

PUBLIC

SAP Build Process Automation

**Cost Center Change Multiple (BAPI)**



# Contents

<b>Change Log</b> .....	<b>3</b>
<b>Overview</b> .....	<b>4</b>
<b>Prerequisites</b> .....	<b>4</b>
<b>Capabilities</b> .....	<b>4</b>
<b>Procedure</b> .....	<b>5</b>
<b>Project Configuration</b> .....	<b>5</b>
<b>Project Execution Flow</b> .....	<b>6</b>
<b>Project Overview</b> .....	<b>7</b>
<b>Release &amp; Deploy Project</b> .....	<b>8</b>
<b>Modes of Execution</b> .....	<b>8</b>
<b>Debugging Automation</b> .....	<b>8</b>
<b>Dependencies</b> .....	<b>9</b>

# Change Log

Version	Date	Description
1.0.0	May 19 <sup>th</sup> , 2023	Document created
1.0.1	March 22 <sup>nd</sup> , 2024	Document updated



# Overview

This document provides the information about the configuration & execution of BAPI (Business Application programming Interface) business project. This project 'Cost Center Change Multiple (BAPI)' is used for changing one or more cost centers using data provided in the input excel file.

SAP BAPI is a standard interface to the business object models in SAP products. BAPIs are the primary method through which customer code and third-party applications interact with SAP products. BAPIs wrap the internal layers of SAP's business object model to ensure that all business logic, validations, and authorization checks are executed properly when accessing or changing business objects (For more information [click here](#)).

This package consists of list of sub automations including *Cost Center Change Multiple - Start, BAPI Execution & Excel Update*. It also includes BAPI action group, data types to handle BAPI inputs & system details and input excel file. The BAPI Action group is a collection of actions that is responsible for execution of BAPI functions & return of BAPI response.

The actions from BAPI action group also contains following features:

- The BAPI Action group allows us to execute multiple operations such as set connection, open connection, execute BAPI, commit transaction & close connection as a single automation activity.
- It provides the data type for BAPI input & BAPI response that contains all the input, output & test parameters for BAPI function BAPI\_COSTCENTER\_CHANGEMULTIPLE (for Cost Center Change Multiple).

## Prerequisites

- The project template requires SAP Build Process Automation subscription or CPEA contract. Follow the [setup and configuration section](#).
- The setup of desktop agent can be checked under "Configure Automation Capabilities" section from the above link.
- Desktop agent version greater than 3.15.23.
- For usage of BAPI methods in an automation please check the prerequisites [here](#).
- Configure SAP connection (SSO or Basic based on your requirement). For more information you can check following links:
  - o [BAPI Connection System](#)
  - o [Best Practices for SAP BAPI Activities](#)

## Capabilities

This business project consists of the following capabilities.

- This project has the capability of reading input excel file & converting the data into BAPI input data format.
- This project can establish connection to the SAP system, executes the BAPI function with the use of BAPI action group.
- It can read the BAPI response & update the required response data to the output excel file.

Name	Description
BAPI function name: BAPI_COSTCENTER_CHANGEMULTIPLE	This BAPI function on execution updates the values for cost center.
Business project name: Cost Center Change Multiple (BAPI)	This is used for Cost Center Change Multiple based on the input data provided in input configuration file.

## Procedure

Followings provides the project setup & configuration details along with project features & execution details:

### Project Configuration

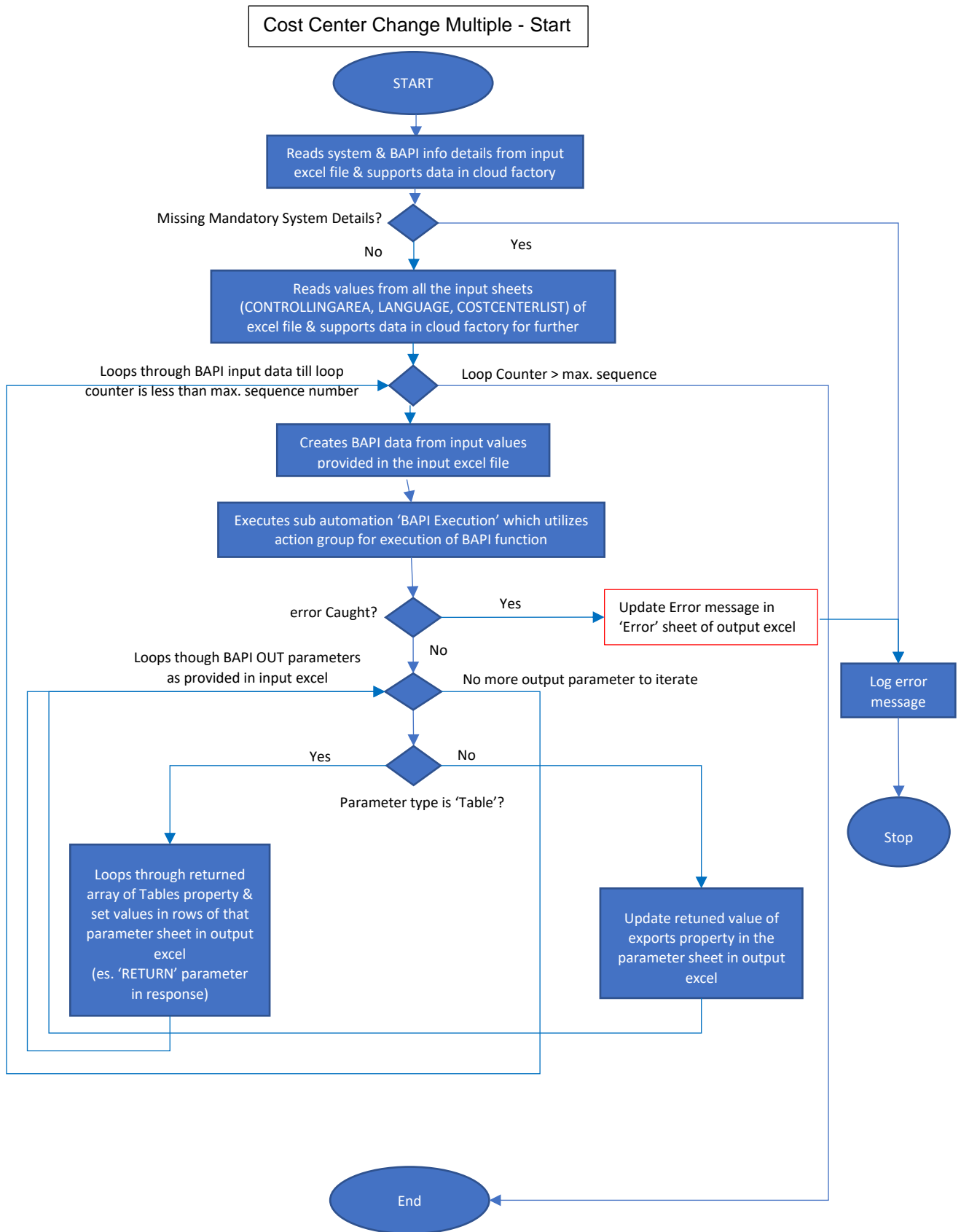
Following steps can be followed for the setup of business project:

- Navigate to the Store in your SAP build process automation application.
- In the Search bar, type “Cost Center Change Multiple (BAPI)”.
- Click “**Create from Template**” button to create the new project that will be available in the lobby.
- After successful project creation, project will be opened.
- Download the input configuration file configBAPI\_CostCenterChangeMultiple.xlsx from the business project.
- Input configuration file contains information like system details, BAPI info, input parameters (CONTROLLINGAREA, LANGUAGE, COSTCENTERLIST) & output parameters (RETURN) details:
  - System details sheet contains information of the backend ECC system like system id, app server, instance number, client and SNC.
  - BAPI Info sheet contains information like function module, Import, table & export details.

### Notes:

- Provided input excel file contains test data & must be updated with new data as per the requirement.
- ‘Error’ sheet needs to be provided in the input excel file (as provided in sample input file) to capture any error.
- Input parameters needs to be provided in the same format as in sample input excel file (With first row as property name for which you can refer to the action group **costCenterChangeMultiple** in the project, second row must be the description (or NA) & data values need to be provided in subsequent rows)  
Make sure to add input excel path in the input Parameter while creating automation trigger for deployed package (For more info check section: ‘Modes of Execution’).

# Project Execution Flow



## Project Overview

- The project execution starts with sub automation *Cost Center Change Multiple-Start*.
- Automation reads the input file for the system data & BAPI info data from input excel file & supports the obtained data in the project for further execution.
- Automation performs the check for mandatory system details.

### **Mandatory System Details missing:**

- Logs error message about missing system details.
- Automation stops the execution.

### **Mandatory System Details provided:**

- Automation continues with execution.
- Automation loops through BAPI input sheets (CONTROLLINGAREA, LANGUAGE, COSTCENTERLIST) & reads the required data & supports it for further execution.
- Automation loops through BAPI input data for all the sequence number as provided in input excel:
  - Automation creates the required data format from the provided input data.
  - Automation utilizes BAPI input data variable created through action group for BAPI execution & its values are updated with created BAPI data in previous step.
  - Sub automation 'BAPI Execution' is called that performs actual BAPI execution via action group functionality.
  - Action group of 'BAPI Execution' sub automation establishes the connection to SAP system, executes the BAPI function (i.e., BAPI\_COSTCENTER\_CHANGEMULTIPLE), commits the BAPI transaction, closes the SAP connection & finally returns the BAPI response.
  - For any exception/error during the BAPI execution, automation performs followings or else it continues with further execution:

### **Execution Error:**

- Automation catches the error & logs the error message.
- Automation updates the error message in the 'Error' sheet of the output excel file.
- Automations stops the execution.
- Automation loops through BAPI output parameters (RETURN) as provided in BAPI Info sheet of input excel file:
  - Automation checks if output parameter is of type 'Tables' or not.
  - If property is a Table, automation loop through the array values of the property & updates the rows of the output.
  - If there is any response (error message) from BAPI execution, it will be updated in the output excel sheet otherwise for successful execution we will get empty RETURN parameter.
  - If property is not a Table (i.e., it's an Exports property), automation updates the output sheet with the property values.
- Automation increments the loop counter for next sequence number & continues with the execution.
- Finally, automation ends with completion of execution.

## Release & Deploy Project

The project must be released and deployed for the usage.

- For information on project release click [here](#).
- For information on project deployment click [here](#).

**Note:** During project deployment you can select the default environment “public” or might create your environment (For more information [check here](#)).

## Modes of Execution

Once you have your project deployed successfully, you will have following two options to trigger the automation bot:

- **Standard/Unattended execution:**

Unattended execution of the project can be created by selecting scheduled option during the automation trigger & providing required scheduling information.

- **Attended execution:**

The package has a project pane with a menu item for triggering attended execution. Attended execution can be created by selecting attended option during automation trigger & your automation will be available as menu item within the agent menu in the system.

- Automation “Cost Center Change Multiple – Start” starts the execution of project which requires path of the input excel file that must be provided by clicking “Input parameter” while creating the automation trigger.
- Please [click here](#) to create the automation trigger.

## Debugging Automation

Automations can be tested in the automation editor & its debugging feature can be utilized to test your automation flow & data values.

For detailed information please check [here](#).



## Dependencies

The table below details the versions used to generate this business project:

Component	Version
Desktop Agent	3.15.23
Factory/Tenant	2304.04
SDK Core	1.36.29
SDK Excel	1.36.29
SDK BAPI	1.36.29
Application	Version
Saplogon.exe	Release 800
Microsoft Excel	Version 2102