

```
[8.0 README]
dsn_sample "/u0/unform_test/_oem/cp/cp7/uf8/readme-cp7-v80.txt"
```

```
# To view this readme section inside the design tool without
# color and syntax error red-mark lines, TURN OFF syntax
# highlighting by pressing the color palette on the ribbon
# menu above!!
```

UnForm 8.0 Rule-file and supporting demo files for CP7

Including these integrated features:
(as of 08/03/2010)

- Archiving preconfigured with standard properties
- Scanning barcodes on forms
- Statement with attached scanned proof of delivery copies out of archive
- PDF hyperlinks to invoices in archived statement
- Electronic form delivery by email or fax
- Page X of Y counters on forms and reports

```
-----
1
-----
```

"cp7-v80.zip" File

Extract or place the contents into the main unform server folder.

Contents:

- cp7-v80.rul file
- cp7-demo-80.sh for Linux/Unix, build archive demo data
- cp7-demo-80.vbs for Windows, build archive demo data
- cp7-txt/ folder containing sample text files
- cp7-scan/ folder containing sample scans and the CP_DOCS image manager job definition.

```
-----
2
-----
```

New "utility" rulesets in the cp7.5-v80.rul file which support the new features:

```
[8.0 zMrg Configurator]
[8.0 zMrg ArcProps]
[8.0 zMrg PDFLinks v2]
[8.0 EDM2 DELIM]
[8.0 EDM6 BOOK]
[8.0 ED TAG PARSE]
[8.0 ED Selective Copy Printing]
[zMrgMultiPageXofY]
[zMrgSinglePageXofY]
```

```
-----
3
-----
```

TURNING FEATURES OFF AND ON:

Top global section of the rule file has:

```
merge "8.0 zMrg Configurator"
```

to include the global constants that turn the new features ON/OFF by

feature and ruleset.

See the constants in "8.0 zMrg Configurator" for which forms are archive-enabled, deliver-enabled, or scanning barcode-enabled, and to turn them on/off.

4

For each CP7 "enhancement" rule-set:

Original code blocks were moved to the bottom of the rule-set so that timing of text movement would be performed AFTER get() functions needed for archiving and deliver properties.

Where applicable, the following code sections were inserted with processing code at the BOTTOM of existing rule-sets, but BEFORE the original code block section.

```
# --- 8.0 ARCHIVING -----  
# --- PDF HYPERLINK FOR ARCHIVING ----  
# --- 8.0 DELIVER -----  
# --- PAGE X of Y FEATURE -----  
# --- ATTACH SCANNED IMAGES -----  
# --- SCANNING BARCODE -----
```

Here's a cross-reference between the new rule-set sections and the new utility rule-sets that are merged:

```
# --- 8.0 ARCHIVING -----  
merge "8.0 zMrg Arcprops"  
  
# --- PDF HYPERLINK FOR ARCHIVING ----  
merge "8.0 zMrg PDFLinks v2"  
  
# --- 8.0 DELIVER -----  
merge "8.0 EDM2 DELIM" (global EDEL_METHOD)  
(or) merge "8.0 EDM6 BOOK"  
(which in turn does: merge "8.0 ED TAG PARSE")  
merge "8.0 Selective Copy Printing"  
  
# --- PAGE X of Y FEATURE -----  
Forms: merge "zMrgMultiPageXofY"  
Reports: merge "zMrgSinglePageXofY"  
  
# --- ATTACH SCANNED IMAGES -----  
(No merges)  
(Statement forms only)  
  
# --- SCANNING BARCODE -----  
(No merges)
```

5

ARCHIVING notes:

Archiving properties for each form and report are maintained centrally in the "8.0 zMrg ArcProps" rule-set. This model is used to facilitate mass re-indexing processes. By centralizing archive properties by form or report type in a single rule-set, a mass re-indexing rule-set will be able to access property assignments for multiple doc types in the same place that the new property assignments are coded for print jobs.

"8.0 zMrgArcProps" rule-set Maintenance:

To change the library names created on your system:

Change the following local constant to zero:

```
local EARC_LIBCOABRSUB=1
```

And/or change or make null (") this library company abgreivation prefix:

```
local EARC_COABREVSUJ="CP7V80"
```

See the individual form "case" statements for each ruleset to modify the earca_library\$ variable to change the library name from "Sales", "Accounting", etc.

IMPORTANT NOTES FOR SITES UPGRADING FROM v71 eArchive:

A

The document date property is now defaulted to the document date printed in the raw text rather than the date processed through UnForm. Previously, a reprint of a document on a different date would have changed the document date to the current date, or a post-dated or future-dated document would have archived with the current date, rather than the date printed on the document. It may be necessary to script a special conversion rule-set to update previously archived documents for date consistency.

B

The below referenced paramater in "8.0 zMrg Configurator" should be set to 'Y' if you are updating from a -v71 rule file and want to maintain the consistency of date categories.

```
# v7.x Archive date category compatibility
global EARCA_71=N
```

Leaving this parameter as 'N' will create date categories where the month segment is in 'MM(mmm)' format, e.g. '12(Dec)' rather than just the numeric month. It is possible to create a rule-set script that could do a mass update/conversion of the old date categories to the new format.

```
-----
6
-----
```

DELIVER notes:

8.0 uses the new DELIVER command.

Address retrieval methods can be chosen by the integrator.

The EDEL_METHOD constant:

Choose either the delimited line method (8.0 EDM2 DELIM) or the address book method (8.0 EDM6 BOOK) by removing the #-sign from one of these lines in the "8.0 zMrg Configurator" rule-set:

```
global EDEL_METHOD="8.0 EDM2 DELIM"
# global EDEL_METHOD="8.0 EDM6 BOOK"
```

If the address retrieved contains an @-sign an EMAIL will be submitted.

If the address retrieved does NOT contain @-sign, a FAX will be initiated.

You can customize tags for each form individually within each rule-set, in the routine underneath the merge for the address book.

You must configure/maintain the deliver.ini file for email tag and fax settings.

You must configure/maintain the prog/mailcall.ini file for email smtp server settings.

More information is available on *.ini configuration in uniform80.pdf.

If you want to use a different method contact SDSI support for assistance and/or see the samples/eDeliver80.rul for more examples.

Delimited Line Method (8.0 EDM2 DELIM):

Form raw text output contains a delimited line printed by the CP7 application which contains address and other tag information. The integrator uses CP form maintenance capabilities to place the data there in the format expected by UnForm.

Address Book Method (8.0 EDM6 BOOK):

Addresses are retrieved from an address book named "cp7" using an address entity ID somewhere on the form, e.g. customer number, vendor number, etc.

The delimited line method is the default for compatibility with previous CP versions.

Additional delimited line method information:

EMAIL delimited line in raw text format:

```
email|copynumber|emailaddress|fromaddress|subject|body|attachment|ccaddress|
      bccaddress|otherheader|login|password|mimetype|printcopy
```

The fromaddress can be left blank if a default from address is set up in the mailcall.ini file found in your UnForm prog/ directory. Login and password are only needed for authentication and, like the fromaddress, can be setup in the mailcall.ini file.

If no email line exists, no emailing will be done.

FAX delimited line in raw text format:

```
fax|copynumber|faxnumber|from|subject|body|attachment|cover|toname|
      tocompany|fromcompany|||printcopy
```

If no fax line exists or the fax command in deliver.ini is not setup, no faxing will be done.

In both cases the first three elements are required, all others are optional, but if populated must maintain their delimited position in the string, E.g. email|copynumber|emailaddress||subject|||ccaddress|||printcopy. The printcopy element allows for more dynamic control of a printed copy when an electronic copy is specified, for instance when certain customers require both a printed and emailed copy, while some don't. If printcopy is 0 (zero), the printed copy will be suppressed.

Additional address book method information:

The "cp7" address book can be defined in the UnForm browser interface. You can import addresses from an appropriately mapped CSV file to populate the address book.

See the "8.0 EDM6 BOOK" merged rule-set to see a hierarchy for looking up address info that demonstrates how to look up addresses with or without document types, suffixes or prefixes. The routines here are for demonstration purposes -- you can modify the address lookup logic to suit your needs, and/or contact SDSI for assistance.

If you want to use a method other than the address book method, contact SDSI about tailoring one of the pre-developed merge rule-sets for other methods that are contained in the samples/eDeliver8.0.rul file. Contact SDSI for possible updates to these templates.

SCANNING BARCODE notes:

See the "BCOD_" series of contents in "8.0 zMrg Configurator" to turn scanning barcodes ON or OFF by form.

This line in the scanning barcode section opens up 7 rows at the bottom of the form for placement of barcodes, and causes the rest of the form to resize with a maintenance of its aspect ratio:

```
rows$=str(MAXROWS+7)
```

This line allows you to specify which copy of a form gets a barcode:

```
bcod_Copies$="2" # which copy(s) get barcode
```

To include the barcode on multiple copies, add the copy numbers to the string, e.g. `bcod_Copies$="123"`. Note that this syntax maxes out at 9 copies, so the `pos()` function would need to be modified if copy numbers above 9 need to be referenced.

The `bcod_docty$` variable places an abbreviated doctype code on the form that is mapped to the full archive document type in the image manager job definition that automates property index assignment on scanned documents. IF YOU CHANGE A DOCUMENT TYPE IN THE ARCHIVING RULE-SET, YOU MUST CHANGE THE CORRESPONDING ENTRY IN THE SCANNING JOB DEFINITION, IF THAT DOCUMENT WILL BE SCANNED AND UPLOADED.

```
-----  
8  
-----
```

SCANNING JOB DEFINITION notes:

A scanning job definition is provided called "CP_DOCS.ufz" in the "cp7-scan" folder. Copy this file to the scanning/JobDefs folder of the image manager workstation. It has been designed to detect and process all the barcoded sxe forms using a single job, including header/attachment jobs where attachment pages receive the same indexes as the header page. Demo scanned images are provided for importing, both regular scans with no attachment pages, as well as ones with attachment pages.

```
-----  
9  
-----
```

DEMO BUILD SCRIPT notes:

****CAUTION**** THESE SCRIPTS ARE DESTRUCTIVE TO PREVIOUSLY ARCHIVED DOCUMENTS. PLEASE READ THIS CAREFULLY BEFORE USING THEM.

- cp7-demo-80.sh for Linux/Unix
- cp7-demo-80.vbs for Windows

Execute the appropriate script from the server folder where the zip file was extracted. It builds demo archives from the text samples contained in the "cp7-txt/" folder.

WHEN LAUNCHED, THE SCRIPTS START BY REMOVING THE SAMPLE LIBRARIES, SO DO NOT RUN THESE SCRIPTS ON A LIVE SYSTEM THAT USES THE SAME LIBRARY NAMES AS THE DEMO!!

See the Archive notes for how to change the library names for the live system.

Or modify the library name prefix at the top of the scripts to

reference a non-live set of libraries.

The build scripts will also create PDF's of the documents in the text file folder. If deliver feature is enabled and an address retrieval method is properly set up, emails or faxes will be attempted, assuming proper configuration of deliver.ini and mailcall.ini.

10

ATTACH SCANNED IMAGE notes:

The AR statement rule-sets contain a routine to retrieve scanned copies of invoices from the archive and attach them to the statement.

There is a global to turn this feature ON or OFF in the constants merged rule-sets.

By default, for each invoice listed on a statement, the routine looks for document images based on the following variables:

dotypes\$="Scan":

- The image sub-id begins with the word "Scan"
- change this if you change the sub-id in the scan job definition

tktplib\$="CP7V80_Sales":

- change this in the ruleset to match your system

tktdocty\$="Invoice":

- change this if you change the doc type for archived invoices

Imagemagick and/or Ghostscript must be available for this feature to work.

By default images are tiled 2 across, 2 down in portrait orientation, as seen in the below command:

```
images {images$},across 2, down 2
```

Remove the across and down arguments to attach full page images.