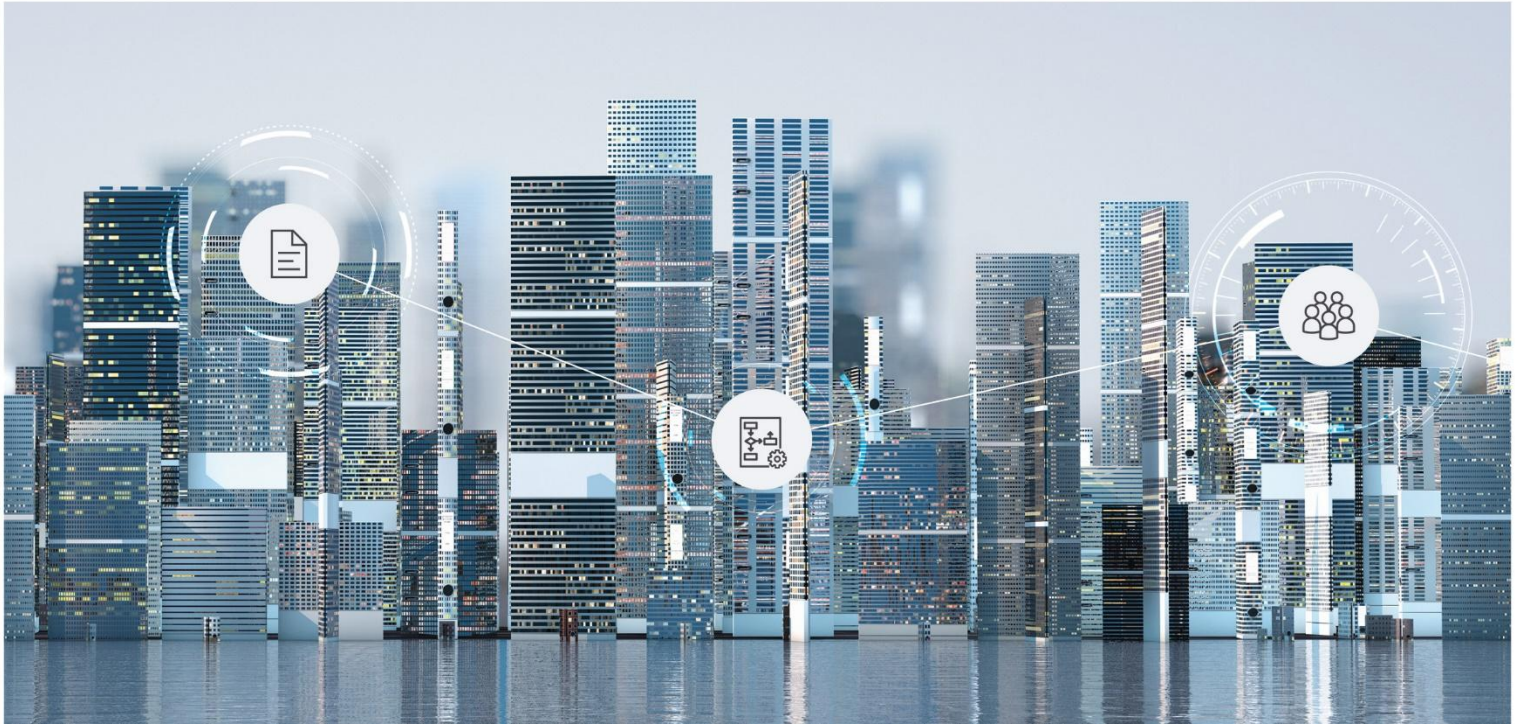
Fabasoft



White Paper

Fabasoft Folio Bulk Jobs

Fabasoft Folio 2025 Update Rollup 1

Copyright © Fabasoft R&D GmbH, Linz, Austria, 2025.

All rights reserved. All hardware and software names used are registered trade names and/or registered trademarks of the respective manufacturers.

No rights to our software or our professional services, or results of our professional services, or other protected rights can be based on the handing over and presentation of these documents.

Contents

1 Introduction 4

2 Software and Hardware Requirements 4

3 General 4

4 Definition of a Bulk Job 4

4.1 Settings Tab 5

4.2 Objects Tab 6

4.3 Log Tab 7

5 Use Case Implementation 7

1 Introduction

This document describes how to update Fabasoft Folio objects using the software component *Bulk Jobs*. This includes the configuration of property modifications, executing actions, defining the affected objects and logging the process.

2 Software and Hardware Requirements

System environment: All information contained in this document implicitly assumes a Microsoft Windows environment.

Supported platforms: For detailed information on supported operating systems and software see the software product information on the Fabasoft distribution media.

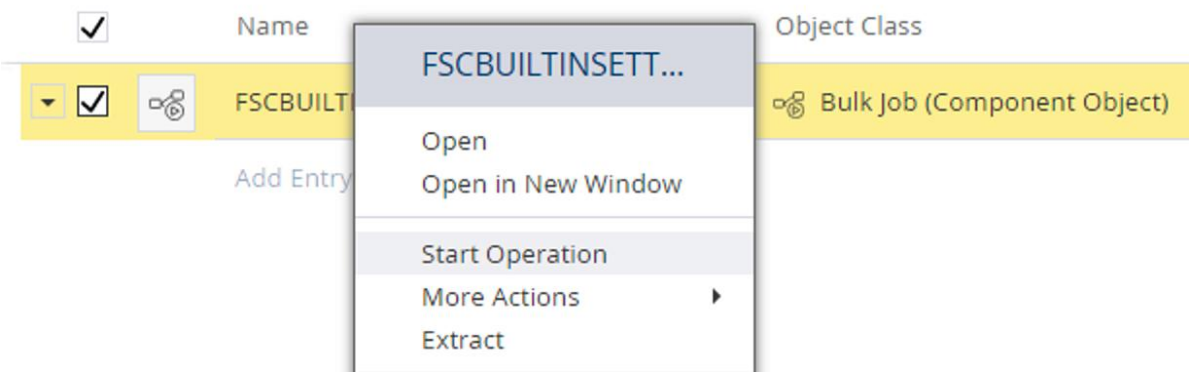
3 General

The functionality of the software component *Bulk Jobs* can be executed by using either the object class *Bulk Job (Component Object)* (`FSCBULK@1.1001:BulkOperation`) or the object class *Bulk Job* (`FSCBULK@1.1001:BulkOperationBasicObject`).

Both object classes provide the same functionality but the user interface of the *Bulk Job* is reduced and thus simpler to use than the counterpart as component object. This white paper describes the functionality using the object class *Bulk Job (Component Object)* exemplary.

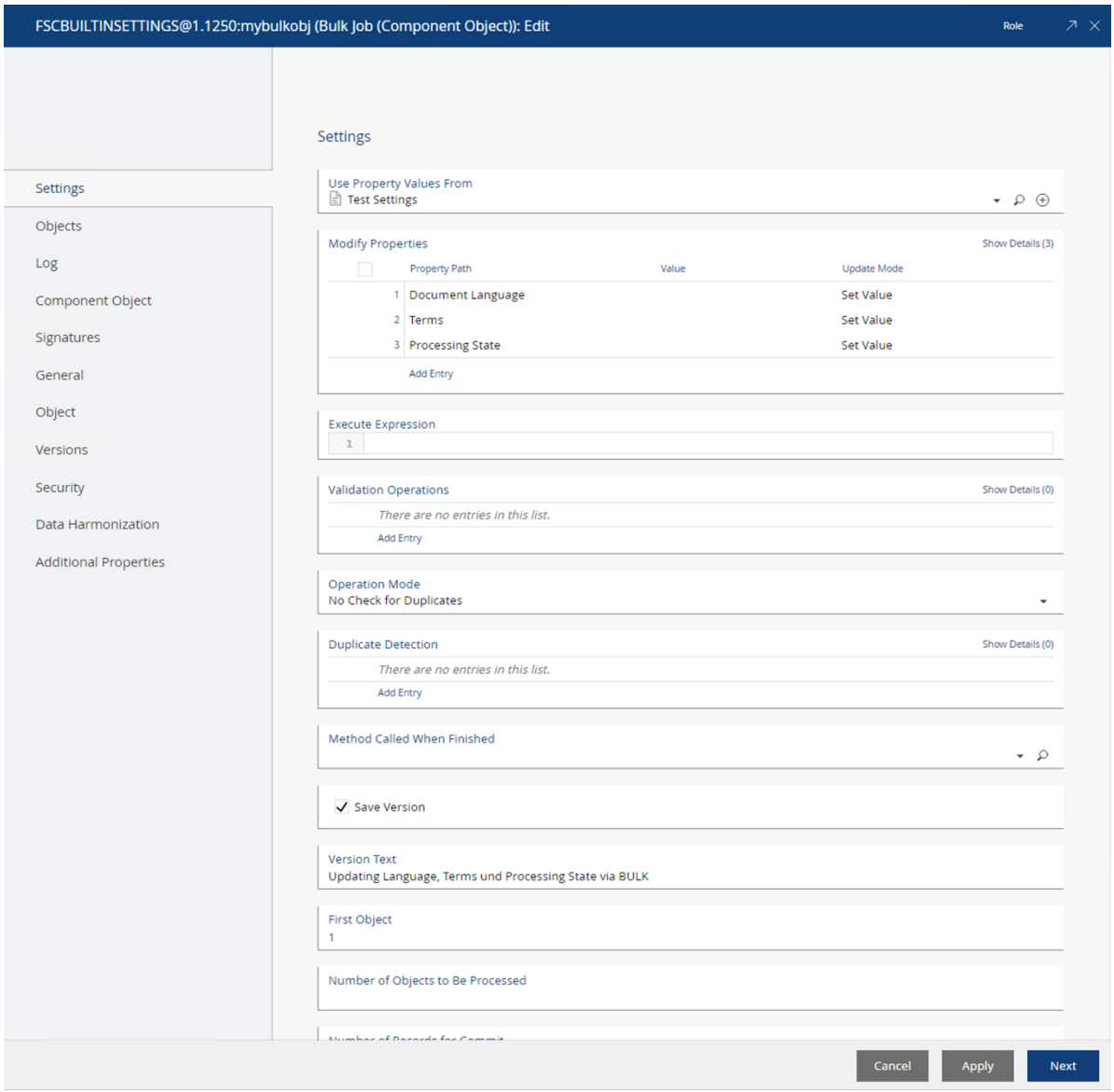
4 Definition of a Bulk Job

A bulk job can be executed via the context menu “Start Operation”.



Alternativly the action `FSCBULK@1.1001:StartOperationSync` can be used to execute a bulk operation using *Automated Tasks*.

4.1 Settings Tab



- *Use Property Values From*
  This property can be used to specify an object that can be used to specify property values, which will be copied from this object to the objects processed by the bulk job. Alternativly, property values can be defined in the property *Modify Properties*.
- *Modify Properties*
  Use this property to define which properties should be modified by the bulk job. If the *Value* is not specified the value will be used from the object defined in *Use Property Values From*.
- *Execute Expression*
  Use this property to define a Fabasoft Folio Expression that should be executed on each object of the bulk job. The current affected object is accessible via the variable `cooobj` or the global scope variable `::obj`. Additionaly the current bulk job object can be accessed via the global scope variable `::bulkobj`.

- *Validation Operations*
  Use this property to define action and parameters, which should be called on each object of the bulk job.
- *Operation Mode*
  Use this property to define how the duplicate check should occur. Note that the mode *Compare to All objects (Dual Pass)* is only available if the object selection is based on an object class.
- *Duplicate Detection*
  Use this property to define properties that should be used to perform a duplicate check.
- *Method Called When Finished*
  Use this property to execute an action when the bulk job has finished.
- *Save Version*
  Set this property to true to create an object version of each object included in the bulk job.
- *Version Text*
  If *Save Version* is set to true, in this field the version text can be specified.
- *First Object*
  In this property specify the record from which starts the bulk process.
- *Number of Objects to Be Processed*
  Use this property to determine the maximum number of records to be processed
- *Number of Records for Commit*
  Use this property to determine the maximum number of records to be processed in one transaction.
- *Skip Setting Modification Properties*
  Use this property to determine whether the properties *Last Change on/at* and *Last Change by* are updated for the objects that are changed during the bulk job.
- *Simulation Mode (No Changes)*
  Set this property to true to simulate the process.

Attention: Fabasoft app.ducx Expressions or actions specified in the properties *Execute Expression* or *Validation Operation* which commit local transactions can not be simulated.

4.2 Objects Tab

- *Search Form / Select Result / Object Class*
  In this field a search form, select result or an object class can be specified. Objects returned from this selection are the objects that are affected by the bulk job.
- *Or Object List*
  If the property value of *Search Form / Select Result / Object Class* contains a search form this object list shows the objects returned by the query.
  If the property value of *Search Form / Select Result / Object Class* is empty this field can be used to define the objects that should be affected by the bulk job.
  This property is also accessible via the tree view.
- *Select Clause*
  Use this field to define additional query restrictions if the property value *Search Form / Select Result / Object Class* contains an object class.

- *Select Objects*
  Use this field to set additional restrictions based on the last execution time, if the property value *Search Form / Select Result / Object Class* contains an object class.
- *Last Start Time*
  The date defined in this property is used in conjunction with the *Select Objects* modes: "Only Objects Created After Last Start Time" or "Only Objects Updated After Last Start Time".
- *Query Scope*
  Use this field to define a query scope, if the property value *Search Form / Select Result / Object Class* contains an object class

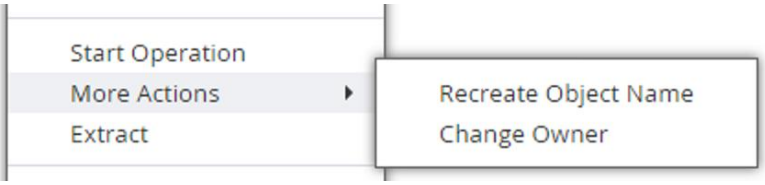
4.3 Log Tab

- *Logging Mode*
  In this field the scope of the created log file will be specified. Working with bulk jobs the *Logging Modes* "No Log", "Error Log" and "Detailed Log" are available. By default the mode "Error Log" will be used.
- *Folder for Logs*
  Log Objects created by the execution of the bulk job will be stored in a folder object referenced in this property.
- *Additional Object Properties in Log*
  Use this field to specify additional properties those values should be written to the log.
- *Display Layout*
  Use this this field to specifiy a XSL transformation that should be used to view the log. By default the XSL transformation *Default XSL Transformation for Bulk Jobs* will be used.

5 Use Case Implementation

The object classes *Bulk Job (Component Object)* and *Bulk Job* can be extended by solution specific bulk processes. The infrastructure of these object classes can be used to implement specific bulk operations using the base functionality provided by the software component `FSCBULK`.

Specific bulk implementations are available via the context menu "More Actions".



The only thing to do is to implement an action using the prototype `BulkActionPrototype` on the object classes *Bulk Job (Component Object)* and/or *Bulk Job*. The multinlingual name of this action is used as menu entry text.

Example

```
usecase ChangeOwner(parameters as BulkActionPrototype) {
  mlname = { }
  variant BulkOperation, BulkOperationBasicObject {
    impl = expression {
      obj.ObjectLock(true, true);
```

```
      User newowner = cooobj.newobjowner;
      Group newownergroup = cooobj.newobjgroup;
      if (newowner && newowner.active) {
        obj.objowner = newowner;
      }
      if (newownergroup && newownergroup.active) {
        obj.objowngroup = newownergroup;
      }
    }
  }
}
```

Additionaly it is possible to provide an action specific GUI when starting the bulk job. Therefore add a solution specific form for the action implemented above.

Example

```
extend class BulkOperation {
  forms {
    ChangeOwner {
      FormSetOwners;
    }
  }
}
```