

IDoc based adapterless communication between SAP NetWeaver Application Server (SAP NetWeaver AS) and Microsoft BizTalk Server

Authors

Jürgen Daiberl, SAP Program Manager CTSC, Global SAP Alliance, Microsoft Corporation

Tilo Böttcher, SAP Program Manager CTSC, Global SAP Alliance, Microsoft Corporation

Summary

This white paper demonstrates how to integrate Microsoft BizTalk Server and SAP NetWeaver Application Server without using additional adapters. Microsoft BizTalk Server uses the HTTP IDoc XML Interface SAP NetWeaver Application Server offers for sending and retrieving IDocs in XML format. The paper contains a walk-through for setting up the SAP NetWeaver Application Server to enable them to post and retrieve IDocs in XML.

Applies to

- Microsoft BizTalk Server 2004 SP1 or Microsoft BizTalk Server 2006
- SAP® Web Application Server Release 6.20 and 6.30
- SAP® NetWeaver Application Server

Keywords

Microsoft BizTalk Server, SAP NetWeaver Application Server, SAP NetWeaver AS, ALE, IDoc

Audience

Technical consultants, Architects, Developers, IT Managers

Contact

This document is provided to you by the Collaboration Technology Support Center Microsoft, a joint team from SAP and Microsoft that drives interoperability. For feedback or questions you can contact the CTSC at ctsc@sap.com or ctsc@microsoft.com. Please check the .NET interoperability area in the SAP Developer Network (<http://sdn.sap.com>) for any updates or further information.

This document is a common publication by SAP and Microsoft ("Co-Editors") who have both contributed to its content and retain respective rights therein.

The information contained in this document represents the current view of the Co-Editors on the issues discussed as of the date of publication. Because the Co-Editors must respond to changing market conditions, it should not be interpreted to be a commitment on the part of the Co-Editors, and the Co-Editors cannot guarantee the accuracy of any information presented after the date of publication.

This document is for informational purposes only. NEITHER OF THE CO-EDITORS MAKES ANY WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of the Co-Editors.

Either Co-Editor may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from the respective Co-Editor(s), the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, any example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, email address, logo, person, place or event is intended or should be inferred.

2005 Microsoft Corporation. All rights reserved.

2005 SAP AG. All rights reserved. Microsoft, Windows, Outlook, and PowerPoint and other Microsoft products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Microsoft Corporation.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, 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 in several other countries all over the world. 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.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Contents

Authors	1
Summary	1
Applies to	1
Keywords	1
Audience	1
Contact	2
Contents	3
Introduction	4
Architecture	5
The Step-By-Step Solution	6
Configuration	6
BizTalk Server Configuration	6
SAP NetWeaver AS	6
Developing the BizTalk Server Solution for sending IDocs to SAP	13
BizTalk Server Solution for retrieving IDocs from SAP	19
Conclusion	20
Limitations	20
References	21
Code Snippets	22
ASP Page for receiving IDocs	22
IDoc Sample	23
Table of Figures	24

Introduction

In the scenario described in this document Microsoft BizTalk Server 2004/2006 connects to SAP NetWeaver via HTTP for sending and receiving IDocs (Standard SAP format for electronic data interchange between systems (Intermediate Document). in XML format.

The ALE (Application Link Enabling) Distribution Model for retrieving and processing IDocs can then be used in SAP in order to process the incoming messages or in order to send the messages out to BizTalk Server. For more information about the ALE Distribution Model see the SAP internal information on <http://sdn.sap.com> or <http://help.sap.com> (e.g. for ECC 5.0 you can find a detailed information at http://help.sap.com/saphelp_erp2005/helpdata/en/0b/2a60bb507d11d18ee90000e8366fc2/frameset.htm).

The walkthrough describes how to set up the SAP NetWeaver Application Server (SAP NetWeaver AS) in order to send and retrieve IDocs via HTTP. In addition you can read how to get the necessary document schemas for BizTalk Server and what has to be configured on the BizTalk Server side for sending and receiving IDocs from the SAP NetWeaver AS.

For the walkthrough for sending an IDoc from BizTalk Server to the SAP system the basic IDoc FLCUSTOMER_CREATEFROMDATA01 will be used. This IDoc creates a new customer in the flight data application. The flight data application was created by SAP to demonstrate the SAP integration technologies and is intended for use in trainings and demos. For more information about the Flight model see <http://help.sap.com>.

For the walkthrough for receiving IDocs in BizTalk Server via HTTP from a SAP system we also use the IDoc FLCUSTOMER_CREATEFROMDATA01. For both scenarios we assume that a travel agency sends the customer information to an airline before the flight booking takes place. The next figure illustrates a possible business scenario.



Figure 1 Overview Business Scenario

The next screenshot shows information about the function module BAPI_FLCUST_CREATEFROMDATA in the BAPI Explorer. Here you can also get more information about the input and output parameter (see Documentation Tab) and you can call the function module in order to test the input parameter you want to use later for sending the IDoc to the SAP system.

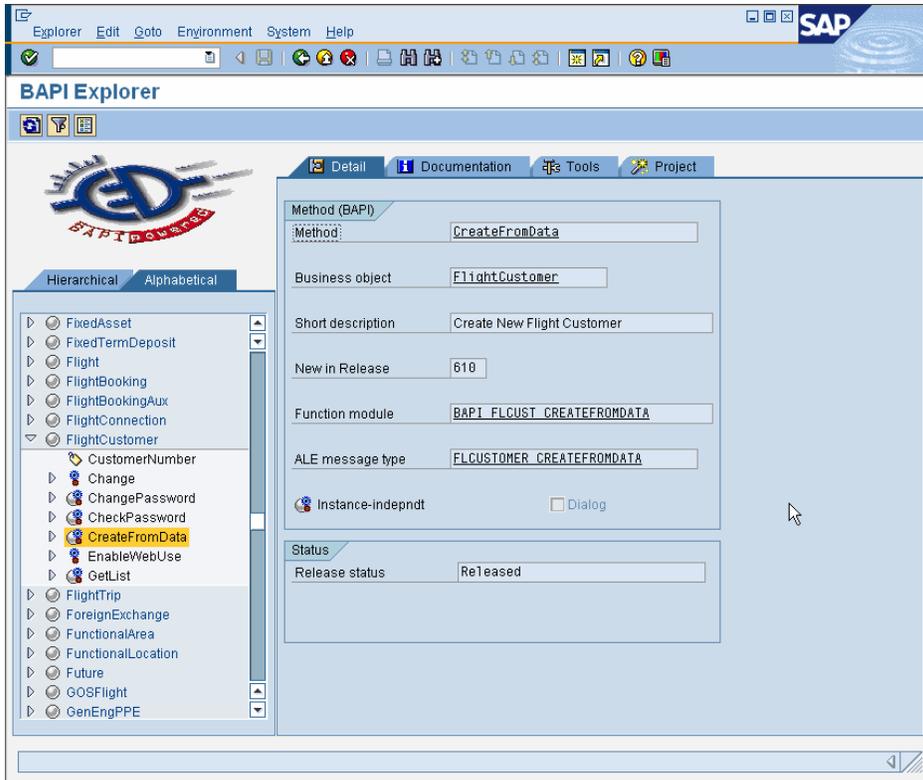


Figure 2 Create Flight Customer BAPI in transaction BAPI

Architecture

SAP Web AS 6.20 added the capability to exchange IDocs with external Systems over HTTP. Since SAP NetWeaver AS 6.40 the necessary services are pre-configured while older SAP Web AS versions require additional configuration steps,.

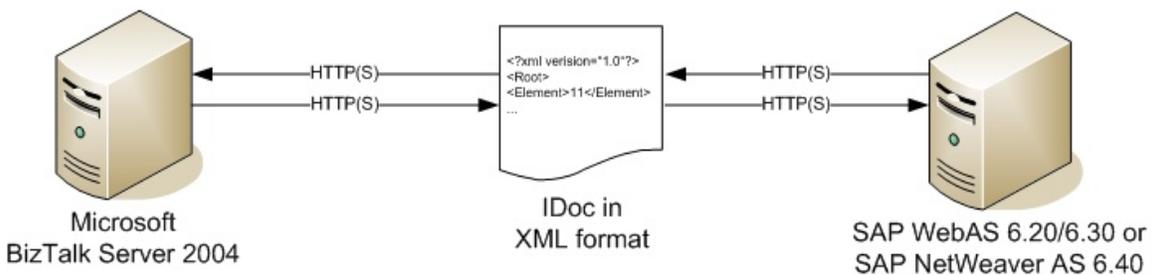


Figure 3 Scenario Overview

SAP NetWeaver AS and Microsoft BizTalk Server also allow the use of HTTP(S) for secure communication between the systems. This document will not cover the HTTP(S) scenario, for detailed information of how to setup a HTTP(S) communication in SAP NetWeaver AS and BizTalk Server please refer to the product documentation from SAP and Microsoft.

The Step-By-Step Solution

This chapter describes how to set up Microsoft BizTalk Server 2004/2006 and SAP NetWeaver AS 6.40 to exchange IDocs via HTTP without any adapter.

After setting up both systems like described in this paper you are able to send and receive IDocs from BizTalk Server to SAP NetWeaver Application Server.

The described solution also works for SAP Web AS 6.20 and SAP Web AS 6.30 but in this case additional configuration steps (creating the service in SICF) are necessary. As the modules for retrieving and sending IDocs in XML format over HTTP were implemented in SAP Web AS since version 6.20 these need additional configuration to function. A short description of what to do in these versions can be found in the chapter [SAP NetWeaver AS](#).

Configuration

BizTalk Server Configuration

Before receiving IDocs from the SAP system via HTTP it is necessary to configure the HTTP adapter of BizTalk Server. The HTTP adapter consists of two ports - a receive and a send port. The HTTP receive port is a Microsoft® Internet Information Services (IIS) Internet Server Application Programming Interface (ISAPI) extension that the IIS process hosts, and controls the receive locations that use the HTTP port.

A detailed description of how to set up the adapter can be found on <http://msdn.microsoft.com> (see http://msdn.microsoft.com/library/default.asp?url=/library/en-us/operations/htm/ebiz_ops_adapt_file_wkmm.asp).

SAP NetWeaver AS

As mentioned above the Service for retrieving IDocs in XML format is already preconfigured in SAP NetWeaver AS 6.40. In most cases the service is not started. For starting the service in SAP NetWeaver AS 6.40 use transaction SICF. The service can be found in SAP – BC – IDOC_XML.

SAP Web AS 6.20 and 6.30 only:

Note: If SAP Web AS 6.20 or 6.30 is used it is necessary to create a new service for retrieving IDocs via HTTP. After creating a new service the handler CL_HTTP_IDOC_REQUEST must be added manually to the handler list. This handler is responsible for retrieving the IDocs via HTTP and forwarding the incoming IDocs to the ALE distribution.

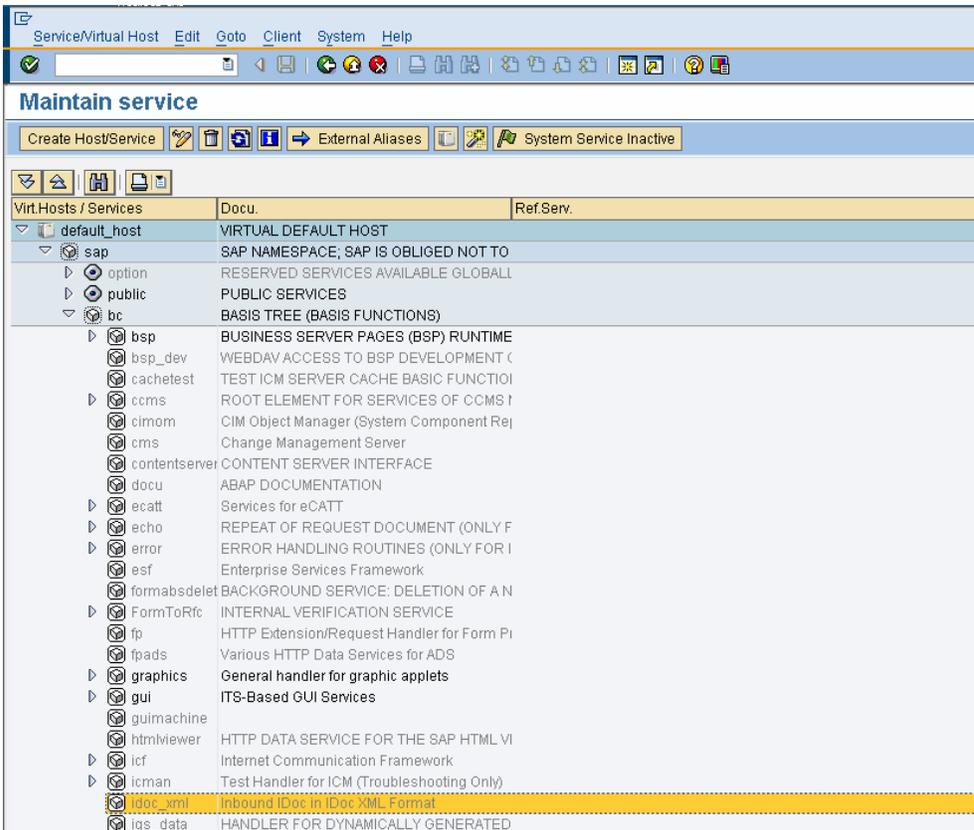


Figure 4 Transaction SICF

Double-click the IDOC_XML node and check the standard settings.

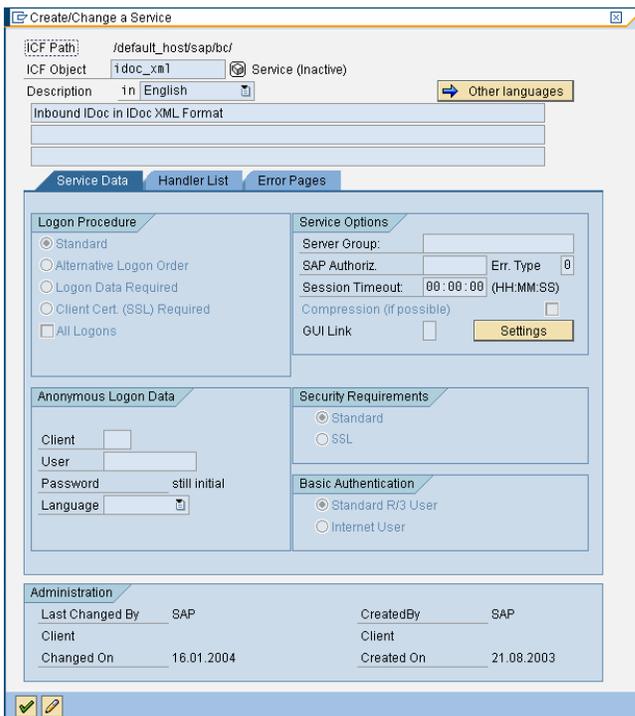


Figure 5 Configuration Screen IDOC_XML

On the tab Handler List you will find the handler assigned to the service, in our case the handler CL_HTTP_IDOC_REQUEST is defined.

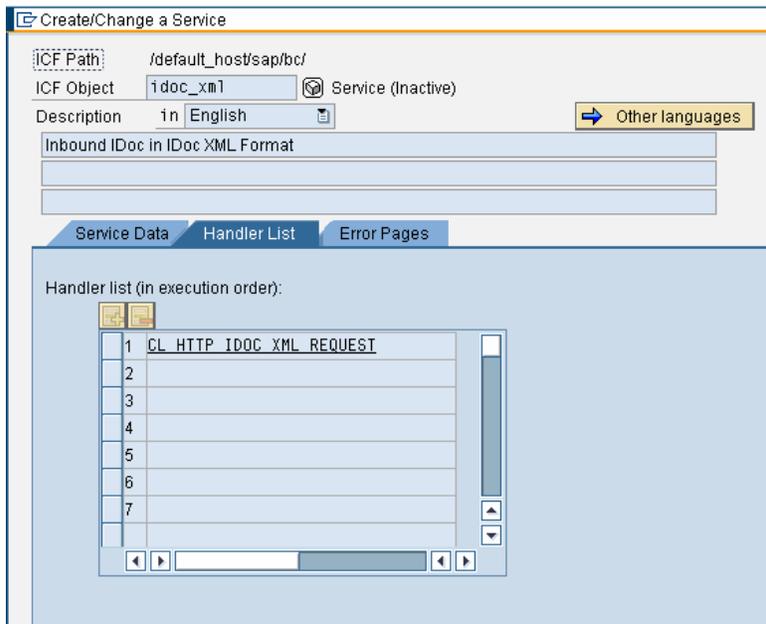


Figure 6 Handler List

After validating and customizing the settings of the service idoc_xml close the configuration window and start the service. For starting a service in SICF right-click the node and choose 'Activate Service'.

Important: Every time you change the settings of the service you will have to restart the service after the changes are made.

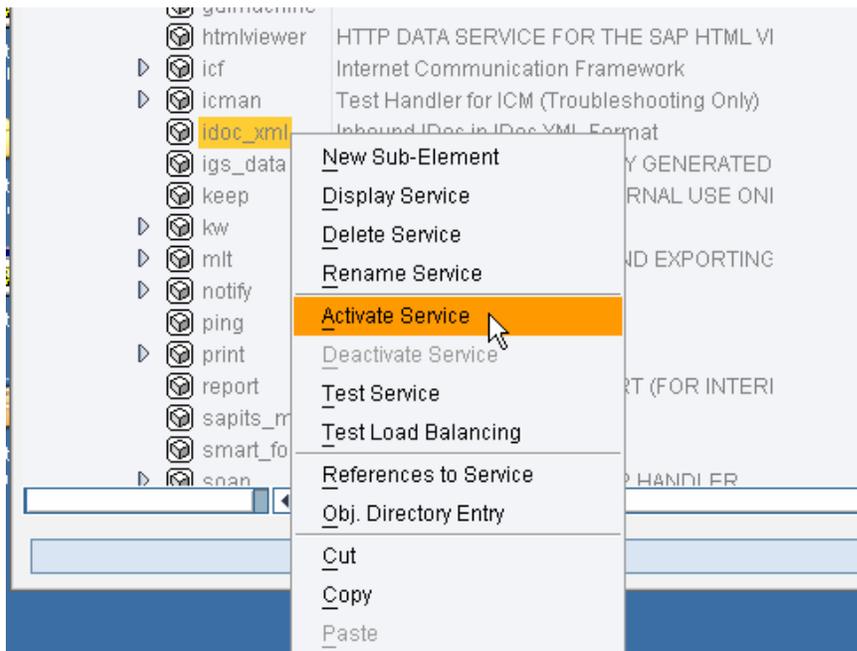


Figure 7 Activate Service IDoc_XML

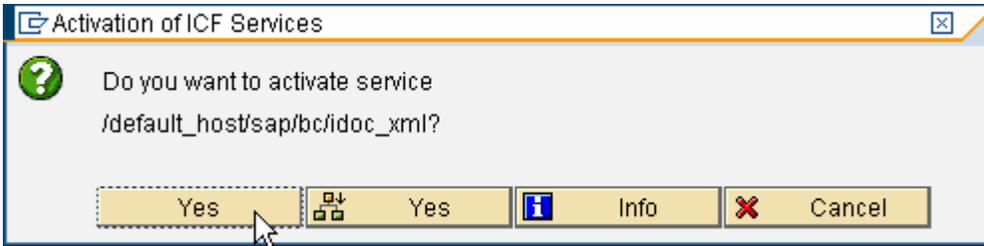


Figure 8 Confirm activation of service IDoc_XML

After activating the service you can post IDocs in XML format to the URL [http://\[Name of your machine\]:\[Port Number\]/sap/bc/idoc_xml](http://[Name of your machine]:[Port Number]/sap/bc/idoc_xml), an example is http://localhost:8000/sap/bc/idoc_xml.

For retrieving IDocs in SAP, a distribution scenario must be configured including the port definitions and partner profiles. For the basic settings we will describe the set up of the necessary ports and logical systems. The set up of the partner profiles won't be described in this scenario because they depend on whether the SAP system receives or sends IDocs.

First set up a logical system from which we want to receive the IDocs. Use transaction SALE and choose *Define Logical System*.

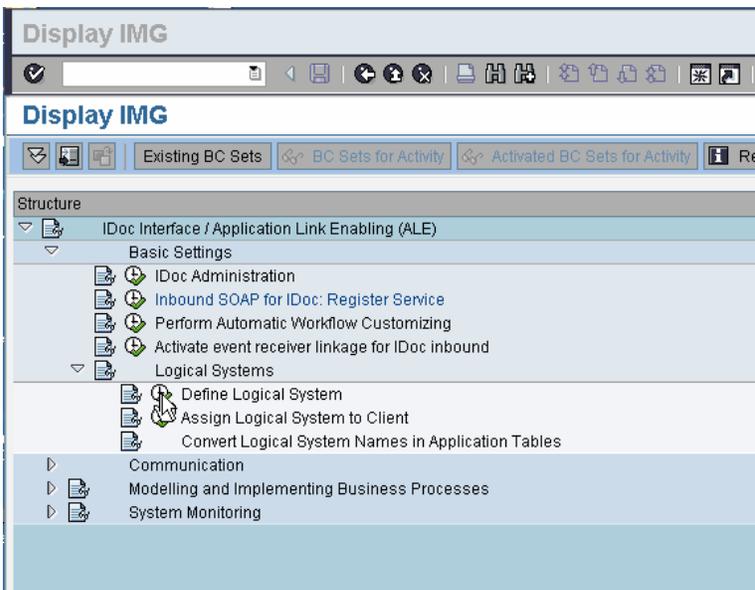


Figure 9 Define Logical System

For our scenario we will define a logical System BTS with the name BizTalk Server.

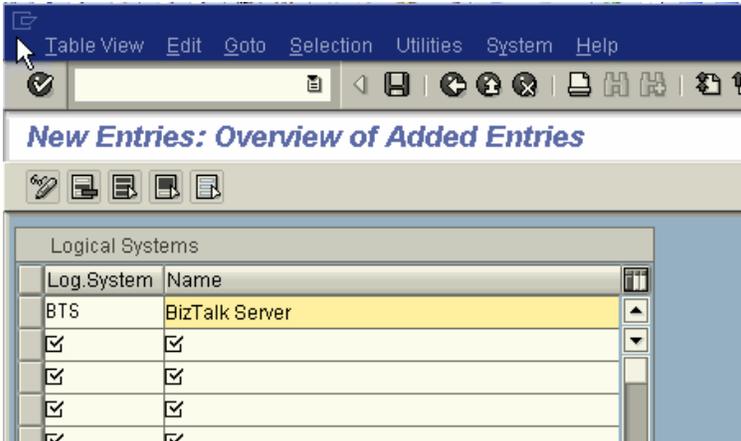


Figure 10 New Logical System

Afterwards we create a new RFC Destination in transaction SM59; we will use the settings shown in Figure 9. After entering the name for the RFC Destination, the Connection Type, and a Description press SAVE to persist and activate the settings for the HTTP Connection.

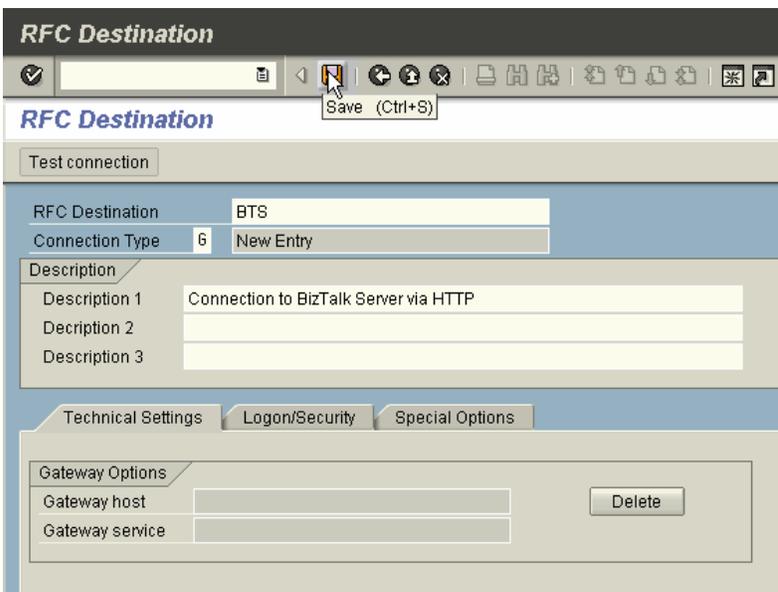


Figure 11 Setup RFC Destination

Now we can enter the settings for the HTTP connection as shown in Figure 10. Use the name of your server which hosts the BizTalk Server HTTP Receive DLL (BTSHTTPReceive.dll, see caption [BizTalk Server Configuration](#)) as the target host name, enter the name of your virtual directory to the BTSHTTPReceive.dll as Path Prefix and enter the port of your IIS which you use for incoming HTTP requests as Service Number.

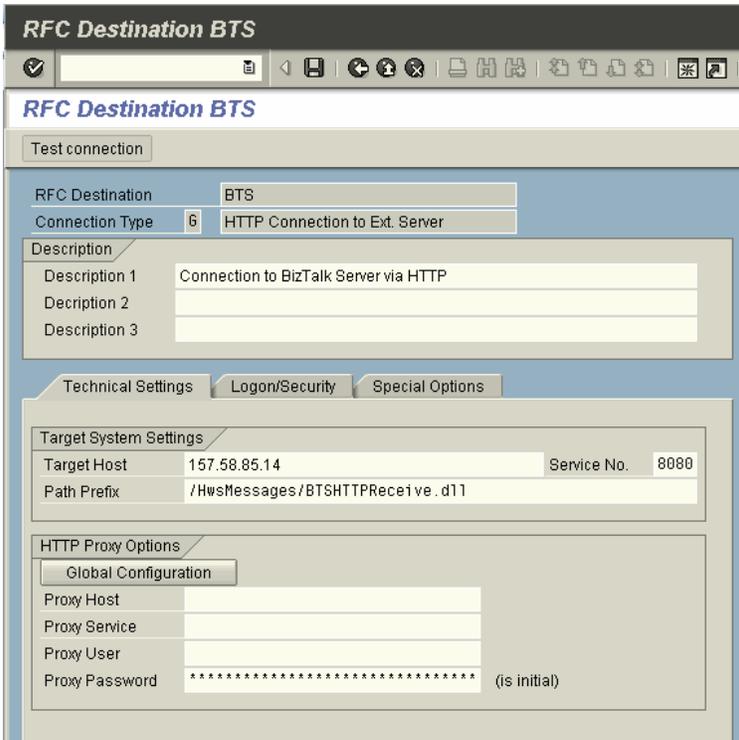


Figure 12 Setting HTTP Connection

On the Logon/Security tab you can define the Logon Procedure and the Logon credentials. It is necessary that the Logon credentials are Windows Logon credentials with the necessary permissions.

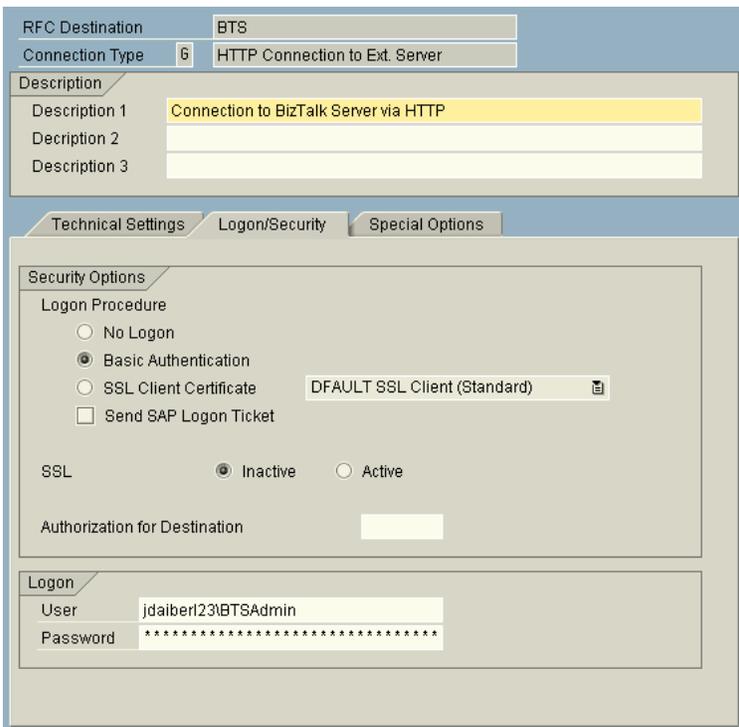


Figure 13 Security Settings HTTP Connection

Once complete, you can test the connection by clicking TEST CONNECTION. An HTTP500 server error will occur because the BTSHTTPRecieve.dll expects to receive a document.

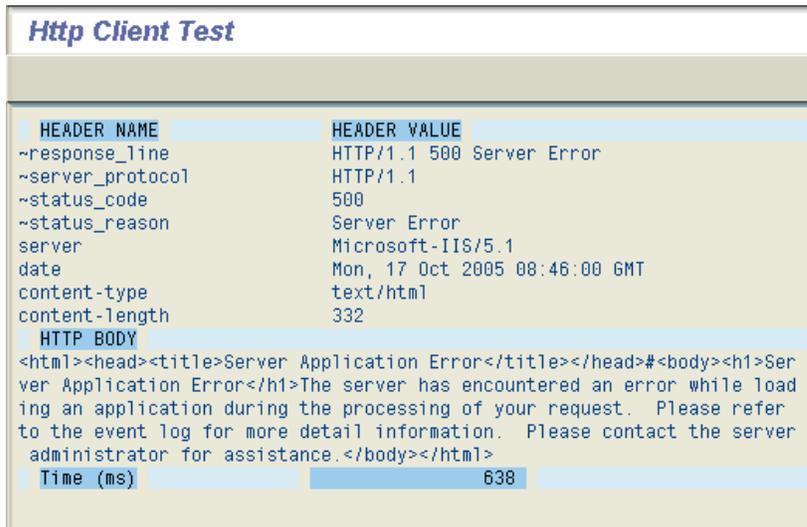


Figure 14 Result of Test HTTP Connection

Configure a port for processing IDocs. For creating a port we use transaction WE21 and create a new XML HTTP Port with the following parameters:

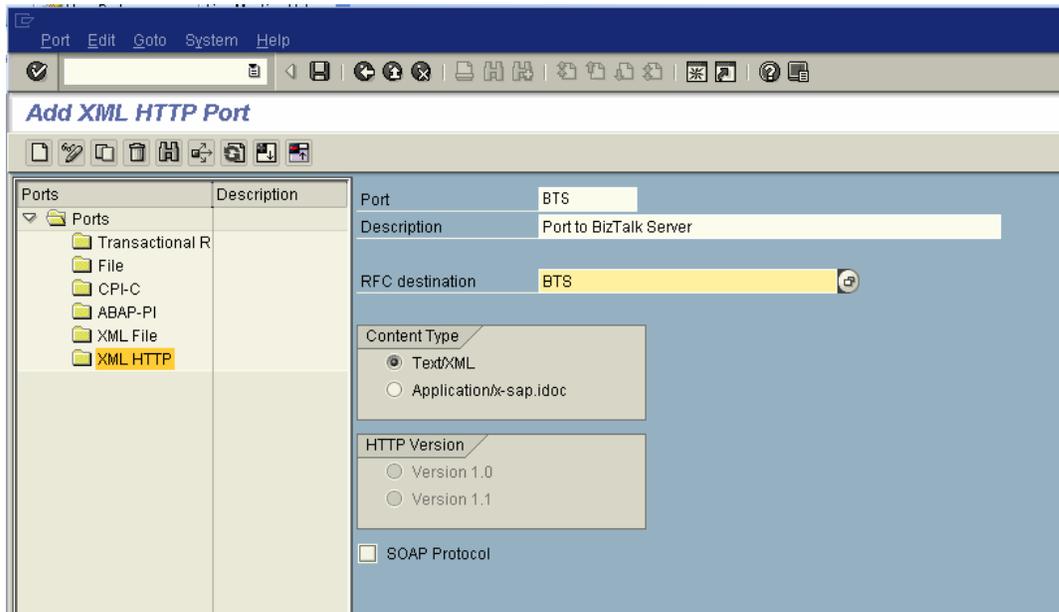


Figure 15 Settings for XML HTTP Port

After creating the XML HTTP Port your SAP NetWeaver Application Server is ready to send and retrieve IDocs via HTTP.

If you send an IDoc to the service running in the SAP NetWeaver AS, the IDoc will be forwarded to the receiver system specified in the IDoc Header. In transaction BD87

(IDoc Monitoring) you will find an error message that no distribution model is defined for this type of information.

Use WE19 (the IDoc test tool) to send out an IDoc to your BizTalk Server. The SAP System will generate an IDoc and post this to the HTTP Receive port on the BizTalk Server side.

You can find a code snippet for a simple ASPX page which writes the incoming IDoc to a folder on your PC in chapter [Code Snippets](#). To use this solution, create a C# ASP project in visual Studio 2003 and deploy the solution to your Web Server where you want to send the IDoc. Then change the target address of you connection specified in transaction SM59 to the address of the ASPX page. The ASPX page will receive the IDoc and will write the IDoc to the folder specified in the code (in the example it is `c:\temp\ldoc.xml`). The output will look like the IDoc shown in chapter [IDoc Sample](#). This sample can also be used for sending an IDoc from BizTalk Server to SAP but in this case the sender and receiver information in the header must be changed according to the settings in the receiving IDoc system. For information regarding the definition of the elements, please refer to the IDoc documentation in transaction WE60.

Developing the BizTalk Server Solution for sending IDocs to SAP

This chapter describes how to setup BizTalk Server to send IDocs from BizTalk Server 2006 via HTTP to a SAP system. The steps described in [BizTalk Server Configuration](#) are required.

Create an empty BizTalk Server project and name it BTS2SAP.

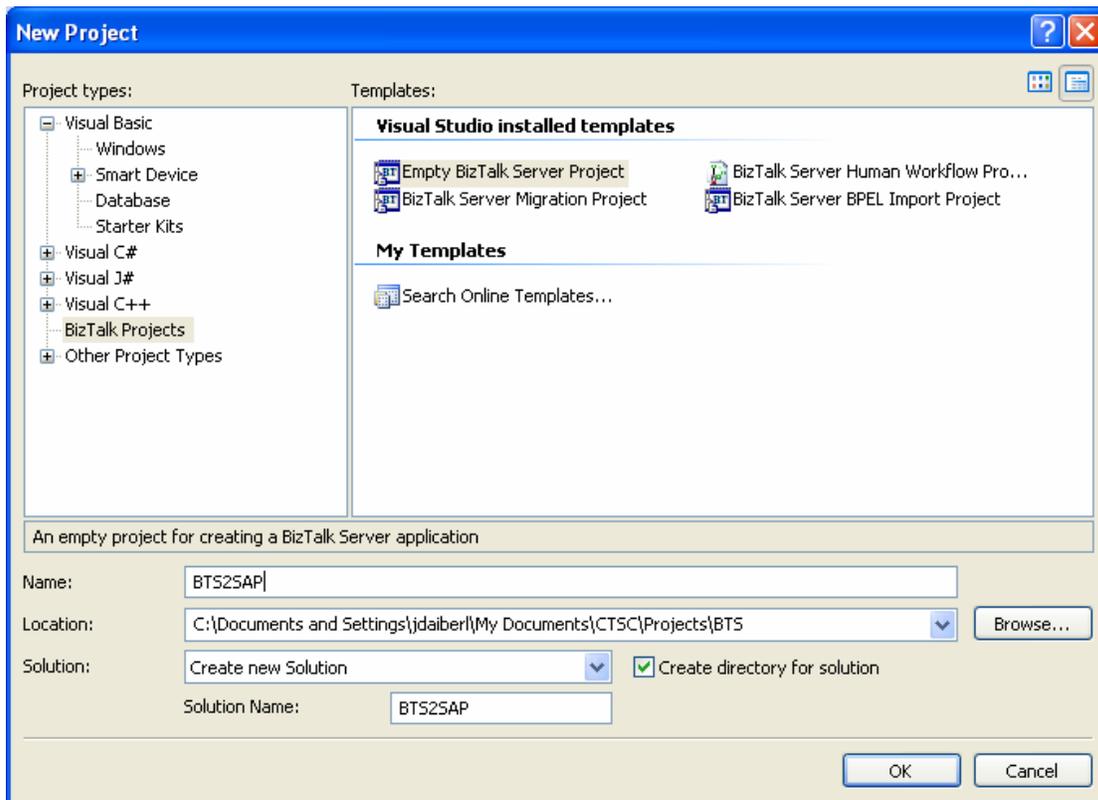


Figure 16 New BizTalk Server Project

Generate the schemas for the IDoc being sent to the SAP system. Transaction WE60 allows the generation of a DTD for an IDoc or you can create an XML schema file; BizTalk Server can then import such a DTD or you can add the XSD file to your project.

For a description of the single IDoc fields and which fields are mandatory you should also check the IDoc documentation of your SAP System or refer to transaction WE60 and display a documentation of your IDoc message type as HTML document.

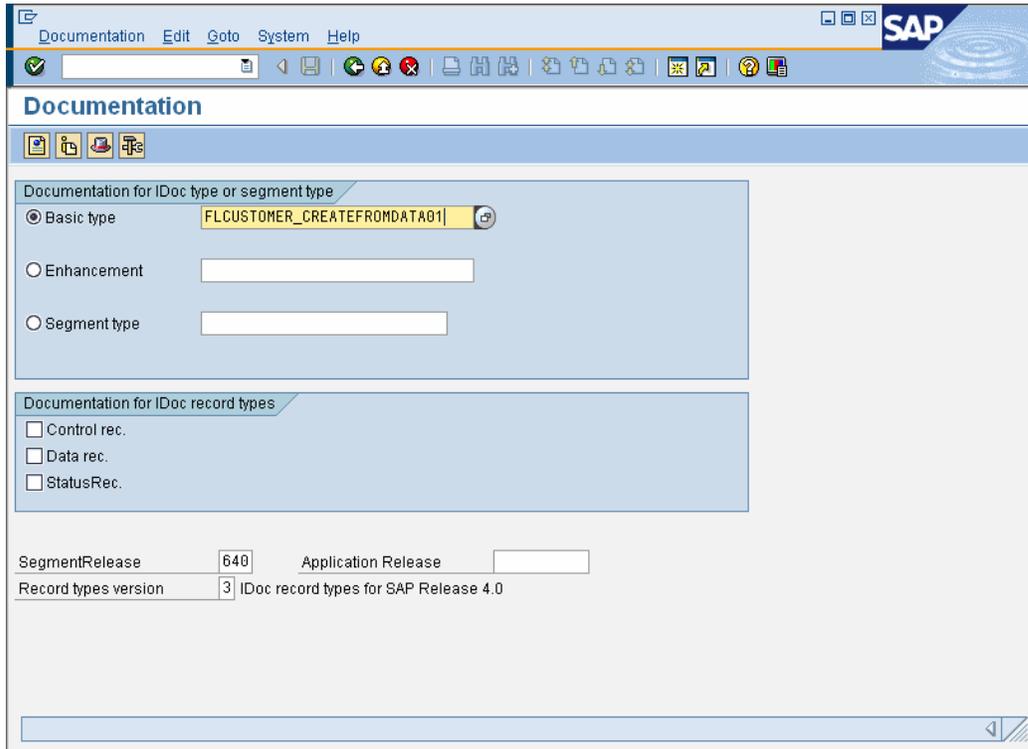


Figure 17 Transaction WE60 - IDoc Documentation

If you choose *Documentation* → *XML Schema*, SAP will generate the XSD Schema for the IDoc and will present the Schema in the SAP GUI. Before displaying the XML Schema a message box will pop up asking you whether you want to generate documentation for a Unicode file or not. Press Yes.

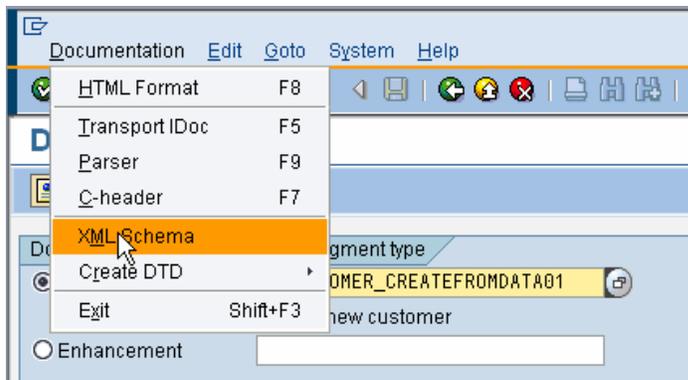


Figure 18 Generate XML Schema

The next figure shows the generated XML Schema.

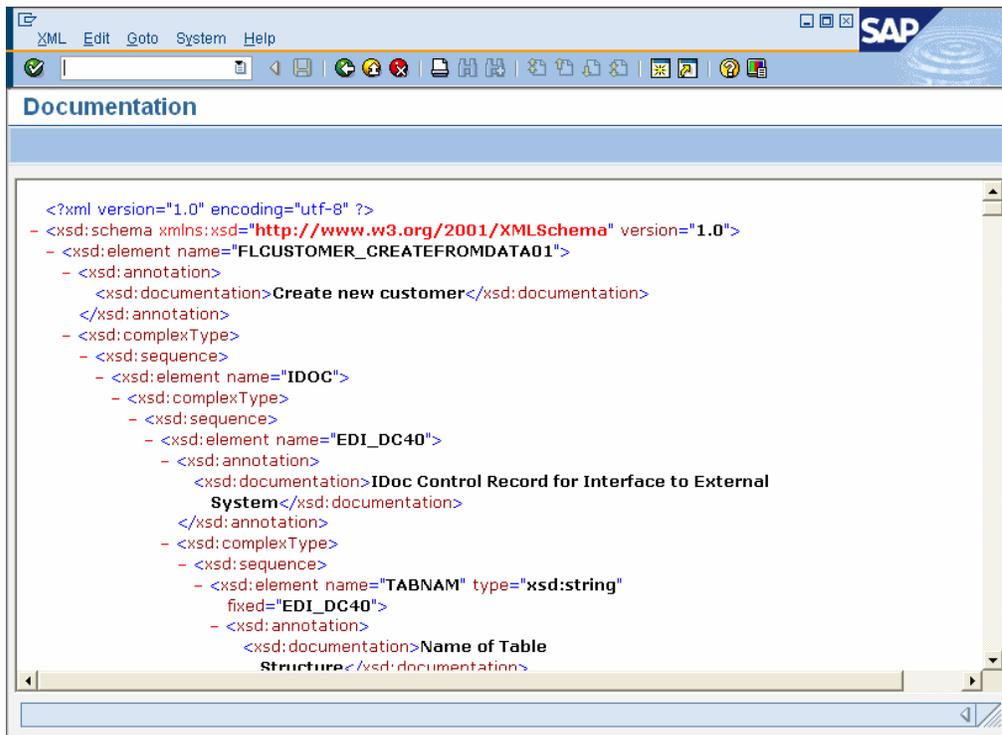


Figure 19 Generated XML Schema

The generated XML Schema can be downloaded, choose *XML* → *Download* and specify a download location on our PC, e.g. the folder where you created the BizTalk Server Project. Choose as file extension *.xsd.

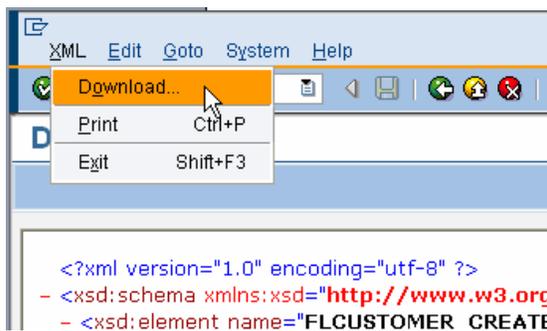


Figure 20 Download generated Schema

In Visual Studio open the Solution Explorer and add the generated XML Schema to your project.

Note: For adding an existing item to your project right-click in the Solution Explorer on your project and choose *Add* → *Existing Item* and then choose the file you downloaded from SAP.

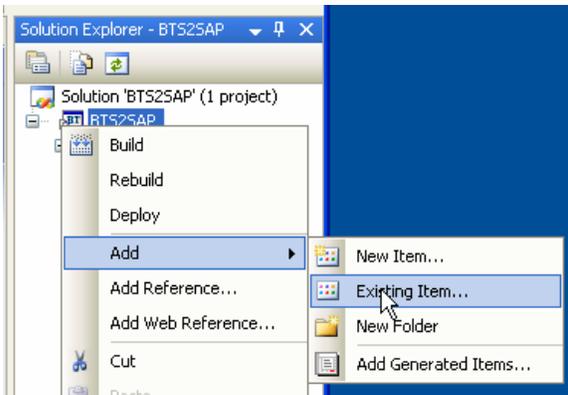


Figure 21 Add XML Schema to BizTalk Solution

BizTalk Server will add the Schema file to your project and you can then open the Schema in Visual Studio.

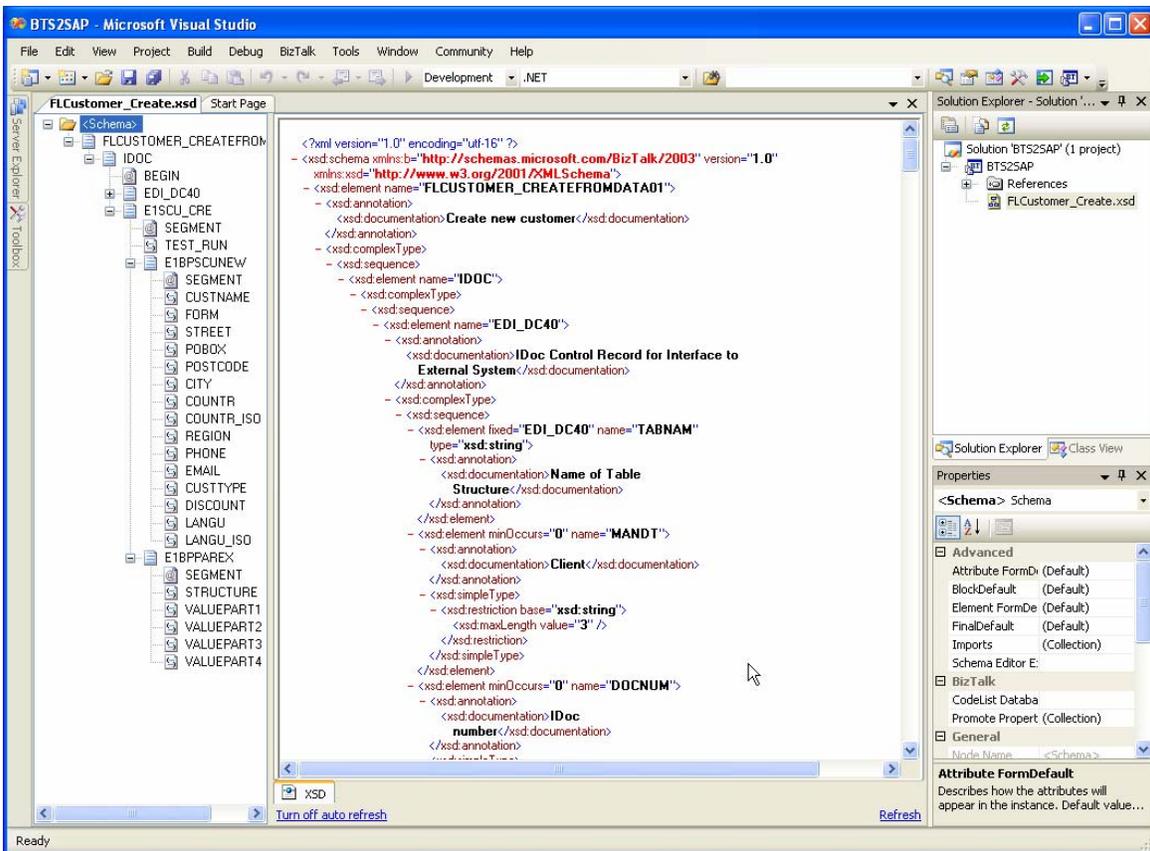


Figure 22 XSD Schema for IDoc FLCUSTOMER_CREATEFROMDATA01

For this walkthrough we will use the same outbound document as the inbound document, no mapping will take place. As this is purely a messaging scenario, one doesn't need to create an orchestration, and instead you could create just the send and receive ports and associated pipelines.

We only add an orchestration because that is the point where you normally would do the mapping from your incoming document to the outgoing IDoc and where you will fill the

header information like the IDoc number and the sender and receiver information. For the BizTalk Server Project create the following artifacts:

- Orchestration with
 - Receive Shape
 - Send Shape
 - Receive Port (e.g. from file)
 - Send Port (HTTP Port to the Service running in SAP NetWeaver AS, see chapter [SAP NetWeaver AS](#))

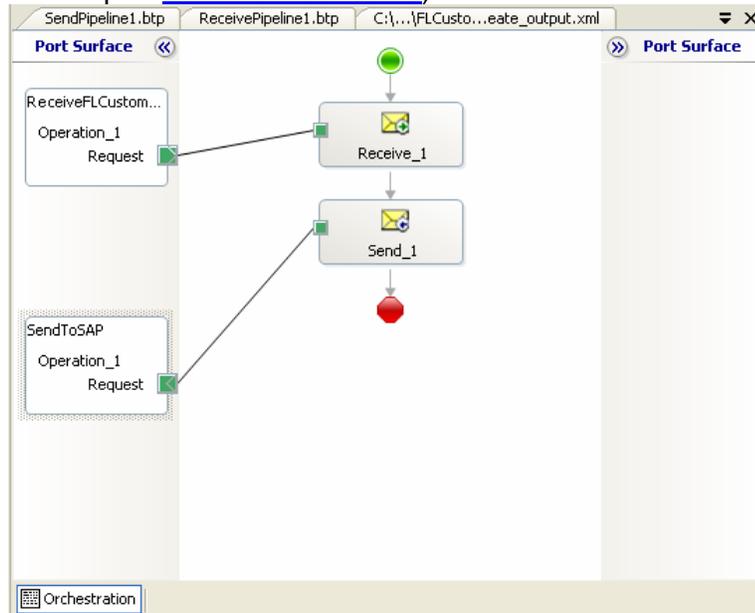


Figure 23 Orchestration Example

- Receive Pipeline with XML disassembler and XML validation
- Send Pipeline with XML assembler

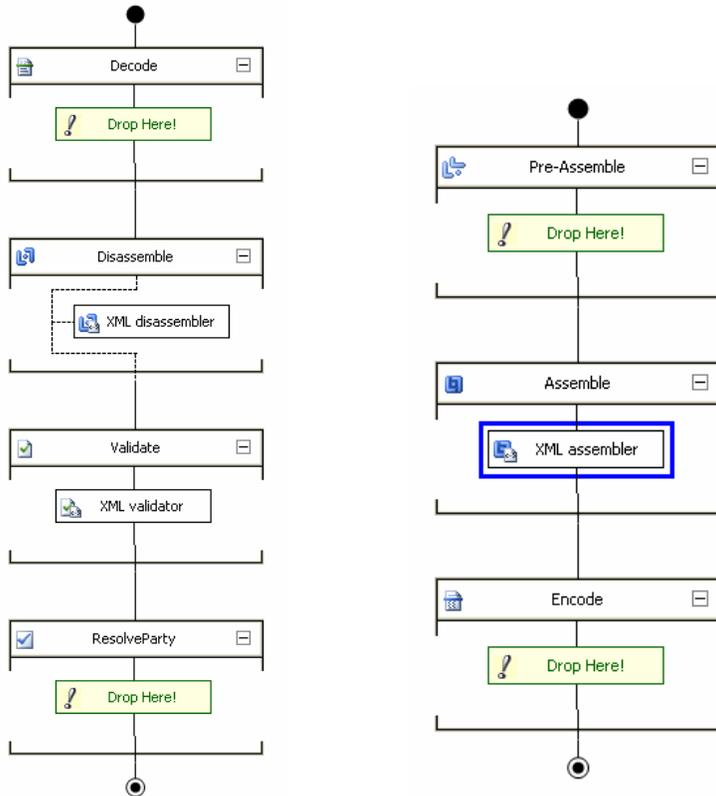


Figure 24 Send and Receive Pipeline

After deploying the BizTalk Server solution you should be able to send IDocs to your SAP system.

An example of an FLCUSTOMER_CREATEFROMDATA01 IDoc can be found in the chapter [IDoc Sample](#). You can use this sample for testing purposes but values in the IDoc header (sender and receiver system) must be set according to the settings in your environment. In addition the IDoc number must be set every time you send an IDoc to the SAP system as the number must be unique per system. In a production environment it is recommended to use a functoid in a map to generate this IDoc number automatically.

In order to check the incoming and outgoing IDocs in SAP, use transaction BD87.

IDocs	IDoc status	Number
IDoc selection		
Changed on is in the range 31.01.2006 to 01.02.2006		
SAP ECC 5.0		21
IDocs in outbound processing		20
IDoc in inbound processing		1

Figure 25 Status Monitor for IDoc

BizTalk Server Solution for retrieving IDocs from SAP

For retrieving IDocs from SAP you can use the same scenario described in the previous chapter. The only difference is that the receive port will now receive the messages via HTTP and the send port will write the IDocs to a directory on your machine. The receive port has to point to the location where your BTSHTTPReceive.dll is running.

The send pipeline and the receive pipeline can be the same as in the solution for sending IDocs from BizTalk server to SAP.

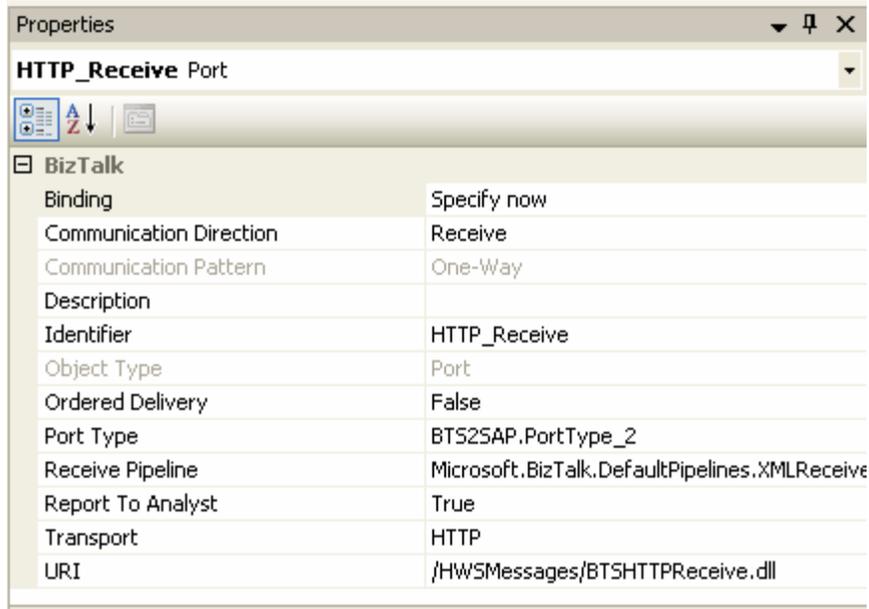


Figure 26 Settings HTTP Receive Port

As in the previous scenario an orchestration is not necessary for this solution but in the in real projects the orchestration is the place where the mapping of the incoming IDoc in XML format to the required outgoing format for your target application will take place.

After deploying the BizTalk Server solution you are able to send IDocs via HTTP from SAP to your BizTalk environment. You can use transaction WE19 in your SAP System for generating and sending out an IDoc. For information about WE19 see http://help.sap.com/saphelp_erp2004/helpdata/en/dc/6b80f643d711d1893e0000e8323c4f/frameset.htm.

After sending out an IDoc to a partner system the outgoing message can be seen in the IDoc monitor, transaction BD87.

Conclusion

This paper describes an integration scenario between Microsoft BizTalk 2004 and higher, SAP NetWeaver AS without additional adapters.

For high availability, one should consider load-balancing the IDoc receiver/sender tier from the message processing tier.

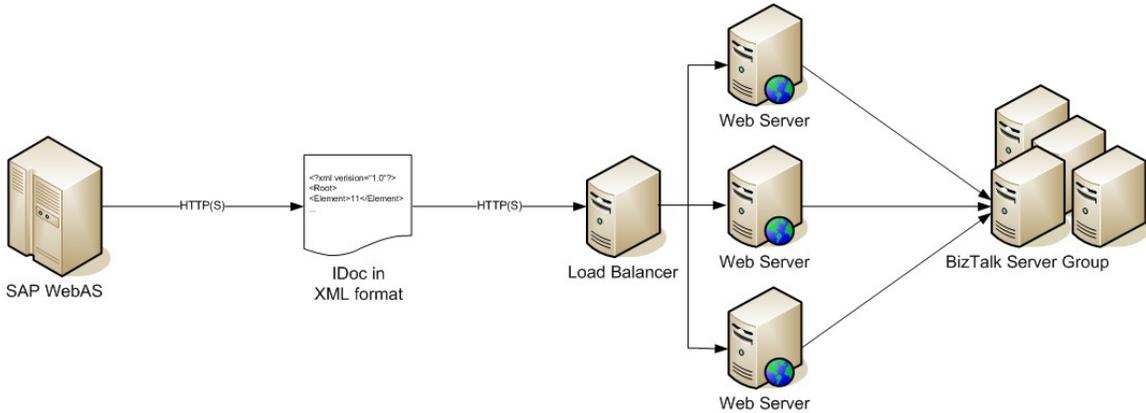


Figure 27 High Availability Scenario

Limitations

The walkthrough described in this whitepaper applies to SAP Web Application Server 6.20, 6.30 and SAP NetWeaver Application Server 6.40. BizTalk scenarios apply to a BizTalk 2004 SP1 installation or a BizTalk Server 2006 installation.

Various tests have shown that the solution described in this paper can also be used for importing and exporting a large number of IDocs in a very short time depending on the architecture of the BizTalk landscape, the complexity of the mapping, and the size of the IDocs.

References

- Microsoft BizTalk Online Help <http://www.microsoft.com/biztalk/techinfo/productdoc/>
- SAP Online Help <http://help.sap.com>
- Understanding Microsoft's Integration Technologies
<http://www.msnusers.com/biztalkserverstuff/Documents/Understanding%20MS%20Integration%20Technologies%2C%201.0.doc>
- SAP Developer Network <http://www.sdn.sap.com>
- Microsoft Developer Network <http://msdn.microsoft.com>

Code Snippets

ASP Page for receiving IDocs

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;

namespace WebApplication1
{
    /// <summary>
    /// Summary description for WebForm1.
    /// </summary>
    public class WebForm1 : System.Web.UI.Page
    {
        private void Page_Load(object sender, System.EventArgs e)
        {
            try
            {
                if (Request != null)
                {
                    if (Request.ContentLength > 0)
                    {
                        Request.SaveAs(@"c:\temp\IDoc.xml", false);
                    }
                }
            }
            catch (Exception exc)
            {
                throw exc;
            }
        }

        #region Web Form Designer generated code
        override protected void OnInit(EventArgs e)
        {
            //
            // CODEGEN: This call is required by the ASP.NET Web Form Designer.
            //
            InitializeComponent();
            base.OnInit(e);
        }

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.Load += new System.EventHandler(this.Page_Load);
        }
        #endregion
    }
}
```

IDoc Sample

```
<?xml version="1.0" encoding="UTF-8" ?>
<FLCUSTOMER_CREATEFROMDATA01 >
  <IDOC BEGIN="1">
    <EDI_DC40 SEGMENT="1">
      <TABNAM>EDI_DC40</TABNAM>
      <MANDT>000</MANDT>
      <DOCNUM>000000000198032</DOCNUM>
      <DOCREL>640</DOCREL>
      <STATUS>30</STATUS>
      <DIRECT>1</DIRECT>
      <OUTMOD>2</OUTMOD>
      <IDOCTYP>FLCUSTOMER_CREATEFROMDATA01</IDOCTYP>
      <MESTYP>FLCUSTOMER_CREATEFROMDATA</MESTYP>
      <STDMES>FLCUST</STDMES>
      <SNDPOR>SAPTLO</SNDPOR>
      <SNDPRT>LS</SNDPRT>
      <SNDPRN>ECC5</SNDPRN>
      <RCVPOR>BTS</RCVPOR>
      <RCVPRT>LS</RCVPRT>
      <RCVPRN>BTS</RCVPRN>
      <CREDAT>20060201</CREDAT>
      <CRETIM>044000</CRETIM>
    </EDI_DC40>
    <E1SCU_CRE SEGMENT="1">
      <E1BPSCUNEW SEGMENT="1">
        <CUSTNAME>MANFRED MUSTER</CUSTNAME>
        <STREET>MUSTERSTR. 88</STREET>
        <POSTCODE>88888</POSTCODE>
        <CITY>MUSTERSTADT</CITY>
        <COUNTR>DE</COUNTR>
        <COUNTR_ISO>D</COUNTR_ISO>
        <REGION>BAY</REGION>
        <PHONE>0888-888999</PHONE>
      </E1BPSCUNEW>
      <E1BPPAREX SEGMENT="1" />
    </E1SCU_CRE>
  </IDOC>
</FLCUSTOMER_CREATEFROMDATA01 >
```

Table of Figures

Figure 1 Overview Business Scenario	4
Figure 2 Create Flight Customer BAPI in transaction BAPI	5
Figure 3 Scenario Overview	5
Figure 4 Transaction SICF	7
Figure 5 Configuration Screen IDOC_XML	7
Figure 6 Handler List.....	8
Figure 7 Activate Service IDoc_XML	8
Figure 8 Confirm activation of service IDoc_XML	9
Figure 9 Define Logical System	9
Figure 10 New Logical System	10
Figure 11 Setup RFC Destination	10
Figure 12 Setting HTTP Connection	11
Figure 13 Security Settings HTTP Connection	11
Figure 14 Result of Test HTTP Connection	12
Figure 15 Settings for XML HTTP Port	12
Figure 16 New BizTalk Server Project	13
Figure 17 Transaction WE60 - IDoc Documentation.....	14
Figure 18 Generate XML Schema	14
Figure 19 Generated XML Schema	15
Figure 20 Download generated Schema.....	15
Figure 21 Add XML Schema to BizTalk Solution	16
Figure 22 XSD Schema for IDoc FLCUSTOMER_CREATEFROMDATA01	16
Figure 23 Orchestration Example	17
Figure 24 Send and Receive Pipeline.....	18
Figure 25 Status Monitor for IDoc	18
Figure 26 Settings HTTP Receive Port.....	19
Figure 27 High Availability Scenario	20