Batch Import User Guide

TIBCO EBX Professional Services

Table of Contents

[1. Overview 1](#_Toc2877824)

[2. Design 1](#_Toc2877825)

[3. Implementation 4](#_Toc2877826)

[4. Services 4](#_Toc2877827)

[1. SOAP Request 4](#_Toc2877828)

[2. REST Request 5](#_Toc2877829)

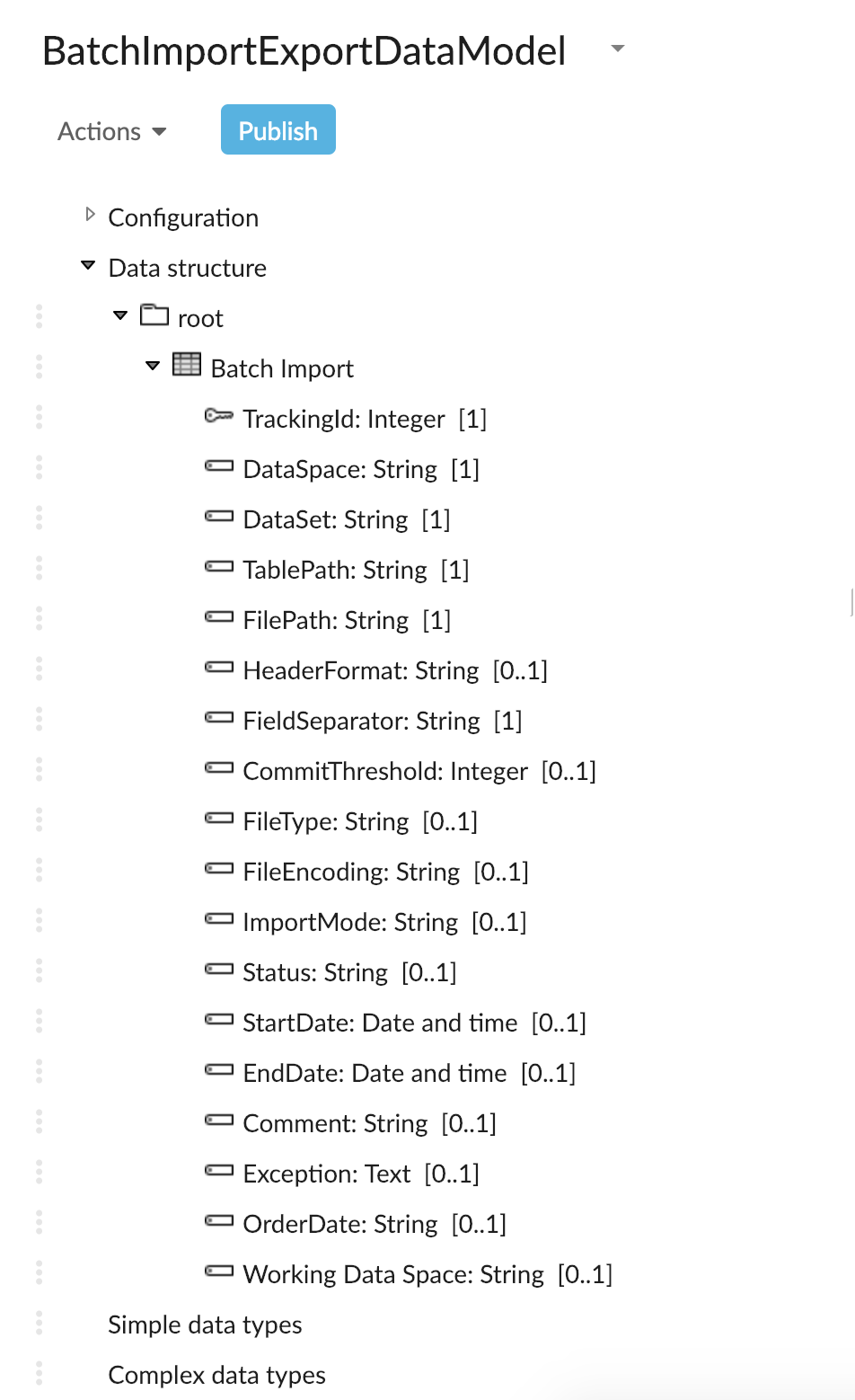
[3. CSV template 6](#_Toc2877830)

## Overview

Batch Import is an integration utility that allows importing data from files (csv and xml) into the EBX tables in batches, without a user having to manually import each file. Typically, these are files generated from external source systems, that need to be loaded into EBX. Each csv or xml file is expected to contain data for 1 table. The structure of the csv and xml files must match the structure of the tables defined in the EBX data model.

## Design

The Batch Import utility has its own data model called the “BatchImportExportDataModel.xsd”, that is available in the PSWorkspace’s company-mdm module. This data model has 1 table defined called “Batch Import”. Here is the structure of the table.



The table below provides the description for each of the fields.

|  |  |  |
| --- | --- | --- |
| **Field name** | **Field description** | **Valid values (if applicable)** |
| TrackingId | Primary Key - Auto generated integer field |  |
| DataSpace | The target dataspace of the table to import the csv/xml file |  |
| DataSet | The target dataset of the table to import the csv/xml file |  |
| TablePath | The path to the table in the dataset/dataspace to import the csv/xml file |  |
| FilePath | The path to the source file (csv/xml) that needs to be imported into the table. Typically an absolute path to the file in the server |  |
| HeaderFormat | Format of the header. This is an enum field that matches the out-of-the-box Import CSV options. The default value is LABEL | LABEL, XPATH, NONE |
| FieldSeparator | The field separator to use to import the file. This is similar to the out-of-the-box Import CSV options.  Typically a comma. You could also use Semicolon, Space, Tab or any other character |  |
| CommitThreshold | This an optional field is used to specify an intermediate commit threshold if needed. Sets the maximum size of the transaction so that intermediate commits occur during the execution of the procedure.  If this is not specified, then the entire file will be imported as a single transaction |  |
| FileType | Specifies the file type (csv or XML). This is an enum field | csv, XML |
| FileEncoding | Specifies the character encoding to use for the exported file (usually UTF-8, which is the default value set in the data model) |  |
| ImportMode | Specifies the Import Mode to be used while importing the file. This is an enum field that matches the out-of-the-box Import CSV options. The default value is O (Insert or Update) | I - Insert mode U - Update mode O - Insert or Update R - Replace |
| The below fields are “Output” fields, populated by the Batch Import process | | |
| Status | Specifies the status of the Import operation. This field is derived in the table trigger | Started, Succeeded, Failed |
| StartDate | Specifies the start date/time of the Import operation. This field is derived in the table trigger |  |
| EndDate | Specifies the end date/time of the Import operation. This field is derived in the table trigger |  |
| Comment | This field is a placeholder to populate any comments observed during the import operation. Could be something that need to be notified to the system making the request to Import file(s) |  |
| Exception | In the case of a failure in the Import, this field captures any exception that occurred during the process. The Status in this case would be set to Failed |  |
| OrderDate | This field is currently not being used |  |
| Working Data Space | The entire Import operation takes place in a child dataspace (created off of the DataSpace mentioned above), which then gets merged to the target dataspace. This field captures the child dataspace identifier. |  |

## Implementation

When a record is created in the Batch Import table, the handleAfterCreate method of the table trigger creates a child dataspace of the DataSpace specified (the target dataspace where the file needs to be imported) and uses the EBX API to import the file into the specified table in the child dataspace. The child dataspace is then merged to the target dataspace. If the import was successful, the Status gets updated to “Succeeded”, else “Failed” with the reason for failure being captured in the Exception field. The start date/time and end date/time gets captured as well, along with the name of the child dataspace.

For any post-import process that needs to be executed, extend the BatchImport trigger and override the executePostImportExtension(AfterCreateOccurrenceContext) method. For example, you may want to launch a workflow at the end of the import process.

## Services

Although the record in the Batch Import table could be created from within the UI, this utility is typically used by the external source systems that do not necessarily have EBX users. They will however have a Service User that has write access to the Batch Import table. These sections provide sample SOAP request and REST request to create records in the Batch Import table.

### SOAP Request

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:ebx-schemas:dataservices\_1.0" xmlns:sec="http://schemas.xmlsoap.org/ws/2002/04/secext" xmlns:urn1="urn:ebx-schemas:deployment\_1.0">

<soapenv:Header>

<sec:Security>

<UsernameToken>

<Username>admin</Username>

<Password>admin</Password>

</UsernameToken>

</sec:Security>

</soapenv:Header>

<soapenv:Body>

<urn:insert\_BatchImport>

<branch>CommonReferenceMasterDataSpace</branch>

<instance>BatchImportExportDataSet</instance>

<data>

<root>

<BatchImport>

<DataSpace>OrganizationMasterDataSpace</DataSpace>

<DataSet>OrganizationDataSet</DataSet>

<TablePath>/root/Organization</TablePath>

<FilePath>/Users/Bala/Downloads/Organization.csv</FilePath>

<HeaderFormat>Label</HeaderFormat>

<FieldSeparator>,</FieldSeparator>

<CommitThreshold>2000</CommitThreshold>

<FileType>csv</FileType>

<FileEncoding>UTF-8</FileEncoding>

<ImportMode>O</ImportMode>

</BatchImport>

</root>

</data>

</urn:insert\_BatchImport>

</soapenv:Body>

</soapenv:Envelope>

### REST Request

<http://localhost:9393/ebx-dataservices/rest/data/v1/BCommonReferenceMasterDataSpace/BatchImportExportDataSet/root/BatchImport?login=admin&password=admin&updateOrInsert=true>

{

"content": {

"DataSpace": {

"content": "OrganizationMasterDataSpace"

},

"DataSet": {

"content": "OrganizationDataSet"

},

"TablePath": {

"content": "/root/Organization"

},

"FilePath": {

"content": "/Users/Bala/Downloads/Organization.csv"

},

"HeaderFormat": {

"content": "XPATH"

},

"FieldSeparator": {

"content": ","

},

"CommitThreshold": {

"content": 2000

},

"FileType": {

"content": "csv"

},

"FileEncoding": {

"content": "UTF-8"

},

"ImportMode": {

"content": "O"

}

}

}

### CSV template

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DataSpace | DataSet | TablePath | FilePath | HeaderFormat | FieldSeparator | CommitThreshold | FileType | FileEncoding | ImportMode |
| OrganizationMasterDataSpace | OrganizationDataSet | /root/Organization | /Users/Bala/Downloads/Organization.csv | Label | , | 2000 | csv | UTF-8 | O |