Using the New SNAP Features

IOF Release 8E contains significant SNAP enhancements for both end users and Rexx application developers.

Snapping Data in html Format (at the terminal)

Specify “html” in the new CONVERT field on the SD, SU, and SS panels to request that the snapped data be converted to a file that can be displayed by a web browser.

This option will enclose the output in simple html header and trailer records and map any potential html codes in the output to transparent html equivalents.

To snap data in html format to a disk data set:

```
SD              (display the SD panel for specifying a disk snap target)
“html”           (value in the “Convert” field)
ENTER           (allocate and open the snap target data set)
SNAP …          (various snap commands to copy data)
SNAPCLOS        (close the snap target data set)
```

Snapping Data in HTML Format (in a Rexx exec)

A Rexx exec can specify the new CONVERT parm on the SS, SD, and SU commands to set up a snap target data set in html format. To snap data in html format to a disk data set:

```
SD  DSN(…) …  CONVERT(HTML)
SNAP …                          (various SNAP commands to copy data)
SNAPCLOS
```

The CONVERT(HTML) parm encloses the output data in simple html header and trailer records and maps any potential html codes in the output to transparent html equivalents. If the Rexx exec wishes to control the html header and trailer information it can omit the CONVERT parm and supply its own html header and trailer records:

```
SD  DSN(…)   …                 (no CONVERT parm)
SNAPTEXT CHAR  html-header-codes
SNAPTEXT  CHAR html-header-codes
…
SNAP  …    HTMLTEXT              (various SNAP commands to copy data)
SNAPTEXT CHAR  html-trailer-codes
SNAPTEXT  CHAR html-trailer-codes
…
SNAPCLOS
```

The HTMLTEXT parm on the SNAP commands will cause SNAP to map any potential html codes in the output to transparent html equivalents.

Snap any allocated data set into the current snap target

A Rexx exec can use the new SNAPDD command to copy any allocated data set into the current snap target, making all of the necessary record format and carriage control conversions.

To copy the data set currently allocated to DD name INPUT into the current snap target data set:

```
SNAPDD  INPUT
```

Convert any sequential data set into html format

Using the features above, to convert a sequential data set to html format, an IOF Rexx exec can:

```
TSO ALLOC  DA(input.data)  FILE(INFILE) SHR
SD  DSN(output.data)  … CONVERT(html)
SNAPDD  INFILE
SNAPCLOS
```

It should be clear that there a number of powerful applications that can be built using this new technology.

Single line command to specify snap target and snap job

New shortcut line commands are provided that allow you to specify the snap target data set and snap a job with a single command.

To establish a disk snap target data set and snap a job just use the "SD" line command beside the job on the IOF Job List.

To establish a sysout snap target data set and snap a job just use the "SS" line command beside the job on the IOF Job List.

To establish an HFS file system snap target data set and snap a job just use the "SU" line command beside the job on the IOF Job List.

Create Base 64 Output for Binary Data

To copy a data set and convert it to base 64 format:

```
TSO ALLOC  DA(input.data)  FILE(INFILE) SHR
SD  DSN(output.data)  …
SNAPDD  INFILE  BASE64
SNAPCLOS
```

Simulate Carriage Control in SNAP Output

To use html codes to simulate carriage control when creating an html data set, specify "html" in the CC field of the SS, SD, or SU interface panels. In a Rexx exec, specify CC(html) on the SS, SD, or SU command that creates the snap target data set.

Specify "ascii" in the CC field of the SS/SD/SU interface panels or specify CC(ascii) on the SS/SDSU commands to map sysout carriage control into a format that is acceptable to a PC text browser.

For even more detailed control, a Rexx exec can explicitly specify the text strings to be used to simulate line feeds and form feeds in the output snap target data set. To accomplish this, the exec can specify the following on an SS, SD, or SU command:

```
 .... CC(TEXT)  FF(form-feed-string/skip-count)  LF(line-feed-string)
```

or

```
... CC(TEXT) FFHEX(hex-form-feed-string) LFHEX(hex-line-feed-string)
```

Access Attributes, Record Count, and Total Size of Open Snap Target

Rexx execs can now obtain information about the current snap target data set after it has been opened. There are session attributes that can be fetched with the TSICOPY command:

```
TSICOPY NAME(attr) FROM(SESSION) TO(REXX)
```

Here are the attributes:

SNAPDDNM	DD name of snap target data set
SNAPTYPE	Type of snap target (SS/SD/SU/SC)
SNAPCTYP	If SNAPTYPE is "SC", this is type of actual snap on client
SNAPLREC	LRECL of snap target
SNAPBLKS	BLKSIZE of snap target
SNAP$REC	# of records in latest snap
SNAP$BYT	# of bytes in latest snap
SNAP$MAX	Max lrecl in latest snap
SNAP#REC	Total records in snap target
SNAP#BYT	Total bytes in snap target
SNAP#MAX	Max lrecl in snap target

Defer Disposition of Sysout Snap Target Data Set

In Rexx execs is can be useful to create a sysout snap target data set but then decide to delete it if errors are encountered in the exec processing. To accomplish this, an exec can specify the DONTFREE parm on the SS command that defines the sysout snap target, which will prevent the eventual SNAPCLOS function from freeing the sysout data set.

A simple example:

```
address IOF
"SS ... DONTFREE"
"TSICOPY NAME(SNAPDDNM) FROM(SESSION) TO(REXX)"
...
various snap commands...
...
SNAPCLOS
```

To free the sysout data set and keep it:

```
address TSO "FREE FILE("snapddnm")"
```

To free the sysout data set and delete it:

```
address TSO "FREE FILE("snapddnm") DELETE"
```