

Context API

# Table of Contents

1. Introduction	2
1.1. Infrastructure	2
1.2. Requests	2
1.3. Responses	3
1.4. Timestamps	3
2. Authentication	4
2.1. Creating a new client	4
2.2. Retrieving an access token	4
2.3. Doing a request with an access token	5
3. Creating context items	6
3.1. Mandatory information	6
3.1.1. Type and workflows	7
3.1.2. Events	7
3.1.3. Component	7
3.2. Additional information	8
3.2.1. Keywords	8
3.2.2. Fields & other properties	8
3.3. Examples	8
4. Context item search	11
4.1. Version 1	11
4.2. Version 2	13
4.3. Available filters	14
4.4. Period and interval filter	15
4.5. Type filter	16
4.6. Component filter	16
4.7. Description filter	17
4.8. Keyword filter	17
4.9. Created by filter	18
4.10. Custom field and other property filter	18
4.10.1. Numeric field filter	18
4.10.2. String field filter	19
4.10.3. Enumeration field filter	19
4.10.4. Other property filter	20
4.11. Approval filter	20
4.12. Duration filter	20
4.13. Current state filter	21
4.14. Combining filters	21
5. Reactive Search endpoints	23

5.1. Minimal Search	23
5.2. Websocket	23
5.2.1. Channels	23
6. Loading a context view	25
6.1. Getting the view	25
6.2. Enriching the view	25
6.2.1. Filter validation	25
6.3. Performing the search	25
6.3.1. Paged search API	25
6.3.2. Reactive endpoints	26
7. Context type access rules	27
7.1. Subject type priorities	27
8. Context item approvals	29
9. Context item attachments	30
10. Context item comments	31
11. Context item history	32
12. Resources	33
12.1. AccessRules	33
12.1.1. Create new	33
12.1.2. Delete specific	33
12.1.3. Get specific	34
12.1.4. Get all for type	34
12.1.5. Update specific	35
12.2. Analytics	35
12.2.1. Get all available aggregation operations	35
12.2.2. Calculate aggregations	36
12.3. Approvals	37
12.3.1. Add approval	37
12.3.2. Get all for item	37
12.3.3. Revoke approval	38
12.4. AssetType	38
12.4.1. Add types to collection for assets	38
12.4.2. Delete types from collection for assets	39
12.4.3. Get types for assets	40
12.4.4. Update entire collection of types for assets	40
12.5. Attachments	41
12.5.1. Create new	41
12.5.2. Delete specific	42
12.5.3. Download	42
12.5.4. Get all for item	43
12.5.5. Get specific	43

12.5.6. Update specific .....	44
12.6. Comments .....	45
12.6.1. Create new coment on a context item .....	45
12.6.2. Delete specific .....	45
12.6.3. Get all comments for context item .....	46
12.6.4. Get all comments for context items in bulk .....	47
12.6.5. Get specific comment .....	47
12.6.6. Update specific .....	48
12.7. ContextData .....	48
12.7.1. Create new context data .....	48
12.7.2. Delete specific context data .....	49
12.7.3. Get all context data .....	49
12.7.4. Get a specific context data .....	50
12.7.5. Search context data .....	51
12.7.6. Update specific context data .....	51
12.8. ContextDataEvent .....	52
12.8.1. Create new .....	52
12.8.2. Delete specific .....	53
12.8.3. Get all for item .....	53
12.8.4. Get specific .....	54
12.8.5. Update specific .....	55
12.9. ContextDataReference .....	55
12.9.1. Add a new context data reference to a context data .....	55
12.9.2. Delete specific .....	56
12.9.3. Get all for context data references for a specific context data .....	56
12.9.4. Get specific .....	57
12.9.5. Update specific .....	58
12.10. ContextItem .....	59
12.10.1. Create new .....	59
12.10.2. Create new in batch .....	59
12.10.3. Delete specific .....	60
12.10.4. Delete bulk by definition .....	60
12.10.5. Delete bulk by identifiers .....	61
12.10.6. Get specific .....	61
12.10.7. Search context items .....	62
12.10.8. Minimal search (streaming) .....	63
12.10.9. Update specific .....	64
12.11. ContextItemV2 .....	64
12.11.1. Returns the number of items matching the provided filters. ....	64
12.11.2. Search context items V2 .....	65
12.12. ContextView .....	65

12.12.1. Enrich a saved context view .....	65
12.13. DataImportController .....	66
12.13.1. Delete a file. ....	66
12.13.2. Download a file by id. ....	66
12.13.3. Download the generated error file for a specific file. ....	66
12.13.4. List all imported files. ....	67
12.13.5. Get a file. ....	68
12.13.6. Upload a file. ....	68
12.14. Export .....	68
12.14.1. Check existence. ....	68
12.14.2. Remove all expired .....	69
12.14.3. Create new .....	69
12.14.4. Get all. ....	70
12.14.5. Get export. ....	70
12.15. FailedEventFrameSynchronization .....	71
12.15.1. Re-sync one specific contextItem .....	71
12.15.2. Search .....	71
12.16. Fields .....	72
12.16.1. Create new .....	72
12.16.2. Delete specific .....	73
12.16.3. Get all. ....	73
12.16.4. Get specific .....	74
12.16.5. Retrieve a list of fields linked to provided types. ....	74
12.16.6. Search fields. ....	75
12.16.7. Update specific .....	76
12.17. History .....	76
12.17.1. Get audit trail for item. ....	76
12.18. Keywords .....	77
12.18.1. Get all. ....	77
12.18.2. Search keywords. ....	78
12.18.3. Validate a list of keywords .....	78
12.19. PropertyKeyMigration .....	79
12.19.1. Download the Property key migration dry-run report .....	79
12.19.2. Get field migrated keys uniqueness .....	79
12.19.3. Get field dry-run results .....	80
12.19.4. Get monitor dry-run results. ....	81
12.19.5. Get migration status. ....	81
12.19.6. Get view dry-run results .....	82
12.19.7. Update phase status .....	82
12.20. PublicDownloadGenerator .....	83
12.20.1. Generate .....	83

12.21. Source	83
12.21.1. Get specific source	83
12.21.2. Search	84
12.21.3. Start historical sync	85
12.21.4. Update specific source	85
12.22. SourceSynchronization	86
12.22.1. Do a manual cleanup of one datasource	86
12.22.2. Re-sync specific context item	86
12.22.3. Re-sync list of context items	87
12.22.4. Re-sync list of context items	87
12.23. SourceSynchronizationInBackground	88
12.23.1. Cancel historical background sync job	88
12.23.2. Get background sync job	88
12.23.3. Search background sync jobs	89
12.23.4. Update background sync job status	89
12.24. Triggers	90
12.24.1. Create new	90
12.24.2. Delete specific	91
12.24.3. Get all for type	91
12.24.4. Get specific	92
12.24.5. Update specific	92
12.25. Type	93
12.25.1. Create new	93
12.25.2. Delete specific context data type	93
12.25.3. Get all context data types	94
12.25.4. Get specific	95
12.25.5. Search context data types	95
12.25.6. Search context data types using a POST request	96
12.25.7. Update specific	97
12.26. Workflow	98
12.26.1. Create new	98
12.26.2. Delete specific workflow	98
12.26.3. Get all workflows	99
12.26.4. Gets specific	99
12.26.5. Search for workflows	100
12.26.6. Search for workflow states	100
12.26.7. Update workflow	101
12.26.8. Update workflow states	102
13. Models	103
13.1. <i>AccessRule</i>	103
13.2. <i>AggregationDefinition</i>	103

13.3. <i>AggregationOperationRepresentationModel</i> .....	103
13.4. <i>AggregationResultRepresentationModel</i> .....	104
13.5. <i>Approval</i> .....	104
13.6. <i>ApprovalFilter</i> .....	105
13.7. <i>ApprovalFilterAllOf</i> .....	108
13.8. <i>Asset</i> .....	108
13.9. <i>AssetModel</i> .....	109
13.10. <i>AssetType</i> .....	109
13.11. <i>AssetTypeModel</i> .....	109
13.12. <i>Attachment</i> .....	110
13.13. <i>AttachmentModel</i> .....	110
13.14. <i>AuditTrail</i> .....	110
13.15. <i>BackgroundSyncJobRepresentationModel</i> .....	111
13.16. <i>BackgroundSyncJobStatusModel</i> .....	112
13.17. <i>BulkDeleteByFiltersModel</i> .....	113
13.18. <i>BulkDeleteByIdentifiersModel</i> .....	113
13.19. <i>BulkDeleteResult</i> .....	114
13.20. <i>BulkResyncByFiltersModel</i> .....	114
13.21. <i>BulkResyncByIdentifiersModel</i> .....	114
13.22. <i>CollectionModelAggregationOperationRepresentationModel</i> .....	115
13.23. <i>CollectionModelAggregationResultRepresentationModel</i> .....	115
13.24. <i>CollectionModelLinkedField</i> .....	115
13.25. <i>ColumnSettingsView</i> .....	115
13.26. <i>Comment</i> .....	116
13.27. <i>CommentModel</i> .....	116
13.28. <i>ComponentFilter</i> .....	116
13.29. <i>ComponentFilterAllOf</i> .....	119
13.30. <i>ComponentValue</i> .....	119
13.31. <i>ContextData</i> .....	120
13.32. <i>ContextDataModel</i> .....	120
13.33. <i>ContextDataType</i> .....	121
13.34. <i>ContextDataTypeModel</i> .....	121
13.35. <i>ContextEvent</i> .....	122
13.36. <i>ContextEventModel</i> .....	122
13.37. <i>ContextItem</i> .....	123
13.38. <i>ContextItemComponent</i> .....	123
13.39. <i>ContextItemComponentModel</i> .....	124
13.40. <i>ContextItemEventModel</i> .....	124
13.41. <i>ContextItemModel</i> .....	125
13.42. <i>ContextView</i> .....	125
13.43. <i>ContextViewData</i> .....	126

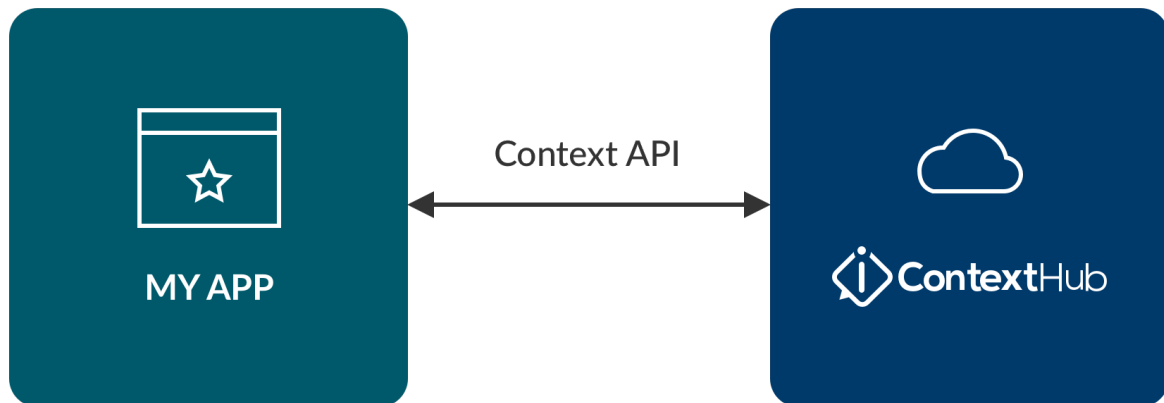
13.44. <i>CountIndependentSearchRequestModel</i> .....	126
13.45. <i>CountRepresentationModel</i> .....	127
13.46. <i>CreatedByFilter</i> .....	127
13.47. <i>CreatedByFilterAllOf</i> .....	130
13.48. <i>CreatedDateFilter</i> .....	130
13.49. <i>CreatedDateFilterAllOf</i> .....	133
13.50. <i>CurrentStateFilter</i> .....	134
13.51. <i>CurrentStateFilterAllOf</i> .....	137
13.52. <i>DataImport</i> .....	137
13.53. <i>DataModel</i> .....	138
13.54. <i>DataReference</i> .....	138
13.55. <i>DataReferenceModel</i> .....	138
13.56. <i>DescriptionFilter</i> .....	138
13.57. <i>DescriptionFilterAllOf</i> .....	141
13.58. <i>DurationCondition</i> .....	141
13.59. <i>DurationFilter</i> .....	141
13.60. <i>DurationFilterAllOf</i> .....	144
13.61. <i>EnumerationFieldFilter</i> .....	144
13.62. <i>EnumerationFieldFilterAllOf</i> .....	147
13.63. <i>ErrorDetail</i> .....	147
13.64. <i>ErrorMessage</i> .....	147
13.65. <i>Export</i> .....	150
13.66. <i>ExportModel</i> .....	150
13.67. <i>FailedEventFrameSynchronization</i> .....	151
13.68. <i>Field</i> .....	151
13.69. <i>FieldFilter</i> .....	152
13.70. <i>FieldModel</i> .....	155
13.71. <i>FieldModelColumn</i> .....	155
13.72. <i>FieldUniquenessRepresentationModel</i> .....	156
13.73. <i>Filter</i> .....	156
13.74. <i>GeneralColumn</i> .....	159
13.75. <i>I18nMessage</i> .....	159
13.76. <i>IdListModel</i> .....	159
13.77. <i>IntervalFilter</i> .....	159
13.78. <i>IntervalFilterAllOf</i> .....	162
13.79. <i>Keyword</i> .....	162
13.80. <i>KeywordFilter</i> .....	162
13.81. <i>KeywordFilterAllOf</i> .....	165
13.82. <i>KeywordModel</i> .....	165
13.83. <i>Link</i> .....	165
13.84. <i>LinkModel</i> .....	165



13.85. <i>LinkedField</i> .....	166
13.86. <i>LiveIntervalFilter</i> .....	166
13.87. <i>MinimalContextItem</i> .....	169
13.88. <i>NumericCondition</i> .....	169
13.89. <i>NumericFieldFilter</i> .....	170
13.90. <i>NumericFieldFilterAllOf</i> .....	173
13.91. <i>PageDataImport</i> .....	173
13.92. <i>PageMetadata</i> .....	174
13.93. <i>Pageable</i> .....	174
13.94. <i>PagedModelAttachment</i> .....	174
13.95. <i>PagedModelAuditTrail</i> .....	174
13.96. <i>PagedModelBackgroundSyncJobRepresentationModel</i> .....	175
13.97. <i>PagedModelComment</i> .....	175
13.98. <i>PagedModelContextData</i> .....	175
13.99. <i>PagedModelContextDataType</i> .....	175
13.100. <i>PagedModelContextEvent</i> .....	175
13.101. <i>PagedModelContextItem</i> .....	176
13.102. <i>PagedModelDataReference</i> .....	176
13.103. <i>PagedModelExport</i> .....	176
13.104. <i>PagedModelFailedEventFrameSynchronization</i> .....	176
13.105. <i>PagedModelField</i> .....	177
13.106. <i>PagedModelKeyword</i> .....	177
13.107. <i>PagedModelPropertyKeyMigrationFieldData</i> .....	177
13.108. <i>PagedModelPropertyKeyMigrationMonitorData</i> .....	177
13.109. <i>PagedModelPropertyKeyMigrationViewData</i> .....	177
13.110. <i>PagedModelSource</i> .....	178
13.111. <i>PagedModelString</i> .....	178
13.112. <i>PagedModelTriggerDefinition</i> .....	178
13.113. <i>PagedModelWorkflow</i> .....	178
13.114. <i>Parameter</i> .....	179
13.115. <i>PeriodFilter</i> .....	179
13.116. <i>PeriodFilterAllOf</i> .....	182
13.117. <i>PermissionUserDetails</i> .....	183
13.118. <i>PropertyFieldFilter</i> .....	184
13.119. <i>PropertyFieldFilterAllOf</i> .....	187
13.120. <i>PropertyKeyMigrationField</i> .....	187
13.121. <i>PropertyKeyMigrationFieldData</i> .....	187
13.122. <i>PropertyKeyMigrationFieldFilter</i> .....	188
13.123. <i>PropertyKeyMigrationMonitorData</i> .....	188
13.124. <i>PropertyKeyMigrationPhaseRepresentationModel</i> .....	188
13.125. <i>PropertyKeyMigrationPhaseStatusModel</i> .....	189

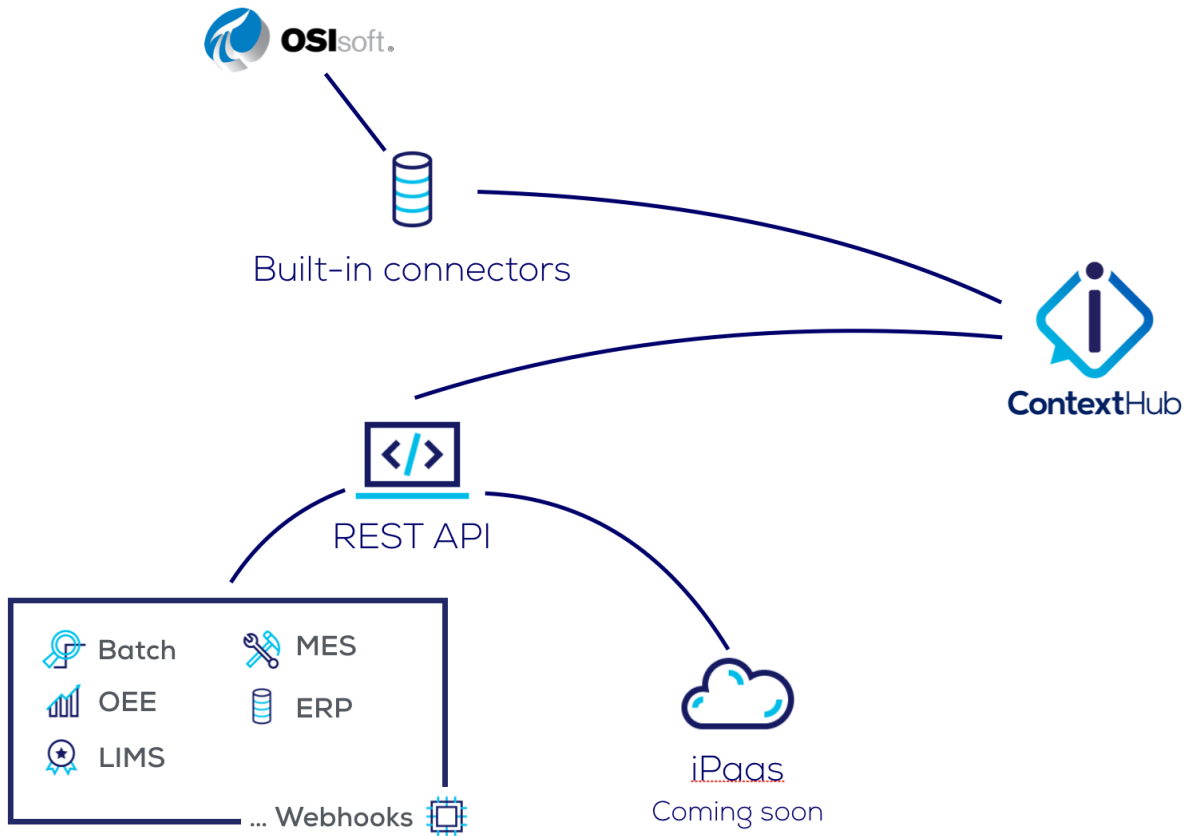
13.126. <i>PropertyKeyMigrationPropertyFieldFilter</i> .....	189
13.127. <i>PropertyKeyMigrationStatusRepresentationModel</i> .....	189
13.128. <i>PropertyKeyMigrationViewData</i> .....	189
13.129. <i>ReactiveSearchRequestModel</i> .....	190
13.130. <i>SearchParametersModel</i> .....	191
13.131. <i>SearchRequestModel</i> .....	191
13.132. <i>Sort</i> .....	192
13.133. <i>Source</i> .....	192
13.134. <i>SourceModel</i> .....	193
13.135. <i>SourceSynchronizationIntervalModel</i> .....	193
13.136. <i>StringFieldFilter</i> .....	193
13.137. <i>StringFieldFilterAllOf</i> .....	196
13.138. <i>TaskSubmissionResult</i> .....	196
13.139. <i>TimeSeriesDefinitionRepresentation</i> .....	196
13.140. <i>TokenMetaData</i> .....	197
13.141. <i>TokenizedPageModelContextItem</i> .....	197
13.142. <i>TriggerDefinition</i> .....	197
13.143. <i>TriggerDefinitionModel</i> .....	198
13.144. <i>TypeAccessRuleModel</i> .....	198
13.145. <i>TypeFilter</i> .....	199
13.146. <i>TypeFilterAllOf</i> .....	202
13.147. <i>UserDetails</i> .....	202
13.148. <i>ValidationResultMessage</i> .....	202
13.149. <i>Workflow</i> .....	203
13.150. <i>WorkflowModel</i> .....	203
13.151. <i>WorkflowStates</i> .....	203
13.152. <i>WorkflowStatesModel</i> .....	204

Based on simple REST principles, the TrendMiner Context API endpoints return JSON metadata about context items, types, and other related objects in the TrendMiner platform.



# Chapter 1. Introduction

## 1.1. Infrastructure



Context can be pulled into the TrendMiner platform through multiple pathways. One is the REST API described in the developer documentation here. The other is by hooking TrendMiner to your iPaas system.

ContextHub supports the contextualization of tags, assets and attributes, or, components as we call them in general. All information is contained within an entity called a context item. Which makes the */item* endpoint the main endpoint to create and search for context in the platform.

## 1.2. Requests

The TrendMiner Web API is based on REST principles. Data resources are accessed via standard HTTPS requests in UTF-8 format to an API endpoint. Where possible, the web API uses appropriate HTTP verbs for each action:

Table 1. REST

Method	Action
GET	Retrieves resources
POST	Creates resources
PUT	Changes and/or replaces resources or collections

Method	Action
DELETE	Deletes resources

## 1.3. Responses

The web API returns all response data as a JSON object. Have a look at the definitions for a description of all retrievable objects.

## 1.4. Timestamps

Timestamps are returned in [ISO 8601](#) format as [Coordinated Universal Time \(UTC\)](#) with a zero offset: YYYY-MM-DDTHH:MM:SSZ.



- ① The access token is to be used in every request towards our services (see example below). A new token needs to be fetched when the token expires after 5 minutes.

## 2.3. Doing a request with an access token

*Example request with access token*

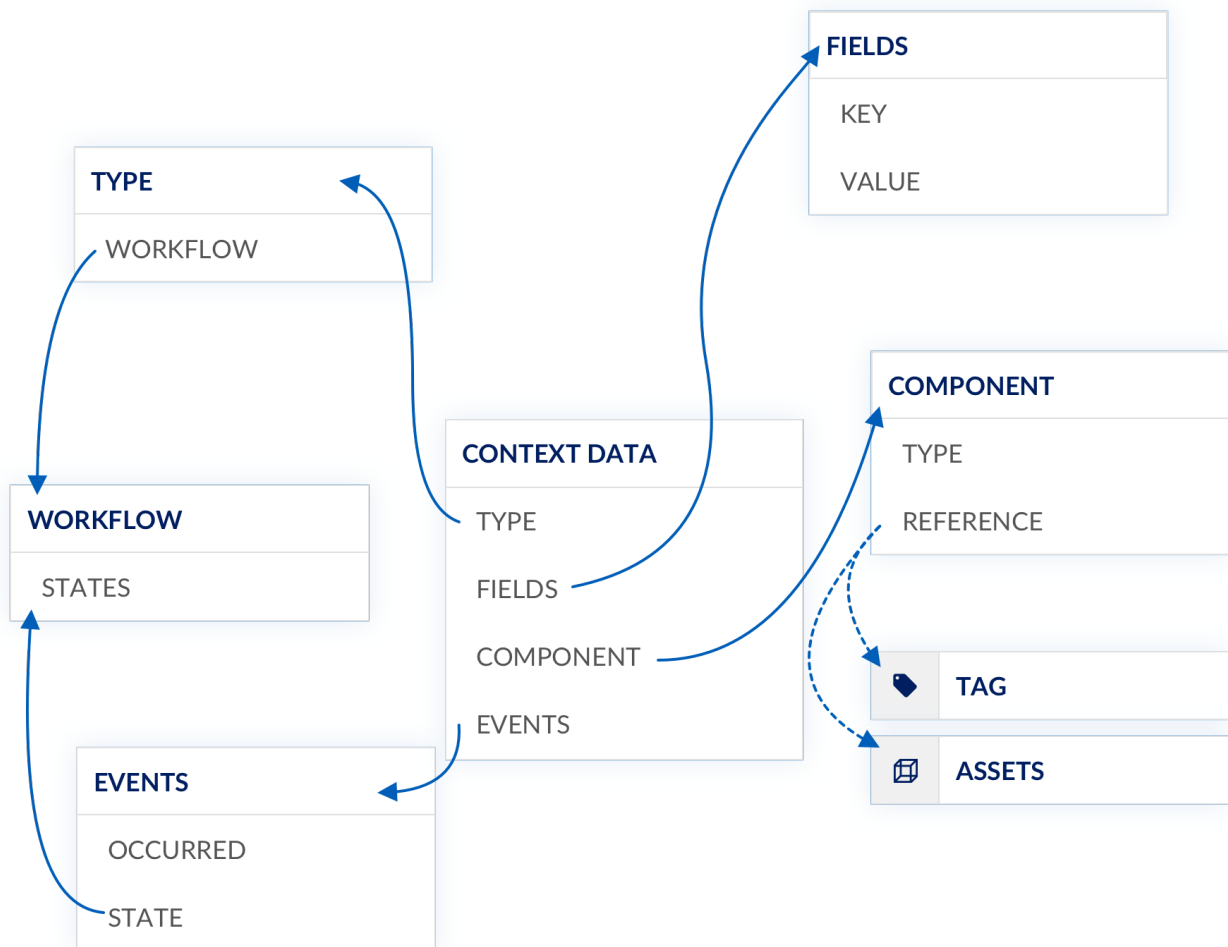
```
curl --request GET \  
  --url 'https://YOUR_DOMAIN/context/item/d0a58479-4120-4683-a143-42f5e7ac249e' \  
  --header 'Authorization: Bearer ACCESS_TOKEN'
```

# Chapter 3. Creating context items

Creating context items can be done in two distinct ways:

- using the [Context Item API](#), to create an item, its events and its component in one request,
- using the [Context Data API](#), to create an item, while additional requests to other APIs are needed for events and component.

This section will focus on the first, and will explain the necessary parts needed to fire your request.



To create a context item, three mandatory parts need to be provided:

- the type of the context item, which needs to exist up front,
- at least one event, which is defined by the used type,
- a component on which this item occurs.

## 3.1. Mandatory information



### 3.1.1. Type and workflows

A type indicates what kind of context item we're dealing with. To create a context item, the type already needs to exist. It can be created using the type API or using ContextHub.

Every type can be assigned a workflow. A workflow dictates the different phases a context item of a certain type can go through, from start to completion.

In case a context item is created using a type that is not assigned a workflow, the context item will be considered instantaneous and can only have one stateless event.

**TIP** For more detailed information on the API, see [Type](#) and [Workflow](#).

By default, TrendMiner has a few predefined types and workflow. Examples are

- **ANOMALY** with states **Started** and **Ended**,
- **INFORMATION** without workflow,
- **OPERATIONAL** with states **In progress**, **Paused**, and **Ended**.

### 3.1.2. Events

An event is the occurrence of a notable *thing*—event, action, incident, ...—and refers to a specific point in time.

In case the type of the context item is assigned a workflow, the events on the context item need to have a state. In this case, the earliest occurring event's state must be the start event, as defined in the workflow. When assigning an event state which is the workflow's end state, this event must reflect the latest time the workflow operates.

Regardless of the type, an event needs to be sent.

### 3.1.3. Component

A component is the reference to a tag, asset or attribute. Since context items are all about contextualizing these, a component is mandatory.

Because tags or assets and attributes are not part of the context domain, we need identify them within their own respective domains.

#### Tags

Tags are mainly used when there is no asset framework available to structure tags in a logical tree. The identifier to be used during context item creation is the name of the tag.

#### Assets & attributes

Assets and attributes are used to create a logical structure. While an asset is purely a virtual concept, an attribute will internally link to a tag. Assets can be found using the [Assets API](#). The identifier to be used during context item creation is the **nodeId**.

## 3.2. Additional information

Besides the mandatory parts for creating context items, additional information can be added to the item to provide more details.

A description can be used to give more information on the event. Any text can be written here. Next to a description, keywords and additional fields can be provided as well.

### 3.2.1. Keywords

Keywords are used to make searching on context items easier.

Context items can contain as many keywords as needed, as long as they are shorter than 100 characters and do not contain commas.

### 3.2.2. Fields & other properties

Fields and properties can contain any data to further describe a context item. While their purpose is the same, there is a small distinction.

Fields are properties with a predefined data type. A field's type can be either **STRING**, **NUMERIC** or **ENUMERATION**. The latter can have predefined values. More advanced searching capabilities are available when using fields.

Fields are created independently of context item types, and are linked to them through ContextHub. In ContextHub, when creating or editing context items, fields can be modified as well. To find out which fields are used on a type, use the [Type API](#).

Other properties are properties without a predefined data type. When present on a context item, they will be shown in ContextHub. They are however not editable through ContextHub, only through the API.

Both fields and other properties need to be filled in the **fields** part of the request body in a **"key": "value"** manner.

## 3.3. Examples

All these parts need to be combined to create a new context item. For detailed information, refer to the [Context Item API](#) or the [Context Data API](#).

#### TIP

For more detailed information on building context items without using the context item API, see [Context Data API](#), [Context Data Event API](#) and [Context Data Reference API](#).

*Example: Creating a context item with a workflow referencing a tag*

```
POST https://trendminer.company.net/context/item HTTP/1.1
```

```
Content-Type: application/json
```

```
{
  "description": "Elevated temperature detected",
  "keywords": [
    "temperature"
  ],
  "type": {
    "identifier": "ANOMALY"
  },
  "components": [
    {
      "type": "TAG",
      "reference": "BA:LEVEL.1"
    }
  ],
  "fields": {
    "criticality": "HIGH",
    "temperature": 87.2
  },
  "events": [
    {
      "state": "Started",
      "occurred": "2019-11-14T08:31:58.000Z"
    }
  ]
}
```

*Example: Creating a context item without workflow with bare minimum data provided*

```
POST https://trendminer.company.net/context/item HTTP/1.1
```

```
Content-Type: application/json
```

```
{
  "type": {
    "identifier": "INFORMATION"
  },
  "components": [
    {
      "type": "ASSET",
      "reference": "8397fef9-ad29-4ee5-bf04-72a11c0b233c"
    }
  ],
  "events": [
    {
      "occurred": "2019-11-13T13:55:07.000Z"
    }
  ]
}
```

# Chapter 4. Context item search

Context items can be searched using the API's paged search endpoint. The search endpoint will always need at least one filter to limit the timeframe to look at.

## 4.1. Version 1

**WARNING** | Version 1 of the search endpoint is deprecated. Please use version 2.

The version 1 search endpoint is a paged API. The response will include a metadata block with the current page information.

*Search endpoint version 1*

```
POST /context/item/search?page=0&size=20
```

*Example: A version 1 search request body using multiple filters.*

```
{
  "filters": {
    {
      "periodType": "LAST_EIGHT_HOURS",
      "period": "PT8H",
      "live": false,
      "type": "PERIOD_FILTER"
    },
    {
      "type": "COMPONENT_FILTER"
      "components": [
        {
          "identifier": "e56b9148-b109-4495-9e84-9cbf866201bc",
          "type": "ASSET",
          "path": "e56b9148b10944959e849cbf866201bc",
          "include": "SELF"
        }
      ]
    }
  }
}
```

*Example: A version 1 search response.*

```
{
  "links": [...],
  "content": [
    {
      "permissions": [
        "ASSET_READ_CONTEXT_ITEM"
      ]
    }
  ]
}
```

```

    ],
    "identifier": "de04ed72-f135-400a-bc11-65104e8c7998",
    "shortKey": "901-C6",
    "externalId": "dc165337-5e6e-11ec-8a12-000d3ac26207",
    "description": "Something happened here.",
    "events": [
      {
        "identifier": "1d4f153e-3342-4c1a-ac47-6deed0152422",
        "occurred": "2021-12-16T12:51:00Z",
        "state": "Started",
        "links": [...]
      }
    ],
    "lastModifiedDate": "2022-05-09T02:00:03.454513Z",
    "createdBy": "833af9d7-b2fc-40a6-bb38-08a5d2ced6ca",
    "createdDate": "2021-12-16T12:54:29.640088Z",
    "userDetails": {
      "identifier": "833af9d7-b2fc-40a6-bb38-08a5d2ced6ca",
      "username": "service-account-tm-context"
    },
    "components": [],
    "type": {
      "identifier": "6e77cfbc876f4b4aae8ec2372f7bdd0f",
      "name": "PI Event Frame",
      ...
      "color": "4A75E2",
      "icon": "flow--line"
    },
    "workFlowDescribed": true,
    "fields": {
      "tm_EF_SYNC_MISSING_DATA_REFERENCE": true,
      "tm_EF_SYNC_MISSING_DATA_REFERENCE_RETRIES": 130,
      "EVENT_FRAME_NAME": "EF20211216-001"
    },
    "keywords": [],
    "startEventDate": "2021-12-16T12:51:00Z",
    "lastEventState": "Started",
    "totalDuration": "PT3450H54M32S",
    "links": [...]
  },
  ...
],
"page": {
  "size": 20,
  "totalElements": 697,
  "totalPages": 35,
  "number": 0
}
}

```







- the description filter
- the keyword filter
- the created by filter
- the custom field or other property filter
- the approvals filter
- the duration filter
- the current state filter

Each of these filters require their own input data.

To specify a more precise search query, the filters can be combined. Combining filters will return only those context items that match the criteria of every specified filter.

## 4.4. Period and interval filter

The period filter creates a moving window of time, starting from the current time, and going back a predefined portion of time.

Table 2. Period options

Key	Description
LAST_HOUR	1 hour
LAST_TWO_HOURS	2 hours
LAST_EIGHT_HOURS	8 hours
LAST_DAY	1 day
LAST_WEEK	7 days
LAST_FOUR_WEEKS	28 days

When using the period filter, one also has the option to have item be sent back as they are created. To do so, the `live` flag needs to be set.

*Example: Period filter searching the last 8 hours*

```
{
  "type": "PERIOD_FILTER",
  "periodType": "LAST_EIGHT_HOURS",
  "live": false
}
```

Alternatively, when searching for a pre-defined time frame, the interval filter can be used to specify the range, using start and end date. Use the ISO date format in UTC timezone to specify the dates.

*Example: Interval filter searching pre-defined time frame*

```
{
  "type": "INTERVAL_FILTER",
  "startDate": "2019-08-01T22:00:00.000Z",
  "endDate": "2019-08-02T21:59:00.000Z"
}
```

## 4.5. Type filter

The type filter can be used to return only those context items of the specified types. The filter accepts a list of type identifiers.

*Example: Type filter search for context items of type ANOMALY or INFORMATION*

```
{
  "type": "TYPE_FILTER",
  "types": [
    "ANOMALY",
    "INFORMATION"
  ]
}
```

## 4.6. Component filter

The component filter is used to limit the results to context items linked to a certain asset, attribute or tag. Multiple components can be specified.

When searching for tags, the identifier to specify is the tag's name or its UUID. The UUID can be found from querying `tm-datasource timeseries` API.

For assets and attributes, it is also possible to search for context items linked to certain assets or attributes, or any of the components in the tree above the specified one. When doing so, specifying the path to the current asset or attribute is required. This path can be found on the structure resource of the node. Check out the [Assets API](#) for more information on structures, nodes and paths.

When searching for context items created on specific assets or attributes, it is sufficient to provide the `nodeId`. However, it is also possible to search for context items that are created on the parents or children of an asset or attribute. To do this, we need to provide the filter with the path information and the relative type. This path can be found on the node resource. Check out the [Assets API](#) for more information on nodes and paths

## Relative types

```
SELF: Only include the node. DEFAULT
ANCESTORS: Include context items created on the node and on its parents. PATH is
required
DESCENDANTS: Include context items created on the node and on all of its children.
PATH is required
```

*Example: Component filter search for a specified attribute and its descendants*

```
{
  "type": "COMPONENT_FILTER",
  "components": [
    {
      "identifier": "51f0d710-246f-4a89-9264-25df45df53e0",
      "type": "ATTRIBUTE",
      "path": "64dc37890dd147629bd1151a6b0729cd.51f0d710246f4a89926425df45df53e0",
      "include": "DESCENDANTS"
    }
  ]
}
```

## 4.7. Description filter

Searching for certain words or phrases in the description of context items is possible using the description filter. The filter allows for specifying multiple words to look for. Wildcards can be used as well: an asterisk (\*) will match zero to any range of characters.

*Example: Description filter matching items containing the words 'clogged' or 'phase \*.b', having \* match anything*

```
{
  "type": "DESCRIPTION_FILTER",
  "values": [
    "clogged",
    "phase *.b"
  ],
}
```

## 4.8. Keyword filter

Upon creating of a context item, keywords can be specified. Using the keyword filter, searching can be limited to items having the provided keywords. Multiple keywords can be specified.

Example: Keyword filter matching items having the keyword 'overflow'

```
{
  "type": "KEYWORD_FILTER",
  "keywords": [
    "overflow"
  ]
}
```

## 4.9. Created by filter

Using the created by filter, only context items created by the specified users are returned. Multiple users can be specified.

To specify the users, use their identifiers.

Example: Created by filter matching items created by the specified user

```
{
  "type": "CREATED_BY_FILTER",
  "users": [
    "ea02ab87-ebbf-4f7b-af6d-8df819abe0"
  ]
}
```

## 4.10. Custom field and other property filter

As there are three different types of custom field, there are also three different types of custom field filter: numeric, string and enumeration. The custom field filter can only be used on existing custom fields. The value for `field` needs to be the identifier of the custom field. Specifying a custom field that does not exist will result in an error.

### 4.10.1. Numeric field filter

The numeric field filter searches for context items having the specified custom field filled in with a number matching the specified criteria. These criteria are simple mathematical conditions:

Table 3. Condition operators

Operator	Description
<code>EQUAL</code>	item value = specified value
<code>NOT_EQUAL</code>	item value != specified value
<code>LESS_THAN</code>	item value < specified value
<code>LESS_THAN_OR_EQUAL</code>	item value ≤ specified value
<code>GREATER_THAN</code>	item value > specified value
<code>GREATER_THAN_OR_EQUAL</code>	item value ≥ specified value

One can specify up to two conditions.

*Example: Numeric field filter searching for items with field 'loss\_value' greater than or equal to 12*

```
{
  "type": "NUMERIC_FIELD_FILTER",
  "field": "loss_value",
  "conditions": [
    {
      "operator": "GREATER_THAN_OR_EQUAL",
      "value": 12
    }
  ]
}
```

### 4.10.2. String field filter

The string field filter searches for context items having the specified custom field filled in with a string value matching the specified value. String matching is done in the same way as the [description filter](#). Multiple values can be specified.

*Example: String field filter searching for items with field 'client' matching any word starting with 'Trend'*

```
{
  "type": "STRING_FIELD_FILTER",
  "field": "client",
  "values": [
    "Trend*"
  ]
}
```

### 4.10.3. Enumeration field filter

The enumeration field filter searches for context items having the specified custom field filled in with the one of the specified values. These values need to be present in the custom field's definition. Specifying a value that does not exist in the custom field definition will result in error.

*Example: Enumeration field filter searching for items with field 'cause' being either 'Human error' or 'Late maintenance'*

```
{
  "type": "ENUMERATION_FIELD_FILTER",
  "field": "cause",
  "values": [
    "Human error",
    "Late maintenance"
  ]
}
```

#### 4.10.4. Other property filter

Other properties on a context item can be present when a sync puts them there or when using the API. These properties can have names or values not defined as custom field. To search on these properties, the other properties filter can be used. This filter will perform string matches in the same way as the [description filter](#). Multiple values can be specified.

The key to specify is the actual property's name.

*Example: Other property filter searching for items with property color having values matching 'brown' or 'green'*

```
{
  "type": "PROPERTY_FIELD_FILTER",
  "key": "color",
  "values": [
    "brown",
    "green"
  ]
}
```

#### 4.11. Approval filter

When approvals are enabled on context item types, users can mark a context item as approved to demonstrate acknowledgement. Using the approval filter, we can specify if we want to see context items either approved or not.

*Example: Approval filter searching for unapproved context items*

```
{
  "type": "APPROVAL_FILTER",
  "withApprovals": false
}
```

#### 4.12. Duration filter

Context items with types that are assigned a workflow have a duration. The duration is calculated from initiation of the start event until the occurrence of an end event, or, in case there is none yet, until now. Using the duration filter, we can apply a simple mathematical operator when searching for context items that have a duration. See also [Condition operators](#).

Duration is provided in standard ISO Duration format

*Example: Duration filter searching for context items longer than one and a half hours*

```
{
  "type": "DURATION_FILTER",
  "conditions": [
    {
      "operator": "GREATER_THAN",
      "value": "PT1H30M"
    }
  ]
}
```

## 4.13. Current state filter

Context items with types that are assigned a workflow, are locked into a specific state at any time. Using the current state filter, we can search for context items of any specified state.

*Example: Current state filter searching for item state "In progress"*

```
{
  "type": "CURRENT_STATE_FILTER",
  "states": [
    "In progress"
  ]
}
```

## 4.14. Combining filters

The request model for searches allows for combining any of the filters above. A search will only return context items that match all filters.

*Example: Different filters combined in a search request body.*

```
{
  "filters": [
    {
      "type": "PERIOD_FILTER",
      "periodType": "LAST_EIGHT_HOURS",
      "live": false
    },
    {
      "type": "CREATED_BY_FILTER",
      "users": [
        "ea02ab87-ebbf-4f7b-af6d-8dff819abe0"
      ]
    },
    {
      "type": "NUMERIC_FIELD_FILTER",
      "field": "loss_value",
      "conditions": [
        {
          "operator": "GREATER_THAN_OR_EQUAL",
          "value": 12
        }
      ]
    }
  ]
}
```



# Chapter 5. Reactive Search endpoints

The Context API has two reactive endpoints:

- a websocket using STOMP,
- a minimal search streaming endpoint.

In contrast with the regular paged search API, items for which the user does not have the required permissions are not returned in the results

Both of these searches have some optional configuration:

Table 4. Configuration on the `ReactiveSearchRequestModel` for both the `websocket` and `minimalSearch`

Parameter name	Default value	Description
<code>limit</code>	unlimited	Maximum number of context items you want to receive.
<code>bufferSize</code>	200	Maximum number of context items per message. Messages are pushed after 1 second in case the buffer is not full.
<code>delay</code>	0	Delay in milliseconds between two data pushes.

## 5.1. Minimal Search

The minimal search endpoint returns only the context item's identifier, the component it is linked to and its start and end date.

A request sent to the minimal search endpoint will return a stream.

## 5.2. Websocket

When the same level of detail in context items as in a regular HTTP search is required, the websocket approach can be used. The Context API's websocket uses the Simple Text Orientated Messaging Protocol, in short STOMP.

Using the websocket, it's possible to setup a live search. Live searches will return context items in scope as soon as they are created, for as long as the websocket is open.

For more information on how to use the websocket using STOMP, please visit the [STOMP specification](#).

### 5.2.1. Channels

The websocket provides three channels:

Channel	Description
<code>/queue/searchresults/</code>	Returns the context items. Messages contain lists of <code>ContextItemResource</code> 's with a maximum item count of <code>bufferSize</code> .
<code>/queue/searcherrors/</code>	Returns errors in case any occur.
<code>/queue/searchfeedback/</code>	Returns feedback if needed. Context will send a <code>NO_MORE_DATA</code> message when there is no more data available. Only in live mode newly created data will still be pushed.

# Chapter 6. Loading a context view

A context item view consistent of a set of filters that are used to search context items, the type of the view and additional options depending on the view type. This section describes the steps to load the data produced by a context view using the API.

## 6.1. Getting the view

Context views are saved in the work-organiser. Use the work-organiser API to get the saved item, which will contain the identifier of the context view.

## 6.2. Enriching the view

The saved view in work-organiser still needs to be enriched with additional data before it is usable. An example of this is that only the identifier of a type in a type filter is saved, rather than the entire object including its name and description. The enrichment will also validate the filters, as certain objects could have been removed.

**TIP** For more detailed information on the API, see [Context View](#).

### 6.2.1. Filter validation

When calling the enrich endpoint the views filters will be validated. Filters that are invalid will be marked with a flag `"valid": false`.

Prior to calling the search endpoint with the retrieved filter data, any filter issues should be resolved. Calling the search end point with invalid filters will result in a **400 Bad Request**.

## 6.3. Performing the search

After enrichment and validation of the filter a search can be triggered to return the data for this view. This can be done in two ways:

- using the paged API,
- using a reactive streaming endpoint.

**TIP** For more detailed information on the API, see [Context Item](#).

### 6.3.1. Paged search API

This end point performs a search with the provided filters and returns a list of results. These results can contain "redacted" results, meaning if there are context items for which the requester does not have the required permissions to view, they will not contain all the data. More information about searching can be found in the context item search section and in the context item section.

### **6.3.2. Reactive endpoints**

See the section about reactive endpoints for more search options.

# Chapter 7. Context type access rules

Each context type has its own set of access rules, which determines what users can do with that context type. An access rules consist of

- a list of permissions, which describes what can be done,
- a subject, which defines who the access rule will apply to.

There are 3 different permissions:

CONTEXT_TYPE_READ	user can read context items of this type
CONTEXT_TYPE_WRITE	user can create and modify context items of this type
CONTEXT_TYPE_DELETE	user can delete context items of this type

For the sake of readability, going forward the prefix CONTEXT\_TYPE\_ will be omitted.

In the current release, the **READ** permission is not enforced, though it is highly recommended already giving users this permission when defining access rules with higher levels of control.

The above permissions can be granted to subjects, the subject types:

GUEST	unauthenticated users
EVERYONE	all authenticated users
GROUP	all users in a given group
USER	a given user

In the current release, the subject types **GUEST** and **GROUP** cannot be used yet.

When defining an access rule on a type, a combination of a subject and a set of permissions needs to be provided. When using the **USER** subject type, a user id also needs to be provided.

**TIP** For more detailed information on the API, see [Access Rules](#).

## 7.1. Subject type priorities

Access rules are prioritized by subject type, meaning an access rule defined on **USER** has precedence over an access rule defined on **EVERYONE** (and so on when the other subject types will be implemented).

### Example 1. Access rules priorities

The following access rules exist:

- permissions **READ**, **WRITE** and **DELETE** on type ANOMALY for subject type **EVERYONE**,
- permissions **READ** and **WRITE** on type ANOMALY for subject type **USER** for user Bob.

This will result in Bob having the permissions **READ** and **WRITE** for that type, while other users also have the **DELETE** permission.

When a new context type is created, a new access rule with subject type **EVERYONE** and permissions **READ** and **WRITE** is automatically created.

**NOTE** | Users with administrator rights will always have all possible permissions available.

# Chapter 8. Context item approvals

Approvals can be set on any context item for which approvals are enabled on its type.

A user can add an approval on a context item and delete said approval as often as needed, though only one approval per user per context item is possible at the same time. An approval will only be valid as long as certain properties of the context item do not change.

Changes (either using ContextHub or the API) that will remove all approvals from a context item include:

- updating the description,
- adding, updating or removing a keyword,
- adding, updating or removing an event,
- updating values for custom fields or other properties,
- and adding or removing an attachment.

Comments do not affect approvals in any way.

Approvals can be enabled or disabled per context item type through ContextHub's config pages or using the type API.

**TIP** | For more detailed information on the API, see [Approvals](#).

# Chapter 9. Context item attachments

To every context item for which the user has view and read permissions, attachments can be uploaded. For this, all mime-types and extensions are allowed. File size is limited to 20MB.

Updating the name, mine-type and extension after creation are only possible through the Context API.

**NOTE** | The attachments on context items are exported when a backup is created.

**TIP** | For more detailed information on the API, see [Attachments](#).



# Chapter 10. Context item comments

Anyone with the permissions to read a certain context item can read all existing comments and add comments.

Modifying or removing own comments is always possible, and will not have an influence on potentially present approvals. Users with admin privileges can remove or modify any comment.

**TIP** For more detailed information on the API, see [Comments](#).

# Chapter 11. Context item history

A history of any changes made to tracked objects is kept on any context item for which history is enabled on its type.

Tracked objects on a context item are:

- the context item itself, its components and its events,
- attachments and comments,
- and approvals.

Creating new, updating, or removing any of the above tracked objects on a history enabled context item will be tracked. This trail can be read by anyone with access to the context item and contains amongst the type of object also the user making the changes, the before and after values, and the date when the changes occurred.

Audit trail can be enabled or disabled per context item type through ContextHub's config pages or using the type API.

**TIP** | For more detailed information on the API, see [History](#).

# Chapter 12. Resources

## 12.1. AccessRules

### 12.1.1. Create new

POST /type/{typeId}/accessrule

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>typeId</b>	The identifier of the ContextItem type.	String
<b>Body required</b>	<b>TypeAccessRuleModel</b>	The access rule model.	<a href="#">TypeAccessRuleModel</a>

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	AccessRule
404	The resource was not found	AccessRule
201	Created	AccessRule
400	Validation failed	ErrorMessage

#### Produces

- application/json
- application/hal+json

### 12.1.2. Delete specific

DELETE /type/{typeId}/accessrule/{accessRuleId}

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>typeId</b>	The identifier of the ContextItem type.	String
<b>Path required</b>	<b>accessRuleId</b>	The identifier of the access rule.	String

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	

204	Deleted	
404	The resource was not found	
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

### 12.1.3. Get specific

GET /type/{typeId}/accessrule/{accessRuleId}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>typeId</b>	The identifier of the ContextItem type.	String
<b>Path required</b>	<b>accessRuleId</b>	The identifier of the access rule.	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	AccessRule
404	The resource was not found	AccessRule
200	OK	AccessRule
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

### 12.1.4. Get all for type

GET /type/{typeId}/accessrule

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>typeId</b>	The identifier of the ContextItem type.	String

### Responses

HTTP Code	Description	Schema
-----------	-------------	--------

401	Unauthorized	List[ <a href="#">AccessRule</a> ]
404	The resource was not found	List[ <a href="#">AccessRule</a> ]
200	OK	List[ <a href="#">AccessRule</a> ]
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.1.5. Update specific

PUT /type/{typeId}/accessrule/{accessRuleId}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>typeId</b>	The identifier of the ContextItem type.	String
<b>Path required</b>	<b>accessRuleId</b>	The identifier of the access rule.	String
<b>Body required</b>	<b>TypeAccessRuleModel</b>	The access rule model.	<a href="#">TypeAccessRuleModel</a>

### Responses

HTTP Code	Description	Schema
401	Unauthorized	AccessRule
404	The resource was not found	AccessRule
201	Created	AccessRule
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.2. Analytics

### 12.2.1. Get all available aggregation operations

GET /analytics/aggregations/{identifier}

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>identifier</b>	Identifier of the field or name of the property	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	CollectionModelAggregationOperationRepresentationModel
404	No aggregations possible on this property or field	ErrorMessage
200	Ok	CollectionModelAggregationOperationRepresentationModel
400	Validation failed	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.2.2. Calculate aggregations

POST /analytics/aggregations/{identifier}/aggregate

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>identifier</b>	Identifier of the field or name of the property	String
<b>Query required</b>	<b>aggregation</b>	Name of the aggregation	
<b>Body required</b>	<b>SearchRequestModel</b>	Search request model	<a href="#">SearchRequestModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	CollectionModelAggregationResultRepresentationModel
404	One or more requested aggregations are not possible on this property or field	ErrorMessage

200	Ok	CollectionModelAggregationResultRepresentationModel
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.3. Approvals

### 12.3.1. Add approval

POST /data/{contextItemId}/approval

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextItemId</b>	The identifier of the ContextItem.	String

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	Approval
403	No rights on this context item	Approval
400	Validation failed	ErrorMessage
201	Approved	Approval
404	EntityModel(s) not found!	ErrorMessage

### Produces

- application/json
- application/hal+json

### 12.3.2. Get all for item

GET /data/{contextItemId}/approval

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextItemId</b>	The identifier of the ContextItem.	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	List[ <a href="#">Approval</a> ]
404	EntityModel(s) not found	ErrorMessage
403	No rights on this context item	List[ <a href="#">Approval</a> ]
200	Ok	List[ <a href="#">Approval</a> ]
400	Validation failed	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.3.3. Revoke approval

DELETE /data/{contextItemId}/approval

## Parameters

Type	Name	Description	Schema
<i>Path required</i>	<b>contextItemId</b>	The identifier of the ContextItem.	String

## Responses

HTTP Code	Description	Schema
204	Revoked	
401	Unauthorized	
403	No rights on this context item	
400	Validation failed	ErrorMessage
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.4. AssetType

### 12.4.1. Add types to collection for assets

PATCH /asset-type



## Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>AssetTypeModel</b>	The model containing the types to add to given assets.	<a href="#">AssetTypeModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	String
404	The resource was not found!	ErrorMessage
204	No Content	String
400	Validation failed	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.4.2. Delete types from collection for assets

DELETE /asset-type

## Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>AssetTypeModel</b>	The model containing the types to delete from given assets.	<a href="#">AssetTypeModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	String
404	The resource was not found!	ErrorMessage
204	No Content	String
400	Validation failed	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.4.3. Get types for assets

POST /asset-type

#### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.
[String]	<b>Body</b> <i>required</i>	<b>AssetModel</b>	

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	AssetType
200	Ok	AssetType

#### Produces

- application/json
- application/hal+json

### 12.4.4. Update entire collection of types for assets

PUT /asset-type

#### Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>AssetTypeModel</b>	The model containing the type collection to set for given assets.	<a href="#">AssetTypeModel</a>

#### Responses

HTTP Code	Description	Schema
-----------	-------------	--------

401	Unauthorized	String
404	The resource was not found!	ErrorMessage
204	No Content	String
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.5. Attachments

### 12.5.1. Create new

POST /data/{contextDataId}/attachments

#### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>contextDataId</b>	The identifier of the context data for which the attachment is created.	String
<b>Query</b> <i>required</i>	<b>name</b>	The name of the attachment.	
<b>Query</b> <i>optional</i>	<b>extension</b>	The extension of the attachment.	
<b>Query</b> <i>optional</i>	<b>type</b>	The mimeType of the attachment.	

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	Attachment
403	Forbidden!	ErrorMessage
201	Created	Attachment
400	Validation failed	ErrorMessage
404	EntityModel(s) not found!	ErrorMessage

### Produces

- /

## 12.5.2. Delete specific

DELETE /data/{contextDataId}/attachments/{identifier}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data the attachment belongs to	String
<b>Path required</b>	<b>identifier</b>	The identifier of the attachment to be deleted.	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	String
403	Forbidden	String
200	Ok	String
404	EntityModel(s) not found!	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.5.3. Download

GET /data/{contextDataId}/attachments/{identifier}/download

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data for which the attachment is created.	String
<b>Path required</b>	<b>identifier</b>	The identifier of the attachment.	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	
403	Forbidden!	ErrorMessage
201	Created	
404	EntityModel(s) not found!	ErrorMessage

## Produces

- /

### 12.5.4. Get all for item

GET /data/{contextDataId}/attachments

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data to fetch all attachments from.	String
<b>Query optional</b>	<b>page</b>	Zero-based page index (0..N)	
<b>Query optional</b>	<b>size</b>	The size of the page to be returned	
<b>Query optional</b>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelAttachment
403	Forbidden!	ErrorMessage
200	Ok	PagedModelAttachment
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.5.5. Get specific

GET /data/{contextDataId}/attachments/{identifier}

#### Parameters

Type	Name	Description	Schema
------	------	-------------	--------

<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data for which the attachment is created.	String
<b>Path required</b>	<b>identifier</b>	The identifier of the attachment.	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	Attachment
403	Forbidden!	ErrorMessage
201	Created	Attachment
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.5.6. Update specific

PUT /data/{contextDataId}/attachments/{identifier}

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data the attachment belongs to	String
<b>Path required</b>	<b>identifier</b>	The identifier of the attachment to be updated.	String
<b>Body required</b>	<b>AttachmentModel</b>	The attachment model.	<a href="#">AttachmentModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	Attachment
403	Forbidden	Attachment
200	Ok	Attachment
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json

- application/hal+json

## 12.6. Comments

### 12.6.1. Create new coment on a context item

POST /data/{contextDataId}/comments

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data for which the comment is created.	String
<b>Body required</b>	<b>CommentModel</b>	The comment model.	<i>CommentModel</i>

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	Comment
201	Created	Comment
400	Validation failed	ErrorMessage
404	EntityModel(s) not found!	ErrorMessage

#### Produces

- application/json
- application/hal+json

### 12.6.2. Delete specific

DELETE /data/{contextDataId}/comments/{identifier}

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data the comment belongs to	String
<b>Path required</b>	<b>identifier</b>	The identifier of the comment to be deleted.	String

#### Responses

HTTP Code	Description	Schema
-----------	-------------	--------

401	Unauthorized	String
403	Forbidden	String
200	Ok	String
404	EntityModel(s) not found!	ErrorMessage

### Produces

- application/json
- application/hal+json

### 12.6.3. Get all comments for context item

GET /data/{contextDataId}/comments

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data to fetch all comments from.	String
<b>Query optional</b>	<b>page</b>	Zero-based page index (0..N)	
<b>Query optional</b>	<b>size</b>	The size of the page to be returned	
<b>Query optional</b>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelComment
200	Ok	PagedModelComment
404	EntityModel(s) not found!	ErrorMessage

### Produces

- application/json
- application/hal+json



## 12.6.4. Get all comments for context items in bulk

POST /data/comments

### Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>IdListModel</b>	object containing a list of context item identifiers	<i>IdListModel</i>

### Responses

HTTP Code	Description	Schema
200	OK	Map[ <i>Comment</i> ]

### Produces

- application/json
- application/hal+json

## 12.6.5. Get specific comment

GET /data/{contextDataId}/comments/{identifier}

### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>contextDataId</b>	The identifier of the context data the comment belongs to	String
<b>Path</b> <i>required</i>	<b>identifier</b>	The identifier of the comment to be fetched.	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	Comment
200	Ok	Comment
404	EntityModel(s) not found!	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.6.6. Update specific

PUT /data/{contextDataId}/comments/{identifier}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data the comment belongs to	String
<b>Path required</b>	<b>identifier</b>	The identifier of the comment to be updated.	String
<b>Body required</b>	<b>CommentModel</b>	The updated comment model.	<a href="#">CommentModel</a>

### Responses

HTTP Code	Description	Schema
401	Unauthorized	Comment
404	The resource was not found!	ErrorMessage
403	Forbidden	Comment
200	Ok	Comment
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.7. ContextData

### 12.7.1. Create new context data

POST /data

### Parameters

Type	Name	Description	Schema
<b>Body required</b>	<b>ContextDataModel</b>	The context data model.	<a href="#">ContextDataModel</a>

### Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextData

404	The resource was not found!	ErrorMessage
201	Created	ContextData
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.7.2. Delete specific context data

DELETE /data/{contextDataId}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data to be deleted.	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	String
200	Ok	String
404	EntityModel(s) not found!	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.7.3. Get all context data

GET /data

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	

<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelContextData
200	Ok	PagedModelContextData

## Produces

- application/json
- application/hal+json

## 12.7.4. Get a specific context data

GET /data/{contextDataId}

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data to be fetched.	String
<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	

## Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextData
200	Ok	ContextData
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.7.5. Search context data

GET /data/search

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>required</i>	<b>query</b>	The search query	
<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelContextData
400	Bad Request	ErrorMessage
200	Ok	PagedModelContextData

### Produces

- application/json
- application/hal+json

## 12.7.6. Update specific context data

PUT /data/{contextDataId}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data to be updated.	String

<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	
<b>Body</b> <i>required</i>	<b>ContextDataModel</b>	The context data model.	<a href="#">ContextDataModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextData
404	The resource was not found!	ErrorMessage
200	Ok	ContextData
400	Validation failed	ErrorMessage

## Produces

- application/json
- application/hal+json

# 12.8. ContextDataEvent

## 12.8.1. Create new

POST /data/{contextDataId}/event

### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>contextDataId</b>	The identifier of the context data for which the event is created.	String
<b>Body</b> <i>required</i>	<b>ContextEventModel</b>	The event model.	<a href="#">ContextEventModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextEvent
201	Created	ContextEvent
400	Validation failed	ErrorMessage
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.8.2. Delete specific

DELETE /data/{contextDataId}/event/{eventId}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data linked to the event to be deleted.	String
<b>Path required</b>	<b>eventId</b>	The identifier of the event to delete	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	String
200	Ok	String
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.8.3. Get all for item

GET /data/{contextDataId}/event

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data to fetch all events from.	String
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	

<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.
---------------------------------	-------------	---	---

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelContextEvent
200	Ok	PagedModelContextEvent
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.8.4. Get specific

GET /data/{contextDataId}/event/{eventId}

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data linked to the event to be fetched.	String
<b>Path required</b>	<b>eventId</b>	The identifier of the event to be fetched	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextEvent
200	Ok	ContextEvent
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json



## 12.8.5. Update specific

PUT /data/{contextDataId}/event/{eventId}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data linked to the event to be updated.	String
<b>Path required</b>	<b>eventId</b>	The identifier of the event that will be updated	String
<b>Body required</b>	<b>ContextEventModel</b>	The updated event model.	<a href="#">ContextEventModel</a>

### Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextEvent
404	The resource was not found!	ErrorMessage
200	Ok	ContextEvent
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.9. ContextDataReference

### 12.9.1. Add a new context data reference to a context data

POST /data/{contextDataId}/reference

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data for which the data reference is created.	String
<b>Query optional</b>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	

<b>Body</b> <i>required</i>	<b>DataReferenceModel</b>		<i>DataReference Model</i>
--------------------------------	---------------------------	--	----------------------------

## Responses

HTTP Code	Description	Schema
401	Unauthorized	DataReference
201	Created	DataReference
400	Validation failed	ErrorMessage
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.9.2. Delete specific

DELETE /data/{contextDataId}/reference/{referenceId}

## Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>contextDataId</b>	The identifier of the context data linked to the reference to be deleted.	String
<b>Path</b> <i>required</i>	<b>referenceId</b>	The identifier of the data reference to be deleted.	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	String
404	The resource was not found!	ErrorMessage
204	Deleted	String

## Produces

- application/json
- application/hal+json

### 12.9.3. Get all for context data references for a specific context data

GET /data/{contextDataId}/reference

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data to fetch all the data references from.	String
<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelDataReference
200	Ok	PagedModelDataReference
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.9.4. Get specific

GET /data/{contextDataId}/reference/{referenceId}

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data linked to the reference to be fetched.	String
<b>Path required</b>	<b>referenceId</b>	The identifier of the data reference to be fetched.	String

<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	
---------------------------------	--------------------------------	--	--

## Responses

HTTP Code	Description	Schema
401	Unauthorized	DataReference
200	Ok	DataReference
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.9.5. Update specific

PUT /data/{contextDataId}/reference/{referenceId}

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context data linked to the reference to be updated.	String
<b>Path required</b>	<b>referenceId</b>	The identifier of the data reference that will be updated.	String
<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	
<b>Body</b> <i>required</i>	<b>DataReferenceModel</b>		<i>DataReference Model</i>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	DataReference
404	The resource was not found!	ErrorMessage
200	Ok	DataReference
400	Validation failed	ErrorMessage

## Produces

- application/json

- application/hal+json

## 12.10. ContextItem

### 12.10.1. Create new

POST /item

#### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	
<b>Body</b> <i>required</i>	<b>ContextItemModel</b>	The context item model	<a href="#">ContextItemModel</a>

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextItem
404	The resource was not found!	ErrorMessage
201	Created	ContextItem
400	Validation failed	ErrorMessage

#### Produces

- application/json
- application/hal+json

### 12.10.2. Create new in batch

POST /item/batch

#### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>async</b>	Indicates if the create in bulk should run async	
<b>Body</b> <i>required</i>	<b>ContextItemModel</b>	Array of context item models to create in batch.	<a href="#">ContextItemModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	String
404	The resource was not found!	ErrorMessage
201	Created	String
400	Validation failed	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.10.3. Delete specific

DELETE /item/{contextDataId}

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextDataId</b>	The identifier of the context item to delete	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	String
200	Ok	String
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.10.4. Delete bulk by definition

DELETE /item/batch/filters

## Parameters

Type	Name	Description	Schema
<b>Body required</b>	<b>BulkDeleteByFiltersModel</b>	Model containing set of filters.	<a href="#">BulkDeleteByFiltersModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	String
202	Accepted: process is running in the background	String
403	Forbidden	String
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.10.5. Delete bulk by identifiers

DELETE /item/batch

### Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>BulkDeleteByIdentifiersModel</b>	Model containing list of identifiers.	<a href="#">BulkDeleteByIdentifiersModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	BulkDeleteResult
403	Forbidden	BulkDeleteResult
200	Ok	BulkDeleteResult
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.10.6. Get specific

GET /item/{contextDataId}

### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>contextDataId</b>	The identifier of the context item to be fetched.	String

<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	
---------------------------------	--------------------------------	--	--

## Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextItem
200	Ok	ContextItem
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.10.7. Search context items

POST /item/search

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>		
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.
[String]	<b>Body</b> <i>required</i>	<b>SearchRequestModel</b>	Model containing the filter objects

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelContextItem



400	Bad Request	ErrorMessage
200	Ok	PagedModelContextItem

### Produces

- application/json
- application/hal+json

## 12.10.8. Minimal search (streaming)

POST /item/minimalSearch

### Parameters

Type	Name	Description	Schema
Query <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	
Query <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
Query <i>optional</i>	<b>size</b>	The size of the page to be returned	
Query <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.
[String]	<b>Body</b> <i>required</i>	<b>ReactiveSearchRequestModel</b>	Model containing the filter objects.

### Responses

HTTP Code	Description	Schema
401	Unauthorized	
200	OK	List[ <a href="#">MinimalContextItem</a> ]
400	Bad Request	ErrorMessage

### Produces

- application/stream+json
- application/json

## 12.10.9. Update specific

PUT /item/{contextDataId}

### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>contextDataId</b>	The identifier of the context item that will be updated	String
<b>Query</b> <i>optional</i>	<b>useTimeSeriesIdentifier</b>	Indicates if time series identifier should be returned for tags references	
<b>Body</b> <i>required</i>	<b>ContextItemModel</b>	The updated context item model.	<a href="#">ContextItemModel</a>

### Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextItem
404	The resource was not found!	ErrorMessage
200	Ok	ContextItem
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.11. ContextItemV2

### 12.11.1. Returns the number of items matching the provided filters.

POST /v2/item/search/count

### Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>SearchRequestModel</b>	Model containing the filters	<a href="#">SearchRequestModel</a>

### Responses

HTTP Code	Description	Schema
401	Unauthorized	CountRepresentationModel

400	Bad Request	ErrorMessage
200	Ok	CountRepresentationModel

### Produces

- application/json

## 12.11.2. Search context items V2

POST /v2/item/search

### Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>CountIndependentSearchRequestModel</b>	Model containing the filter objects and paging information	<a href="#">CountIndependentSearchRequestModel</a>

### Responses

HTTP Code	Description	Schema
401	Unauthorized	TokenizedPageModelContextItem
400	Bad Request	ErrorMessage
200	Ok	TokenizedPageModelContextItem

### Produces

- application/json

## 12.12. ContextView

### 12.12.1. Enrich a saved context view

GET /view/{identifier}/enriched

### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>identifier</b>	The identifier of the context view to be fetched.	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextView

200	Ok	ContextView
404	EntityModel(s) not found!	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.13. DataImportController

### 12.13.1. Delete a file.

`DELETE /import/{id}`

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>id</b>	The id of the file to delete.	String

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	
204	No content	

### 12.13.2. Download a file by id.

`GET /import/{id}/download`

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>id</b>	The id of the file to download.	String

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	
404	Resource(s) not found!	
200	Ok	

### 12.13.3. Download the generated error file for a specific file.

`GET /import/{id}/errors`

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>id</b>	The id of the file for which the error file should be retrieved.	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	
404	Resource(s) not found!	
200	Ok	

## 12.13.4. List all imported files.

GET /import

## Parameters

Type	Name	Description	Schema
<b>Query</b> <i>required</i>	<b>pageable</b>		
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PageDataImport
200	Ok	PageDataImport

## Produces

- /

### 12.13.5. Get a file.

GET /import/{id}

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>id</b>	The id of the file to retrieve.	String

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	DataImport
404	Resource(s) not found!	DataImport
200	Ok	DataImport

#### Produces

- /

### 12.13.6. Upload a file.

POST /import

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>name</b>	The name of the file.	String
<b>Path required</b>	<b>type</b>	The type of the file.	String

#### Responses

HTTP Code	Description	Schema
200	OK	DataImport

#### Produces

- /

## 12.14. Export

### 12.14.1. Check existence

HEAD /export/{identifier}

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>identifier</b>	The identifier of the export.	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	
204	No Content	
404	EntityModel not found	
410	Gone	

## 12.14.2. Remove all expired

DELETE /export/cleanup

### Parameters

Type	Name	Description	Schema
------	------	-------------	--------

### Responses

HTTP Code	Description	Schema
403	Forbidden	
204	Clean up is successful	

## 12.14.3. Create new

POST /export

### Parameters

Type	Name	Description	Schema
<b>Body required</b>	<b>ExportModel</b>	The export creation model.	<a href="#">ExportModel</a>

### Responses

HTTP Code	Description	Schema
401	Unauthorized	TaskSubmissionResult
202	Accepted.	TaskSubmissionResult
404	View not found!	ErrorMessage
403	Forbidden!	ErrorMessage

400	Validation failed	ErrorMessage
-----	-------------------	--------------

### Produces

- application/json
- application/hal+json

## 12.14.4. Get all

GET /export

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelExport
200	Ok	PagedModelExport

### Produces

- application/json
- application/hal+json

## 12.14.5. Get export

GET /export/{identifier}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>identifier</b>	The identifier of the export.	String



## Responses

HTTP Code	Description	Schema
401	Unauthorized	
404	EntityModel not found	
410	Gone	
200	Ok	

## 12.15. FailedEventFrameSynchronization

### 12.15.1. Re-sync one specific contextItem

POST /source/sync/failed/{failedItemId}

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>failedItemId</b>	The identifier of the context item to be re-synced	UUID

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	
200	ok	
400	Validation failed	ErrorMessage

#### Produces

• /

### 12.15.2. Search

GET /source/sync/failed

#### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>query</b>	The search query	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	

<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelFailedEventFrameSynchronization
400	Bad Request	PagedModelFailedEventFrameSynchronization
200	Ok	PagedModelFailedEventFrameSynchronization

## Produces

- /

# 12.16. Fields

## 12.16.1. Create new

POST /fields

## Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>FieldModel</b>	The custom field model.	<i>FieldModel</i>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	Field
201	Created	Field
400	Validation failed	ErrorMessage

## Produces

- application/json

- application/hal+json

## 12.16.2. Delete specific

DELETE /fields/{identifier}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>identifier</b>	The identifier of the custom field to delete	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	String
200	Ok	String
404	EntityModel(s) not found!	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.16.3. Get all

GET /fields

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

### Responses

HTTP Code	Description	Schema
-----------	-------------	--------

401	Unauthorized	PagedModelField
200	Ok	PagedModelField

### Produces

- application/json
- application/hal+json

## 12.16.4. Get specific

GET /fields/{identifier}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>identifier</b>	The identifier of the custom field to be fetched.	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	Field
200	Ok	Field
404	EntityModel(s) not found!	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.16.5. Retrieve a list of fields linked to provided types.

POST /fields/linked

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>fieldTypes</b>	Allowed field types	
<b>Body</b> <i>required</i>	<b>string</b>	List of type identifiers.	[string]

### Responses

HTTP Code	Description	Schema
-----------	-------------	--------

401	Unauthorized	CollectionModelLinkedField
400	Bad Request	ErrorMessage
200	Ok	CollectionModelLinkedField

### Produces

- application/json
- application/hal+json

## 12.16.6. Search fields

GET /fields/search

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>required</i>	<b>query</b>	The search query	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelField
400	Bad Request	ErrorMessage
200	Ok	PagedModelField

### Produces

- application/json
- application/hal+json

## 12.16.7. Update specific

PUT /fields/{identifier}

### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>identifier</b>	The identifier of the custom field that will be updated	String
<b>Body</b> <i>required</i>	<b>FieldModel</b>	The updated custom field model.	<i>FieldModel</i>

### Responses

HTTP Code	Description	Schema
401	Unauthorized	Field
404	The resource was not found!	ErrorMessage
200	Ok	Field
400	Validation failed	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.17. History

### 12.17.1. Get audit trail for item

GET /history/{contextDataId}

### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>contextDataId</b>	The identifier of the ContextItem.	String
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	

<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.
---------------------------------	-------------	---	---

## Responses

HTTP Code	Description	Schema
401	Unauthorized	ErrorMessage
403	Forbidden	ErrorMessage
200	Ok	PagedModelAuditTrail
400	Validation failed	ErrorMessage
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

# 12.18. Keywords

## 12.18.1. Get all

GET /keywords

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

## Responses

HTTP Code	Description	Schema
-----------	-------------	--------

401	Unauthorized	PagedModelKeyword
200	Ok	PagedModelKeyword

### Produces

- application/json
- application/hal+json

## 12.18.2. Search keywords

GET /keywords/search

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>required</i>	<b>query</b>	The search query	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelKeyword
400	Bad Request	ErrorMessage
200	Ok	PagedModelKeyword

### Produces

- application/json
- application/hal+json

## 12.18.3. Validate a list of keywords

POST /keywords



## Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>KeywordModel</b>	A collection of keyword models.	<a href="#">KeywordModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	List[ <a href="#">Keyword</a> ]
400	Validation failed	ErrorMessage
204	No content	List[ <a href="#">Keyword</a> ]

## Produces

- application/json
- application/hal+json

# 12.19. PropertyKeyMigration

## 12.19.1. Download the Property key migration dry-run report

GET /migration/property-key/report

## Parameters

Type	Name	Description	Schema
------	------	-------------	--------

## Responses

HTTP Code	Description	Schema
401	Unauthorized	
404	Resource Not Found	
200	Ok	

## 12.19.2. Get field migrated keys uniqueness

GET /migration/property-key/field-uniqueness

## Parameters

Type	Name	Description	Schema
------	------	-------------	--------

## Responses

HTTP Code	Description	Schema
401	Unauthorized	FieldUniquenessRepresentationModel
200	Ok	FieldUniquenessRepresentationModel

## Produces

- application/json
- application/hal+json

### 12.19.3. Get field dry-run results

GET /migration/property-key/fields

## Parameters

Type	Name	Description	Schema
Query <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
Query <i>optional</i>	<b>size</b>	The size of the page to be returned	
Query <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelPropertyKeyMigrationFieldData
200	Ok	PagedModelPropertyKeyMigrationFieldData

## Produces

- application/json
- application/hal+json

## 12.19.4. Get monitor dry-run results

GET /migration/property-key/monitors

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelPropertyKeyMigrationMonitorData
200	Ok	PagedModelPropertyKeyMigrationMonitorData

### Produces

- application/json
- application/hal+json

## 12.19.5. Get migration status

GET /migration/property-key/status

### Parameters

Type	Name	Description	Schema
------	------	-------------	--------

### Responses

HTTP Code	Description	Schema
401	Unauthorized	PropertyKeyMigrationStatusRepresentationModel
200	Ok	PropertyKeyMigrationStatusRepresentationModel

## Produces

- application/json
- application/hal+json

## 12.19.6. Get view dry-run results

GET /migration/property-key/views

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelPropertyKeyMigrationViewData
200	Ok	PagedModelPropertyKeyMigrationViewData

## Produces

- application/json
- application/hal+json

## 12.19.7. Update phase status

POST /migration/property-key/status

### Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>PropertyKeyMigrationPhaseStatusModel</b>	Object holding the status for the waitingForApproval phase	<a href="#">PropertyKeyMigrationPhaseStatusModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	String
204	No content	String

## Produces

- application/json
- application/hal+json

# 12.20. PublicDownloadGenerator

## 12.20.1. Generate

POST /download/generate

### Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>LinkModel</b>	The model containing the requested link	<a href="#">LinkModel</a>

## Responses

HTTP Code	Description	Schema
200	Ok	DataModel

## Produces

- application/json

# 12.21. Source

## 12.21.1. Get specific source

GET /source/{sourceIdentifier}

### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>sourceIdentifier</b>	The identifier of the source to be synced	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	Source
404	Resource(s) not found!	ErrorMessage
200	Ok	Source

## Produces

• /

## 12.21.2. Search

GET /source/search

## Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>query</b>	The search query	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelSource
400	Bad Request	PagedModelSource
200	Ok	PagedModelSource

## Produces

• /

### 12.21.3. Start historical sync

POST /source/{sourceIdentifier}/sync

#### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>sourceIdentifier</b>	The identifier of the source to be synced	String
<b>Body</b> <i>required</i>	<b>SourceSynchronizationIntervalModel</b>	Model containing the interval.	<a href="#">SourceSynchronizationIntervalModel</a>

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	
202	Accepted	
400	Validation failed	ErrorMessage

#### Produces

- /

### 12.21.4. Update specific source

PUT /source/{sourceIdentifier}

#### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>sourceIdentifier</b>	The identifier of the source to be updated	String
<b>Body</b> <i>required</i>	<b>SourceModel</b>	Model containing the source. Only enabled flag can be changed.	<a href="#">SourceModel</a>

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	Source
404	Resource(s) not found!	Source
200	Ok	Source
400	Validation failed	ErrorMessage

## Produces

• /

# 12.22. SourceSynchronization

## 12.22.1. Do a manual cleanup of one datasource

POST /source/sync/cleanup

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>required</i>	<b>datasourceId</b>	The identifier of the datasource	

### Responses

HTTP Code	Description	Schema
401	Unauthorized	
202	Accepted	
400	Validation failed	ErrorMessage

## Produces

• /

## 12.22.2. Re-sync specific context item

POST /source/sync/{contextItemIdentifier}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>contextItemIdentifier</b>	The identifier of the context item to be re-synced	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextItem
204	No Content	ContextItem
200	Ok	ContextItem
400	Validation failed	ErrorMessage



## Produces

- /

### 12.22.3. Re-sync list of context items

POST /source/sync/filters

#### Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>BulkResyncByFiltersModel</b>	Model containing the list of filters.	<a href="#">BulkResyncByFiltersModel</a>

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	
202	Accepted	
400	Validation failed	ErrorMessage

## Produces

- /

### 12.22.4. Re-sync list of context items

POST /source/sync

#### Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>BulkResyncByIdentifiersModel</b>	Model containing the list of context item identifiers.	<a href="#">BulkResyncByIdentifiersModel</a>

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	
202	Accepted	
400	Validation failed	ErrorMessage

## Produces

- /

## 12.23. SourceSynchronizationInBackground

### 12.23.1. Cancel historical background sync job

DELETE /source/sync/background/{syncJobIdentifier}

#### Parameters

Type	Name	Description	Schema
<i>Path required</i>	<b>syncJobIdentifier</b>	The identifier of the sync job	String

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	BackgroundSyncJobRepresentationModel
400	Bad Request	BackgroundSyncJobRepresentationModel
404	Resource(s) not found!	ErrorMessage
200	Ok	BackgroundSyncJobRepresentationModel

#### Produces

- /

### 12.23.2. Get background sync job

GET /source/sync/background/{syncJobIdentifier}

#### Parameters

Type	Name	Description	Schema
<i>Path required</i>	<b>syncJobIdentifier</b>	The identifier of the sync job	String

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	BackgroundSyncJobRepresentationModel
404	Resource(s) not found!	ErrorMessage
200	Ok	BackgroundSyncJobRepresentationModel

## Produces

- /

### 12.23.3. Search background sync jobs

GET /source/sync/background

#### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>query</b>	The search query	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelBackgroundSyncJobRepresentationModel
200	Ok	PagedModelBackgroundSyncJobRepresentationModel

## Produces

- /

### 12.23.4. Update background sync job status

PUT /source/sync/background/{syncJobIdentifier}

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>syncJobIdentifier</b>	The identifier of the sync job	String

<b>Body</b> <i>required</i>	<b>BackgroundSyncJobStatusModel</b>	Model containing the new status.	<a href="#">BackgroundSyncJobStatusModel</a>
--------------------------------	-------------------------------------	----------------------------------	--

## Responses

HTTP Code	Description	Schema
401	Unauthorized	BackgroundSyncJobRepresentationModel
404	Resource(s) not found!	ErrorMessage
200	Ok	BackgroundSyncJobRepresentationModel

## Produces

- /

# 12.24. Triggers

## 12.24.1. Create new

POST `/type/{typeId}/triggers`

## Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>typeId</b>	The identifier of the context data type for which the trigger is created.	String
<b>Body</b> <i>required</i>	<b>TriggerDefinitionModel</b>	The trigger model.	<a href="#">TriggerDefinitionModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	TriggerDefinition
403	Forbidden!	ErrorMessage
201	Created	TriggerDefinition
400	Validation failed	ErrorMessage
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.24.2. Delete specific

DELETE /type/{typeId}/triggers/{identifier}

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>typeId</b>	The identifier of the context data type the trigger is attached to.	String
<b>Path required</b>	<b>identifier</b>	The identifier of the trigger to delete	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	String
403	Forbidden!	ErrorMessage
200	Ok	String
404	EntityModel(s) not found!	ErrorMessage

### Produces

- application/json
- application/hal+json

## 12.24.3. Get all for type

GET /type/{typeId}/triggers

### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>typeId</b>	The identifier of the context data type the trigger is attached to.	String

### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelTriggerDefinition
403	Forbidden!	ErrorMessage
200	Ok	PagedModelTriggerDefinition
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.24.4. Get specific

GET /type/{typeId}/triggers/{identifier}

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>typeId</b>	The identifier of the context data type the trigger is attached to.	String
<b>Path required</b>	<b>identifier</b>	The identifier of the trigger to be fetched.	String

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	TriggerDefinition
403	Forbidden!	ErrorMessage
200	Ok	TriggerDefinition
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.24.5. Update specific

PUT /type/{typeId}/triggers/{identifier}

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>typeId</b>	The identifier of the context data type the trigger is attached to.	String
<b>Path required</b>	<b>identifier</b>	The identifier of the trigger to update	String
<b>Body required</b>	<b>TriggerDefinitionModel</b>	The updated trigger model.	<a href="#">TriggerDefiniti onModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	TriggerDefinition
403	Forbidden!	ErrorMessage
400	Validation failed	ErrorMessage
204	No content	TriggerDefinition
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

# 12.25. Type

## 12.25.1. Create new

POST /type

## Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>ContextDataTypeModel</b>	The context date type model.	<a href="#">ContextDataTy peModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextDataType
201	Created	ContextDataType
400	Validation failed	ErrorMessage
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.25.2. Delete specific context data type.

DELETE /type/{identifier}

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>identifier</b>	The identifier of the type to delete	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	String
200	Ok	String
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.25.3. Get all context data types

GET /type

## Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>assetId</b>	When provided, only types allowed for this asset are returned.	
<b>Query</b> <i>optional</i>	<b>hasPermission</b>	Type permissions the user needs to have. Possible values: CONTEXT_TYPE_WRITE or CONTEXT_TYPE_DELETE. When CONTEXT_TYPE_DELETE is provided, CONTEXT_TYPE_WRITE is automatically assumed.	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	



<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.
---------------------------------	-------------	---	---

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelContextDataTy pe
200	Ok	PagedModelContextDataTy pe

## Produces

- application/json
- application/hal+json

### 12.25.4. Get specific

GET /type/{identifier}

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>identifier</b>	The identifier of the type to be fetched	String

## Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextDataType
200	Ok	ContextDataType
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

### 12.25.5. Search context data types.

GET /type/search

## Parameters

Type	Name	Description	Schema
<b>Query</b> <i>required</i>	<b>query</b>	The search query	
<b>Query</b> <i>optional</i>	<b>hasPermission</b>	Type permissions the user needs to have. Possible values: CONTEXT_TYPE_WRITE or CONTEXT_TYPE_DELETE. When CONTEXT_TYPE_DELETE is provided, CONTEXT_TYPE_WRITE is automatically assumed.	
<b>Query</b> <i>optional</i>	<b>includeDeleted</b>	Whether to include deleted types or not.	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelContextDataTy pe
400	Bad Request	ErrorMessage
200	Ok	PagedModelContextDataTy pe

## Produces

- application/json
- application/hal+json

### 12.25.6. Search context data types using a POST request.

POST /type/search

## Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>SearchParametersModel</b>	Search request model.	<a href="#">SearchParametersModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelContextDataType
400	Bad Request	ErrorMessage
200	Ok	PagedModelContextDataType

## Produces

- application/json
- application/hal+json

## 12.25.7. Update specific

PUT /type/{identifier}

## Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>identifier</b>	The identifier of the type that will be updated	String
<b>Body</b> <i>required</i>	<b>ContextDataTypeModel</b>	The updated context date type model.	<a href="#">ContextDataTypeModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	ContextDataType
404	The resource was not found!	ErrorMessage
200	Ok	ContextDataType
400	Validation failed	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.26. Workflow

### 12.26.1. Create new

POST /workflow

#### Parameters

Type	Name	Description	Schema
<b>Body</b> <i>required</i>	<b>WorkflowModel</b>	The workflow model.	<a href="#">WorkflowModel</a>

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	Workflow
201	Created	Workflow
400	Validation failed	ErrorMessage

#### Produces

- application/json
- application/hal+json

### 12.26.2. Delete specific workflow.

DELETE /workflow/{identifier}

#### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>identifier</b>	The identifier of the work flow to delete.	String

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	String
204	No content	String
404	EntityModel(s) not found!	ErrorMessage

#### Produces

- application/json
- application/hal+json

### 12.26.3. Get all workflows

GET /workflow

#### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelWorkflow
200	Ok	PagedModelWorkflow

#### Produces

- application/json
- application/hal+json

### 12.26.4. Gets specific

GET /workflow/{identifier}

#### Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>identifier</b>	The identifier of the workflow to be fetched.	String

#### Responses

HTTP Code	Description	Schema
401	Unauthorized	Workflow
200	Ok	Workflow
404	EntityModel(s) not found!	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.26.5. Search for workflows

GET /workflow/search

### Parameters

Type	Name	Description	Schema
<b>Query</b> <i>required</i>	<b>query</b>	The search query	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

### Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelWorkflow
400	Bad Request	ErrorMessage
200	Ok	PagedModelWorkflow

## Produces

- application/json
- application/hal+json

## 12.26.6. Search for workflow states

GET /workflow/states/search

### Parameters

Type	Name	Description	Schema
------	------	-------------	--------

<b>Query</b> <i>required</i>	<b>query</b>	The search query	
<b>Query</b> <i>optional</i>	<b>page</b>	Zero-based page index (0..N)	
<b>Query</b> <i>optional</i>	<b>size</b>	The size of the page to be returned	
<b>Query</b> <i>optional</i>	<b>sort</b>	Sorting criteria in the format: property(,asc	desc). Default sort order is ascending. Multiple sort criteria are supported.

## Responses

HTTP Code	Description	Schema
401	Unauthorized	PagedModelString
400	Bad Request	ErrorMessage
200	Ok	PagedModelString

## Produces

- application/json
- application/hal+json

## 12.26.7. Update workflow

PUT /workflow/{identifier}

### Parameters

Type	Name	Description	Schema
<b>Path</b> <i>required</i>	<b>identifier</b>	The identifier of the workflow that will be updated.	String
<b>Body</b> <i>required</i>	<b>WorkflowModel</b>	The updated workflow model.  Can only add new intermediate states to the workflow model and update the name.  Deletion of states is not allowed. Start- and end-states can not be updated.  Identifier can not be updated	<a href="#">WorkflowModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	Workflow
404	The resource was not found!	ErrorMessage
200	Ok	Workflow
400	Validation failed	ErrorMessage

## Produces

- application/json
- application/hal+json

## 12.26.8. Update workflow states

PUT /workflow/{identifier}/states

## Parameters

Type	Name	Description	Schema
<b>Path required</b>	<b>identifier</b>	The identifier of the workflow that will be updated.	String
<b>Body required</b>	<b>WorkflowStatesModel</b>	The updated workflow states model.	<a href="#">WorkflowStatesModel</a>

## Responses

HTTP Code	Description	Schema
401	Unauthorized	WorkflowStates
404	The resource was not found!	ErrorMessage
200	Ok	WorkflowStates
400	Validation failed	ErrorMessage

## Produces

- application/json
- application/hal+json



# Chapter 13. Models

## 13.1. *AccessRule*

Represents an access rule definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the access rule.	
objectId	X	String	The identifier of the object.	
paths	X	List of <a href="#">[string]</a>	An array of paths towards the object.	
subjectId	X	String	The identifier of the subject.	
subjectType	X	String	The type of the subject: USER, GROUP, EVERYBODY, GUEST.	<i>Enum:</i> GUEST, EVERYONE, GROUP, USER,
permissions	X	Set of <a href="#">[string]</a>	A set of permissions the subject has on the specified object. Can be an empty set.	<i>Enum:</i>
userDetailsResource		PermissionUserDetails		
links		List of <a href="#">Link</a>		

## 13.2. *AggregationDefinition*

The aggregations that are defined on this field

Field Name	Required	Type	Description	Format
name		String		

## 13.3. *AggregationOperationRepresentationModel*

Field Name	Required	Type	Description	Format
name		String		

Field Name	Required	Type	Description	Format
queryType		String		Enum: MEDIAN, MOST_COMMON, LEAST_COMMON, REGULAR,
returnType		String		Enum: NUMERIC, STRING, DURATION,
links		List of <i>Link</i>		

### 13.4. AggregationResultRepresentationModel

Field Name	Required	Type	Description	Format
operation		AggregationOperationRepresentationModel		
value		Object		
links		List of <i>Link</i>		

### 13.5. Approval

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the approval.	
createdDate		Date	The date when the approval was given.	date-time
createdBy		String	The approver's user id.	
userDetails		UserDetails		
links		List of <i>Link</i>		

## 13.6. *ApprovalFilter*

Represents an Approval filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
withApprovals		Boolean	Filter context items with or with an approvals	

## 13.7. ApprovalFilterAllOf

Field Name	Required	Type	Description	Format
withApprovals		Boolean	Filter context items with or with an approvals	

## 13.8. Asset

Represents an asset

Field Name	Required	Type	Description	Format
identifier		UUID	The asset's identifier.	uuid
paths		List of <a href="#">string</a>	The paths towards this asset starting from the root.	
longPaths		List of <a href="#">string</a>	The old longer paths towards this asset starting from the root.	
parentNames		List of <a href="#">string</a>	The names of the parents, in the same order as the paths.	
name		String	The asset's name.	
description		String	The asset's description.	
type		String	The asset's type.	
dataType		String	The asset's data type, if the Asset is an ATTRIBUTE.	
data		String	The asset's data, if the Asset is an ATTRIBUTE. (tagName or UUID)	
timeSeriesDefinition		TimeSeriesDefinitionRepresentation		
deleted		Boolean	Indication whether this asset is deleted or not.	
externalId		String	The asset's identifier in the external system.	

Field Name	Required	Type	Description	Format
template		String	The asset's template.	
templateId		String	The asset's template identifier.	
source		Source		
permissions		List of <a href="#">[string]</a>	The asset's permissions.	<i>Enum:</i>
links		List of <a href="#">Link</a>		

## 13.9. AssetModel

Collection of assets

Field Name	Required	Type	Description	Format
assets		Set of <a href="#">[UUID]</a>	Set of asset identifiers	uuid

## 13.10. AssetType

Represents a mapping between types and their allowed assets

Field Name	Required	Type	Description	Format
types		PagedModelContextDataType		
inheritance		String	Inheritance type of the rule.	<i>Enum:</i> ALL, SELF, INHERITED,
inherited		List of <a href="#">Asset</a>	Assets of which the rules were inherited	
links		List of <a href="#">Link</a>		

## 13.11. AssetTypeModel

Collections of assets and types

Field Name	Required	Type	Description	Format
assets		Set of <a href="#">[UUID]</a>	Set of asset identifiers	uuid
types		Set of <a href="#">[string]</a>	Set of type identifiers	

## 13.12. Attachment

Represents a attachment definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the attachment	
name	X	String	The name of the attachment	
type		String	The mine type of the attachment	
extension		String	The extension of the attachment	
createdDate		Date	The date when the attachment was created	date-time
lastModifiedDate		Date	The date of the last modification of the attachment	date-time
createdBy		String	The user who created the attachment	
lastModifiedBy		String	The user who made the latest update to the attachment	
links		List of <a href="#">Link</a>		

## 13.13. AttachmentModel

Represents an attachment definition

Field Name	Required	Type	Description	Format
name	X	String	The name of the attachment	
extension		String	The extension of the attachment	
type		String	The type of the attachment	

## 13.14. AuditTrail

Represents an audit trail definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the audit trail	
contextDataIdentifier		String	The identifier of the context item	
changedItemIdentifier		String	The identifier of the changed entity	
changedItemType		String	The type of the changed entity	
changedDate		Date	The date at which the changes occurred	date-time



Field Name	Required	Type	Description	Format
changedBy		String	The identifier of the user who made the changes	
action		String	The action performed	
changes		List of <a href="#">[map]</a>	The changes	
userDetails		UserDetails		
links		List of <a href="#">Link</a>		

### 13.15. *BackgroundSyncJobRepresentationModel*

Represents an interval to synchronize in the background

Field Name	Required	Type	Description	Format
identifier		UUID	The identifier of this sync job	uuid
source		Source		
type		String	Type of background sync	<i>Enum:</i> EXCESSIVE, HISTORICAL, CLEANUP_NIGHTLY, CLEANUP_MANUAL, CLEANUP_AFTER_AF,
status		String	Status of this background sync.	<i>Enum:</i> WAITING_FOR_SCHEDULING, SCHEDULED, WAITING, RUNNING, DONE, CANCELLED, FAILED,

Field Name	Required	Type	Description	Format
intervalStart		Date	Lower end of the interval to be synced	date-time
intervalEnd		Date	Upper end of the interval to be synced	date-time
intervalCount		Integer	Number of items in the interval to be synced	int32
intervalPage		Integer	Current page of the interval being synced	int32
submitDate		Date	Date this background interval was submitted to the queue	date-time
startDate		Date	Date the synchronization process started	date-time
endDate		Date	Date the synchronization process ended	date-time
holdDate		Date	Date the synchronization process was put on hold	date-time
resumeDate		Date	Date the synchronization process resumed	date-time
holdCount		Integer	Number of times the synchronization process was put on hold	int32
lastSyncedItemDate		Date	Last modified date of the last synced item	date-time
exceptionMessage		String	Exception message, should one have occurred in the current interval	
data		Map of <a href="#">[object]</a>	Additional data related to the job	
cancelable		Boolean		
links		List of <a href="#">Link</a>		

## 13.16. *BackgroundSyncJobStatusModel*

Represents a status for a background sync interval.

Field Name	Required	Type	Description	Format
status		String	Status of this background sync.	Enum: WAITING_FOR_SCHEDULING, SCHEDULED, WAITING, RUNNING, DONE, CANCELLED, FAILED,

### 13.17. *BulkDeleteByFiltersModel*

Represents the definition for a context item bulk delete by filters action

Field Name	Required	Type	Description	Format
filters	X	List of <a href="#">ComponentFilter</a> , <a href="#">CreatedByFilter</a> , <a href="#">CreatedDateFilter</a> , <a href="#">CurrentStateFilter</a> , <a href="#">DescriptionFilter</a> , <a href="#">DurationFilter</a> , <a href="#">EnumerationFieldFilter</a> , <a href="#">IntervalFilter</a> , <a href="#">KeywordFilter</a> , <a href="#">LiveIntervalFilter</a> , <a href="#">NumericFieldFilter</a> , <a href="#">PeriodFilter</a> , <a href="#">PropertyFieldFilter</a> , <a href="#">StringFieldFilter</a> , <a href="#">TypeFilter</a> >	List of filters to which context items have to match to be deleted. Mutually exclusive with identifiers field.	
createdBefore	X	Date	Date before which context items need to be created to be deleted when using the filters. Mutually exclusive with identifiers field. Will default to request date.	date-time

### 13.18. *BulkDeleteByIdentifiersModel*

Represents the definition for a context item bulk delete by identifiers action

Field Name	Required	Type	Description	Format
identifiers	X	List of <a href="#">[UUID]</a>	List of context item identifiers to be deleted. Mutually exclusive with the other fields.	uuid

### 13.19. *BulkDeleteResult*

Represents the result of a bulk delete action.

Field Name	Required	Type	Description	Format
deleted		Integer	The number of deleted context items.	int32
remaining		Integer	The number of remaining context item created before the provided date.	int32

### 13.20. *BulkResyncByFiltersModel*

Represents the definition for a context item resync by filters action

Field Name	Required	Type	Description	Format
filters	X	List of <a href="#">ComponentFilter, CreatedByFilter, CreatedDateFilter, CurrentStateFilter, DescriptionFilter, DurationFilter, EnumerationFieldFilter, IntervalFilter, KeywordFilter, LiveIntervalFilter, NumericFieldFilter, PeriodFilter, PropertyFieldFilter, StringFieldFilter, TypeFilter</a> >	List of filters to which context items have to match to be resynced.	
createdBefore	X	Date	Date before which context items need to be created to be resync when using the filters. Will default to request date.	date-time

### 13.21. *BulkResyncByIdentifiersModel*

Represents the definition for a context item resync by identifiers action

Field Name	Required	Type	Description	Format
identifiers	X	List of <a href="#">[UUID]</a>	List of context item identifiers to be resynced.	uuid

### 13.22.

#### *CollectionModelAggregationOperationRepresentationModel*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">AggregationOperationRepresentationModel</a>		

### 13.23.

#### *CollectionModelAggregationResultRepresentationModel*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">AggregationResultRepresentationModel</a>		

### 13.24. *CollectionModelLinkedField*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">LinkedField</a>		

### 13.25. *ColumnSettingsView*

Settings for the table view (frontend related)

Field Name	Required	Type	Description	Format
general		List of <a href="#">GeneralColumn</a>		
fields		List of <a href="#">FieldModelColumn</a>		

## 13.26. Comment

Represents a comment definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the comment	
content	X	String	The content of the comment	
createdDate		Date	The date when the comment was created	date-time
lastModifiedDate		Date	The date of the last modification of the comment	date-time
createdBy		String		
lastModifiedBy		String		
userDetails		UserDetails		
links		List of <a href="#">Link</a>		

## 13.27. CommentModel

Represents a comment definition

Field Name	Required	Type	Description	Format
content	X	String	The comment content	

## 13.28. ComponentFilter

Represents a Component filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,



Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
mode		String		
components		List of <a href="#">ComponentValue</a>		
includeAncestors		Boolean		
componentResources		List of <a href="#">ContextItemComponent</a>		
numComponents		Integer		int32
resolvedComponents		Set of <a href="#">[string]</a>		

### 13.29. *ComponentFilterAllOf*

Field Name	Required	Type	Description	Format
mode		String		
components		List of <a href="#">ComponentValue</a>		
includeAncestors		Boolean		
componentResources		List of <a href="#">ContextItemComponent</a>		
numComponents		Integer		int32
resolvedComponents		Set of <a href="#">[string]</a>		

### 13.30. *ComponentValue*

Field Name	Required	Type	Description	Format
identifier		String		
path		String		
type	X	String		

Field Name	Required	Type	Description	Format
include		String		<i>Enum:</i> SELF, ANCESTORS, DESCENDANTS,

### 13.31. ContextData

Represents a context data definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the context data	
shortKey		String	The short key of the context event	
description		String	The description of the context data	
externalId		String	The external identifier of the context data	
type		ContextDataType		
keywords		Set of <a href="#">[string]</a>	The keywords for this context data	
fields		Map of <a href="#">[object]</a>	The optional fields as their key and values	
permissions		Set of <a href="#">[string]</a>	Permissions on the context data	
links		List of <a href="#">Link</a>		

### 13.32. ContextDataModel

Represents a context data definition

Field Name	Required	Type	Description	Format
type	X	ContextDataType Model		
description		String	The description of the context data	
externalId		String	The external id of the context data	
keywords	X	Set of <a href="#">[string]</a>	The keywords the context data is tagged with	
fields		Map of <a href="#">[object]</a>	The optional fields as their key and values	

## 13.33. ContextDataType

Represents a context data type definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the context data type	
name		String	The name of the context data type	
workflow		Workflow		
color		String	The color of the context data type in hex value	
icon		String	The icon name of the context data type	
externalId		String	The external id of the context data type	
deleted		Boolean	Boolean that indicates if the context data type is deleted	
deletable		Boolean	Boolean that indicates if the context data type is deletable	
approvalsEnabled		Boolean	Boolean that indicates whether approvals are enabled or not	
auditTrailEnabled		Boolean	Boolean that indicates whether audit trail is enabled or not	
fields		List of <i>Field</i>		
permissions		Set of [string]	Set of permissions the current user has on this type	
fieldLabelIdentifier		String	The id of the custom field that will be used as a label on gantt view	
source		Source		
links		List of <i>Link</i>		

## 13.34. ContextDataTypeModel

Represents a context data type definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the context data type. If empty, one will be generated	
source		SourceModel		
externalId		String	The identifier of the type in the external source system	
name	X	String	The name of the context data type	

Field Name	Required	Type	Description	Format
workflow		WorkflowModel		
color		String	The color of the context data type in hex value	
icon	X	String	The icon name of the context data type	
fields		List of <i>FieldModel</i>	The custom fields linked to this context data type	
deletable		Boolean	If the ContextDataType is deletable or not	
approvalsEnabled		Boolean	Whether the ContextDataType has approvals enabled or not	
auditTrailEnabled		Boolean	Whether the ContextDataType has audit trail enabled or not	
fieldLabelIdentifier		UUID	The id of the custom field that will be used as a label on gantt view	uuid

### 13.35. ContextEvent

Represents a context event definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the context event	
occurred		Date	The timestamp when this event took place	date-time
state		String	The state of this event	
links		List of <i>Link</i>		

### 13.36. ContextEventModel

Represents a context event definition

Field Name	Required	Type	Description	Format
state		String	The state of the event, can be stateless if the context item does not describe a workflow	
occurred	X	Date	The date on which the event occurred	date-time

## 13.37. ContextItem

Field Name	Required	Type	Description	Format
permissions		Set of <a href="#">[string]</a>	Permissions on the context data	
identifier		String	The identifier of the context item	
shortKey		String	The short key of the context item	
externalId		String	The external system's original id	
description		String	The description of the context item	
events		List of <a href="#">ContextEvent</a>	The events of the context item	
lastModifiedDate		Date	The date of the most recent event of the context item	date-time
createdBy		String	The uuid of the user who created the context item	
createdDate		Date	The date this context item was created	date-time
userDetails		UserDetails		
components		List of <a href="#">ContextItemComponent</a>	The components for which the context item is registered	
type		ContextDataType		
workFlowDescribed		Boolean	True if the context item type has a workflow	
fields		Map of <a href="#">[object]</a>	Contains the optional fields as their key and values	
keywords		Set of <a href="#">[string]</a>	The keywords the context data is tagged with	
startEventDate		Date	The date of the start event of the context item	date-time
endEventDate		Date	The date of the end event	date-time
lastEventState		String	The state of the most recent event of the context item	
totalDuration		String	The total duration of the context item	
links		List of <a href="#">Link</a>		

## 13.38. ContextItemComponent

The component for which the context item is registered

Field Name	Required	Type	Description	Format
identifier		String		
name		String		
description		String		
type		String		
reference		String		
parentName		String		
path		String		
permissions		Set of <a href="#">[string]</a>		
deleted		Boolean		
include		String		<i>Enum:</i> SELF, ANCESTORS, DESCENDANTS,

### 13.39. *ContextItemComponentModel*

Represents a data reference definition

Field Name	Required	Type	Description	Format
type	X	String	The data reference type	
reference	X	String	The id of the referenced data	
identifier		String	The identifier of the data reference	

### 13.40. *ContextItemEventModel*

Represents a context event definition

Field Name	Required	Type	Description	Format
state		String	The state of the event, can be stateless if the context item does not describe a workflow	
occurred	X	Date	The date on which the event occurred	date-time
identifier		String	The identifier of the context event	

## 13.41. ContextItemModel

Represents a context item definition

Field Name	Required	Type	Description	Format
description		String	The description of the context item	
externalId		String	The external system's original id	
events	X	List of <a href="#">ContextItemEventModel</a>	The events of the context item	
components		List of <a href="#">ContextItemComponentModel</a>	The components of the context item	
type	X	ContextDataTypeModel		
keywords		Set of <a href="#">[string]</a>	The keywords the context data is tagged with	
fields		Map of <a href="#">[object]</a>	Contains the optional fields as their key and values	

## 13.42. ContextView

Represents a context view definition

Field Name	Required	Type	Description	Format
description		String		
folder		Boolean		
identifier		String		
lastModifiedDate		Date		date-time
name		String		
owner		String		
ownerUuid		String		
shared		Boolean		
data		ContextViewData		
permissions		List of <a href="#">[string]</a>		
parentId		String		
links		List of <a href="#">Link</a>		

## 13.43. *ContextViewData*

Field Name	Required	Type	Description	Format
filters		List of <a href="#">ComponentFilter</a> , <a href="#">CreatedByFilter</a> , <a href="#">CreatedDateFilter</a> , <a href="#">CurrentStateFilter</a> , <a href="#">DescriptionFilter</a> , <a href="#">DurationFilter</a> , <a href="#">EnumerationFieldFilter</a> , <a href="#">IntervalFilter</a> , <a href="#">KeywordFilter</a> , <a href="#">LiveIntervalFilter</a> , <a href="#">NumericFieldFilter</a> , <a href="#">PeriodFilter</a> , <a href="#">PropertyFieldFilter</a> , <a href="#">StringFieldFilter</a> , <a href="#">TypeFilter</a> >	List of filters	
viewType		String	Type of the view (frontend related)	
columnSettings		ColumnSettingsView		
ganttSettings		Object	Settings for the gantt view (frontend related)	
sortSettings		Object	Sort settings	
scatterSettings		Object	Scatter settings	
gridSettings		Object	Grid settings	

## 13.44. *CountIndependentSearchRequestModel*

Represents a search request definition used in the count independent search.



Field Name	Required	Type	Description	Format
filters		List of <a href="#">ComponentFilter</a> , <a href="#">CreatedByFilter</a> , <a href="#">CreatedDateFilter</a> , <a href="#">CurrentStateFilter</a> , <a href="#">DescriptionFilter</a> , <a href="#">DurationFilter</a> , <a href="#">EnumerationFieldFilter</a> , <a href="#">IntervalFilter</a> , <a href="#">KeywordFilter</a> , <a href="#">LiveIntervalFilter</a> , <a href="#">NumericFieldFilter</a> , <a href="#">PeriodFilter</a> , <a href="#">PropertyFieldFilter</a> , <a href="#">StringFieldFilter</a> , <a href="#">TypeFilter</a> >		
sortDirection		String	The sorting direction.	
sortProperties		List of <a href="#">[string]</a>	List of properties to sort on. The order of the list is the sorting order. First sort on Index 0 then on 1 to n-1.	
fetchSize		Integer	Number of items to fetch.	int32
continuationToken		String	Token generated by the service that represent a marker for where the previous page ended.	

### 13.45. *CountRepresentationModel*

Field Name	Required	Type	Description	Format
totalCount		Long		int64

### 13.46. *CreatedByFilter*

Represents a CreatedBy filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <i>ValidationResultMessage</i>		
order		Integer		int32
users	X	List of [string]		
userDetails		List of <i>UserDetails</i>		

### 13.47. *CreatedByFilterAllOf*

Field Name	Required	Type	Description	Format
users		List of [string]		
userDetails		List of <i>UserDetails</i>		

### 13.48. *CreateDateFilter*

Represents a created date filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <i>ValidationResultMessage</i>		
order		Integer		int32
createdDate	X	Date		date-time
operator	X	String		<i>Enum:</i> EQUAL, NOT_EQUAL, LESS_THAN, LESS_THAN_OR_EQUAL, GREATER_THAN, GREATER_THAN_OR_EQUAL,

### 13.49. CreatedDateFilterAllOf

Field Name	Required	Type	Description	Format
createdDate		Date		date-time
operator		String		<i>Enum:</i> EQUAL, NOT_EQUAL, LESS_THAN, LESS_THAN_OR_EQUAL, GREATER_THAN, GREATER_THAN_OR_EQUAL,

## 13.50. *CurrentStateFilter*

Represents a Current state filter definition





Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
states		List of <a href="#">string</a>		
mode		String		

### 13.51. *CurrentStateFilterAllOf*

Field Name	Required	Type	Description	Format
states		List of <a href="#">string</a>		
mode		String		

### 13.52. *DataImport*

Field Name	Required	Type	Description	Format
id		String		
name		String		
type		String		
created		Date		date-time
createdBy		String		
modified		Date		date-time
status		String		<i>Enum:</i> UPLOADED, COMPLETED, COMPLETED_WITH_ERRORS, FAILED,
errors		Long		int64
errorMessage		String		
parameters		List of <a href="#">Parameter</a>		

Field Name	Required	Type	Description	Format
errorsFileName		String		
fileName		String		

### 13.53. *DataModel*

Field Name	Required	Type	Description	Format
data		String	the signed link data	

### 13.54. *DataReference*

Represents a data reference definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the data reference	
type	X	String	The data reference type	
reference	X	String	The id of the referenced data	
links		List of <i>Link</i>		

### 13.55. *DataReferenceModel*

Represents a data reference definition

Field Name	Required	Type	Description	Format
type	X	String	The data reference type	
reference	X	String	The id of the referenced data	

### 13.56. *DescriptionFilter*

Represents a Description filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <i>ValidationResultMessage</i>		
order		Integer		int32
values		List of [string]		
mode		String		

### 13.57. *DescriptionFilterAllOf*

Field Name	Required	Type	Description	Format
values		List of [string]		
mode		String		

### 13.58. *DurationCondition*

Represents a duration condition definition

Field Name	Required	Type	Description	Format
value		String		
operator	X	String		<i>Enum:</i> EQUAL, NOT_EQUAL, LESS_THAN, LESS_THAN_OR_EQUAL, GREATER_THAN, GREATER_THAN_OR_EQUAL,

### 13.59. *DurationFilter*

Represents a value filter definition





Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
conditions	X	List of <a href="#">DurationCondition</a>		

### 13.60. *DurationFilterAllOf*

Field Name	Required	Type	Description	Format
conditions		List of <a href="#">DurationCondition</a>		

### 13.61. *EnumerationFieldFilter*

Represents an Enumeration Field filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
fieldIdentifier		UUID		uuid
field		String		
values		List of <a href="#">[string]</a>		
customField		Field		
mode		String		

### 13.62. *EnumerationFieldFilterAllOf*

Field Name	Required	Type	Description	Format
fieldIdentifier		UUID		uuid
field		String		
values		List of <a href="#">[string]</a>		
customField		Field		
mode		String		

### 13.63. *ErrorDetail*

Field Name	Required	Type	Description	Format
defaultMessage		String		

### 13.64. *ErrorMessage*

Represents a tm-context exception



Field Name	Required	Type	Description	Format
statusCode		String	The httpStatus code of the exception	Enum: 100 CONTINUE, 101 SWITCHING_PROTOCOLS, 102 PROCESSING, 103 CHECKPOINT, 200 OK, 201 CREATED, 202 ACCEPTED, 203 NON_AUTHORITATIVE_INFORMATION, 204 NO_CONTENT, 205 RESET_CONTENT, 206 PARTIAL_CONTENT, 207 MULTI_STATUS, 208 ALREADY_REPORTED, 226 IM_USED, 300 MULTIPLE_CHOICES, 301 MOVED_PERMANENTLY, 302 FOUND, 302 MOVED
				19

Field Name	Required	Type	Description	Format
errorCode		String	The specific error code of the exception	
message		String	The exception message	
details		Object	The details of the exception	

## 13.65. *Export*

Represents an export

Field Name	Required	Type	Description	Format
identifier		UUID		uuid
type		String		<i>Enum:</i> CSV_CO MMA_SE PARATE D, CSV_SE MICOLO N_SEPA RATED, XLSX, ZIP,
downloadName		String		
createdBy		String		
createdDate		Date		date- time
lastDownloadDate		Date		date- time
links		List of <i>Link</i>		

## 13.66. *ExportModel*

Represents an export creation definition

Field Name	Required	Type	Description	Format
viewIdentifier	X	String	The identifier of the view to be exported	



Field Name	Required	Type	Description	Format
type	X	String	The type of the export	<i>Enum:</i> CSV_CO MMA_SE PARATE D, CSV_SE MICOLO N_SEPA RATED, XLSX, ZIP, CSV_CO MMA_SE PARATE D,CSV_S EMICOL ON_SEP ARATED ,XLSX,
timezone		String	The timezone in which to display all times	
hostname		String	The TrendMiner hostname	

### 13.67. *FailedEventFrameSynchronization*

Represents a source and its synchronization status

Field Name	Required	Type	Description	Format
identifier		UUID	The identifier of the failed event frame synchronization	uuid
source		Source		
externalId		String	The identifier of the event frame in the source system	
failureDate		Date	The date when the failure of syncing occurred	date-time
exceptionMessage		String	The failure's exception message	
links		List of <i>Link</i>		

### 13.68. *Field*

Represents a custom field definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the field	
propertyKey		String	The technical identifier of the field	
name	X	String	The name of the field	
type	X	String	The field type, must be one of these values: STRING, NUMERIC, ENUMERATION	
placeholder		String	The placeholder of the field, an example value for what this field's value might look like	
options		List of <a href="#">[string]</a>	The possible options of the field, only applicable when type Enumeration is chosen	
aggregations		List of <a href="#">AggregationDefinition</a>	The aggregations that are defined on this field	
links		List of <a href="#">Link</a>		

## 13.69. *FieldFilter*



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
fieldIdentifier		UUID		uuid
field		String		
order		Integer		int32

## 13.70. FieldModel

Represents a custom field definition

Field Name	Required	Type	Description	Format
identifier		UUID	The identifier of the field	uuid
propertyKey	X	String	The property key of the field	
name	X	String	The name of the field	
type	X	String	The field type, must be one of these values: STRING, NUMERIC, ENUMERATION	
placeholder		String	The placeholder of the field, an example value for what this field's value might look like	
options		List of <a href="#">[string]</a>	The possible options of the field, only applicable when type Enumeration is chosen	

## 13.71. FieldModelColumn

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the field	
propertyKey		String	The technical identifier of the field	
name	X	String	The name of the field	
type	X	String	The field type, must be one of these values: STRING, NUMERIC, ENUMERATION	
placeholder		String	The placeholder of the field, an example value for what this field's value might look like	

Field Name	Required	Type	Description	Format
options		List of <a href="#">[string]</a>	The possible options of the field, only applicable when type Enumeration is chosen	
aggregations		List of <a href="#">AggregationDefinition</a>	The aggregations that are defined on this field	
order		Integer		int32
width		Integer		int32
links		List of <a href="#">Link</a>		

## 13.72. *FieldUniquenessRepresentationModel*

Field Name	Required	Type	Description	Format
unique		Boolean		

## 13.73. *Filter*

Represents the base Filter



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,



Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32

## 13.74. GeneralColumn

Field Name	Required	Type	Description	Format
name		String		
order		Integer		int32
width		Integer		int32
aggregations		List of <a href="#">AggregationDefinition</a>	The aggregations that are defined on this column	

## 13.75. I18nMessage

Field Name	Required	Type	Description	Format
code		ErrorDetail		
message		String		
parameters		Map of <a href="#">[object]</a>		

## 13.76. IdListModel

Model for a List containing UUID

Field Name	Required	Type	Description	Format
uuidList		List of <a href="#">[UUID]</a>		uuid

## 13.77. IntervalFilter

Represents a Interval filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
startDate	X	Date	The start of the filter segment	date-time
endDate	X	Date	The end of the filter segment	date-time
intervalType	X	String	The intervalType: EVENT(default) or CREATED_DATE	

### 13.78. *IntervalFilterAllOf*

Field Name	Required	Type	Description	Format
startDate		Date	The start of the filter segment	date-time
endDate		Date	The end of the filter segment	date-time
intervalType		String	The intervalType: EVENT(default) or CREATED_DATE	

### 13.79. *Keyword*

Represents a keyword definition

Field Name	Required	Type	Description	Format
keyword		String	The keyword	
links		List of <a href="#">Link</a>		

### 13.80. *KeywordFilter*

Represents a Keyword filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
keywords		List of <a href="#">[string]</a>		
mode		String		

### 13.81. *KeywordFilterAllOf*

Field Name	Required	Type	Description	Format
keywords		List of <a href="#">[string]</a>		
mode		String		

### 13.82. *KeywordModel*

Field Name	Required	Type	Description	Format
keyword	X	String	The string value representing a keyword	

### 13.83. *Link*

Field Name	Required	Type	Description	Format
rel		String		
href		String		
hreflang		String		
media		String		
title		String		
type		String		
deprecation		String		
profile		String		
name		String		

### 13.84. *LinkModel*

Field Name	Required	Type	Description	Format
link	X	String	The requested link.	
validity		Long	Validity of the requested token, in seconds.	int64

### 13.85. *LinkedField*

Represents a custom field definition with an indication of usage over provided types

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the field	
propertyKey		String	The technical identifier of the field	
name	X	String	The name of the field	
type	X	String	The field type, must be one of these values: STRING, NUMERIC, ENUMERATION	
placeholder		String	The placeholder of the field, an example value for what this field's value might look like	
options		List of <a href="#">[string]</a>	The possible options of the field, only applicable when type Enumeration is chosen	
aggregations		List of <a href="#">AggregationDefinition</a>	The aggregations that are defined on this field	
all		Boolean	Indicates whether the field is present on all or just some provided types.	
links		List of <a href="#">Link</a>		

### 13.86. *LiveIntervalFilter*





Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32

### 13.87. *MinimalContextItem*

Field Name	Required	Type	Description	Format
permissions		Set of <a href="#">[string]</a>	Permissions on the context data	
identifier		String	The identifier of the context item	
components		List of <a href="#">ContextItemComponent</a>	The component for which the context item is registered	
startEventDate		Date	The date of the start event of the context item	date-time
endEventDate		Date	The date of the end event	date-time
links		List of <a href="#">Link</a>		

### 13.88. *NumericCondition*

Represents a numeric condition definition

Field Name	Required	Type	Description	Format
value	X	Double		double
operator	X	String		<i>Enum:</i> EQUAL, NOT_EQUAL, LESS_THAN, LESS_THAN_OR_EQUAL, GREATER_THAN, GREATER_THAN_OR_EQUAL,

## 13.89. *NumericFieldFilter*

Represents a Numeric Field filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
fieldIdentifier		UUID		uuid
field		String		
conditions		List of <a href="#">NumericCondition</a>		
customField		Field		
mode		String		

### 13.90. *NumericFieldFilterAllOf*

Field Name	Required	Type	Description	Format
fieldIdentifier		UUID		uuid
field		String		
conditions		List of <a href="#">NumericCondition</a>		
customField		Field		
mode		String		

### 13.91. *PageDataImport*

Field Name	Required	Type	Description	Format
totalPages		Integer		int32
totalElements		Long		int64
first		Boolean		
last		Boolean		
numberOfElements		Integer		int32
pageable		Pageable		
sort		Sort		
size		Integer		int32
content		List of <a href="#">DataImport</a>		
number		Integer		int32

Field Name	Required	Type	Description	Format
empty		Boolean		

### 13.92. PageMetadata

Field Name	Required	Type	Description	Format
size		Long		int64
totalElements		Long		int64
totalPages		Long		int64
number		Long		int64

### 13.93. Pageable

Field Name	Required	Type	Description	Format
pageNumber		Integer		int32
unpaged		Boolean		
pageSize		Integer		int32
paged		Boolean		
sort		Sort		
offset		Long		int64

### 13.94. PagedModelAttachment

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">Attachment</a>		
page		PageMetadata		

### 13.95. PagedModelAuditTrail

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">AuditTrail</a>		
page		PageMetadata		



## 13.96.

### *PagedModelBackgroundSyncJobRepresentationModel*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">BackgroundSyncJobRepresentationModel</a>		
page		PageMetadata		

## 13.97. *PagedModelComment*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">Comment</a>		
page		PageMetadata		

## 13.98. *PagedModelContextData*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">ContextData</a>		
page		PageMetadata		

## 13.99. *PagedModelContextDataType*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">ContextDataType</a>		
page		PageMetadata		

## 13.100. *PagedModelContextEvent*

Field Name	Required	Type	Description	Format
links		List of <i>Link</i>		
content		List of <i>ContextEvent</i>		
page		PageMetadata		

### 13.101. *PagedModelContextItem*

Field Name	Required	Type	Description	Format
links		List of <i>Link</i>		
content		List of <i>ContextItem</i>		
page		PageMetadata		

### 13.102. *PagedModelDataReference*

Field Name	Required	Type	Description	Format
links		List of <i>Link</i>		
content		List of <i>DataReference</i>		
page		PageMetadata		

### 13.103. *PagedModelExport*

Field Name	Required	Type	Description	Format
links		List of <i>Link</i>		
content		List of <i>Export</i>		
page		PageMetadata		

### 13.104. *PagedModelFailedEventFrameSynchronization*

Field Name	Required	Type	Description	Format
links		List of <i>Link</i>		
content		List of <i>FailedEventFrameSynchronization</i>		
page		PageMetadata		

### 13.105. *PagedModelField*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">Field</a>		
page		PageMetadata		

### 13.106. *PagedModelKeyword*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">Keyword</a>		
page		PageMetadata		

### 13.107. *PagedModelPropertyKeyMigrationFieldData*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">PropertyKeyMigrationFieldData</a>		
page		PageMetadata		

### 13.108. *PagedModelPropertyKeyMigrationMonitorData*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">PropertyKeyMigrationMonitorData</a>		
page		PageMetadata		

### 13.109. *PagedModelPropertyKeyMigrationViewData*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		

Field Name	Required	Type	Description	Format
content		List of <a href="#">PropertyKeyMigrationViewData</a>		
page		PageMetadata		

### 13.110. *PagedModelSource*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">Source</a>		
page		PageMetadata		

### 13.111. *PagedModelString*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">[string]</a>		
page		PageMetadata		

### 13.112. *PagedModelTriggerDefinition*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">TriggerDefinition</a>		
page		PageMetadata		

### 13.113. *PagedModelWorkflow*

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		
content		List of <a href="#">Workflow</a>		
page		PageMetadata		

## 13.114. *Parameter*

Field Name	Required	Type	Description	Format
name		String		
value		String		

## 13.115. *PeriodFilter*

Represents a period filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
periodType		String	The period type	Enum: ONE_HOUR, LAST_HOUR, TWO_HOURS, LAST_TWO_HOURS, EIGHT_HOURS, LAST_EIGHT_HOURS, ONE_DAY, LAST_DAY, SEVEN_DAYS, LAST_WEEK, LAST_FOUR_WEEKS, THIRTY_DAYS, INVALID ,
period		String	The period in String format	ISO8601
live		Boolean	Indicates to start a live search	

### 13.116. *PeriodFilterAllOf*



Field Name	Required	Type	Description	Format
periodType		String	The period type	Enum: ONE_HOUR, LAST_HOUR, TWO_HOURS, LAST_TWO_HOURS, EIGHT_HOURS, LAST_EIGHT_HOURS, ONE_DAY, LAST_DAY, SEVEN_DAYS, LAST_WEEK, LAST_FOUR_WEEKS, THIRTY_DAYS, INVALID ,
period		String	The period in String format	ISO8601
live		Boolean	Indicates to start a live search	

### 13.117. *PermissionUserDetails*

Represents a subset of user details

Field Name	Required	Type	Description	Format
userId		String	User's id.	
userName		String	User's username.	
firstName		String	User's first name.	
lastName		String	User's last name.	
admin		Boolean	Indication whether user is an admin or not.	

## 13.118. *PropertyFieldFilter*

Represents a String Field filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
fieldIdentifier		UUID		uuid
field		String		
key	X	String		
values	X	List of <a href="#">[string]</a>		

### 13.119. *PropertyFieldFilterAllOf*

Field Name	Required	Type	Description	Format
fieldIdentifier		UUID		uuid
field		String		
key		String		
values		List of <a href="#">[string]</a>		

### 13.120. *PropertyKeyMigrationField*

Field Name	Required	Type	Description	Format
original		String		
migrated		String		

### 13.121. *PropertyKeyMigrationFieldData*

Represents the changes to be performed on a field

Field Name	Required	Type	Description	Format
identifier		UUID		uuid
name		String		
type		String		
propertyKey		String		
migratedKey		String		
ambiguous		Boolean		
createdByDetails		UserDetails		

Field Name	Required	Type	Description	Format
links		List of <a href="#">Link</a>		

### 13.122. *PropertyKeyMigrationFieldFilter*

Field Name	Required	Type	Description	Format
original		FieldFilter		
migrated		FieldFilter		

### 13.123. *PropertyKeyMigrationMonitorData*

Represents the changes to be performed on a monitor

Field Name	Required	Type	Description	Format
identifier		UUID		uuid
monitorIdentifier		String		
monitorOwner		String		
monitorOwnerId entifier		String		
monitorName		String		
monitorOwnerDet ails		UserDetails		
fields		List of <a href="#">PropertyKeyMigrationField</a>		
links		List of <a href="#">Link</a>		

### 13.124. *PropertyKeyMigrationPhaseRepresentationModel*

Field Name	Required	Type	Description	Format
startDate		Date		date-time
endDate		Date		date-time

Field Name	Required	Type	Description	Format
status		String		Enum: WAITING, RUNNING, FINISHED, FAILED, SKIPPED, '

### 13.125. *PropertyKeyMigrationPhaseStatusModel*

Field Name	Required	Type	Description	Format
waitingForApproval		String		

### 13.126. *PropertyKeyMigrationPropertyFieldFilter*

Field Name	Required	Type	Description	Format
original		PropertyFieldFilter		
migrated		PropertyFieldFilter		

### 13.127. *PropertyKeyMigrationStatusRepresentationModel*

Field Name	Required	Type	Description	Format
dryRun		PropertyKeyMigrationPhaseRepresentationModel		
waitingForApproval		PropertyKeyMigrationPhaseRepresentationModel		

### 13.128. *PropertyKeyMigrationViewData*

Represents the changes to be performed on a view

Field Name	Required	Type	Description	Format
identifier		UUID		uuid
viewIdentifier		String		
viewOwner		String		
viewOwnerIdentifier		String		
viewOwnerDetails		UserDetails		
viewName		String		
fieldFilters		List of <a href="#">PropertyKeyMigrationFieldFilter</a>		
propertyFilters		List of <a href="#">PropertyKeyMigrationPropertyFieldFilter</a>		
ganttGroupBy		PropertyKeyMigrationField		
ambiguous		Boolean		
links		List of <a href="#">Link</a>		

### 13.129. *ReactiveSearchRequestModel*

Represents a reactive search request definition

Field Name	Required	Type	Description	Format
filters		List of <a href="#">ComponentFilter</a> , <a href="#">CreatedByFilter</a> , <a href="#">CreatedDateFilter</a> , <a href="#">CurrentStateFilter</a> , <a href="#">DescriptionFilter</a> , <a href="#">DurationFilter</a> , <a href="#">EnumerationFieldFilter</a> , <a href="#">IntervalFilter</a> , <a href="#">KeywordFilter</a> , <a href="#">LiveIntervalFilter</a> , <a href="#">NumericFieldFilter</a> , <a href="#">PeriodFilter</a> , <a href="#">PropertyFieldFilter</a> , <a href="#">StringFieldFilter</a> , <a href="#">TypeFilter</a> >		
sort		Sort		
pageSize		Integer		int32
delay		Integer		int32



Field Name	Required	Type	Description	Format
limit		Integer		int32
bufferSize		Integer		int32
useTimeSeriesIdentifier		Boolean	Boolean to indicate if Tag need to be resolved to UUIDs or not, default false	
dataSort		Sort		

### 13.130. *SearchParametersModel*

Represents search parameters

Field Name	Required	Type	Description	Format
query	X	String	Search query	
hasPermission		String	Type permissions the user needs to have. Possible values: CONTEXT_TYPE_WRITE or CONTEXT_TYPE_DELETE. When CONTEXT_TYPE_DELETE is provided, CONTEXT_TYPE_WRITE is automatically assumed.	
includeDeleted		Boolean	Whether to include deleted types or not.	
page		Integer		int32
size		Integer		int32
sort		List of <a href="#">[string]</a>		

### 13.131. *SearchRequestModel*

Represents a search request definition

Field Name	Required	Type	Description	Format
filters		List of <a href="#">ComponentFilter</a> , <a href="#">CreatedByFilter</a> , <a href="#">CreatedDateFilter</a> , <a href="#">CurrentStateFilter</a> , <a href="#">DescriptionFilter</a> , <a href="#">DurationFilter</a> , <a href="#">EnumerationFieldFilter</a> , <a href="#">IntervalFilter</a> , <a href="#">KeywordFilter</a> , <a href="#">LiveIntervalFilter</a> , <a href="#">NumericFieldFilter</a> , <a href="#">PeriodFilter</a> , <a href="#">PropertyFieldFilter</a> , <a href="#">StringFieldFilter</a> , <a href="#">TypeFilter</a> >		

### 13.132. Sort

Field Name	Required	Type	Description	Format
unsorted		Boolean		
sorted		Boolean		
empty		Boolean		

### 13.133. Source

Represents a source and its synchronization status

Field Name	Required	Type	Description	Format
identifier		UUID	The identifier of the source	uuid
name		String	The name of the source	
externalId		String	The identifier of the source in tm-datasource	
enabled		Boolean	Indication whether the LIVE synchronization for this source is enabled or not	
currentIntervalStart		Date	Lower end of the current interval being synced	date-time
currentIntervalEnd		Date	Upper end of the current interval being synced	date-time
currentIntervalCount		Integer	Number of items in the current interval being synced	int32

Field Name	Required	Type	Description	Format
currentIntervalPage		Integer	Current page of the current interval being synced	int32
lastSyncedItemDate		Date	Last modified date of the last synced item	date-time
exceptionMessage		String	Exception message, should one have occurred in the current interval	
syncPageSize		Integer	The sync page size	int32
links		List of <a href="#">Link</a>		

### 13.134. *SourceModel*

The source to which this workflow belongs

Field Name	Required	Type	Description	Format
identifier		UUID	The source's identifier.	uuid
enabled		Boolean	Indication whether live sync should be enabled or not.	

### 13.135. *SourceSynchronizationIntervalModel*

Field Name	Required	Type	Description	Format
startDate	X	Date	The start date of the interval to sync.	date-time
endDate		Date	The end date of the interval to sync. Current date will be taken if left empty.	date-time
cancelable		Boolean		

### 13.136. *StringFieldFilter*

Represents a String Field filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
fieldIdentifier		UUID		uuid
field		String		
values		List of <a href="#">[string]</a>		
customField		Field		
mode		String		

### 13.137. *StringFieldFilterAllOf*

Field Name	Required	Type	Description	Format
fieldIdentifier		UUID		uuid
field		String		
values		List of <a href="#">[string]</a>		
customField		Field		
mode		String		

### 13.138. *TaskSubmissionResult*

Field Name	Required	Type	Description	Format
running		Integer	Number of tasks currently being executed.	int32
queued		Integer	Number of tasks currently queued.	int32

### 13.139. *TimeSeriesDefinitionRepresentation*

TimeSeries definition to which the data points. Can be empty, even if data is present.

Field Name	Required	Type	Description	Format
datasourceId		UUID		uuid
ownerId		String		
externalId		String		
name		String		

Field Name	Required	Type	Description	Format
description		String		
type		String		
units		String		
interpolationType		String		
deleted		Boolean		
plotDataUrl		String		
indexDataUrl		String		
links		List of <i>Link</i>		
id		UUID		uuid

### 13.140. *TokenMetaData*

Field Name	Required	Type	Description	Format
continuationToken		String		

### 13.141. *TokenizedPageModelContextItem*

Field Name	Required	Type	Description	Format
content		List of <i>ContextItem</i>		
page		TokenMetaData		

### 13.142. *TriggerDefinition*

Represents a trigger definition

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the trigger	
state		String	The triggering state, an empty state means the trigger should fire on all possible states.	
type		String	The trigger type	<i>Enum:</i> EXPORT _DASHB OARD,
configuration		Map of <i>[string]</i>	The configuration for the trigger. Depends on the selected type	
links		List of <i>Link</i>		

## 13.143. *TriggerDefinitionModel*

Represents a trigger definition

Field Name	Required	Type	Description	Format
state		String	The workflow state causing this trigger to fire. An empty state means the trigger should fire on all possible states.	
type	X	String	The trigger type.	<i>Enum:</i> EXPORT_DASHBOARD,
configuration	X	Map of <a href="#">[string]</a>	The configuration for the trigger. Depends on the selected type.	

## 13.144. *TypeAccessRuleModel*

Object used for creating and updating an access rule

Field Name	Required	Type	Description	Format
identifier		String	The identifier of the access rule.	
subjectId	X	String	The identifier of the subject.	
subjectType	X	String	The type of the subject.	<i>Enum:</i> GUEST, EVERYONE, GROUP, USER, USER, GROUP, EVERYONE, GUEST,
permissions	X	Set of <a href="#">[string]</a>	A set of permissions the subject has on the specified object.	<i>Enum:</i> CONTEXT_TYPE_READ, CONTEXT_TYPE_WRITE, CONTEXT_TYPE_DELETE,



## 13.145. *TypeFilter*

Represents a type filter definition



Field Name	Required	Type	Description	Format
type		String		<i>Enum:</i> INTERVAL_FILTER, LIVE_INTERVAL_FILTER, PERIOD_FILTER, COMPONENT_FILTER, TYPE_FILTER, KEYWORD_FILTER, STRING_FIELD_FILTER, ENUMERATION_FIELD_FILTER, NUMERIC_FIELD_FILTER, PROPERTY_FIELD_FILTER, CREATED_BY_FILTER, DESCRIPTION_FILTER, DURATION_FILTER, APPROVAL_FILTER, CURRENT_STATE_FILTER, CREATED_DATE_FILTER, EXTERNAL_FILTER,

Field Name	Required	Type	Description	Format
valid		Boolean		
validationResultMessages		List of <a href="#">ValidationResultMessage</a>		
order		Integer		int32
types	X	List of [string]		
typeResources		List of <a href="#">ContextDataType</a>		

### 13.146. *TypeFilterAllOf*

Field Name	Required	Type	Description	Format
types		List of [string]		
typeResources		List of <a href="#">ContextDataType</a>		

### 13.147. *UserDetails*

The details of the user who created the context item

Field Name	Required	Type	Description	Format
identifier		String	The uuid of the user who created the context item	
username		String	The name of the user who created the context item	
firstName		String	The first name of the user who created the context item	
lastName		String	The last name of the user who created the context item	
links		List of <a href="#">Link</a>		

### 13.148. *ValidationResultMessage*

Field Name	Required	Type	Description	Format
key		String		
message		I18nMessage		

## 13.149. *Workflow*

Represents a work flow definition

Field Name	Required	Type	Description	Format
identifier		UUID	The identifier of the workflow	uuid
name	X	String	The name of the workflow	
startState	X	String	The start state of the workflow	
endState	X	String	The end state of the workflow	
states	X	Set of <a href="#">[string]</a>	The possible states of the workflow	
deletable		Boolean	Boolean that indicates if the workflow is deletable	
source		Source		
links		List of <a href="#">Link</a>		

## 13.150. *WorkflowModel*

Represents a work flow definition

Field Name	Required	Type	Description	Format
identifier		UUID	The identifier of the workflow	uuid
source		SourceModel		
externalId	X	String	The identifier of the workflow in the external source system	
name	X	String	The name of the workflow	
startState	X	String	The start state of the workflow	
endState	X	String	The end state of the workflow	
states	X	Set of <a href="#">[string]</a>	The possible states of the workflow	
deletable		Boolean	If the workflow is deletable or not	

## 13.151. *WorkflowStates*

Represents a work flow states definition

Field Name	Required	Type	Description	Format
states	X	Set of <a href="#">[string]</a>	The possible states of the work flow	
links		List of <a href="#">Link</a>		

## 13.152. *WorkflowStatesModel*

Represents the possible states for a work flow definition

Field Name	Required	Type	Description	Format
states	X	Set of <a href="#">string</a>	The possible states of the work flow	

[Download PDF version](#) | [Download OpenAPI specifications](#)