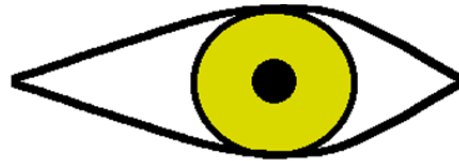# I Didn't Know I Could Do THAT with CANDE !!!

(Part 3 – Text and Queue Manipulation)

Universe 2014

Doug Dobson

GoldEye Software, Inc

I didn't know I could do that with CANDE (Part three – Command Queue Manipulation)

Prerequisite

Before any of this will work your CANDE administrator must enter the ?DEPTH command to set the number of commands that will be saved. The maximum is 20. So enter:

```
?DEPTH 20
```

Logon

```
#LX100:5260 CANDE SSR 56.1 (56.150.0012) AT MOODY
#YOU ARE CCF/JAVAP3001/1/CANDE/1(18)
#SESSION 6870  16:10:37 Monday, October 6, 2014 (PDT)
#
#FILE (DOBSOD561)CANDE/STARTUP ON WORK
#
  WFL MYJOB(JOBSUMMARY = SUPPRESSED,NOJOBSUMMARYIO = TRUE)
#RUNNING 6871
#BOT 6872  (DOBSOD561)WFLCODE
#EOT 6872  (DOBSOD561) (DOBSOD561)WFLCODE
#
  TERM DOWN 5; CONTINUE = \;  ESCAPE HEX = @ SQUASHED
#LINE = 80, PAGE = 30, MAXOUTPUT = 3984, MAXINPUT = 6000, SCREEN, WAIT,
    WRAPAROUND = FALSE, FORMSCAPABLE = FALSE, TRAILINGBLANKS = FALSE, DOWN =
    5, "\"
#CONTINUATION CHARACTER = \
#ESCAPE HEXADECIMAL = @ (SQUASHED)
```

My CANDE/STARTUP FILE looks like:

```
00000100  WFL MYJOB(JOBSUMMARY = SUPPRESSED,NOJOBSUMMARYIO = TRUE)
00000200  TERM DOWN 5; CONTINUE = \;  ESCAPE HEX = @ SQUASHED
```

The ?SHOW command

?SHOW ALL
```
 [4]DO CANDE/STARTUP*
 [3]  WFL MYJOB(JOBSUMMARY = SUPPRESSED,NOJOBSUMMARYIO = TRUE)*
 [2]  TERM DOWN 5;*
 [1]CONTINUE = \; *
 <0>ESCAPE HEX = @ SQUASHED*
#
```
Notice that CANDE split up the commands from line 200. As it executed each command it put it into my saved text queue.

The ?EDIT command

I want to change my escape character to "#" so I'll edit the last command.

?EDIT .@.#
```
ESCAPE HEX = # SQUASHED*
#
```
This command has an <edit spec> that says "replace the @ with #. Note that unlike the REPLACE command you don't repeat the delimiter between the <target text> and the <new text>. For documentation on the <edit specs> see the FIX command.

The ?REPEAT command

Now we can do the last command that we edited with the ?REPEAT command.

?REP
```
#
#ESCAPE HEXADECIMAL = # (SQUASHED)
```

We can combine the EDIT & REPEAT by putting the <edit specs> after the ?REPEAT. So let's put it back to @

?rep .#.@
```
ESCAPE HEX = @ SQUASHED*
#
#ESCAPE HEXADECIMAL = @ (SQUASHED)
```
Notice that it did the EDIT, displayed the result and then did the command.

The ?RETRIEVE command

The ?RETRIEVE command pulls items from the saved queue and moves them to the end. I'll move the TERM command at [2] to the end.

```
?SH ALL
 [4]DO CANDE/STARTUP*
 [3]  WFL MYJOB(JOBSUMMARY = SUPPRESSED,NOJOBSUMMARYIO = TRUE)*
 [2]  TERM DOWN 5;*
 [1]CONTINUE = \; *
 <0>ESCAPE HEX = @ SQUASHED*
#

?RET 2
#
?SH ALL
 [4]DO CANDE/STARTUP*
 [3]  WFL MYJOB(JOBSUMMARY = SUPPRESSED,NOJOBSUMMARYIO = TRUE)*
 [2]CONTINUE = \; *
 [1]ESCAPE HEX = @ SQUASHED*
 <0>  TERM DOWN 5;*
#
```

Now we can ?EDIT and ?REPEAT the TERM command just as we did with the ESCAPE command above.

?RETRIEVE, ?REPEAT and ?EDIT in one command

We can combine the retrieve and repeat (and edit) into one command. Let's change our continuation character [2] above.

```
?RET 2 REP '\'~
CONTINUE = ~; *
#
#CONTINUATION CHARACTER = ~
```

?BRK, ?DS and ?GO

If a command or a program fails the message "#QUEUED INPUT PENDING" is shown. Entering "?GO" will cause CANDE to resume with the next command. Entering a non-control command will discard everything in the queue.

If a program or command is terminated with ?DS the queue will be put into a pending state and can be restarted with the ?GO command. Many commands and some programs are setup to handle the ?BRK command. That will cause them to close their files and terminate gracefully. The queue will then continue processing the next command.

Consider the following DO file.

```
RUN CRASH/TEST;
FILES {SO+SECTORS}  ;
RUN  DELAY/TASKVALUE/SECS;VALUE=10;
```

```
FILES {SO+SECTORS};
RUN  DELAY/TASKVALUE/SECS;VALUE =1;

DO DO/CANDE/QUEUE/DEMO
#FILE (DOBSOD)DO/CANDE/QUEUE/DEMO ON WORK
#
 RUN CRASH/TEST;
#RUNNING 7242
#P-DS @ 00000350.
#ET=0.0 PT=0.0 IO=0.0
# QUEUED INPUT PENDING
?SH A
 [6]DO CANDE/STARTUP*
 [5]  WFL MYJOB(JOBSUMMARY = SUPPRESSED,NOJOBSUMMARYIO = TRUE)*
 [4]  TERM DOWN 5;*
 [3]CONTINUE = \; *
 [2]ESCAPE HEX = @ SQUASHED*
 [1]do DO/CANDE/QUEUE/DEMO*
 <0> RUN CRASH/TEST;*
D(1) FILES {SO+SECTORS}  ;*
D(2) RUN  DELAY/TASKVALUE/SECS;VALUE=10;*
D(3) FILES {SO+SECTORS};*
D(4) RUN  DELAY/TASKVALUE/SECS;VALUE =1;*
#
```

After the run failed the queue stopped. We can see what is in the queue with the ?SHOW ALL
command. Now continue the queue with ?GO. When the FILES command shows the first page kill it ?DS
and show the queue.

```
?GO
#
 FILES {SO+SECTORS}  ;
= ON WORK {SO + SECTORS}

      File Name            Filekind     Records      Sectors    CreationTime
 --------------------------+------------+------------+------------+----------
 FORM/BAE                DATA                  0        0  04/26/2005
 … ETC.

?DS
#
# QUEUED INPUT PENDING
```

Now continue with ?GO. When the files command shows the first screen terminate it with ?BRK. The
next command will be done.

```
?go
#
 RUN  DELAY/TASKVALUE/SECS;VALUE=10;
```

```
#RUNNING 7243
#ET=10.0 PT=0.0 IO=0.0
 FILES {SO+SECTORS};

?BRK
.ok.
#
 RUN  DELAY/TASKVALUE/SECS;VALUE =1;
#RUNNING 7244
#ET=1.0 PT=0.0 IO=0.0
```

Forcing the queue to stop (?WAIT)

While a command or task is executing you can enter ?WAIT. The queue will enter a waiting state when the current command or task completes. You can now manipulate the queue as you wish.

Other fun stuff (?ENTER, ?PURGE, ?TAKE)

?PURGE does what you would expect it to do.

?ENTER allows creating new entries or editing existing entries in either the saved text or the input queue.

?TAKE works similar to ?RETRIEVE except on the input queue rather than the saved text. It allows you to reorder or edit items in the input queue.