**SAP Integration Suite   
Cloud Integration - Technical Specification  
 iFlow Name : EDI\_850\_TO\_IDOC\_1809\_ORDERS**

Version: 1.0

Author: Generated by AI

Date: 2025-10-27

# Table of Contents

1. Change History

2. Overview

3. High level iFlow Design

4. Message Flow

5. Technical Description

5.1. Main Integration Process

5.2. Local Integration Process

5.3. Sender

5.4. Receiver

5.5. Mappings

5.6. Security

5.7. Groovy Scripts

5.8. Error Handling & Logging

6. Version and Metadata

7. Appendix

# 1. Change History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description** |
| 1.0 | 2025-10-27 | Generated by AI | Initial version |

# 2. Overview

This technical specification document serves as a blueprint for the EDI\_850\_TO\_IDOC\_1809\_ORDERS iFlow. It details the iFlow's configuration, including properties like namespace mappings, HTTP session handling, logging behavior, and security settings. The document specifies the involved sender and receiver endpoints, along with the integration process. It defines the configuration of the IDOC sender adapter, including the transport protocol, message protocol, and authentication details. The primary use of this document is to provide comprehensive information for deployment, maintenance, troubleshooting and auditing the iFlow.

# 3. High level iFlow Design

The iFlow `EDI\_850\_TO\_IDOC\_1809\_ORDERS` processes messages from a Sender to a Receiver system. It begins with a Start Event, followed by a Content Modifier that enriches the message with a pre-defined EDI 850 document as the message body. The enriched message is then transformed from EDI to XML format using the EDI to XML Converter, utilizing the `ASC-X12\_850\_004010.xsd` schema. Subsequently, a Message Mapping step using `MM.mmap` transforms the XML structure. Finally, the transformed message reaches the End Event.

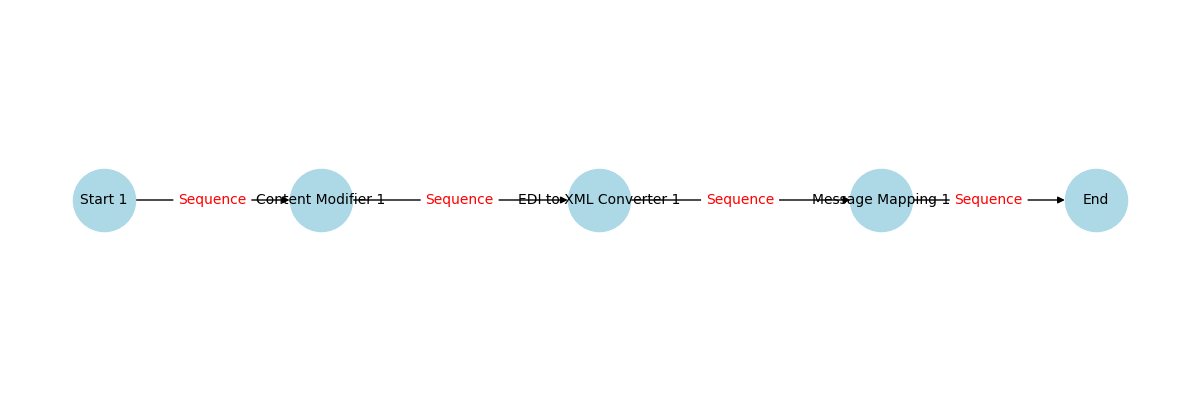


Figure: High level BPMN iFlow message and sequence flow

# 4. Message Flow

This iFlow message flow, identified as "MessageFlow\_69" with the name "IDOC," represents an IDOC sender adapter configuration. It utilizes the HTTP transport protocol and the IDoc SOAP message protocol for communication with the address "/IDOC\_SRI27". The adapter is configured to handle XML character issues by throwing exceptions. Sender authentication is role-based, requiring the "ESBMessaging.send" user role, with a maximum body size of 40 and a maximum attachment size of 100. The system is configured as a "Sender".

|  |  |  |
| --- | --- | --- |
| **Source** | **Target** | **Name** |
| Sender | Start 1 | IDOC |

# 5. Technical Description

## 5.1. Main Integration Process

The iFlow `Process\_1` named "Integration Process" starts with a `StartEvent\_66` and ends with an `EndEvent\_2`. The integration flow first executes `CallActivity\_14`, a Content Modifier of type Enricher with component version 1.6 that inserts a constant value into the message body. It then converts the EDI message to XML using `CallActivity\_17`, an EDI to XML Converter using X12 schema `ASC-X12\_850\_004010.xsd`. Finally, `CallActivity\_4` maps the message using `MM.mmap` message mapping. The iFlow has a transaction timeout of 30 seconds and transactional handling is set to "Not Required".

|  |  |  |
| --- | --- | --- |
| **Component Name** | **Key** | **Value** |
| Integration Process | Transaction Timeout | 30 |
| Integration Process | Component Version | 1.2 |
| Integration Process | Cmd Variant Uri | ctype::FlowElementVariant/cname::IntegrationProcess/version::1.2.1 |
| Integration Process | Transactional Handling | Not Required |

### endEvent End Properties

|  |  |
| --- | --- |
| **Key** | **Value** |
| Component Version | 1.1 |
| Cmd Variant Uri | ctype::FlowstepVariant/cname::MessageEndEvent/version::1.1.0 |

### callActivity Content Modifier 1 Properties

|  |  |
| --- | --- |
| **Key** | **Value** |
| Body Type | constant |
| Property Table |  |
| Header Table |  |
| Wrap Content | ISA\*00\* \*00\* \*ZZ\*0011223456 \*ZZ\*999999999 \*990320\*0157\*U\*00401\*000000015\*0\*P\*>~ GS\*PO\*0011223456\*999999999\*950120\*0147\*5\*X\*004010~ ST\*850\*000000001~ BEG\*00\*SA\*95018017\*\*\*950118~ N1\*SE\*UNIVERSAL WIDGETS~ N3\*375 PLYMOUTH PARK\*SUITE 205~ N4\*IRVING\*TX\*75061~ N1\*ST\*JIT MANUFACTURING~ N3\*BUILDING 3B\*2001 ENTERPRISE PARK~ N4\*JUAREZ\*CH\*\*MEX~ N1\*AK\*JIT MANUFACTURING~ N3\*400 INDUSTRIAL PARKWAY~ N4\*INDUSTRIAL AIRPORT\*KS\*66030~ N1\*BT\*JIT MANUFACTURING~ N2\*ACCOUNTS PAYABLE DEPARTMENT~ N3\*400 INDUSTRIAL PARKWAY~ N4\*INDUSTRIAL AIRPORT\*KS\*66030~ PO1\*001\*4\*EA\*330\*TE\*IN\*525\*VN\*X357-W2~ PID\*F\*\*\*\*HIGH PERFORMANCE WIDGET~ SCH\*4\*EA\*\*\*\*002\*950322~ CTT\*1\*1~ SE\*20\*000000001~ GE\*1\*5~ IEA\*1\*000000015~ |
| Component Version | 1.6 |
| Activity Type | Enricher |
| Cmd Variant Uri | ctype::FlowstepVariant/cname::Enricher/version::1.6.1 |

### callActivity EDI to XML Converter 1 Properties

|  |  |
| --- | --- |
| **Key** | **Value** |
| Tradacoms Source Encoding | ISO-8859-1 |
| X12 Source Encoding | ISO-8859-1 |
| Edifact Source Encoding | ISO-8859-1 |
| Tradacoms Header Name |  |
| Tradacoms Conversion Preference | No |
| Tradacoms Edi Schema Source | IntegrationProject |
| Component Version | 2.6 |
| Edifact Header Name |  |
| Edifact Envelope Truncator | true |
| Edifact Decimal Character | fromIncomingPayload |
| Edifact Target Root Element | interchange |
| X12 Edi Schema Source | IntegrationProject |
| X12 Header Name |  |
| X12 Envelope Truncator | false |
| Tradacoms Schema Table |  |
| Edifact Edi Schema Source | IntegrationProject |
| Activity Type | EDItoXMLConverter |
| Cmd Variant Uri | ctype::FlowstepVariant/cname::EDItoXMLConverter/version::2.6.0 |
| X12 Schema Table | <row><cell id='x12SchemaName'>/xsd/ASC-X12\_850\_004010.xsd</cell></row> |
| Edifact Target Encoding | ISO-8859-1 |
| Edifact Schema Table |  |

### startEvent Start 1 Properties

|  |  |
| --- | --- |
| **Key** | **Value** |
| Cmd Variant Uri | ctype::FlowstepVariant/cname::MessageStartEvent |

### callActivity Message Mapping 1 Properties

|  |  |
| --- | --- |
| **Key** | **Value** |
| Mappinguri | dir://mmap/src/main/resources/mapping/MM.mmap |
| Mappingname | MM |
| Mapping Source Value |  |
| Mapping Type | MessageMapping |
| Mapping Reference | static |
| Mappingpath | src/main/resources/mapping/MM |
| Component Version | 1.3 |
| Activity Type | Mapping |
| Cmd Variant Uri | ctype::FlowstepVariant/cname::MessageMapping/version::1.3.1 |
| Message Mapping Bundle Id |  |

### callActivity Message Mapping 2 Properties

|  |  |
| --- | --- |
| **Key** | **Value** |
| Mappinguri | dir://mmap/src/main/resources/mapping/EDI\_850.mmap |
| Mappingname | EDI\_850 |
| Mapping Source Value |  |
| Mapping Type | MessageMapping |
| Mapping Reference | static |
| Mappingpath | src/main/resources/mapping/EDI\_850 |
| Component Version | 1.3 |
| Activity Type | Mapping |
| Cmd Variant Uri | ctype::FlowstepVariant/cname::MessageMapping/version::1.3.1 |
| Message Mapping Bundle Id |  |

## 5.2. Local Integration Process

The SAP iFlow "Process\_1" defines a main integration process that executes sequentially. This process starts with a content modifier named "ContentModifier\_Start" likely used for initial configuration. Next, an "IntegrationProcessCall" step invokes a local integration process identified as "Process\_2". Finally, another content modifier, "ContentModifier\_End", is executed, presumably for final data manipulation or logging. The entire sequence ensures that the configuration is applied before the nested local process executes, and finishing touches occur at the end.  
  
<!-- Summarizing XML snippets is challenging. The description captures what's logically gleaned from the naming convention of the process steps. A better summary would require knowing the details of "Process\_2" and the specifics actions performed by content modifiers. Without more context I am making assumptions about their general purpose. -->

No process with id='Process\_1' found.

## 5.3. Sender

This sender component within the SAP iFlow utilizes the IDoc adapter (version 1.4) over HTTP protocol (version 1.8.1) to receive IDoc messages via SOAP (version 1.8.1) at address `/IDOC\_SRI27`. Authentication is role-based, requiring the `ESBMessaging.send` user role. The system identified as "Sender" acts as an external system sending IDoc data. Maximum body and attachment size are configured.

|  |  |
| --- | --- |
| **Key** | **Value** |
| Component Type | IDOC |
| Description |  |
| Address | /IDOC\_SRI27 |
| Maximum Body Size | 40 |
| Component N S | sap |
| Maximum Attachment Size | 100 |
| Component Version | 1.4 |
| Name | IDOC |
| Xml Character Handling | throwException |
| Transport Protocol Version | 1.8.1 |
| Component S W C V Name | external |
| System | Sender |
| Transport Protocol | HTTP |
| Cmd Variant Uri | ctype::AdapterVariant/cname::sap:IDOC/tp::HTTP/mp::IDoc SOAP/direction::Sender/version::1.4.4 |
| User Role | ESBMessaging.send |
| Sender Auth Type | RoleBased |
| Message Protocol | IDoc SOAP |
| Message Protocol Version | 1.8.1 |
| Component S W C V Id | 1.8.1 |
| Direction | Sender |
| Client Certificates |  |

## 5.4. Receiver

The Receiver section in this SAP iFlow XML (represented by `<ReceiverProperties>`) currently indicates a default or undefined receiver configuration. There are no specified receiver components within the XML. Consequently, no receiver-specific adapters, communication channels, or target systems are explicitly defined. This suggests the iFlow may be incomplete or configured to dynamically determine the receiver at runtime based on message content. Further configuration outside this XML snippet is likely required to establish concrete routing and communication with a recipient system.

## 5.5. Mappings

The SAP iFlow contains two mapping activities. The first uses a Message Mapping named "MM", located at `dir://mmap/src/main/resources/mapping/MM.mmap`. The second uses a Message Mapping named "EDI\_850", located at `dir://mmap/src/main/resources/mapping/EDI\_850.mmap`. Both mappings are referenced statically and are of type "MessageMapping". They both appear to be standard message mappings with a component version of 1.3 and flowstep variant version 1.3.1. No specific data transformation logic is defined within the XML; the mapping details reside in the referenced `.mmap` files.

### Mapping Activity 1 Properties

|  |  |
| --- | --- |
| **Key** | **Value** |
| mappinguri | dir://mmap/src/main/resources/mapping/MM.mmap |
| mappingname | MM |
| mappingSourceValue |  |
| mappingType | MessageMapping |
| mappingReference | static |
| mappingpath | src/main/resources/mapping/MM |
| componentVersion | 1.3 |
| activityType | Mapping |
| cmdVariantUri | ctype::FlowstepVariant/cname::MessageMapping/version::1.3.1 |
| messageMappingBundleId |  |

### Mapping Activity 2 Properties

|  |  |
| --- | --- |
| **Key** | **Value** |
| mappinguri | dir://mmap/src/main/resources/mapping/EDI\_850.mmap |
| mappingname | EDI\_850 |
| mappingSourceValue |  |
| mappingType | MessageMapping |
| mappingReference | static |
| mappingpath | src/main/resources/mapping/EDI\_850 |
| componentVersion | 1.3 |
| activityType | Mapping |
| cmdVariantUri | ctype::FlowstepVariant/cname::MessageMapping/version::1.3.1 |
| messageMappingBundleId |  |

## 5.6. Security

The iFlow's security configuration, defined within the `Collaboration\_1` element, disables HTTP session handling and CORS. Exception handling is set not to return exceptions to the sender. The sender authentication type for the `IDOC` message flow is role-based, requiring the `ESBMessaging.send` role. Basic authentication is disabled on the sender endpoint. Server tracing and `accessControlAllowCredentials` are disabled. Logging is configured to capture all events.

|  |  |
| --- | --- |
| **Key** | **Value** |
| Namespace Mapping |  |
| Http Session Handling | None |
| Access Control Max Age |  |
| Return Exception To Sender | false |
| Log | All events |
| Cors Enabled | false |
| Exposed Headers |  |
| Component Version | 1.2 |
| Allowed Header List |  |
| Server Trace | false |
| Allowed Origins |  |
| Access Control Allow Credentials | false |
| Allowed Headers |  |
| Allowed Methods |  |
| Cmd Variant Uri | ctype::IFlowVariant/cname::IFlowConfiguration/version::1.2.4 |

## 5.7. Groovy Scripts

This SAP iFlow named "Integration Process" (EDI\_850\_TO\_IDOC\_1809\_ORDERS) processes EDI 850 purchase orders and converts them to IDOC format. The flow starts at "Start 1" and proceeds to "Content Modifier 1," which enriches the message with hardcoded EDI content. "EDI to XML Converter 1" then transforms the EDI data into XML format based on the ASC-X12\_850\_004010 schema. Finally, "Message Mapping 1" applies a message mapping (`MM.mmap`) to generate the final IDOC structure before ending at "End".

No Groovy scripts found in the specified folder.

## 5.8. Error Handling & Logging

The SAP iFlow configuration lacks explicit exception handling logic within the `<Exceptions>` element, implying a reliance on default platform behavior. No custom error processes are defined. Similarly, the configuration does not showcase any logging activities or user-defined message persistence. Consequently, default SAP logging (if enabled at platform level) will capture basic iFlow execution and errors. Deeper insight into specific processing stages or data transformations will not be readily available without customized error handling and logging implementations.

No exception subprocesses found in the iFlow.

# 6. Version and Metadata

|  |  |
| --- | --- |
| **Key** | **Value** |
| componentVersion | 1.3 |
| ComponentNS | sap |
| ComponentSWCVName | external |
| ComponentSWCVId | 1.8.1 |

The SAP iFlow's component version is 1.3, operating under the `sap` namespace. It's associated with a Software Component Version (SWCV) named `external`. This SWCV is identified by the ID `1.8.1`. The iFlow leverages functionality provided by this external component, which has a version of 1.8.1. Essentially, the iFlow metadata specifies the versioning and source of the components it utilizes.

# 7. Appendix

This iFlow processes a message through a series of steps. It starts with a `StartEvent\_66`, followed by a `Content Modifier 1` (Enricher activity) which enriches the message with a constant value. The enriched message is then converted from EDI to XML using `EDI to XML Converter 1`, configured for X12 schema `/xsd/ASC-X12\_850\_004010.xsd`. The XML message is then transformed by `Message Mapping 1`, which utilizes the message mapping `MM.mmap`. Finally, the transformed message reaches the `EndEvent\_2`. This iFlow also defines another `Message Mapping 2` which utilizes the mapping `EDI\_850.mmap`, although it is not actively used in the iFlow.

|  |  |
| --- | --- |
| **Key** | **Value** |
| mappinguri | dir://mmap/src/main/resources/mapping/MM.mmap |
| mappingname | MM |
| mappingType | MessageMapping |
| mappingReference | static |
| mappingpath | src/main/resources/mapping/MM |
| mappinguri | dir://mmap/src/main/resources/mapping/EDI\_850.mmap |
| mappingname | EDI\_850 |
| mappingType | MessageMapping |
| mappingReference | static |
| mappingpath | src/main/resources/mapping/EDI\_850 |