May/June 2008 Volume IX Issue 3

JDEtips Journal

JDEtips

On Reporting, Query, and Business Intelligence

DIY Forms Kick It Up a Notch

By Steve Kriegel

W *Editor's Note: Don't you love when someone says "free" and it really is FREE? Steve Kriegel demonstrates the ten easy steps to create form overlays, a simple way to insert company logos and other graphics onto an AS/400 spool file that can be tied into JD Edwards® – without purchasing expensive software! This technique will provide your reports and forms with a customized, professional look! Read this article, based on Steve's Collaborate 08 presentation, and let your creative side be free!*

Introduction

If you have been around the IBM AS/400 for a long while (and trust me, you have if you still refer to the "System - i" as an AS/400), you may be familiar with the concept of form overlays. If you are not, a form overlay is a simple and absolutely free way to place graphics onto an AS/400 spool file to give it a professional look. In this article, I will take you through ten easy steps to create a form overlay and how you can tie this into your JDE® application.

Business Case

Recently, our company had the request to begin emailing our purchase orders and invoices to our vendors and customers. We had a faxing solution in place, but it would be cumbersome to expand this to a wide range of customers. We also received a quote from one of our vendors to upgrade our solution, but felt we had the expertise and tools in house to satisfy this request. Our email software already had the capability to convert AS/400 spooled output to PDF attachments. We only needed to add the graphics to the R43500 and R42565 reports and interface the resolved output to the email software via the vendor supplied API.

This is where the concept of the IBM form overlay comes in. With this technique, you will be able to take a standard character set spool file (*SCS) and create a new advanced printer data stream (*AFPDS) spool file with a graphical overlay. These overlays can be tailored by the Dream Writer version allowing you to create a different looking document for a PO Requisition, vendor PO, and a PO reprint even though each produces the same spool file (R43500). Or, if you need to produce overlays for invoices that contain brand-specific logos or overlays by branch plant, you can accomplish this with a simple change to the Dream Writer version. Once the programming is in place, all that remains is to configure the system and to maintain the UDC tables.

The concepts and techniques you will see in this article were written for a World® A7.3 system, but are applicable to World A8.1, World A9.1, as well as any AS/400 generated spool file.

> ***A form overlay is a simple and absolutely free way to place graphics onto an AS/400 spool file to give it a professional look.***

The Solution

In order to implement this technique, you will need to set up an output queue to which you will redirect all your reports. When the report status changes from OPN to RDY, it triggers a program that is running in the background to submit a job to reprocess the spool file. This submitted job reads a UDC table and applies the proper form overlay to the spool file.

Step 1: Create a Data Queue

The data queue is only a small 222-character string of memory that will be used (in conjunction with the output queue) to trigger the background job. To create the data queue, type the following command from an IBM command line:

```
CRTDTAQ DTAQ(QGPL/EDISTRIB) MAXLEN(222) TEXT('eDISTRIB Data Queue')
```

Step 2: Create an Output Queue

The output queue is where you redirect all your output. You DO NOT want to attach a printer to this output queue. If you do, you will print your reports before you get a chance to put a form overlay on them. You will also attach the data queue that you created in Step 1 to

JDEtips © 2008 Klee Associates, Inc.

JDEtips

the output queue. To create the output queue, type the following command on an IBM command line:

CRTOUTQ OUTQ(QGPL/EDISTRIB) TEXT('eDISTRIB Output Queue') JOBSEP(0) DTAQ(QGPL/EDISTRIB)

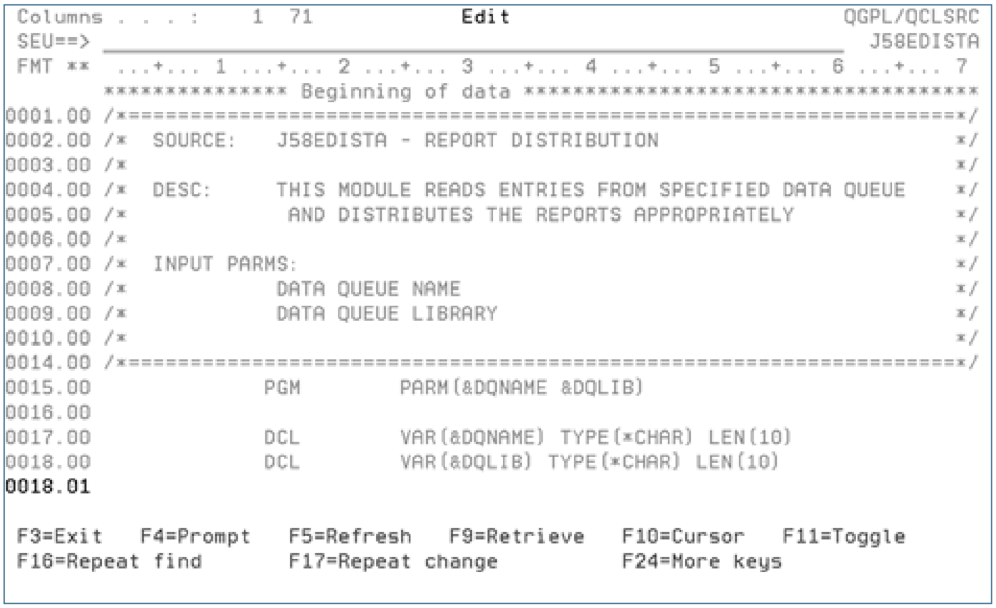
Step 3: Create the Background Program to Monitor the Status of the Output Queue

Now you need to create a CL that will constantly be running in a subsystem (I chose to run ours in QCTL), the sole purpose of which is to listen for entries made to the data queue. This program accepts the parameters of the data queue name and data queue library that you created in Step 1 (see Figure 1).

This program sits and waits on the IBM program QRCVDTAQ (see statement 49.00 in Figure 2). When the status of the spool file on the output queue changes to a RDY status, an entry is written to the data queue and is processed by this program. Statements 54.00 through 63.00 parse out the attributes of the spool file (job name, user ID, job number, spool file name, and spool file number).

Once parsed out, it submits a second job to process the spool file. This second job is submitted on behalf of the original user, so this CL must be running with a user profile that has enough authority. I chose QSECOFR as my profile to run under; however, you may have another profile that has *ALL OBJECT authority you would like to use. Once the SBMJOB command (statement 65.00) is executed, the program loops back to the READNEXT tag at statement 48.00.

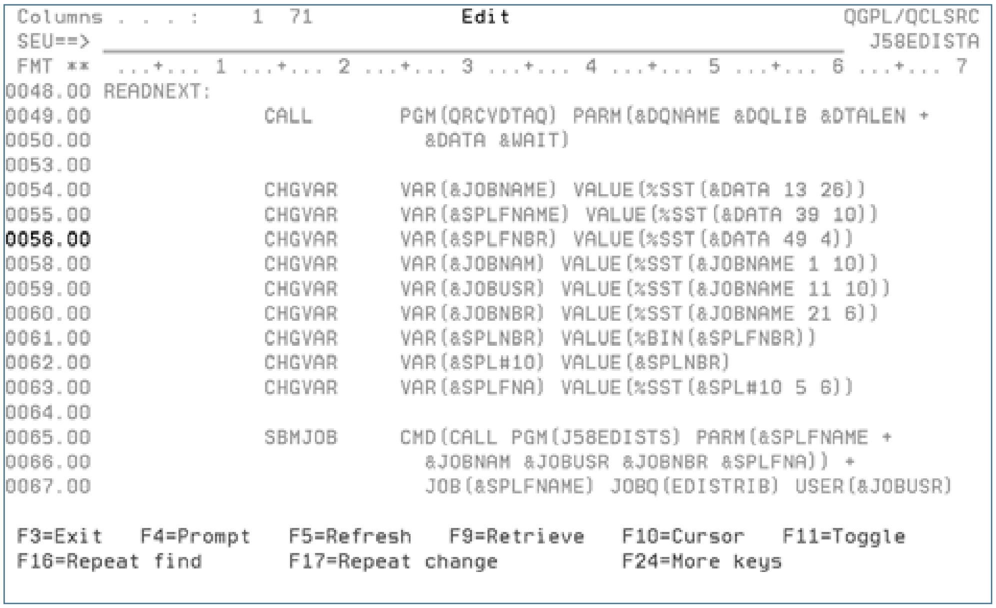
I also created a single threaded job queue to which we will submit the second CL (also named EDISTRB). You may have another job queue that meets this criterion that could be used, but I chose to create one to allocate to this function. Each spool file is generally small, and this second job finishes in a matter of seconds. Even larger spool files take only a few minutes to reprocess, so you will not have jobs stacking up in this job queue.



```
 Columns . . . :     1  71                 Edit                         QGPL/QCLSRC
 SEU==> ____________________________________________________________      J58EDISTA
 FMT **   ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
        *************** Beginning of data ************************************
0001.00 /*==================================================================*/
0002.00 /*  SOURCE:    J58EDISTA - REPORT DISTRIBUTION                         */
0003.00 /*                                                                     */
0004.00 /*  DESC:      THIS MODULE READS ENTRIES FROM SPECIFIED DATA QUEUE     */
0005.00 /*              AND DISTRIBUTES THE REPORTS APPROPRIATELY              */
0006.00 /*                                                                     */
0007.00 /*  INPUT PARMS:                                                       */
0008.00 /*             DATA QUEUE NAME                                         */
0009.00 /*             DATA QUEUE LIBRARY                                      */
0010.00 /*                                                                     */
0014.00 /*==================================================================*/
0015.00              PGM        PARM(&DQNAME &DQLIB)
0016.00
0017.00              DCL        VAR(&DQNAME) TYPE(*CHAR) LEN(10)
0018.00              DCL        VAR(&DQLIB) TYPE(*CHAR) LEN(10)
0018.01

 F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F10=Cursor   F11=Toggle
 F16=Repeat find       F17=Repeat change          F24=More keys
```

Figure 1: Background CL Program (Input Parameters)



```
 Columns . . . :     1  71                 Edit                         QGPL/QCLSRC
 SEU==> ____________________________________________________________      J58EDISTA
 FMT **   ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0048.00 READNEXT:
0049.00              CALL       PGM(QRCVDTAQ) PARM(&DQNAME &DQLIB &DTALEN +
0050.00                           &DATA &WAIT)
0053.00
0054.00              CHGVAR     VAR(&JOBNAME) VALUE(%SST(&DATA 13 26))
0055.00              CHGVAR     VAR(&SPLFNAME) VALUE(%SST(&DATA 39 10))
0056.00              CHGVAR     VAR(&SPLFNBR) VALUE(%SST(&DATA 49 4))
0058.00              CHGVAR     VAR(&JOBNAM) VALUE(%SST(&JOBNAME 1 10))
0059.00              CHGVAR     VAR(&JOBUSR) VALUE(%SST(&JOBNAME 11 10))
0060.00              CHGVAR     VAR(&JOBNBR) VALUE(%SST(&JOBNAME 21 6))
0061.00              CHGVAR     VAR(&SPLNBR) VALUE(%BIN(&SPLFNBR))
0062.00              CHGVAR     VAR(&SPL#10) VALUE(&SPLNBR)
0063.00              CHGVAR     VAR(&SPLFNA) VALUE(%SST(&SPL#10 5 6))
0064.00
0065.00              SBMJOB     CMD(CALL PGM(J58EDISTS) PARM(&SPLFNAME +
0066.00                           &JOBNAM &JOBUSR &JOBNBR &SPLFNA)) +
0067.00                           JOB(&SPLFNAME) JOBQ(EDISTRIB) USER(&JOBUSR)

 F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F10=Cursor   F11=Toggle
 F16=Repeat find       F17=Repeat change          F24=More keys
```

Figure 2: Background CL Program (Parse Spool File Attributes and Submit Job)

This Article Continues…

Subscribers, log in from our main search page to access the full article:

www.JDEtips.com/MyAccess.html

Not a Subscriber? Gain access to our full library of JDE topics:

www.JDEtips.com/JD-Edwards-Library

Visit **www.JDEtips.com** for information on the JDEtips University schedule, private training and consulting, and our Knowledge Express Document Library.

License Information: The use of JDE is granted to JDEtips, Inc. by permission from J.D. Edwards World Source Company. The information on this website and in our publications is the copyrighted work of JDEtips, Inc. and is owned by JDEtips, Inc.

NO WARRANTY: This documentation is delivered as is, and JDEtips, Inc. makes no warranty as to its accuracy or use. Any use of this documentation is at the risk of the user. Although we make every good faith effort to ensure accuracy, this document may include technical or other inaccuracies or typographical errors. JDEtips, Inc. reserves the right to make changes without prior notice.

Oracle and J.D. Edwards EnterpriseOne and World are trademarks or registered trademarks of Oracle Corporation. All other trademarks and product names are the property of their respective owners.

Copyright © by JDEtips, Inc.