncJobEntity
-----	--------------------------------------	---------------

Code	Description	Type
400	If no syncjob data could be loaded from request for update.	
401	If access is not allowed with the given credentials/data.	
404	If no sync job with the given id could be found.	
500	When an internal server occurs during the update of sync job.	

Example

Request

The following is an example of how to call the endpoint:

```
PUT /odata/syncjobs(1)?repository=repo1
Content-Type: application/json

{
  "name": "New SyncJob Name"
}
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "syncJobId": 1,
  "name": "New SyncJob Name"
}
```

Delete Endpoint - DeleteSyncJobById

Deletes a sync job

HTTP Request

One of the following requests should be made:

```
DELETE /odata/syncjobs({key})
DELETE /odata/syncjobs/{key}
```

Path parameters

Name	Description
------	-------------

key	required
-----	----------

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	With value true if deletion was successful otherwise false.	boolean
400	Request contains invalid data.	
401	If access is not allowed with the given credentials/data.	
404	If no sync job with the given id could be found.	
500	When an internal server occurs during the deletion of sync job.	

Example

Request

The following is an example of how to call the endpoint:

```
DELETE /odata/syncjobs(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
HTTP/1.1 200 OK
```

Post Endpoint - EnableSyncJobById

Enables a sync job

HTTP Request

One of the following requests should be made:

```
POST /api/syncjobs/activate({syncJobId})
POST /api/syncjobs/activate/{syncJobId}
```

Path parameters

Name	Description
------	-------------

syncJobId	required. ID of sync job which have to be activated.
-----------	--

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Responses

Code	Description	Type
------	-------------	------

200	Sync job is successfully activated.	
-----	-------------------------------------	--

400	Request contains invalid data.	
-----	--------------------------------	--

401	If access is not allowed with the given credentials/data.	
-----	---	--

500	Activating sync job failed due to an internal server error.	
-----	---	--

Example

Request

The following is an example of how to call the endpoint:

```
POST /api/syncjobs/activate(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
HTTP/1.1 200 OK
```

Post Endpoint - DisableSyncJobById

Deactivates sync job with the provided id.

HTTP Request

One of the following requests should be made:

```
POST /api/syncjobs/deactivate({syncJobId})
POST /api/syncjobs/deactivate/{syncJobId}
```

Path parameters

Name	Description
------	-------------

syncJobId	required. ID of sync job which have to be deactivated.
-----------	--

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Responses

Code	Description	Type
------	-------------	------

200	Sync job is successfully deactivated.	
-----	---------------------------------------	--

400	Request contains invalid data.	
-----	--------------------------------	--

401	If access is not allowed with the given credentials/data.	
-----	---	--

500	Deactivating sync job failed due to an internal server error.	
-----	---	--

Example

Request

The following is an example of how to call the endpoint:

```
POST /api/syncjobs/deactivate(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
HTTP/1.1 200 OK
```

Post Endpoint - RunSyncJob

Starts execution of sync job with provided id.

HTTP Request

One of the following requests should be made:

```
POST /api/syncjobs/run({syncJobId})
POST /api/syncjobs/run/{syncJobId}
```

Path parameters

Name	Description
------	-------------

syncJobId	required. ID of sync job which have to synchronized.
-----------	--

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Responses

Code	Description	Type
------	-------------	------

200	Sync job synchronization started successfully.	
-----	--	--

400	Request contains invalid data.	
-----	--------------------------------	--

401	If access is not allowed with the given credentials/data.	
-----	---	--

500	Starting sync job synchronization failed due to an internal server error.	
-----	---	--

Example

Request

The following is an example of how to call the endpoint:

```
POST /api/syncjobs/run(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
HTTP/1.1 200 OK
```

Post Endpoint - CancelSyncJob

Cancels execution of sync job with provided id.

HTTP Request

One of the following requests should be made:

```
POST /api/syncjobs/cancel({syncJobId})
POST /api/syncjobs/cancel/{syncJobId}
```

Path parameters

Name	Description
------	-------------

syncJobId	required. ID of sync job which have to synchronized.
-----------	--

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Responses

Code	Description	Type
------	-------------	------

202	The cancellation has been accepted, but not yet executed.	
-----	---	--

401	If access is not allowed with the given credentials/data.	
-----	---	--

404	The sync job does not exist or is not being executed.	
-----	---	--

500	Cancellation of the sync job synchronization failed due to an internal server error.	
-----	--	--

Example

Request

The following is an example of how to call the endpoint:

```
POST /api/syncjobs/cancel(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
HTTP/1.1 200 OK
```

Get Endpoint - GetSharePointChildrenById

Fetches the child items for the given SharePoint element.

HTTP Request

One of the following requests should be made:

```
GET /api/syncjobs/sharepoint/subitems
```

Query parameters

Name	Description
shSiteId	optional. Site ID of the sharepoint item of which sub items have to be loaded. If omitted or null all root items will be loaded.
shListId	optional. ID of DocumentLibrary of the sharepoint item.
shFolderId	optional. ID of then Folder of the sharepoint item.
loadMoreToken	optional. Contains information to load more items.
searchWord	optional. Contains the text to search for.

Request headers

Name	Description
	Authorization required. Bearer <i>{token}</i> . See authorization

Responses

Code	Description	Type
200	List of sharepoint items.	SharepointItemResponse
400	Request contains invalid data.	
401	If access is not allowed with the given credentials/data.	
500	Loading sharepoint items failed due to an internal server error.	

Example

Request

The following is an example of how to call the endpoint:

```
GET /api/syncjobs/sharepoint/subitems?shSiteId=123456.sharepoint.com,12345678-1234-1234-123
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "itemList": [
    {
      "name": "List 1",
      "itemType": "List",
      "hasChilds": false,
      "siteId": "123456.sharepoint.com,12345678-1234-1234-1234-123456789000,00000000-1234-1234-1234-123456789000",
      "listId": "b2f13d23-e8fd-4301-8fa6-48d3db14b509"
    },
    {
      "name": "List 2",
      "itemType": "List",
      "hasChilds": true,
      "siteId": "123456.sharepoint.com,12345678-1234-1234-1234-123456789000,00000000-1234-1234-1234-123456789000",
      "listId": "280e9a97-f4d6-487f-bbe5-ae78f4d46803"
    },
    {
      "name": "SubSite 1",
      "itemType": "Site",
      "hasChilds": true,
      "siteId": "123456.sharepoint.com,1a06b175-668a-45ca-8d24-c51c198ce8b9,6bc6c886-c171-4567-890a-bcde-fghijklmnopqrstuv"
    }
  ],
  "loadMoreToken": "dGhpcyBpcyBhbiBleGFtcGxlIHRva2Vu"
}
```

Post Endpoint - GetSharepointItemByUrl

Resolves a sharepoint item from the given url.

HTTP Request

One of the following requests should be made:

```
POST /api/syncjobs/sharepoint/resolveitem
```

Request headers

Name	Description
------	-------------

Authorization required. Bearer *{token}*. See authorization

Content-Type required. application/json

Responses

Code	Description	Type
200	The result from the resolution process.	SharepointItemResponse
400	The given url is invalid.	
401	The user is not logged in.	

Example

Request

The following is an example of how to call the endpoint:

```
POST /api/syncjobs/sharepoint/resolveitem  
  
https://tenant.sharepoint.com/sites/sitename/Library123/Forms/AllItems.aspx
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "itemList": [
    {
      "name": "Library123",
      "itemType": "List",
      "hasChilds": false,
      "siteId": "123456.sharepoint.com,12345678-1234-1234-1234-123456789000,00000000-1234-1
      "listId": "b2f13d23-e8fd-4301-8fa6-48d3db14b509"
    },
  ]
}
```

Get Endpoint - GetSharePointPreviewUrlById

Loads preview URL for the provided SharePoint list collection item.

HTTP Request

One of the following requests should be made:

```
GET /api/syncjobs/sharepoint/listcollitem/preview/url
```

Query parameters

Name	Description
shSiteId	required. SharePoint site id to which the item belongs for which the preview url should be loaded.
shListCollId	required. SharePoint list collection id to which the item belongs for which the preview url should be loaded.
shListCollItemId	required. SharePoint list item id for which the preview url should be loaded.

Request headers

Name	Description
	Authorization required. Bearer <i>{token}</i> . See authorization

Responses

Code	Description	Type
200	SharePoint list collection item preview url loaded successfully.	string
400	Request contains invalid data.	
401	If access is not allowed with the given credentials/data.	

Get Endpoint - GetOneDrivePreviewUrlById

Loads preview URL for the provided OneDrive item.

HTTP Request

One of the following requests should be made:

```
GET /api/syncjobs/onedrive/item/preview/url
```

Query parameters

Name	Description
oneDriveId	required. OneDrive id to which the item belongs for that the preview url should be loaded.
oneDriveFolderId	optional. OneDrive id to which the item belongs for that the preview url should be loaded.
oneDriveItemId	required. OneDrive item id for which the preview url should be loaded.

Request headers

Name	Description
	Authorization required. Bearer <i>{token}</i> . See authorization

Responses

Code	Description	Type
200	OneDrive item preview url loaded successfully.	string
400	Request contains invalid data.	
401	If access is not allowed with the given credentials/data.	

Get Endpoint - GetMyOneDrive

It loads the user's OneDrive drives itself.

HTTP Request

One of the following requests should be made:

```
GET /api/syncjobs/onedrive/me/drives
```

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	List of user's OneDrive drives.	
-----	---------------------------------	--

401	If access is not allowed with the given credentials/data.	
-----	---	--

500	Loading OneDrive drives has failed due to an internal server error.	
-----	---	--

		array OneDriveItemDto
--	--	-----------------------

Get Endpoint - GetOneDriveFoldersById

Loads folders of OneDrive drive with provided id.

HTTP Request

One of the following requests should be made:

```
GET /api/syncjobs/onedrive/drives/{driveId}
GET /api/syncjobs/onedrive/drives/{driveId}
```

Path parameters

Name Description

driveId required

Query parameters

Name Description

oneDriveFolderId optional. OneDrive id to which the item belongs for that the preview url should be loaded.

Request headers

Name Description

Authorization required Bearer *{token}*. See authorization

Responses

Code Description

200 List of folders.

401 If access is not allowed with the given credentials/data.

500 Loading OneDrive drive folders has failed due to an internal server error.

Type

array

OneDriveItemDto

Get Endpoint - GetGroupDrivesById

Loads the group's OneDrives drives.

HTTP Request

One of the following requests should be made:

```
GET /api/syncjobs/onedrive/groups({groupId})/drives
GET /api/syncjobs/onedrive/groups/{groupId}/drives
```

Path parameters

Name	Description
------	-------------

groupId	required
---------	----------

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	List of OneDrive drives of provided group id.	array OneDriveItemDto
401	If access is not allowed with the given credentials/data.	
500	Loading group's OneDrive drives has failed due to an internal server error.	

Get Endpoint - GetOneDriveUserGroups

Loads OneDrive user groups.

HTTP Request

One of the following requests should be made:

```
GET /api/syncjobs/onedrive/groups
```

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	List of OneDrive user groups.	array
-----	-------------------------------	-------

401	If access is not allowed with the given credentials/data.	OneDriveItemDto
-----	---	-----------------

500	Loading OneDrive user groups has failed due to an internal server error.	
-----	--	--

Get Endpoint - GetDriveOfSharePointSiteById

Loads OneDrive drives of provided SharePoint site.

HTTP Request

One of the following requests should be made:

```
GET /api/syncjobs/sharepoint/site/drives({siteId})
GET /api/syncjobs/sharepoint/site/drives/{siteId}
```

Path parameters

Name	Description
------	-------------

siteId	required
--------	----------

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	List of OneDrive drives of provided SharePoint site.	array OneDriveItemDto
-----	--	--------------------------

401	If access is not allowed with the given credentials/data.
-----	---

500	Loading SharePoint site OneDrive drives has failed due to an internal server error.
-----	---

syncjobsdocmasks

Get Endpoint - GetEloDocumentMasksForJobs

Returns list of ELO document doc masks for specified sync jobs.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncjobsdocmasks
GET /odata/syncjobsdocmasks/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

	repository optional. Destination repository key.
--	--

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of the document masks was successful.	array
-----	---	-------

400	Request contains invalid data.	
-----	--------------------------------	--

500	When an server error occurs during the load of sync jobs ELO document masks.	MaskEntity
-----	--	------------

Get Endpoint - GetEloDocumentMaskById

Loads metadata of the document doc mask with the provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncjobsdocmasks({key})
GET /odata/syncjobsdocmasks/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of document mask that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200	If the load of sync job ELO document mask was successful.
400	Request contains invalid data.
500	When an server error occurs during the load of sync job document mask.

Type

array MaskEntity

Update Endpoint - UpdateEloDocumentMaskById

Updates data of sync job document mask with provided id.

HTTP Request

One of the following requests should be made:

```
PUT /odata/syncjobsdocmasks({key})
PUT /odata/syncjobsdocmasks/{key}
```

Path parameters

Name	Description
------	-------------

key	required. Id of sync job elo document mask that should be updated.
-----	--

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Content-Type	required. application/json
--------------	----------------------------

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code	Description	Type
------	-------------	------

200	If the update of sync job ELO document mask was successful.	MaskEntity
-----	---	------------

Code	Description	Type
400	Request contains invalid data.	
500	When an server error occurs during the update of sync job ELO document mask.	

syncjobsfoldermasks

Get Endpoint - GetEloFolderMasksForJobs

Returns list of ELO folder doc masks for specified sync jobs.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncjobsfoldermasks
GET /odata/syncjobsfoldermasks/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository optional.	Destination repository key.
----------------------	-----------------------------

Request headers

Name	Description
------	-------------

Authorization required.	Bearer <i>{token}</i> . See authorization
-------------------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of the folder masks was successful.	array MaskEntity
-----	---	------------------

400	Request contains invalid data.
-----	--------------------------------

500	When an server error occurs during the load of sync jobs ELO folder masks.
-----	--

Get Endpoint - GetEloFolderMaskById

Loads metadata of the folder doc mask with the provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncjobsfoldermasks({key})
GET /odata/syncjobsfoldermasks/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of folder mask that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of sync job ELO folder mask was successful.
 400 Request contains invalid data.
 500 When an server error occurs during the load of sync job folder mask.

Type

array MaskEntity

Update Endpoint - UpdateEloFolderMaskById

Updates data of sync job folder mask with provided id.

HTTP Request

One of the following requests should be made:

```
PUT /odata/syncjobsfoldermasks({key})
PUT /odata/syncjobsfoldermasks/{key}
```

Path parameters

Name Description

key required. Id of sync job elo folder mask that should be updated.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Content-Type required. application/json

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code Description

200 If the update of sync job ELO folder mask was successful.

Type

MaskEntity

Code	Description	Type
400	Request contains invalid data.	
500	When an server error occurs during the update of sync job ELO folder mask.	

syncjobsmasks

Get Endpoint - GetJobMasks

Returns list of all ELO masks.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncjobsmasks
GET /odata/syncjobsmasks/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of masks was successful.	array MaskEntity
-----	--------------------------------------	------------------

400	Request contains invalid data.
-----	--------------------------------

500	When an server error occurs during the load of masks.
-----	---

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/syncjobsmasks?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#SyncJobsMasks",
  "value": [
    {
      "maskId":1,
      "name":"E-Mail",
      "eloMaskId":"2",
      "type":"DocMask",
      "syncJobId":2
    },
    {
      "maskId":2,
      "name":"Workspace",
      "eloMaskId":"31",
      "type":"FolderMask",
      "syncJobId":2
    },
    ...
  ],
  "@odata.nextLink": "Link for the next 100 items"
}
```

Get Endpoint - GetJobMaskById

Loads ELO mask with the provided id.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncjobsmasks({key})
GET /odata/syncjobsmasks/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of sync job elo mask that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of sync job elo mask was successful.
 400 Request contains invalid data.
 500 When an server error occurs during the load of sync job elo mask.

Type

array MaskEntity

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/syncjobsmasks(1)?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "@odata.context": "https://elosyncurl/odata/$metadata#SyncJobsMasks",
  "value": [
    {
      "maskId":1,
      "name":"E-Mail",
      "eloMaskId":"2",
      "type":"DocMask",
      "syncJobId":2
    }
  ]
}
```

Update Endpoint - UpdateJobMaskById

Updates metadata of specific elo mask that is used by an sync job for keywording.

HTTP Request

One of the following requests should be made:

```
PUT /odata/syncjobsmasks({key})
PUT /odata/syncjobsmasks/{key}
```

Path parameters

Name	Description
------	-------------

key	required. Id of mask that should be updated.
-----	--

Query parameters

Name	Description
------	-------------

repository optional.	Destination repository key.
----------------------	-----------------------------

Request headers

Name	Description
------	-------------

Authorization required.	Bearer <i>{token}</i> . See authorization
-------------------------	---

Content-Type required.	application/json
------------------------	------------------

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code	Description	Type
------	-------------	------

200	If the update of sync job elo mask was successful.	MaskEntity
-----	--	------------

Code Description**Type**

- 400 If update contains data for properties which values are immutable or if no data for the update could be loaded from the request.
- 500 When an server error occurs during the update of sync job elo mask.

Example**Request**

The following is an example of how to call the endpoint:

```
PUT /odata/syncjobsmasks(1)?repository=rep01
Content-Type: application/json
```

```
{
  "maskId": 1,
  "type": "DocMask",
  "syncJobId": 2,
  "name": "E-Mail",
  "eloMaskId": "2"
}
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "maskId":1,
  "name":"E-Mail",
  "eloMaskId":"2",
  "type":"DocMask",
  "syncJobId":2
}
```

syncjobtriggerrelations

Get Endpoint - GetSyncJobTriggerRelations

Returns list of all relations between sync jobs and trigger.

HTTP Request

One of the following requests should be made:

```
GET /odata/syncjobtriggerrelations
GET /odata/syncjobtriggerrelations/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository optional.	Destination repository key.
----------------------	-----------------------------

Request headers

Name	Description
------	-------------

Authorization required.	Bearer <i>{token}</i> . See authorization
-------------------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of relations was successful.	array
-----	--	-------

500	When an server error occurs during the load of syncjob and trigger relations.	SyncJobTriggerEntity
-----	---	----------------------

Get Endpoint - GetSyncJobTriggerRelationById

Get the relation between the given sync job and trigger.

HTTP Request

One of the following requests should be made:

```
GET /api/syncjobtriggerrelations({syncjobid},{triggerid})
```

Query parameters

Name	Description
------	-------------

syncjobid	optional. Id of sync job trigger relation that should be loaded.
-----------	--

triggerId	optional
-----------	----------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

	Authorization required. Bearer <i>{token}</i> . See authorization
--	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of sync job trigger relation was successful.	
-----	--	--

500	When an server error occurs during the load of sync job trigger relation.	
-----	---	--

timetrigger

Get Endpoint - GetTimeTriggers

Returns list of all time triggers.

HTTP Request

One of the following requests should be made:

```
GET /odata/timetrigger
GET /odata/timetrigger/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of time triggers was successful.	array TimeTriggerEntity
-----	--	-------------------------

500	When an server error occurs during the load of time triggers.
-----	---

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/timetrigger?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "value": [
    {
      "triggerId": 123,
      "isActive": false,
      "cronJobFormatValue": "...",
    },
    {
      "triggerId": 125,
      "isActive": true,
      "cronJobFormatValue": "...",
    }
  ],
  "@odata.nextLink": "...",
}
```

Get Endpoint - GetTimeTriggerById

Get a time trigger using its id.

HTTP Request

One of the following requests should be made:

```
GET /odata/timetrigger({key})
GET /odata/timetrigger/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of time trigger that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of time trigger was successful.

500 When an server error occurs during the load of time triggers.

Type

array TimeTriggerEntity

Update Endpoint - UpdateTimeTriggerById

Update a time trigger using its id.

HTTP Request

One of the following requests should be made:

```
PUT /odata/timetrigger({key})
PUT /odata/timetrigger/{key}
```

Path parameters

Name Description

key required. Id of time trigger that should be updated.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Content-Type required. application/json

Request body

Information

Only properties with new values need to be specified, unmodified properties are optional and can be skipped.

Information

If readonly properties are specified the response will always be a 400 Bad Request error.

Responses

Code Description

200 If update of time trigger was successful.

404 If no time trigger with the given id could be found.

500 When an internal server occurs during the update of time trigger.

Type

TimeTriggerEntity

Example

Request

The following is an example of how to call the endpoint:

```
PUT /odata/timetrigger(1)
Content-Type: application/json

{
  "cronJobFormatValue": "* */12 * * *"
}
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "triggerId": 1,
  "cronJobFormatValue": "* */12 * * *",
  "isActive": true
}
```

trigger

Get Endpoint - GetTriggers

Returns list of all triggers.

HTTP Request

One of the following requests should be made:

```
GET /odata/trigger
GET /odata/trigger/$count
```

The endpoint supports the following OData query parameters: \$select, \$expand

Query parameters

Name	Description
------	-------------

repository	optional. Destination repository key.
------------	---------------------------------------

Request headers

Name	Description
------	-------------

Authorization	required. Bearer <i>{token}</i> . See authorization
---------------	---

Responses

Code	Description	Type
------	-------------	------

200	If the load of triggers was successful.	array TriggerEntity
-----	---	---------------------

500	When an server error occurs during the load of triggers.
-----	--

Example

Request

The following is an example of how to call the endpoint:

```
GET /odata/trigger?repository=repo1
```

Response

The following example shows the response:

Information

The response object is truncated for clarity. Default properties will be returned from the call.

```
{
  "value": [
    {
      "triggerId": 123,
      "isActive": false
    },
    {
      "triggerId": 125,
      "isActive": true
    }
  ],
  "@odata.nextLink": "...
}
```

Get Endpoint - GetTriggerById

Get a trigger using its id

HTTP Request

One of the following requests should be made:

```
GET /odata/trigger({key})
GET /odata/trigger/{key}
```

The endpoint supports the following OData query parameters: \$select, \$expand

Path parameters

Name Description

key required. Id of trigger that should be loaded.

Query parameters

Name Description

repository optional. Destination repository key.

Request headers

Name Description

Authorization required. Bearer *{token}*. See authorization

Responses

Code Description

200 If the load of trigger was successful.

500 When an server error occurs during the load of triggers.

Type

array TriggerEntity