

How to configure integration of a Z IDOC with SAP MEINT



Applies to:

This article is relevant for SAP ME 5.2.5 and higher releases.

Summary

This how to document explains what configuration changes need to be done when customer would like to send a Z IDOC instead of standard IDOC for an existing configuration delivered by SAP ME.

Author(s): Rajeev Kansal

Company: SAP

Created on: 25th Dec, 2010

Author Bio

Rajeev Kansal has worked for SAP since 1999 and on different project and technologies e.g. SAP Production Revenue & Accounting (Upstream Oil & Gas Solution), Composite development (xIEP) using SAP Netweaver components, SAP Global Data Synchronization using SAP Netweaver and MDM. Rajeev is currently working as development architect in SAP Labs India Pvt Ltd for a product called SAP Manufacturing Execution System.

Table of Contents

Prerequisites	3
Scenario Explanation	3
Procedure	3
Create a new message processing Rule	4
Create new workflow configurations	5
Conclusion	8
Copyright.....	9

Prerequisites

You should have installed SAP ME 5.2.5 along with all dependent components. For installation you can refer to:

<http://service.sap.com/instguides> -> SAP Business Suite Applications -> SAP Manufacturing -> SAP Manufacturing Execution 5.2 -> Installation Guide SAP ME 5.2.

You should have done all development and configuration in ERP system for sending the custom IDOC along with additional field.

Scenario Explanation

SAP ME has delivered standard integration scenarios and their respective integration configuration done in SAP MEINT using SAPMEINT CTC. Now customer would like to integrate his custom Z IDOC instead of standard IDOC name. Below documentation will show an example where in customer is trying to integrate custom MATMAS03 instead of MATMAS03 which has an additional field ZCUST_MATMAS03/IDOC/E1MARAM/E1MARCM/ZTEST) which needs to be mapped to custom data in ME Item as CUSTOM_FIELD. Here is a small screen snapshot of where that field lies in custom IDOC.

```

<?xml version="1.0" encoding="UTF-8"?>
<ZCUST MATMAS03>
  <IDOC BEGIN="1">
    <EDI_DC40 SEGMENT="1">
      <E1MARAM SEGMENT="1">
        <MSGFN>005</MSGFN>
        <MATNR>000000000000200509</MATNR>
        <ERSDA>19971120</ERSDA>
        <ERNAM>PCMN00</ERNAM>
        <LAEDA>20071101</LAEDA>
        <WRKST_NEW>200509</WRKST_NEW>
      <E1MAKTM SEGMENT="1">
        <E1MAKTM SEGMENT="1">
          <E1MAKTM SEGMENT="1">
            <E1MARCM SEGMENT="1">
              <ZTEST>Test Value</ZTEST/>
            <MSGFN>005</MSGFN>
            <WERKS>AT20</WERKS>
            <PSTAT>VEDLBG</PSTAT>
            <LVORM>X</LVORM>
          </E1MAKTM SEGMENT="1">
        </E1MAKTM SEGMENT="1">
      </E1MAKTM SEGMENT="1">
    </E1MARAM SEGMENT="1">
  </EDI_DC40 SEGMENT="1">
</IDOC BEGIN="1">
</ZCUST MATMAS03>

```

Procedure

Before explaining the configuration changes required for above scenario, here is the brief overview of the possible workflows which can happen for any message type.

- Standard Workflow – Standard workflow is built to send the messages received to other systems. Example: MATMAS03 message from SAP to SAPME or Confirmation Message from SAPME to SAP. This is the only workflow which can make a call to other system.
- Split Workflow – This workflow is built to split the message into multiple messages before it is processed by Standard workflow.
- Correlation Workflow – This workflow is built to merge multiple messages into a single message before it is processed by Standard workflow.

For the above scenario description, Split workflow is being used and to achieve the mentioned scenario, customer has to do following changes.

Create a new message processing Rule

If customer is integrating a new custom *message type* instead of standard *message type* then he has to copy the standard processing rule and create a new one for the new custom message type. If only the IDOC type is being changed then this configuration change is not required. Check following steps to make this change.

1. Go to MII Menu -> Message Services -> Processing Rule Editor, select the standard message type processing rule and choose Copy.

Message Processing Rules

Name

- XMIIIDOC01_ZDOC01
- XMIIIDOC01_BOMMAT03
- XMIIIDOC01_CLFMAS02
- XMIIIDOC01_HRMD_A06
- XMIIIDOC01_INVCON02
- XMIIIDOC01_IORDER01
- XMIIIDOC01_LOIPLO01
- XMIIIDOC01_LOIPRO02
- XMIIIDOC01_LOIROU02
- XMIIIDOC01_LOIWCS02
- XMIIIDOC01_MATMAS03**

Rule Name * XMIIIDOC01_MATMAS03

Rule Description XMIIIDOC01_MATMAS03

Server Name * XMIIIDOC01

Message Name * MATMAS

Message Type Web Service IDoc RFC

Processing Type * Transaction Category

Category Add Delete

Category Description

Transaction Visiprise/ERPShopFloorIntegration/services/SyncDOCSer ...

Persist Transaction ALWAYS

Log Level NONE

Parameters

Name	Value
DOC	<input checked="" type="checkbox"/> ReceivedMessageXML

Save New Copy Delete Help

2. Enter the new Rule name, description, Message name and choose Save. You can leave rest of the information as it is.

Message Processing Rules

Name
XMIIDOC01_ZDOC01
XMIIDOC01_BOMMAT03
XMIIDOC01_CLFMAS02
XMIIDOC01_HRMD_A06
XMIIDOC01_INVCON02
XMIIDOC01_IORDER01
XMIIDOC01_LOIPLO01
XMIIDOC01_LOIPRO02
XMIIDOC01_LOIROU02
XMIIDOC01_LOIWCS02
XMIIDOC01_MATMAS03

Rule Name *
 Rule Description
 Server Name *
 Message Name *
 Message Type Web Service IDoc RFC
 Processing Type * Transaction Category
 Category
 Category Description
 Transaction
 Persist Transaction
 Log Level
 Parameters

Name	Value
DOC	<input checked="" type="checkbox"/> ReceivedMessageXML

Create new workflow configurations

Since the IDOC type is different than the standard IDOC type, customer need to copy the existing workflow configuration to Z IDOC type ID.

1. Go to SAP MEINT Configuration -> Workflow Configuration and choose the standard Type ID(IDOC type in case of IDOC messages). For our scenario, it is going to be

MATMAS03.

SAPMEINT Workflow Configuration

Type List

Create Delete

Type ID	Handler Transaction
LOIPL001	Visiprise/ERPShopFloorIntegration/frame/workflow/StdWorkflow/StdWorkflowHandler
Z_CLFMAS02_SAVE_ITEM_CUST_FLDS	Visiprise/ERPShopFloorIntegration/frame/workflow/StdWorkflow/StdWorkflowHandler
MATMAS03	Visiprise/ERPShopFloorIntegration/frame/workflow/SplitWorkflow/SplitWorkflowHandler
HRMD_A06	Visiprise/ERPShopFloorIntegration/frame/workflow/SplitWorkflow/SplitWorkflowHandler
yieldBackflushingRequest	Visiprise/ERPShopFloorIntegration/frame/workflow/StdWorkflow/StdWorkflowHandler
rmaOrderCompleteRequest	Visiprise/ERPShopFloorIntegration/frame/workflow/StdWorkflow/StdWorkflowHandler
LOIWCS02_ME_RES	Visiprise/ERPShopFloorIntegration/frame/workflow/StdWorkflow/StdWorkflowHandler
LOIROU02	Visiprise/ERPShopFloorIntegration/frame/workflow/StdWorkflow/StdWorkflowHandler
equipmentStatusChangeRequest_PROD	Visiprise/ERPShopFloorIntegration/frame/workflow/StdWorkflow/StdWorkflowHandler
scrapBackflushingRequest	Visiprise/ERPShopFloorIntegration/frame/workflow/StdWorkflow/StdWorkflowHandler

Type Details

Modify Copy As

Type ID: MATMAS03

Handler Transaction: Visiprise/ERPShopFloorIntegration/frame/workflow/SplitWorkflow/SplitWorkflowHandler

Handler Transaction Parameters: identifier=/MATMAS03/IDOC/E1MARAM/MATNR;

Retry Limit: 3

- Enter the new type ID and change the handler transaction parameters xpath value to refer to the new Z IDOC type name and save the configuration.

Type Details

Save Cancel

Type ID: * ZCUST_MATMAS03

Handler Transaction: Visiprise/ERPShopFloorIntegration/frame/workflow/SplitWorkflow/SplitWorkflowHandler

Handler Transaction Parameters: identifier=/ZCUST_MATMAS03/IDOC/E1MARAM/MATNR;

Retry Limit: 3

Pre XSLT Transaction: Visiprise/ERPShopFloorIntegration/frame/workflow/EnrichMessageWithSuppPlantData

Pre XSLT Transaction Parameters:

Partner Pre XSLT Transaction:

Partner Pre XSLT Transaction Parameters:

Customer Pre XSLT Transaction:

Customer Pre XSLT Transaction Parameters:

Split XSLT Address: WEB://Visiprise/XSLT/Inbound/Material/MaterialSplit.xslt

Partner Split XSLT Address:

Customer Split XSLT Address:

Pass Handler Transaction:

Pass Handler Transaction Parameters:

Partner Pass Handler Transaction:

Partner Pass Handler Transaction Parameters:

Customer Pass Handler Transaction:

Customer Pass Handler Transaction Parameters:

- As MATMAS03 is of type Split workflow (This workflow splits the incoming messages into multiple messages based on the XSLT provided in the configuration. Handler transaction name is SplitWorkFlowHandler), customer has to do following.
 - Customer need to copy the standard split XSLT file used in workflow definition for MATMAS03 which is shown as below

The screenshot shows the SAP Web Editor interface. On the left, a tree view displays the file structure under 'MaterialSplit.xslt'. The main window shows the XSLT code, with the root node `<Z_MATMAS03>` circled in red. Below the XSLT code, a 'Properties' table is visible.

Refresh	
Name	Value
Name	MaterialSplit.xslt
Full Name	Visiprise/WEB/XSLT/Inbound/Mate...
File Type	Content
Created	2010-11-26T14:15:16
Created By	Administrator
Modified	

This XSLT also derive what would be the root node for the split message i.e. Z_MATMAS03 in above screen snapshot. Hence In the copied XSLT, change that to something like Z_ZCUST_MATMAS03 since workflow configuration done for type Z_MATMAS can't be used as customer want to send an additional field into ME Item custom data. Now provide the new split XSLT file reference in *Customer split XSLT Address* of new Split workflow configuration created as part of step 2.

- b. Copy the workflow configuration for standard split message type ID (Z_MATMAS03 for MATMAS03 integration) to the root node provided in split XSLT for the split messages i.e. Z_ZCUST_MATMAS03. This step would be similar to step 2.
- c. For the newly created workflow configuration for split message i.e. for type ID Z_ZCUST_MATMAS03, customer need to copy the standard request XSLT file used in standard split message(i.e.. Z_MATMAS03 for MATMAS03 integration) which is shown as below.

```
<?xml version="1.0" ?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:sch=
<xsl:template match="/">
  <xsl:apply-templates select="/Z_MATMAS03/IDOC/E1MARAM/E1MARCM/">
</xsl:template>
  <xsl:template match="/Z_MATMAS03/IDOC/E1MARAM/E1MARCM">
    <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
      <soapenv:Header/>
      <soapenv:Body>
        <sch:ItemUpdateRequest_sync>
          <sch:Item>
            <sch:SiteRef>
              <sch:Site>
                <xsl:value-of select="WERKS"/>
              </sch:Site>
            </sch:SiteRef>
            <sch:Item>
              <xsl:call-template name="addItem">
                <xsl:with-param name="item" select="..MATN">
              </xsl:call-template>
            </sch:Item>
          <xsl:if test="(string(..REVLV))">
            <sch:Revision>
              <xsl:value-of select="..REVLV"/>
            </sch:Revision>
          </xsl:if>
        </sch:ItemUpdateRequest_sync>
      </soapenv:Body>
    </soapenv:Envelope>
  </xsl:template>
</xsl:stylesheet>
```

- d. In the copied XSLT, change Z_MATMAS03 to Z_CUST_MATMAS03 since the custom split message type is Z_CUST_MATMAS03. Also If customer has any custom fields which needs to be integrated back to ME system, then those fields need to be mapped to ME structure in this XSLT. As an example, field ZCUST_MATMAS03/IDOC/E1MARAM/E1MARCM/ZTEST needs to be mapped to custom data in ME Item as CUSTOM_FIELD. Below screen snapshot shows the additional line which you need to add in CustomFieldList section of copied XSLT.

```
<sch:CustomFieldList>
  <sch:CustomField>
    <sch:Attribute>STORAGE_LOCATION</sch:Attribute>
    <sch:Value><xsl:value-of select="LGPRO"/></sch:Value>
  </sch:CustomField>
  <sch:CustomField>
    <sch:Attribute>CUSTOM_FIELD</sch:Attribute>
    <sch:Value><xsl:value-of select="ZTEST"/></sch:Value>
  </sch:CustomField>
```

Now provide the new request XSLT file reference in *Customer request XSLT Address*.

In case customer wants to send a Z IDOC for which the standard configuration follows the standard workflow instead of split workflow, then customer need to copy the standard workflow configuration to their Z IDOC type and follow the step from 3.c onward.

Conclusion

By following the above mentioned steps, customer can easily integrate a custom IDOC instead of standard IDOC.

Copyright

© Copyright 2010 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, System i, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, iSeries, pSeries, xSeries, zSeries, eServer, z/VM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects S.A. in the United States and in other countries. Business Objects is an SAP company.

All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.