Integrations

Custom Methods
31.5.2023

Content

1 General information of Sympa custom integrations

Sympa can offer to export/expose the agreed data with the following (alternate) integration options: (see the following chapters for more detailed descriptions of each method):

- configurable CSV file-based output to SFTP,
- Changes XML file-based output to SFTP or
- configurable passive Sympa REST API (.json) endpoint(s).

The customer or a third-party vendor (e.g. integrated system, middleware solution, or customer's IT service provider) must fetch the data and execute the transfer to the third-party system. The customer is responsible for acquiring all necessary licenses, permits, and rights to use third-party vendors' systems in connection with Sympa integrations. Furthermore, the customer is responsible for costs related to the use of agreed third-party vendors' systems and integrations.

All integration implementations require participation from the customer and from the third-party vendor. The customer is responsible for establishing efficient cooperation with the third-party vendors to enable the functioning of the elected integrations as agreed.

Sympa's default work estimates and prices for the integration projects include:

- Specification: online workshop (2-3h) and further clarifications
- Documentation on Sympa's document templates
- Implementation: Sympa REST API and possible file transfer configuration
- Technical testing support (no input of test data)
- Delivery to production use
- Two-week hypercare period after the go-live of production use

⦾ Sympa

2 Sympa Integration API

Sympa passive REST API provides methods for exposing data from the Sympa solution, and for importing data to Sympa for integration purposes. GET Interfaces are configured to return (read-only) data from a specific card base (e.g., employees). Webhooks are currently not supported, and it is not possible to call for changes since last call. If only changed/new data is required, the party calling is responsible for conducting a data comparison. POST & PATCH interfaces can be used to modify data in a specific card base (e.g., employees).

There are no preset/fixed field lists for the interfaces, which means that basically each interface is unique. Configurable interfaces enable to restrict the transferrable fields to only ones required by the calling system, and no unnecessary data is exposed.
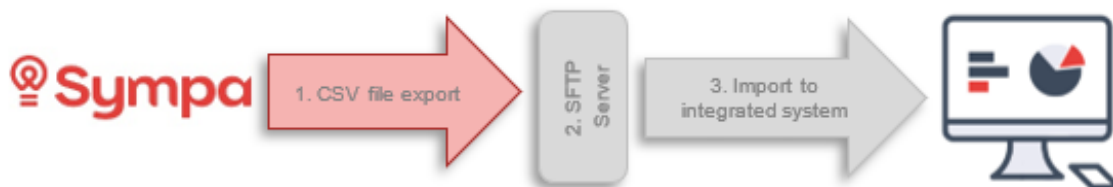
The Sympa Integration API implements part of the Open Data Protocol, Version 4.0/4.01. The results from data queries are returned in JavaScript Object Notation (JSON) format.

Example JSON:

```json
{
    "@odata.context": "http://api.sympahr.net/api/$metadata#DemoExample",
    "value": [
        {
            "Employee_number": "139",
            "Last_name": "Jørgensen",
            "First_names": "Katrine N.",
            "Company_currently_valid": "Bela Denmark A/S",
            "Country_currently_valid": "Denmark",
            "Email_address": "katrine.n.jorgensen@smail.com",
            "Street_address": "Holmevej 81",
            "Postal_code": "1420",
            "City": "København K",
            "First_day_of_employment_currently_valid": "2017-04-03",
            "Last_day_of_employment_currently_valid": null,
            "Salary": [
                {
                    "Salary_category": "Monthly salary",
                    "Currency": "DKK",
                    "Amount": 35500,
                    "Start_date": "2017-04-13",
                    "End_date": null,
                    "Approval": 5
                }
            ]
        }
    ]
}
```

3   Generic CSV

Generic CSV is an integration that can transform data from Sympa's API into CSV format and send the CSV file to an SFTP server for the receiving system. SFTP server can use username + password or key authentication (RSA) login. Generic CSV can be configured to include any supported fields within Sympa, keeping in mind the limitations of CSV as a format (e.g., might require several CSV files).

Sympa's CSV integrations run once per day.

Transfer File Formats and Settings

Separator	• Semicolon (;) - default • Comma (,) • Vertical Bar (|) • TAB (\t) • Tilde (~)
Separator handling	If the data includes the separator (;) or a line break, those are removed by the integration
Wrap data values in quotes	Yes / No
Changes comparison for form data (Transfers only one row for employee per run)	• No comparison (full file / overwrite) • Send whole row if something changed (append) • Send only changed values and ID fields, others with skip characters (*) (append) • Send only changed values and ID fields, with others empty/blank (append)
Changes comparison for table data (Data picked directly from Sympa's table)	• If a new row / change is found, the whole row is sent • No comparison (full file / overwrite)
Over Writing	• Data is written (appended) in the end of the file. The size of the file increases unless the file is deleted or renamed after it has been read by the integrated system. • Old file is overwritten by a new one. • New file with a timestamp is created
Column Headers	• No header line. (Columns are in pre-defined order) • The first line is reserved as the header line (Columns in alphabetical order)
File split (based on values in first column)	Yes / No
Encoding	ISO-8859-1 (ANSI) / UTF-8 (Without BOM)

Date Formatting	For example: d.M.yyyy/dd.MM.yyyy/ddMMyyyy/yyyyMMdd/yyyy-MM-dd… Any date format that follows java SimpleDateFormat can be chosen.
Decimal Separator	, (comma) / .(Dot)
Decimal handling	All numeric values are transferred with two decimals
Timestamp in filename (if used, creates new file with a timestamp)	For example: yyyyMMdd/No
Timestamp in last column (if used, adds a timestamp as last column in each file and row in CSV)	For example: dd.MM.yyyy/yyyy-MM-dd/ yyyyMMdd/No

Transfer files

If in '*Transfer File Formats and Settings*' table it is defined that a date stamp and/or file split per the first column is applied, that timestamp and/or file split value is added to the file name after an underscore _
For example:
SystemNameBasicInfo_yyyyMMdd.csv (=timestamp applied) OR
SystemNameBasicInfo_Company1.csv (=file split applied) OR
SystemNameBasicInfo_Company1_yyyyMMdd.csv (=file split and timestamp applied)

Example CSVS

```
City;Company;Country;Email_address;Employee_number;First_day_of_employment;First_names;Last_day_
København K;Bela Denmark A/S;Denmark;katrine.n.jorgensen@smail.com;139;2017-04-03;Katrine N ;;Jø
```

```
Amount;Company;Currency;Employee_number;End_date;Salary_category;Start_date
35500,00;Bela Denmark A/S;DKK;139;;Monthly salary;2017-04-13
```

4 Changes XML

Changes XML integration offers the possibility to track changes within selected data stored in Sympa. Changes XML integration does not have a fixed field list but can be configured to include any supported fields necessary for the customer and the receiving system. The data structure of the XML will always follow the Sympa environment structure in question. This structure is fixed and receiving party will have to modify it if needed. See example below.
Sympa's XML integrations run once per day. The file is transferred to an SFTP (username + password login) server. It is also possible to notify which fields have been changed in the XML file with a "changed" attribute.

ChangesXML settings

File handling	New files are created with a timestamp (yyyyMMdd_hhmmss) in the beginning of the filename If no data changed since the previous transfer, there will be an empty XML file.
Decimal Separator	.
Date Formatting	yyyy-MM-dd
Encoding	UTF8
Remove CDATA tags	Yes/No
Include changed attribute	Yes/No
Changes comparison	• No comparison, send all data/ • Send whole data (e.g. of an employee) only if something changed

Example xml

```xml
<?xml version='1.0' encoding='UTF-8'?>
<sympa>
    <changes>
        <ID changed="false">139</ID>
        <Last_name changed="false">Jørgensen</Last_name>
        <First_names changed="false">Katrine N </First_names>
        <Company_currently_valid changed="false">Bela Denmark A/S</Company_currently_valid>
        <Country_currently_valid changed="false">Denmark</Country_currently_valid>
        <Email_address changed="false">katrine.n.jorgensen@smail.com</Email_address>
        <Street_address changed="true">Holmevej 79</Street_address>
        <Postal_code changed="true">1421</Postal_code>
        <City changed="false">København K</City>
        <First_day_of_employment_currently_valid changed="false">2017-04-03</First_day_of_employment_currently_valid>
        <Last_day_of_employment_currently_valid changed="false"/>
    </changes>
</sympa>
```