Integration with SAP

How to Automatically Print BarTender Labels using SAP's Auto-ID Infrastructure (AII), IDocs, and SAPscript

SEAGULL
S C I E N T I F I C

Contents

SAP™ (Systems, Applications, and Products in Data Processing) is an enterprise-scale, customizable, workflow application produced by SAP AG of Frankfurt, Germany. It is designed to automate all of the core processes in a large business including order processing, order fulfillment, customer service, supply chain management, and inventory management. It is used by medium and large businesses worldwide. Crucial to SAP's success is its powerful integration features that enable disparate third-party applications to exchange information with each other. BarTender uses a variety of these integration technologies to print data from any of your company's SAP-connected databases onto your labels.

Print using BarTender or SAPscript

When selecting a method of integrating BarTender's label printing functions with SAP, the primary consideration is whether you want to have the print job output by BarTender or SAPscript. (Note that, when printing from BarTender, you can still have SAP trigger and control the print job. In fact, your users don't even have to see BarTender or know that it's running.)

Printing using SAPscript

If you choose to print using SAPscript, this does not allow you to take advantage of the rich, print-time functionality that BarTender provides, such as logging, reprint, and object sizing, etc. It also results in a system that is harder to maintain since format changes and printer configuration changes do not take place immediately, but rather require updating the SAP system. SAPscript printing is covered in the *SAPscript* section below.

Printing using BarTender

When using BarTender as your print engine, you will have several options for implementing the integration. The benefits of one option vs. another will really depend on the skill set of the person doing the integration. All options require at least the Automation edition, except for AII which requires the Enterprise Automation edition.

- AII (Auto ID Infrastructure)
  AII is a part of the NetWeaver platform. It sends XML data packets over a TCP/IP socket that contain the data to print, the name of the format to print, and the name of the printer to print to. Commander receives this data and uses it to instruct BarTender to print labels. AII is particularly well suited for RFID applications. See the *AII (Auto-ID Infrastructure)* section below for details.

- IDoc Files
  Within the ALE component of the SAP system, the user can choose an existing IDoc or create a new IDoc that contains the data to appear on the label. The user also defines

within ALE the transaction(s) that will cause the IDoc to be generated and the destination directory into which to place the IDoc file. See the *IDoc (Intermediate Documents)* section below for details.

- BAPI-Generated Flat, Text Files
  Within SAP, a BAPI can be written using ABAP to export a flat text file. SAPscript-Generated Flat, Fixed-Width Text File A SAPscript report can be designed to generate a flat, fixed-width text file that contains the data to be printed on the label. (SAP can be configured to run this report any time label printing is desired.) Commander can then be configured to detect these files and initiate BarTender print jobs.

- Custom BarTender Integrations
  The SAP system can be configured to make ActiveX calls or execute command lines whenever label printing is desired. This can be used to control BarTender and specify the label format, printer, and data to be printed.

AII (Auto-ID Infrastructure)

This section explains how the SAP Auto-ID Infrastructure (AII) can be integrated with the Enterprise Automation edition of BarTender to automatically print labels.

There are two primary tasks that must be completed in order to implement this label printing integration:

- **Configuring the SAP AII Application.** Label printing integration with SAP AII is based on the generation of an XML file containing all of the information needed to describe an RFID label print job. This application must be configured to create the desired XML file and send it over a specified TCP\IP port. The steps to do this are not documented here but can be found in your SAP documentation. SAP is responsible for helping users get to the point where they are properly generating these XML messages. For additional information on SAP AII and label integration, please see the AII-DC-RFID 1.0.doc file from SAP.

- **Configuring Commander and BarTender.** To handle XML label requests, Commander must be configured both to monitor the port in which the XML files will be received and then respond by executing the desired print job. This white paper documents the steps necessary to accomplish this.

Configuring Commander and BarTender

To handle XML label requests, Commander must be configured both to listen on the TCP/IP port and then respond by executing the desired print job. The following sections document the steps necessary to configure Commander.

Seagull Scientific's Commander Utility

Commander is a utility that allows BarTender to automatically print label jobs in response to certain triggering events from other software. Commander can be set to monitor a TCP/IP socket and begin processing data when triggered. When a trigger event is detected, Commander reads commands and/or data from that trigger and passes them on to BarTender, which executes the print job.

For more extensive information about using the Commander utility, please see the Commander white papers at:

http://www.seagullscientific.com/aspx/whitepapers.aspx

Note: Performance in heavy utilization environments is improved with the Enterprise Automation edition of Commander, as it can launch and communicate with multiple instances of BarTender.

Included Files

Samples files, located in the "BarTender\Formats\SAP AII" sub-folder of your Documents folder, are installed with BarTender as examples of how BarTender can be integrated with SAP AII.

Label Format

A sample label format called **SGTIN-96.btw** is installed. It references the field names that are supplied by SAP AII.

Commander Task List

A sample Commander Task list called **SAP-AII.tl** is provided. It is configured to allow reception of SAP AII XML messages through the TCP/IP port 5171.

Configuring Commander for Integration with SAP AII

To integrate Commander with SAP AII:

1. Install and activate the Enterprise Automation Edition of BarTender.

2. Use the Windows Explorer to browse to the SAP AII samples folder, which should be located in the "BarTender\Formats\SAP AII" sub folder of your Documents folder.

3. Open the "SAP-AII.tl" task list in the SAP AII folder. Commander will launch automatically.

4. A task appears in the Commander screen named **SAP AII (Socket port 5171)**. When you select the SAP AII task check box and click the **Start Detection** button in Commander's tool bar, Commander starts listening for SAP AII messages on port 5171.

Testing the Commander Configuration

In order to confirm that you have Commander properly configured, perform the following test:

1. Configure SAP AII to send label print messages on port 5171.

2. Send a SAP AII message/print job specifying the printer and the BarTender label format.

3. BarTender should respond to the message by automatically printing the label to the printer specified in your SAP AII message.

Diagram of the Print Process

For your reference, the diagram and bulleted steps below illustrate the printing process:
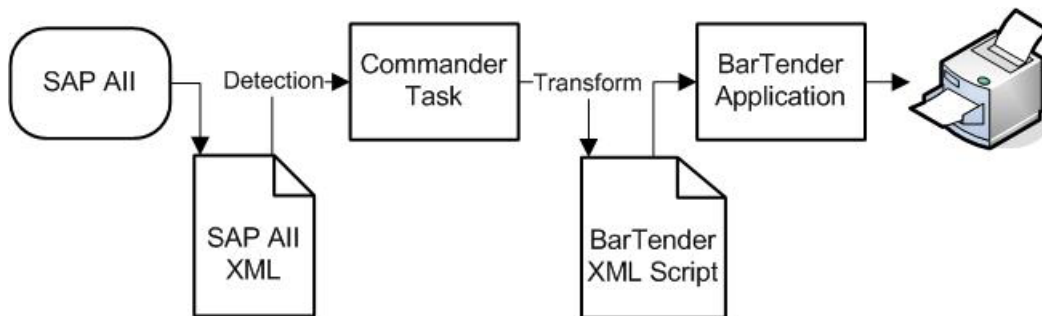
- SAP AII selects the printer and printer format, and evaluates the associated data variables, including a unique ID value. A WriteTagData command message is sent to the Commander using the specified TCP/IP port.

- Commander listens on the specified port and receives the incoming SAP AII message. Sample SAP AII code is displayed below:

```
<?xml version="1.0" encoding="UTF-8"?>
<Command xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Command.xsd">
        <WriteTagData readerID="Writer_Device">
          <item>
             <FieldList format="C.LABEL.PL"
```

```
                    jobName="Writer_Device20040929165746"
                    quantity="1">
                    <Field name="EPC">
                            30740242220403B8000000008</Field>
                    <Field name="EPC_TYPE">SGTIN-96</Field>
                    <Field name="EPC_URN">
                            urn:autoid:tag:sgtin-96:3.5.0037000.065774.8</Field>
                    <Field name="PRODUCT">SGPROD</Field>
                    <Field name="PRODUCT_DESCRIPTION">Test product</Field>
            </FieldList>
        </item>
    </WriteTagData>
</Command>
```

- Commander transforms the data into BarTender XML Script (BTXML) using XSL.

- Commander submits the BTXML to BarTender for execution.

- BarTender sends the format and data to the printer.

- The printer prints a label and optionally writes the ID to the RFID tag embedded in the label.

IDocs (Intermediate Documents)

This section explains how you can integrate IDocs with the one of the two Automation editions of BarTender to automatically print labels.

Overview of SAP R/3 Intermediate Documents (IDocs)

An IDoc is a transactional message, in the form of a pure ASCII file, sent from a SAP-connected application to other applications. Most of an IDoc message consists of fields of data grouped into segments. The segments themselves have a hierarchical relation to each other.

EXAMPLE:

A physician's prescription of a drug for a hospital patient needs to get to the hospital's pharmacy. This could be done by means of an IDoc sent from a bedside application to an application in the pharmacy. Suppose the IDoc has a hierarchy of four levels of segments:

```
Patient Name: Johnson
        Diagnosis: croup
```

```
                    Diagnosis: tibia fracture
                Visit Type: admission
                            Drug:  codeine
                            Drug: amidol
                Visit Type: followup
                            Drug: naprosin
```

Level 1: Contains data that remains constant for years at a time, such as patient name and address.

Level 2: Contains data that remains constant through a given illness, but changes from illness-to-illness; such as primary physician and diagnosis. There can be more than one second level segment for a given patient.

Level 3: Contains data that tend to change from visit-to-visit but remains constant through a particular visit to the hospital; such as visit type (admission or followup) and attending physician. A patient may have more than one hospital visit during an illness.

Level 4: Contains data that tends to change from prescription-to-prescription; such as prescribing physician, medicine, and dosage. More than one medicine may be ordered for a given patient on a given visit.

IDoc Types and Segment Definitions

Since an IDoc is a message, both the sending and receiving applications must conform to a common convention about where, in a given IDoc, each piece of data will be found. To this end, SAP AG has defined several hundred IDoc types and a large number of segment types. And SAP owners can create their own custom IDoc types and segment types.

A sending application must construct an IDoc of a given type in accordance with these definitions; and a receiving application, like BarTender, must conform to the definitions when parsing the IDoc. This means that identifying a parser file is one step in setting up BarTender to use data from IDocs. A parser file for an IDoc type contains the information BarTender needs to parse the IDocs; such as what segments can appear in it, which segments are repeatable, what data fields will appear in each segment, what order the fields will be in, and what length each field will have.

IDoc types have names of six letters and two numerals. SHPMNT01 is an IDoc that embodies a message about shipments. SAP revises the definitions of IDocs from time-to-time, and the two numerals at the end of the name identify the revision.

Segment names may end in three digit version numbers. For example, E2KNA1M001 is a segment for the DEBMAS02 (customer masters) IDoc type.

Converting IDoc Hierarchical Data to Flat Records

Because BarTender views data as organized into tables, it must convert the hierarchies in IDocs to flat records. Fortunately, since data in a child segment is always associated with the data in its parent, any non-branching path through a tree in an IDoc, from a top node to a bottom node can be reconstructed as a record.

In the hospital example above, the path through the tibia fracture and the prescription for naprosin can be collapsed into a flat record like this:

Patient Name	Diagnosis	Visit	Drug
Johnson	tibia fracture	followup	naprosin

The two other complete paths could be collapsed into these records:

Patient Name	Diagnosis	Visit	Drug
Johnson	tibia fracture	admission	codeine
Johnson	tibia fracture	admission	amidol

Master Segment

You will not always need a label for every possible record that could be produced from the data hierarchy in an IDoc. BarTender can be configured at print time to select only some records, but some label formats that draw data from an IDoc will not need every possible record even in the typical case. BarTender uses the concept of the *Master Segment* to help you produce just the records you need from an IDoc.

The Master Segment is the segment whose data is the focus of interest with respect to your labeling needs. BarTender will produce one record for every segment in the IDoc that you designate as the Master Segment.

To continue the hospital example, if the hospital needs a label for every drug prescribed, it would set the Master Segment at the fourth level. Thus, BarTender would produce the three flat records shown above.

Suppose the hospital keeps separate files for every patient visit and wants to generate labels to be used for file folders. Only one label is needed for each visit. So the Master Segment is set to the third level, and BarTender produces two records:

Patient Name	Diagnosis	Visit	Drug
Johnson	tibia fracture	admission	codeine amidol
Johnson	tibia fracture	followup	naprosin

The rule of thumb for deciding what your Master Segment should be is:

In the typical run of labels built from this IDoc,

I'll want one label for every _____.

Steps for Adding an IDoc to Your BarTender System

BarTender's Add Database Connection Wizard takes you through the process of adding a SAP IDoc database to your BarTender system.

1. Launch the wizard by clicking the **Database Connection Setup** button. Then click **Next**.

2. On the **Add Database Connection Wizard - Type** window, select the radio button **SAP R/3 Intermediate Document (IDoc) File**, and then click **Next**.

3. On the **Add Database Connection Wizard - Options** window, select an IDoc type and an IDoc file from which to draw data for your labels. If necessary, add, modify, or remove IDoc types. When you are finished, click **Next**.

4. On the **Add Database Connection Wizard - Master Segment** window, select the **Master Segment** and click **Next**.

5. On the **Add Database Connection Wizard - Fields** window, select the fields on the IDoc from which you want to draw data. If you pick any fields on repeating segments, set the repeating segment rules for each such field.

6. Click **Finish**. You will be returned to the **Database Connection Setup** window. In the tree view of databases available to BarTender, there is now a child, called **SAP Database**, under **All Databases**.

7. Add a bar code or text object to a label and right-click on it. Click **Properties** to open the **Modify . . .** dialog. Click the **Data Source** tab and select **Database Field** as the source and pick the desired IDoc field from the **Use Field** list.

8. Print the label.

Automatically Printing Labels

You can configure your SAP system to automatically create an IDoc file upon certain desired transactions. You can then set up Commander to detect these files and launch BarTender to read the data out of the IDoc and to initiate a print job.

For More Information

For more details about setting up BarTender to use IDocs, refer to the following topics in the online help:

Understanding IDocs

Master Segment

Parser Files

IDoc Types and Segment Definitions

Converting IDoc Hierarchical Data to Flat Records

Database Connection Setup Options Tab for IDocs

Add a new IDoc Type

IDoc Type Definition Window

Modify how BarTender works with an IDoc

Fields Tab for IDocs

Repeating Segment Rules . . . Window

SAPscript

When you print using SAPscript, BarTender simply provides the label formatting commands. SAPscript Then supplies the label data, merges it with the label formatting commands, and sends the resulting print code to the printer. This method of printing does not allow you to take advantage of the rich, print-time functionality that BarTender provides, such as logging, reprint, and object sizing, etc. It also requires you to upload the format every time changes are made. The procedure is as follows:

1. Design a label format in BarTender, with the printer settings set up properly for your desired printer.

2. Export the label format from BarTender in SAPscript ITF (Interchange Text Format). This template consists of a mix of printer commands with place holder tags to mark where the data fields are to be filled in by SAPscript. (The appropriate Seagull printer driver must be available when generating the ITF file. However, not all printer languages are compatible with ITF. Check with Seagull for a list of compatible printer drivers and models.)

3. Import the ITF file into SAPscript.

4. The SAP system can be configured to run the appropriate SAPscript report whenever label printing is desired. The SAPscript reporting engine then replaces the place holder tags with the appropriate label data and sends the resulting code to the desired printer.

SAP is a trademark of SAP Aktiengesellschaft.

Available Seagull White Papers

General White Papers

- The Advantage of Drivers by Seagull

Companion Applications

- Printer Maestro: Enterprise Print Management
- Librarian
- BarTender Security Center
- BarTender Web Print Server

Recent Upgrades

- What's New in the Latest BarTender

Integration White Papers

- Integration Overview
- Commander
- Commander Examples
- BarTender's .NET SDKs
- BarTender's ActiveX Automation Interface
- Exporting Printer Code Templates
- Using BarTender with Remote Desktop Services and Citrix XenApp
- Integration with Oracle's WMS and MSCA
- Integration with IBM WebSphere Sensor Events
- Integration with SAP

Miscellaneous White Papers

- Weighing Scales
- Dynamically Changing Objects at Print-Time using VB Script
- GHS Labeling
- Licensing for BarTender's Automation Editions
- Printing International Characters Using BarTender
- BarTender Software Activation
- Using BarTender's Application Identifier Wizard
- Optimizing Label Printing Performance
- Status Monitor Overview
- Silent Install

For downloadable versions, visit:

www.seagullscientific.com/aspx/white-papers.aspx

SEAGULL
S C I E N T I F I C

w w w . s e a g u l l s c i e n t i f i c . c o m