Merge PDF files in ABAP



Applies to:

SAP Net Weaver 7.0, ABAP. For more information, visit the ABAP homepage.

Summary

This article explains how to merge PDF files using an external non SAP solution from ABAP.

Author: Krisztian Katona

Company: Swisscom

Created on: 06 September 2010

Author Bio



Krisztian Katona is working as SAP developer at Swisscom specialized to technologies as ABAP Objects, WebDynpro and Process Integration.

Table of Contents

3
3
3
3
4
5
6
7
7
8
8
9
10

Introduction

This article will show you how to merge two ore more PDF files into one using ABAP. As the ABAP language or the integrated Adobe Document Services does not support PDF merge operations hence we have to use PDFTK which is an external non SAP solution for PDF manipulation and ZCL_PDF_MERGE an ABAP class to manage the merging process.

How does it work

The PDFTK based merge process consist the following steps:

- The source PDF files are written to a temporary directory on the SAP application server (class ZCL_PDF_MERGE)
- Run PDFTK to merge the PDF files and save the merged file
- Read the merged PDF file and forward it to the ABAP program (class ZCL_PDF_MERGE)

How to set up the environment

This document will describe the process to set up the SAP PDF merging environment on Linux machines. As PDFTK is an open source cross-platform product, it is also possible to set up the environment in different operating systems but it is out of the scope of this article.

Step 1: Installing PDFTK

Download the <u>PDFTK binary</u> and install it onto the SAP application server. Make sure the binary is on the PATH and the user of the SAP installation has the execution right.

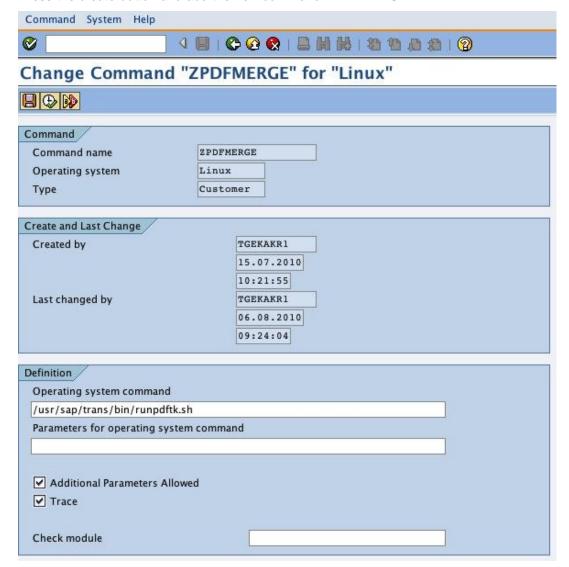
As SAP cannot handle the asterix character when executing external OS commands we have to use a shell script as a transient layer. Create a shell script called *runpdftk.sh* and place it into the application server. The shell script should have the following content:

#!/bin/sh
exec "/usr/local/bin/pdftk" \$1*.pdf cat output \$2

Step 2: Setting up the external execution of PDFTK

Go to transaction SM49.

Press the create button and add the new command ZPDFMERGE



Write the absolute path of the shell script you created in the previous step into the *Operating system command* field.

Check Additional Parameters Allowed

Save your changes

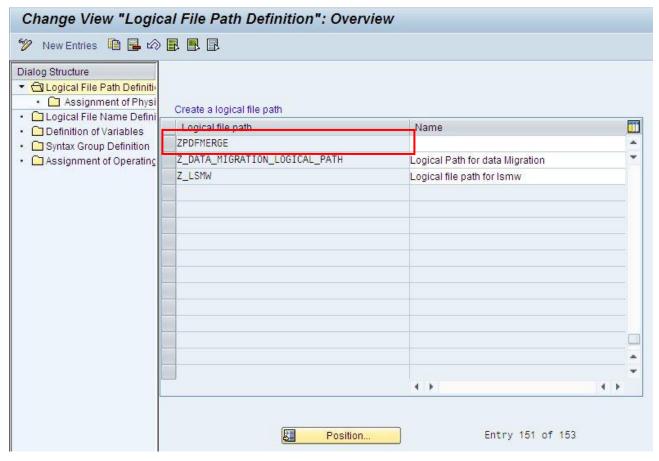
To run external OS commands authorization object S_LOG_COM (COMMAND=ZPDFMERGE OPSYSTEM=Linux HOST=*) is necessary.

Step 3: Define a temporary directory path in SAP

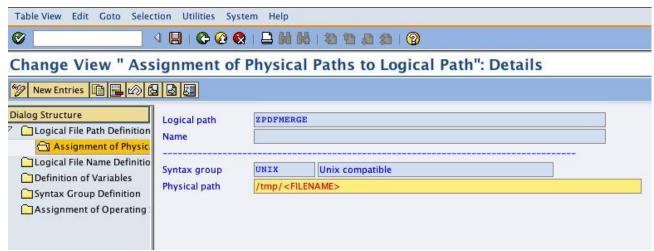
This will be a directory in the application server to store temporary files for the merge operation.

Go to transaction FILE

Click on the New Entries button and add the logical path ZPDFMERGE



Double click on the Assignment of Physical Paths to Logical Path label on the tree structure and add the following Physical path: /tmp/<FILENAME>



Step 4: Set up number range

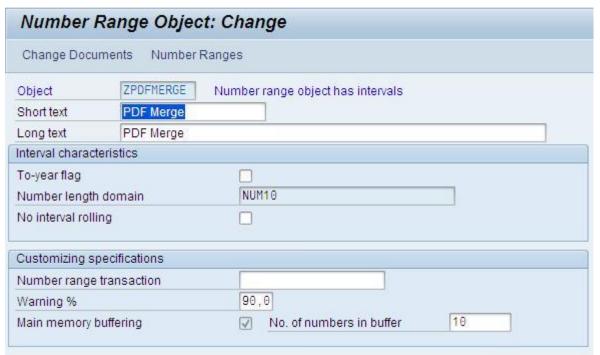
A number range will be used to compose the name of the temporary files.

Go to transaction SNRO

Create number range object ZPDFMERGE.

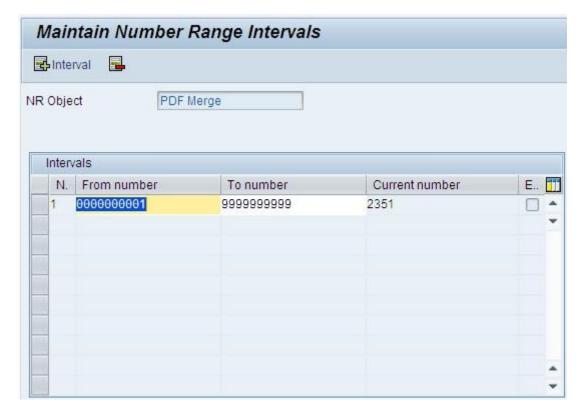


Create a number range (by clicking on the Number ranges label on the toolbar) with the following parameters:



Add new interval to the number range.

You can do this by pressing the number range button on the SNRO screen.



Step 5: Installing ZCL_PDF_MERGE

Download and install ZCL_PDF_MERGE SAP nugget from http://code.google.com/p/sappdfmerge/ and install it using SAPLINK. For documentation on using SAPLINK check this link.

ABAP Code example

Merging individual PDFs into a new PDF

The following code sample illustrates how to use ZCL_PDF_MERGE class for a PDF merge operation.

You can call the ADD_PDF method and add PDF pages one by one in XSTRING format. Finally call the GET_MERGED method

Related Content

For more information, please visit the ABAP homepage.

PDF files in SAP

Offline Interactive Forms Using ABAP

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.