External Tool Integration API in Focused Build
How to integrate external software development tools

TABLE OF CONTENTS

In context of Focused Build an external tool integration API is provided. This API enables you to connect an external software development tool like e.g. Atlassian Jira or Microsoft Team Foundation Server to SAP Solution Manager. To keep the API generic, the communication is based on OData web services.
The Solution Manager side of the API is delivered as part of the ST-OST add-on but the integration on the external tool side is not delivered by SAP due to the possible amount of variations (combination of different tools with different version/setup/customizing).

This document describes the capabilities of the API and provides the necessary technical details to build and integration on the external tool side.
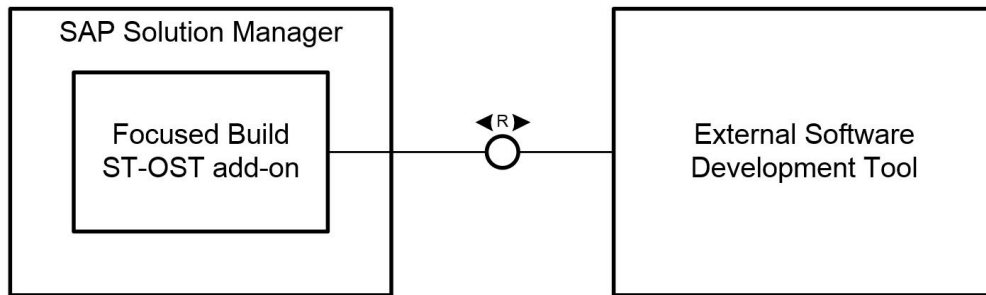
The API consists of two parts:
- The outbound channel, which is calling an OData web service of the external tool in order to create and update work packages/items in the external tool.
- The inbound channel, which is a web service, called by the external tool in order to update the status of an existing work package/item.

In Focused Build the outbound channel is triggered automatically via Change Request Management Framework actions. The action name is S1_CALL_EXT.

The customizing of the API in Solution Manager 7.1 is done via the view cluster /SALM/VC_EXI which is always the entry point for the customizing settings described below in chapter 1, chapter 2 and chapter 3.

As of Solution Manager 7.2 the customizing is part of the Focused Build IMG. The differences in customizing are outlined in chapter 4.

1   PROJECT MAPPING AND CONNECTIVITY

In Focused Build work packages and work items are assigned to cPro projects. The configuration entity for the API is the cPro project level. This means you can have different API configurations per cPro project. This even allows you to send work packages and work items from one cPro project to external tool A while work packages and work items from other cPro projects are synchronized with external tool B.
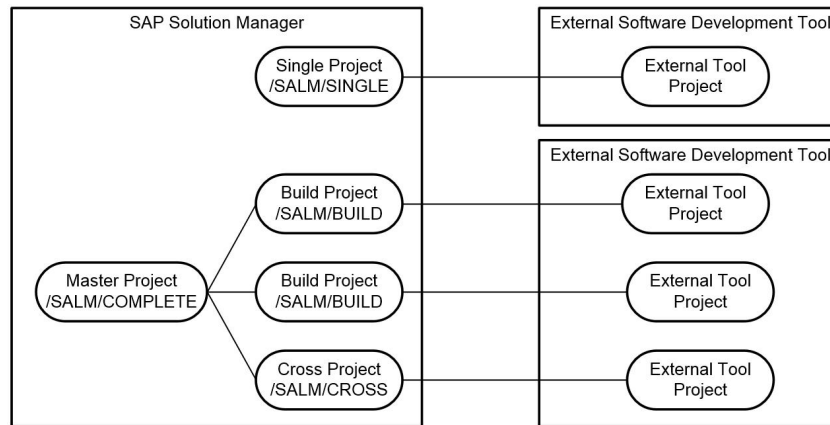
1.1   Project Mapping Customizing in Solution Manager 7.1

The above described API configuration concept is reflected in the API customizing (transaction SM34, view cluster /SALM/VC_EXI). Entry point is the *Project Mapping* where the key of the API configuration set is the cPro *project number*.

On this level you define via the *destination*, which is a RFC destination of type G (HTTP connection to external server), the external tool that is called for the outbound web service.
To generate the URL to the OData web service of the external tool the API uses the host, port and protocol out of the RFC destination and concatenates it with the URIs defined in the project mapping customizing node. To provide full flexibility you could define different URIs for create and update request and also define the HTTP method. In standard OData web service the create call uses the POST method and the update call the PUT method.
In order to update an object in the external tool the OData service normally requires the object ID in the URL. The object ID of the external tool is stored in Solution Manager as EXTERNAL_ID. By masking the EXTERNAL_ID with # characters you can add the EXTERNAL_ID to the URL.

Example:
/plugins/servlet/IssueService.svc/Issues('#EXTERNAL_ID#')
With an EXTERNAL_ID = BRF-123 this will result in an update URL like:
https://externaltool.sap.corp:8083/plugins/servlet/IssueService.svc/Issues('BRF-123')

In case external tool also uses a grouping entity for the work packages and work items like a project and has a specific project entry page URI then you can configure it in the *Jump in URL* field. In case this field is filled the link shows up in the project maintenance UI:

1.2   Connectivity to External Tool

The destination used in the API configuration needs to be created in transaction SM59. It needs to be a connection of type G. On the *technical settings* tab you need to maintain the host (*Target Host*) and https port (*Service No.*) of your external tool. On the *Logon & Security* tab you have to activate SSL. Prerequisite for this is that your Solution Manager has SSL correctly configured and you need to make sure that the selected SSL Client trusts the certificate of your external tool.

For the communication to the external tool we use a technical user which authenticates with Basic Authentication. User and password are maintained also on the *Logon & Security* tab. For security reasons it is therefore important that you setup the connectivity via SSL.

2 CREATE / UPDATE WORKPACKAGE/WORKITEM IN EXTERNAL TOOL

As we want to provide a generic API we use OData web service calls. The API generates out of the work package and work item a JSON payload file which is send to the OData web service of the external tool. The content of the JSON payload file is customizable.

In addition to "simple" properties work packages and work items can also have attachments and attached documents (e.g. specification documents, test cases…). In further course they are only called documents. As the call to the external tool is synchronous we do not want to overload it by sending all documents with this call. Instead the payload JSON includes download references to the documents which then need to be used by the external tool to download the documents and attach it to the corresponding external tool object (see sequence diagram below).
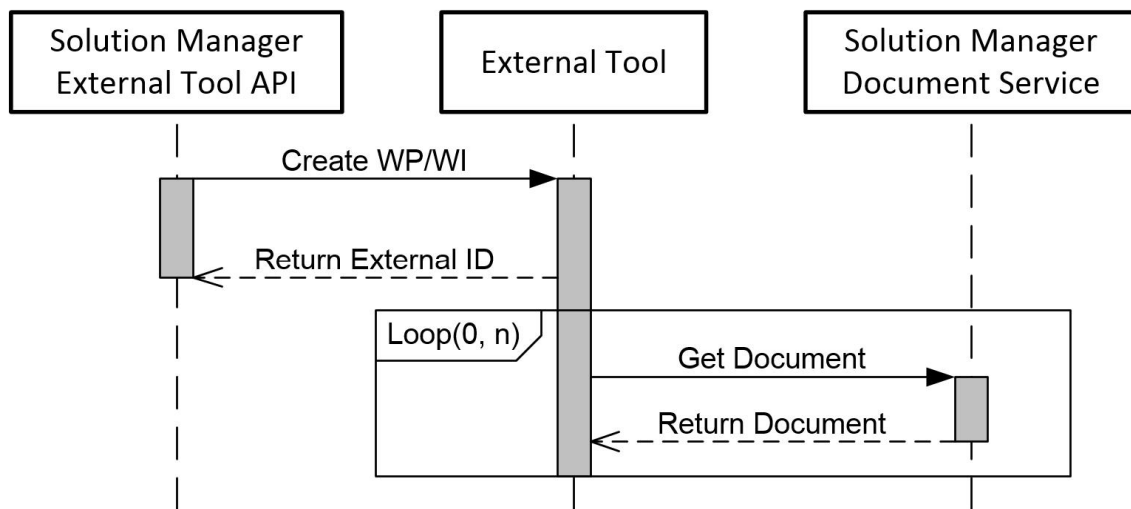
When the create call was successful the external tool needs to return the EXTERNAL_ID (object identifier in the external tool) which is then stored by Solution Manager. The API also uses the EXTERNAL_ID field to decide if the method of the call to the external tool is create or update. In case the EXTERNAL_ID is empty it will be a create call. Otherwise an update call will be done where you can use the EXTERNAL_ID as described in the Project Mapping Customizing section above.

2.1 Process Type Customizing

In Focused Build you have work packages and work items. Technically they are just different business transaction types (process types). In Focused Build the following process types are used:

SolMan Process Type	SolMan Description
S1IT	Work Package
S1CG	Work Item (General Change)
S1MJ	Work Item (Normal Change)

As the API behavior needs to be different depending on the process type you can configure the API behavior for each process type individually. This is reflected in the API customizing accordingly.
In the *Process Types* node you define which business transaction types can be used by the API:

With the *Jump in URL* field you can set a URI which is used to create a direct jump-in link to the object in the external tool. The host, port and protocol information from the destination defined in the project mapping is concatenated with the here defined URI. Also here you can mask the EXTERNAL_ID with # characters to add the EXTERNAL_ID to the URL.
Similar to the project also for the process types the jump-in link is only shown in the work package and work item UI5 application in case the *Jump in URL* field is maintained and the EXTERNAL_ID is not empty.

In the *Process Types* node you also have to maintain a *Partner Function*. The business partner assigned to the process type with the specified partner function is used to decide if the API is triggered. In Focused Build it is normally check against the partner function /SALM/09 (Dev Team). For this business partner the identification number is checked and only in case the ID Type S1EXTI is set with the Identification Number X the outbound channel of the API will be triggered.

2.2   OData Payload Format

The generated JSON payload uses the standard JSON format. In case a property can have more than one value (e.g. WRICEF_ATTRIBUTES, PARTNER, APPOINTMENT…) arrays are used. You find an example in the Appendix section.

2.3   Object Dependencies

In Focused Build there is always a 1:n relation between work packages and work items. The API to create the work package and corresponding work items in the external tool is triggered when the action handover to development is executed. In order to enable the external tool to reflect this 1:n relation Solution Manager uses the EXTERNAL_ID and the PARENT_EXTERNAL_ID. Once EXTERNAL_ID is returned for the work package it is used as the PARENT_EXTERNAL_ID of the corresponding work items.
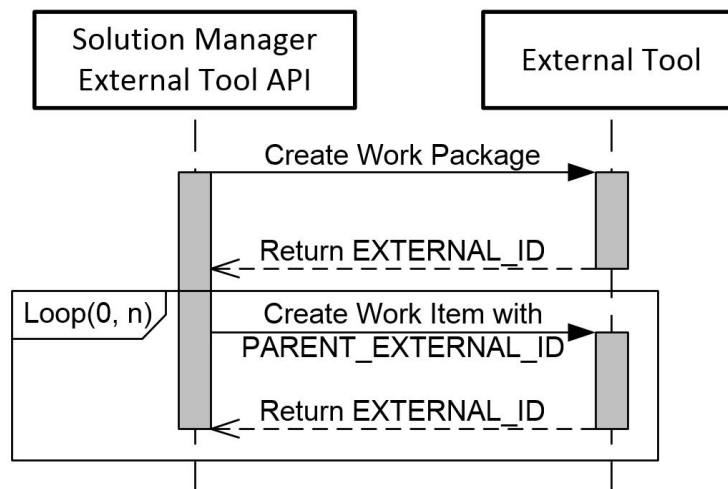
By this the external tool can also build up the correlation between work package and work item.

2.4   Payload Content Customizing

The payload content customizing is structured in different sections per process type.
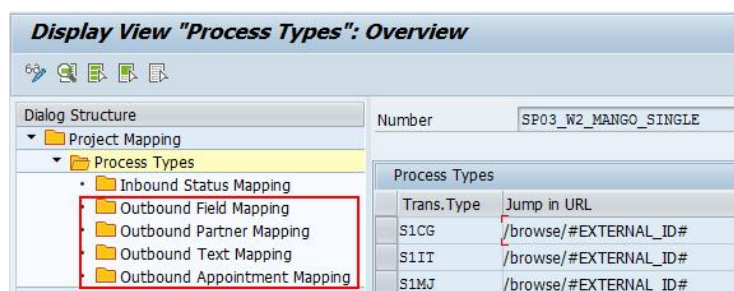
2.4.1 Outbound Field Mapping

In the outbound field mapping you define which attributes of the work package/item are send to the external tool. You can either take the attribute value from the work package/item or overwrite it with a fixed value.
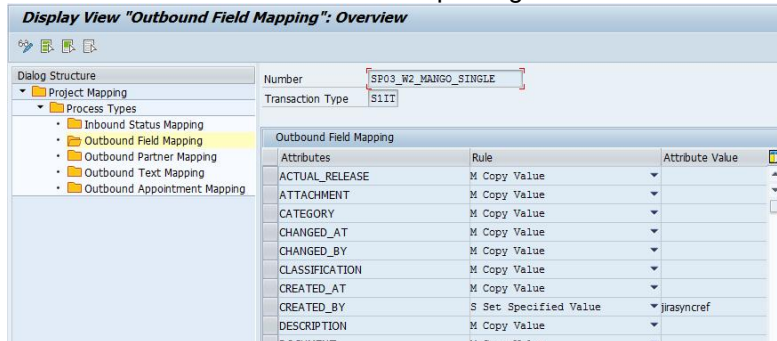
Attribute
The here defined attribute will be part of the JSON send to the OData web service of the external tool.

Rule
The Copy Value rule will send the attribute value of the work package/item to the external tool.
The Set Specific Value rule will send the fixed attribute value as it is defined here in the customizing.

Attribute Value
This is the fixed attribute value which is send in case the Set Specific Value rule is set.

2.4.2 Outbound Partner Mapping

In the outbound partner mapping you define which business partner information is send to the external tool based on the partner function
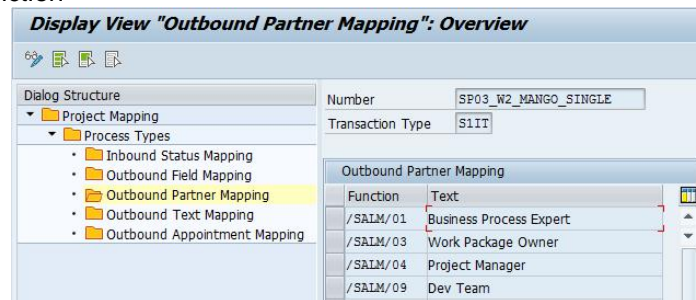
Function
This is the partner function used to identify the business partner. E.g.:
/SALM/01 (Business Process Expert)
/SALM/03 (Work Package Owner)
/SALM/04 (Project Manager)

In case there is more than one business partner per partner function assigned to the work package there is always one main contact. This is then reflected in the parameter PARTNER_MAIN.

2.4.3 Outbound Text Mapping

In the outbound text mapping you define which texts of the work package/item is send to the external tool.
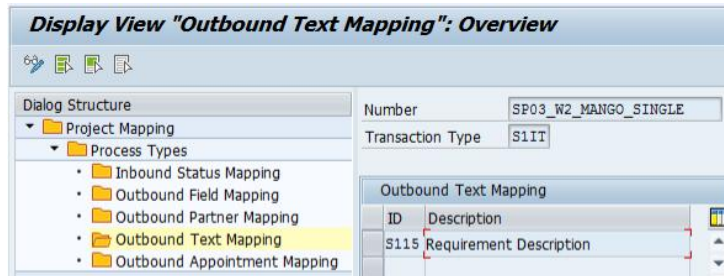
ID
The text of the work package/item with this Text ID is send to the external tool. E.g.:
S115 (requirement description; for work package)
CD99 (overview; for work item)

2.4.4 Outbound Appointment Mapping

In the outbound appointment mapping you define which appointments of the work package/item are send to the external tool.
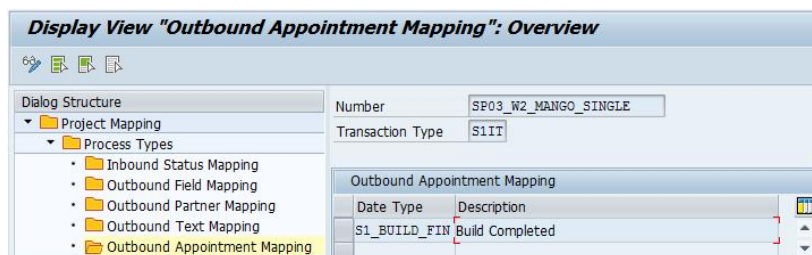
Date Type
The appointment with this date type is send to the external tool. E.g.:
S1_BUILD_FIN (Build Complete)

2.5 Document and Attachment Handling

As already mentioned above the attached documents and attachments are not directly send with the outbound OData web service call. Instead the payload JSON includes the property information of the document and also a download link for the documents. For examples see the Appendix section.

On the external tool side a logic needs to be implemented to download and attach the documents after the work package and work item was created on external tool side.

2.6 OData Response

In case of a successful creation at least the EXTERNAL_ID needs to be returned by the OData web service. See the Appendix section for an example.

2.7   External Tool OData Web Service Requirements

To use the external tool integration API the external tool needs to provide an OData web service which has the following requirements implemented:

- It needs to provide as authentication procedure basic authentication. In addition the web service should be called via HTTPS.
- It needs to accept the JSON payload send by the Solution Manager API.
- It needs to map the Solution Manager fields to the corresponding fields in the external tool.
- It needs to return at least the EXTERNAL_ID in case of a successful object creation.
- It needs to download documents asynchronously after the creation of the object.
- As Solution Manager is the leading system for work packages the corresponding objects on external tool side should not be modified after the creation. Only status updates from via the OData web service call should be done.
- The TICKET_GUID needs to be stored on external tool side to be able to utilize the status update web service of Solution Manager.

3 UPDATE WORKITEM STATUS IN SOLUTION MANAGER

In case a work item is synchronized to an external tool in context of Focused Build the external tool will be the leading system for this work item. To ensure the full capabilities of Focused Build the status changes of work items in the external tool need to be synchronized back to Solution Manager. Therefore Solution Manager provides an OData web service which can be called by the external tool.
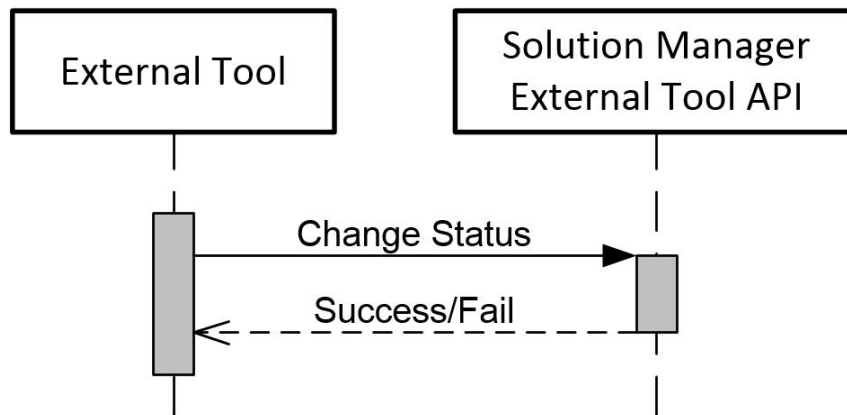
3.1 OData Web Service end point

The OData Web Service end point on Solution Manager side is:
```
https://<SolMan host>:<https port>
/opu/odata/SALM/EXT_INTEGRATION_SRV/StatusSet(Guid='TICKET_GUID')
```

When calling this OData web service the TICKET_GUID is used to identify the work item on Solution Manager side. The TICKET_GUID was provided by Solution Manager in the create JSON payload and needs to be used in in the URL as well as in the OData payload JSON.

3.2 OData Payload Format

The OData payload send to Solution Manager needs to have the following JSON content:

```
{
    "ExtStatus": "EXTERNAL_STATUS",
    "Guid": "TICKET_GUID"
}
```

The EXTERNAL_STATUS can be any string which is then mapped on Solution Manager side to the corresponding status.

3.3 Inbound Status Mapping Customizing

In the *Inbound Status Mapping* customizing you can define the status mapping used when the external tool calls Solution Manager to update the status of a work item. Like for the outbound mapping the status mapping is done per process type.
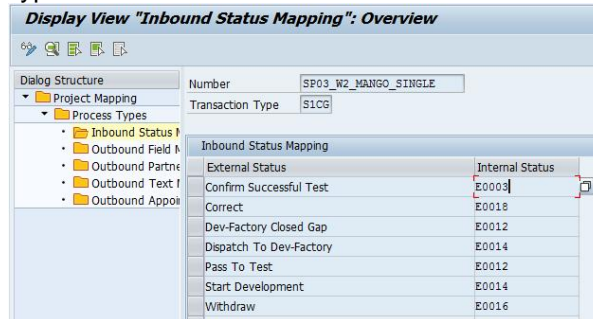
External Status
This is a string send by the external tool.

Internal Status
This is the internal status the work item should change to.

3.4 OData Response

In case of a successful status change on Solution Manager side the OData web service just returns an HTTP 204 status code.
In case the status change fails Solution Manager will return a HTTP 500 status code and provide in the response body further information about the exception. For an example see the Appendix section.

3.5 External Tool OData Client Requirements

To use the status update OData web service the external tool client side needs to implement the following requirements:
- Connectivity data to Solution Manager like web service URL and logon credentials needs to be stored on external tools side.
- It needs to use the correct JSON payload
- It needs to use the TICKET_GUID to identify the work item on.
- In case the status change fails on Solution Manager side either the status change on external tool side needs to be blocked or a queuing mechanism needs to be implemented to re-execute the status change.

4   CUSTOMIZING IN SOLUTION MANAGER 7.2

In Solution Manager 7.2 the customizing is done via transaction SPRO.
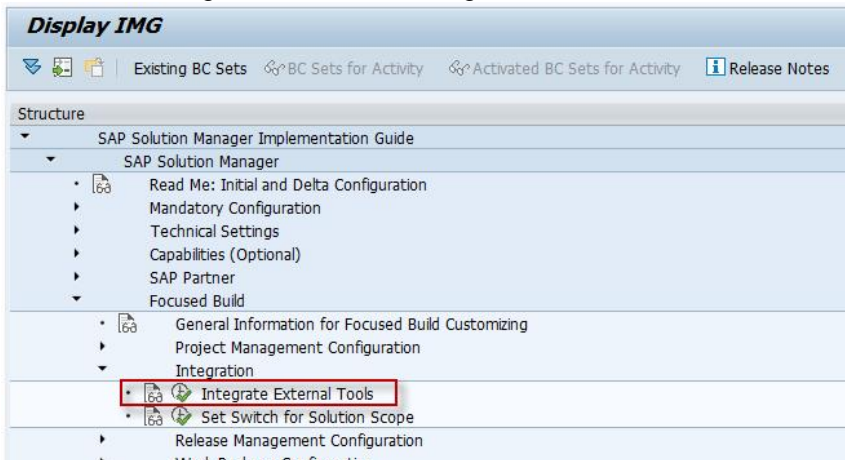
The difference to Solution Manager 7.1 is that there is an additional configuration layer between the field mapping customizing and the mapping to the project.
When you execute the *Integrate External Tools* customizing you get two activities:
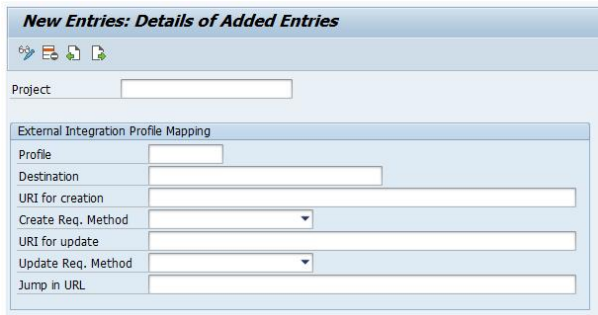
In the *External Tool Integration* activity, you create a profile in which you define the mapping as described in chapter 2 and chapter 3. There is one delivered profile /SALM/FBDF which you can use as an example.
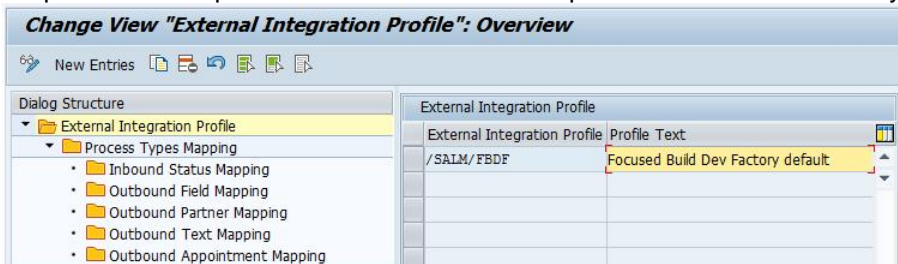
In the *Mapping for external Tool Integration* activity you do the project mapping customizing settings. You have to fill all fields as described in chapter 1. The only additional field is the profile field. Just take the profile you created in the *External Tool Integration* activity.

New Entries: Details of Added Entries

Project   [ ]

External Integration Profile Mapping

Profile	[ ]
Destination	[ ]
URI for creation	[ ]
Create Req. Method	[▼]
URI for update	[ ]
Update Req. Method	[▼]
Jump in URL	[ ]

5 APPENDIX

5.1 Create Payload JSON Example

Payload

```
{
        "SYSTEM_GUID": "00505683226E1EE5AED98BF0E4AD82E1",
        "EXTERNAL_ID": "",
        "PARENT_EXTERNAL_ID": "",
        "TICKET_ID": "8000013587",
        "TICKET_GUID": "00505683226E1EE5BEFB3A231046E491",
        "DESCRIPTION": "Work Package J45_PO Form_01",
        "PROCESS_TYPE": "S1IT",
        "CREATED_AT": "20160406092754",
        "CREATED_BY": "JIRASYNC",
        "CHANGED_AT": "20160705140732",
        "CHANGED_BY": "testuser",
        "STATUS": "E0022",
        "STATUS_TXT": "To be Developed",
        "SOLDOC_PROJECT": "ST_OST_SP2",
        "SOLDOC_PROJECT_TXT": "ST-OST SP02 Master Projekt",
        "PRIORITY": 2,
        "PRIORITY_TXT": "2: High",
        "IMPACT": 0,
        "IMPACT_TXT": "",
        "URGENCY": 0,
        "URGENCY_TXT": "",
        "EFFORT": 24480.00,
        "EFFORT_UNIT": "MIN",
        "PPM_PROJECT_ID": "SP03_W1_MABU_2",
        "PPM_PROJECT_TXT": "MANGO Build Project 2 SP03 Wave 1",
        "PPM_WAVE": "00000000000000000000000000000000",
        "PPM_WAVE_TXT": "",
        "PPM_START_DATE": "00000000",
        "PPM_FINISH_DATE": "00000000",
        "PPM_COMPLETION": 30,
        "PPM_TOTAL_WORK": 0,
        "PPM_TOTAL_WORK_UNIT": "",
        "REQUESTED_RELEASE_NO": "",
        "REQUESTED_RELEASE_CLASS": "",
        "REQUESTED_RELEASE_TYPE": "",
        "REQUESTED_GO_LIVE_DATE": "00000000",
        "FORECAST_RELEASE_NO": "1.0",
        "FORECAST_RELEASE_CLASS": "ST-OST_SP2",
        "FORECAST_RELEASE_TYPE": "S_MAJOR",
        "FORECAST_GO_LIVE_DATE": "20150928",
        "ACTUAL_RELEASE_NO": "1.1",
        "ACTUAL_RELEASE_CLASS": "ST-OST_SP2",
        "ACTUAL_RELEASE_TYPE": "S_MINOR",
        "ACTUAL_GO_LIVE_DATE": "20151005",
        "CLASSIFICATION": "1",
        "CLASSIFICATION_TXT": "WRICEF",
        "WRICEF_ATTRIBUTES": [{
                "KEY": "R",
                "VALUE": "Report"
        },
        {
                "KEY": "F",
                "VALUE": "Form"
        }],
        "PARTNER": [{
                "PARTNER_FCT": "/SALM/01",
                "PARTNER_FCT_TXT": "Business Process Expert",
                "PARTNER_NO": "786",
                "PARTNER_NAME": "Nicolas A",
                "PARTNER_EMAIL": "nicolas.a@test.com",
                "PARTNER_MAIN": ""
        },
        {
                "PARTNER_FCT": "/SALM/03",
                "PARTNER_FCT_TXT": "Work Package Owner",
                "PARTNER_NO": "KF",
```

```
                "PARTNER_NAME": "Frank K",
                "PARTNER_EMAIL": "frank.k@test.com",
                "PARTNER_MAIN": ""
        },
        {
                "PARTNER_FCT": "/SALM/04",
                "PARTNER_FCT_TXT": "Project Manager",
                "PARTNER_NO": "801",
                "PARTNER_NAME": "Michael K",
                "PARTNER_EMAIL": "michael.k@test.com",
                "PARTNER_MAIN": ""
        },
        {
                "PARTNER_FCT": "/SALM/09",
                "PARTNER_FCT_TXT": "Dev Team",
                "PARTNER_NO": "4000",
                "PARTNER_NAME": "External Dev Team 1",
                "PARTNER_EMAIL": "",
                "PARTNER_MAIN": ""
        }],
        "APPOINTMENT": [{
                "APPT_TYPE": "S1_BUILD_FIN",
                "APPT_TYPE_TXT": "200ES1_BUILD_FINBuild Completed",
                "TIMESTAMP": "20160205000000",
                "TIMEZONE": "UTC",
                "IS_DURATION": "",
                "DURATION": 0,
                "TIME_UNIT": ""
        }],
        "TEXT": [{
                "TEXTID": "S115",
                "TEXTID_TXT": "",
                "CHANGED_BY": "KF",
                "CHANGE_DATE": "20160406",
                "CHANGE_TIME": "113420",
                "TEXT": "Requirement_J45_POform_1\\n\\nSAP's PO Form does not meet customer
expectation. Layout need to be changed according to corporate design from customer.\\nIn addition 10
fields need to be added to each PO Line Item and 2 fields to PO header."
        }],
        "CATEGORY": [{
                "GUID": "005056836E761EE4A0A37568FF6E0944",
                "DESCRIPTION": "Industry Specific Solution",
                "LEVEL": 3
        },
        {
                "GUID": "005056836E761EE4A0A37568F9B3C939",
                "DESCRIPTION": "Functional",
                "LEVEL": 2
        },
        {
                "GUID": "005056836E761EE4A0A37568F9038938",
                "DESCRIPTION": "Applications",
                "LEVEL": 1
        }],
        "PROCESS_STRUCT": [{
                "PROJECT_ID": "ST_OST_SP2",
                "NODE_ID": "710FB9CF34604ABBA6FEB2A641F27654",
                "PROCESSSTEP_TXT": "Check Freight Order Update",
                "PROCESS_TXT": " E2E TM-20 Transport Planning Full Truck or Less Truck Load",
                "SCENARIO_TXT": " E2E Processes"
        },
        {
                "PROJECT_ID": "ST_OST_SP2",
                "NODE_ID": "8FE8D320FC66488DB8E7740B94EE54DB",
                "PROCESSSTEP_TXT": "Check Freight Order Update",
                "PROCESS_TXT": " FSB-20-10 Invoice Posting",
                "SCENARIO_TXT": " PH FSB - Freight Settlement & Billing"
        }],
        "DOCUMENT": [{
                "FILENAME": "Functional Specification_J45.docx",
                "TITLE": "Functional Specification_J45.docx",
                "STATUS": "COPY_EDITING",
                "STATUS_TXT": "Copy Editing",
                "DOCU_TYPE": "ZFW",
```

```
            "DOCU_TYPE_TXT": "Functional Specification type WRICEF",
            "AUTHOR": "LZ",
            "AUTHOR_NAME": "LZ",
            "URL":
"http://ldci.sap.corp:50089/sap/bc/solman/SolmanDocuments/200?_CLASS=SOLARGEN&_LOIO=00505683226E1ED6
8CF740FDA3BA396B&LANGUAGE=EN&RELEASE=620&IWB_INDUSTRY=/KWCUST/&TMP_IWB_TRY_OTHER_LANG=X&TMP_IWB_TRY_
OTHER_IND=X&TMP_IWB_TASK=PREVIEW2&"
        },
        {
            "FILENAME": "Test Case 1.docx",
            "TITLE": "Test Case 1.docx",
            "STATUS": "COPY_EDITING",
            "STATUS_TXT": "Copy Editing",
            "DOCU_TYPE": "ZFT",
            "DOCU_TYPE_TXT": "Single Functional Test",
            "AUTHOR": "SR",
            "AUTHOR_NAME": "SR",
            "URL":
"http://ldci.sap.corp:50089/sap/bc/solman/SolmanDocuments/200?_CLASS=SOLARGEN&_LOIO=00505683226E1EE6
8FC20A10CAC24149&LANGUAGE=EN&RELEASE=620&IWB_INDUSTRY=/KWCUST/&TMP_IWB_TRY_OTHER_LANG=X&TMP_IWB_TRY_
OTHER_IND=X&TMP_IWB_TASK=PREVIEW2&"
        }],
        "ATTACHMENT": [{
            "FILENAME": "Attachment",
            "TECHN_FILENAME": "Attachment.docx",
            "EXTENSION": "DOCX",
            "URL":
"http://ldci.sap.corp:50089/sap/bc/contentserver/200?get&pVersion=0046&contRep=CRMORDER&docId=005056
83226E1EE690D75FC4C98E7161&compId=Attachment.docx&accessMode=r&authId=CN%3D*.wdf.sap.corp,OU%3DST7,O
U%3DSAP-
AG,0%3DSAPTrustCommunity,C%3DDE&expiration=20160705161635&secKey=MIIBUwYJKoZIhvcNAQcCoIIBRDCCAUACAQE
xCzAJBgUrDgMCGgUAMAsGCSqGSIb3DQEHATGCAR8wggEbAgEBMG8wYzELMAkGA1UEBhMCREUxHDAaBgNVBAoTE1NBUCBUcnVzdCB
Db21tdW5pdHkxDzANBgNVBAsTB1NBUC1BRzEMMAoGA1UECxMDU1Q3MRcwFQYDVQQDDA4qLndkZi5zYXAuY29ycAIICiAVBwcFKAE
wCQYFKw4DAhoFAKBdMBgGCSqGSIb3DQEJAzELBgkqhkiG9w0BBwEwHAYJKoZIhvcNAQkFMQ8XDTE2MDcwNTE0MTYzNVowIwYJKoZ
IhvcNAQkEMRYEFIyNmDtNqAGFUo5c4fOx8tmdRGAsMAkGByqGSM44BAMEMDAuAhUAztFE0rMHz1JcpfgLOj7Z9%2FC7b%2BsCFQC
PWySPe6Gz21F3EJ6SWliWgfcCiQ%3D%3D"
        },
        {
            "FILENAME": "Link to Requirement - BENS_03_0007",
            "TECHN_FILENAME": "Link to Requirement - BENS_03_0007.URL",
            "EXTENSION": "URL",
            "URL":
"https://test3.signavio.com/p/portal#/glossary/028a5b8b8c65470ab65882f4bc2ac2e9"
        }]
}
```

Successful Response
HTTPS Status Code 201
Example Content:

```
{
    "@odata.context": "$metadata#Issues",
    "SYSTEM_GUID": null,
    "EXTERNAL_ID": "BFD-17",
    "PARENT_EXTERNAL_ID": null,
    "TICKET_ID": null,
    "TICKET_GUID": null,
    "DESCRIPTION": null,
    "PROCESS_TYPE": null,
    "CREATED_AT": null,
    "CREATED_BY": null,
    "CHANGED_AT": null,
    "CHANGED_BY": null,
    "STATUS": null,
    "STATUS_TXT": null,
    "SOLDOC_PROJECT": null,
    "SOLDOC_PROJECT_TXT": null,
    "PRIORITY": null,
    "PRIORITY_TXT": null,
    "IMPACT": null,
    "IMPACT_TXT": null,
    "URGENCY": null,
```

```json
  "URGENCY_TXT": null,
  "EFFORT": null,
  "EFFORT_UNIT": null,
  "PPM_PROJECT_ID": null,
  "PPM_PROJECT_TXT": null,
  "PPM_WAVE": null,
  "PPM_WAVE_TXT": null,
  "PPM_START_DATE": null,
  "PPM_FINISH_DATE": null,
  "PPM_COMPLETION": null,
  "PPM_TOTAL_WORK": null,
  "PPM_TOTAL_WORK_UNIT": null,
  "REQUESTED_RELEASE_NO": null,
  "REQUESTED_RELEASE_CLASS": null,
  "REQUESTED_RELEASE_TYPE": null,
  "REQUESTED_GO_LIVE_DATE": null,
  "FORECAST_RELEASE_NO": null,
  "FORECAST_RELEASE_CLASS": null,
  "FORECAST_RELEASE_TYPE": null,
  "FORECAST_GO_LIVE_DATE": null,
  "ACTUAL_RELEASE_NO": null,
  "ACTUAL_RELEASE_CLASS": null,
  "ACTUAL_RELEASE_TYPE": null,
  "ACTUAL_GO_LIVE_DATE": null,
  "CLASSIFICATION": null,
  "CLASSIFICATION_TXT": null
}
```

5.2   Update Status Payload JSON Example

Called URL
https://<SolMan host>:<https port>
/opu/odata/SALM/EXT_INTEGRATION_SRV/StatusSet(Guid='00505683226E1EE5B5AA9CEE1DADAC68')

Payload

```
{
    "ExtStatus": "Start Development",
    "Guid": "00505683226E1EE5B5AA9CEE1DADAC68"
}
```

Successful Response
HTTP Status Code 204, No Content

Error Response
HTTP Status Code 500
Example Content:

```
{
    "error": {
        "code": "/SALM/EXT_INTEG/003",
        "message": {
            "lang": "en",
            "value": "Status mapping for ext. status Start Development is not existing for ticket
8000013139."
        },
        "innererror": {
            "transactionid": "5779D741E1072697E10000000A4DA024",
            "timestamp": "20160704140612.0949610",
            "Error_Resolution": {
                "SAP_Transaction": "Run transaction /IWFND/ERROR_LOG on SAP NW Gateway hub system
and search for entries with the timestamp above for more details",
                "SAP_Note": "See SAP Note 1797736 for error analysis
(https://service.sap.com/sap/support/notes/1797736)"
            },
            "errordetails": [
                {
                    "code": "/SALM/EXT_INTEG/003",
                    "message": "Status mapping for ext. status Start Development is not existing for
ticket 8000013139.",
                    "propertyref": "",
                    "severity": "error",
                    "target": ""
                },
                {
                    "code": "/IWBEP/CX_SD_GEN_DPC_TECH",
                    "message": "RFC call ended with \"Communication Failure\" exception",
                    "propertyref": "",
                    "severity": "error",
                    "target": ""
                }
            ]
        }
    }
}
```

www.sap.com