



**Sympa HR
Integration API**

Sympa

Sympa Oy

Technopolis Helsinki-Vantaa · Teknobulevardi 3-5 · FI-01530 Vantaa, Finland
+358 290 001 200 · www.sympa.fi · Domicile: Lahti, Finland · Business ID: FI19385975

Table of Contents

1	This Document.....	3
2	Overview	3
3	Using the Api Interfaces	3
3.1	Authentication	3
3.1.1	API Key	3
3.2	Constructing your queries with OData	4
3.2.1	Metadata	4
3.2.2	Queries	4
3.3	Approvals	6
3.4	API Activity Log	7
3.5	Throttling.....	7

1 This Document

This document provides an overview of the Sympa HR Integration API functionality.

2 Overview

Sympa HR Integration API (from here on “API” in this document) provides a method for exposing data from the Sympa HR solution for integration purposes. An administrator can dynamically create interfaces that provide access to basically any data stored in the HR system. Interfaces can, at this point, be configured to return (read-only) data from a specific card base (e.g. employees).

Configuring the API for your organization’s needs can be done by any user with administrator privileges. There are no preset/fixed field lists for the interfaces, which means that basically each interface is unique. This gives admin users the possibility to restrict the transferrable fields to only ones required by the calling system, and no unnecessary data is exposed.

Setting up and configuring an API interface is done at the API panel, sympahr.net/admin/apipanel.

3 Using the API Interfaces

Calls to the API interfaces that have been configured are done by issuing HTTP GET requests to the URLs referring to each interface with the corresponding interface type. The URLs can be found in the list of configured interfaces on the API panel (admin settings).

Example URL: <https://api.sympahr.net/api/testinterface>

3.1 Authentication

Each request to any API interface has to authenticate, using the method specified for the API key being used.

The following examples are based on the following key/secret combinations (note: the test interface and credentials do not point to any live API, they are only used as examples for documentation purposes):

Key: a447eb14e84e4ecf8eae52cfb932a3b3

Secret: 26c1967a07a04c43b8abada48d1379c2

3.1.1 API Key

Authentication with the API interface is done using a unique API key, which enables interaction with one or more API interfaces. The Sympa Integration API supports two types of authentication;

- A plain API key
- Basic Authentication using API key and Secret

The authentication method is chosen for each API key.

ApiKey

If ApiKey authentication is being used, the API key is added as a request header named “X-ApiKey”.

Example header values:

X-ApiKey: a447eb14e84e4ecf8eae52cfb932a3b3

Basic

Using Basic authentication, the API key function as the username, and the Secret as the password.

Example header values:

Authorization: Basic

YtQ0N2ViMTRiODRlNGVjZjhlYWU1MmNmYjZjMmEzYjM6MjZjMTk2N2EwN2EwNGM0M2I4YWJhZGE0OGQxMzc5YzI=

Requests where the authentication fails will get an Http Unauthorized (401) response code. Since unauthenticated requests are not necessarily associated with any specific ApiKey or interface, these failed requests will not be visible in the Api Activity Log. Http Unauthorized (401) will also be returned if the Http verb (GET, PATCH or POST) does not match the configured interface type.

3.2 Constructing your queries with OData

The Sympa Customer API implements part of the [Open Data Protocol, Version 4](#). The supported features are described below.

3.2.1 Metadata

Metadata about the interfaces associated to a particular API key can be accessed by appending `/ $metadata` to the API service root URL (`api.sympahr.net/api`). The metadata is returned as XML, and it lists the available interfaces as well as their field properties.

3.2.2 Queries

The Sympa Integration API implements part of the Open Data Protocol, Version 4. The supported features are described below. The results from all data queries are returned in JavaScript Object Notation (JSON) format. Only the subset of OData query parameters defined below are supported.

- `$top`
- `$skip`
- `$orderby`
- `$filter`
- `$count`
- `errordata` (non-standard, specific for Sympa HR Integration API)
- `since & until` (non-standard, specific for Sympa HR Integration API)

By default, all matches to any query are returned, which can lead to slow performance on large data sets. Therefore, it is recommended that the client request a certain amount of matches to be returned per query.

In this document, we use shortened urls, e.g. url `"api/testinterface?$filter=..."` is just a shortening for `https://api.sympahr.net/api/testinterface?$filter=...`

Top

The query `$top=XX` will return the first XX matches to the query.

Example: `api/testinterface?$top=200`

Skip

The query `skip=XX` will leave out the first XX matches to the query.

Example: `api/testinterface?skip=10`

Example: `api/testinterface?skip=10&top=10`

Order by

The query `orderby=somealias, someotheralias desc` will sort the result set, primarily by the alias "somealias", ascending, and secondarily by the alias "someotheralias", descending.

Example: `api/testinterface?orderby=id`

Example: `api/testinterface?orderby=id desc`

Example: `api/testinterface?orderby=surname,age desc`

Filter

The following subset of OData filters are supported by the API.

And	Gt	Contains
Or	Lt	StartsWith
Not	Ge	EndsWith
Eq	Le	

Filter examples by field type:

- All the field types can be filtered by null or by text
`api/testinterface?filter=not (email eq null)`
`api/testinterface?filter=name eq 'no values'`
- Numeric field
`eq`=equal, `gt`=greater than, `lt` = less than, `ge`=greater than or equal to, `le`=less than or equal to
`api/testinterface?filter=not (number eq null)`
`api/testinterface?filter=number ge 0`
 decimal number as filter
`api/testinterface?filter=number ge 1.14d`
- Dates
`eq`=equal, `gt`=greater than, `lt` = less than, `ge`=greater than or equal to, `le`=less than or equal to
`api/testinterface?filter=not (date eq null)`
`api/testinterface?filter=birthday gt 1980-12-17`
 between `api/testinterface?filter=(date gt 2020-02-29) and (date lt 2020-03-02)`
- Timestamps
 Datacard's created and modified timestamps can also be used as filter.
`api/testinterface?filter=(modified gt 2015-12-31T23:59:59Z)`
 Special case: If you don't specify the time, the timestamp is interpreted as T00:00:00, e.g:
`api/testinterface?filter=(created le 2014-01-01)`
 comparison is done with timestamp (2014-01-01T00:00:00)
 A good idea would be to always specify the time with the as a DateTime attribute.

Note: Datacard's empty created date is shown "0001-01-01T00:00:00" but empty modified date is shown as "null"

- Identification field
Identification field can be used in filter as the "raw" numeric value, **not the formatted value**
e.g. `?$filter=sympaID eq '123'`
Using values that are not numbers (e.g. `sympaID eq 'a123'`) will give an error ("message": "Error with SQM query: MissingFilterField")

Count

The query `$count=true` will add the total count of all matches, after filtering but before skip and top, as a return value named "odata.count".

Example: `api/testinterface?$count=true`

The `@odata.count` value always contains all the results matched by the filter, i.e. top and skip do not affect the count (`api/testinterface?$skip=2&$top=1&$count=true` count value is same as `api/testinterface?$count=true`)

Error data

The query `errordata=true` will add a summary of all errors encountered in the data while querying, as a return value named "odata.errors". Note: Do not use `$`, because this a customized parameter for Sympa HR, not ODATA standard.

Since & until

To show the table rows which validity time overlaps with the given time frame (note: requires effectivity settings in Sympa HR for the tables in question). Use `since`, `until`, `at`. Note: Do not use `$`, because this a customized parameter for Sympa HR, not ODATA standard.

Examples:

`api/testinterface?since=2011-12-29&until=2012-02-21`

`api/testinterface?since=2011-12-29`

`api/testinterface?until=2012-02-21`

Expand

`$expand` query parameter allows filtering rows inside a table. Example:

`api/testinterface?$expand=tablename($filter=modified ge 2011-01-01)`

Note: the `$expand` parameter filters only within the table in question, meaning the result set contains all the datacards (eg. employees) it would contain otherwise, only the table rows are filtered.

You can combine `$expand` with other filters + attributes:

`api/testinterface?$filter=unit eq 'HR'&$expand=tablename($filter=Approval eq 5)`

In this example the datacards (eg. employees) are filtered by unit, and the table rows by the row approval status.

3.3 Approvals

Table row approval information is returned as an integer value in the JSON message:

Undef = 0

WaitingSave = 1

WaitingRequest = 2

WaitingApproval = 3

Declined = 4

Approved = 5

3.4 API Activity Log

The ten most recent requests to your organization's API interfaces, along with their statuses, metadata and any error messages, will be shown in a table on the bottom of the API panel.

3.5 Throttling

The default throttling level is up to 100 requests in 1h time per interface. If a more allowing level is required, please contact your Sympa HR contact person.