

Fun with QSH, call qp2term and RPG ...

Full examples at ...

<http://www.youngiprofessionals.com/wiki/index.php/PASE/SHELL>

Author(s)

Tony “Ranger” Cairns - IBM i PHP / PASE

Agenda



QSH/QP2TERM (shells)

No F4 prompt, but nice pipes w/tricks.



QSH/QP2TERM (parms)

system call lib/mbr, or make my own utilities w/RPG.



QSH/QP2TERM (web)

All runs on the web.



Questions?

Before you start ...

PASE runs ASCII and ILE runs EBCDIC, interactions require a valid CCSID

DSPSYSVAL SYSVAL(QCCSID)

If dspsysval finds your machine 65535 ...

- * CHGSYSVAL SYSVAL(QCCSID) VALUE(37) -- change entire machine
- * CHGUSRPRF USRPRF(ME) CCSID(37) -- change your user profile

What happened IBM???

IBM i's history of 65535 is simply a good world wide setting for years pre-dating PASE arrival on the machine. i suggest moving away from 65535 is needed in a modern world (most clients ASCII), but you can get around to changing your machines in a few more decades by using CHGUSRPRF in meantime.

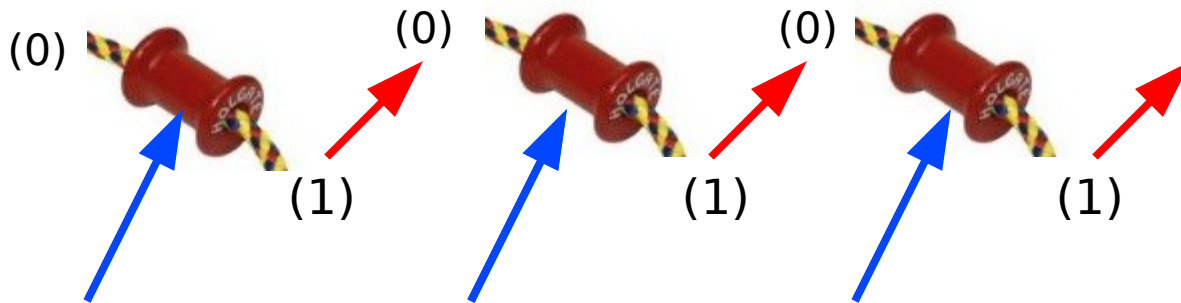
QSH/QP2TERM (shells)

QSH/QP2TERM vs. QCMD:

- * No QCMD PF4 prompting.
- * No QCMD style parameter passing.

QSH/QP2TERM/shells are about pipes ...

- * (0) STDIN/input pipe
- * (1) STOUT/output pipe
- * (2) STDERR/output pipe



> `system wrkactjob | grep -i php-cgi | grep TIMW`

```

xt5250 - lp0264d
QSH Command Entry
$
> /bin/system wrkactjob | /QOpenSys/usr/bin/grep -i php-cgi | /bin/grep TIMW |
/QOpenSys/usr/bin/grep -i bcl
ZEND2 0 QTMHHTTP 797534 QTMHHTTP BCI 2 25 .0
      .0 PGH-php-cgi.bi TIMW 1
ZEND2 0 QTMHHTTP 797535 QTMHHTTP BCI 2 25 .0
      .0 PGH-php-cgi.bi TIMW 1
ZEND2 0 QTMHHTTP 797536 QTMHHTTP BCI 2 25 .0
      .0 PGH-php-cgi.bi TIMW 1
ZEND2 0 QTMHHTTP 797537 QTMHHTTP BCI 2 25 .0
      .0 PGH-php-cgi.bi TIMW 1
ZEND2 0 QTMHHTTP 797538 QTMHHTTP BCI 2 25 .0
      .0 PGH-php-cgi.bi TIMW 1
ZEND2 0 QTMHHTTP 797539 QTMHHTTP BCI 2 25 .0
      .0 PGH-php-cgi.bi TIMW 1
==>
F3=Exit F6=Print F9=Retrieve F12=Disconnect
F13=Clear F17=Top F18=Bottom F21=CL command entry
5250 007/018
    
```

QSH/QP2TERM (IBM i tricks)

Trick #1) PASE loader detects ILE.PGM, appends /QOpenSys ...

```
> /bin/system (SYSTEM.PGM) <PASE loader runs> /QOpenSys/bin/system
```

```
> /bin/grep (GREP.PGM) <PASE loader runs> /QOpenSys/bin/grep
```

Trick #2) PASE wrapper script qsh (qsh_in, qsh_out) ...

```
> qsh -c '/QSYS.LIB/QSHELL.LIB/SYSTEM.PGM dsplibl' | grep -i QS
```

```
QSYS          SYS
```

```
System Library
```

```
QSYS2         SYS
```

```
System Library for CPI's
```

Trick #3) QSH detects PASE program, calls PASE loader ...

```
> easy fred flinstone
```

```
0 /www/zendsvr/htdocs/Samples/SHELL_use/easy - program (argv[0])
```

```
1 fred -- parameter 1 (argv[1])
```

```
2 flinstone -- parameter 2 (argv[2])
```

QSH/QP2TERM (scripting)

Golden rule: ALWAYS provide shebang line 1 (!#)
(shebang for PASE loader and/or qsh know what shell to run)

call qp2term or QSH or ssh -X ibmi ...

```
> wrkactjob-filter1.sh
```

```
#!/Q0penSys/usr/bin/sh
```

```
/bin/system wrkactjob | \
```

```
/Q0penSys/usr/bin/grep -i php-cgi | \
```

```
/bin/grep TIMW | \
```

```
/Q0penSys/usr/bin/grep -i bci
```



```
Type command, press Enter.
```

```
==> IBMJOB CMD(QSH CMD('/home/adc/wrkactjob-filter1.sh'))
```

QSH/QP2TERM (parms)

shell tricks that join shells with the world of QCMD ...

```
> system "call MYLIB/MYPGM LAND('BEDROCK') BEFORE('FRED') TIME('16:53')"
```

Various IBM i commands include OUTPUT(*) for STDOUT ...

```
> system wrkactjob | grep -i php-cgi
      ZEND2          QTMHHTTP      798311      QTMHHTTP      BCI
```

Example: Using *SECOFR profile to DSPJOBLOG JOB() of running java jobs on IBM i system

call qp2term

```
> ps -ef | grep -i java | awk '{print "qsh_out -c \'''/usr/bin/getjobid "$2'\''"}' | sh | awk '{print "qsh_out -c \"system -i \'''/usr/bin/dspjoblog job(\"$5\")'\''\""}' | sh > test.txt
```

```
> head test.txt
```

```
5761SS1 V6R1M0 080215          Display Job Log          LP0264D 02/28/13 14:40:46          Page 1
Job name . . . . . : QSRVMON          User . . . . . : QSYS          Number . . . . . : 552621
Job description . . . . . : QSRVJOB          Library . . . . . : QSYS
```

see command awk is building leave off | sh ...

```
> ps -ef | grep -i java | awk '{print "qsh_out -c \'''/usr/bin/getjobid "$2'\''"}' | sh | awk '{print "qsh_out -c \"system -i \'''/usr/bin/dspjoblog job(\"$5\")'\''\""}'
qsh_out -c "system -i 'dspjoblog job(552621/QSYS/QSRVMON)'"
```

keep chopping off pipes '|' to understand the whole script line ...

```
> ps -ef | grep -i java | awk '{print "qsh_out -c \'''/usr/bin/getjobid "$2'\''"}'
qsh_out -c '/usr/bin/getjobid 20'
```

```
> ps -ef | grep -i java
```

```
qsecofr      20      17      0      Aug 05      - 20:57 /QOpenSys/QIBM/ProdData/JavaVM/jdk50/32bit/jre/bin/jvmStartPase 703
```

QSH/QP2TERM (my own utility parms)

```
> system -i 'call pgm(qgpl/easyrpg) parm('\''1\'\' '\''2\'\' '\''3\'\' '\''4\'\' '\''5\'\' '\''6\'\' '\''7\'\'')'
| awk '{print "-"$1"-"$2"-"$3"-"$4}'
```

<pre>system -i 'call pgm(qgpl/easyrpg) Parm('\''1\'\' '\''2\'\' '\''3\'\' '\''4\'\' '\''5\'\' '\''6\'\' '\''7\'\')'</pre>		<pre>D argv_t DS D D argv * D argc S 10i 0 D myArgv ds D Main pr D Parm01 48a D Parm02 48a D Parm03 48a D Parm04 48a D Parm05 48a D Parm06 48a D Parm07 48a D Parm08 48a argc = %parms; if argc >= 1; myArgv.argv(1) = %addr(Parm01); endif; if argc >= 2; myArgv.argv(2) = %addr(Parm02); Endif; :</pre>	<pre>qualified based(Template) * dim(8) inz(0) likeds(argv_t) extpgm('EASYRPG')</pre>
---	--	---	---

QSH/QP2TERM (my own utility pipes)

```
> system -i 'call pgm(qgpl/easyrpg) parm('\''1\'\' '\''2\'\' '\''3\'\' '\''4\'\' '\''5\'\' '\''6\'\' '\''7\'\'')'
| awk '{print "-"$1-"-$2-"-$3-"-$4}'
```

shells are about pipes ...

* (0) STDIN/input

* (1) STOUT/output

* (2) STDERR/output

* STDIN

```
D readIFS          PR          20I 0 ExtProc('read')
D fd               10I 0 value
D buf              * value
D size             10I 0 value
```

```
rSz = readIFS(0:pos0:%size(buf));
dow rSz > 0;
  rSz = readIFS(0:pos0:%size(buf));
enddo;
```

* STDOUT, STDERR

```
D writeIFS         PR          20I 0 ExtProc('write')
D fd               10I 0 value
D buf              * value
D size             10I 0 value
```

```
myFmt =%trim(%str(myArgv.argv(i):48)) + LF + x'00';
wSz = writeIFS(1:%addr(myFmt):strlen(myFmt));
```

(0)



(1)

QSH/QP2TERM (my own utility link convenience)

QSH – EASYRPG.PGM make a symbolic link for easy command processing ...

```
> ln -sf /qsys.lib/qgpl.lib/EASYRPG.PGM /usr/bin/easyrpg
> chmod 0705 /qsys.lib/qgpl.lib/EASYRPG.PGM
```

call qp2term – easyrpg PASE wrapper ...

```
> chmod 0755 /QOpenSys/usr/bin/easyrpg
> cat /QOpenSys/usr/bin/easyrpg
#!/QOpenSys/usr/bin/ksh
# map base name "qsh_inout" to "qsh" (just run the interpreter)
basename=${0##*/}
if [[ "$basename" = qsh_inout ]] ; then
    basename=qsh
fi
args=''
for temp ; do
    args="$args'"
    # double embedded quotes halved by CL CALL
    while [[ "$temp" = *\'* ]] ; do
        args="$args${temp%*\'}'"
        temp="${temp#*\'}"
    done
    args="$args$temp' "
    shift
done
# Copy environment variables, call utility in this process,
# do not process OS/400 messages or spooled output files
exec /QOpenSys/usr/bin/system -eiqs "CALL QSYS/QP0ZCALL ('/usr/bin/$basename' $args)"
```

QSH/QP2TERM easyrpg available everywhere ...

```
> qsh -c '/usr/bin/easyrpg i am really glad "to finally understand" shell programming' | awk  
'{print "-"$1 "-"$2 "-"$3 "-"$4}'
```

```
> system -i 'call pgm(qgpl/easyrpg) parm('\''i'\'' '\''am'\'' '\''really'\'' '\''glad'\''  
'\''to finally understand'\'' '\''shell'\'' '\''programming'\''')' | awk '{print  
"-"$1 "-"$2 "-"$3 "-"$4}'
```

```
> easyrpg i am really glad "to finally understand" shell programming | awk '{print  
"-"$1 "-"$2 "-"$3 "-"$4}'
```

```
> ls /home/adc | grep -i zz | easyrpg -pipe | grep sh
```

QSH/QP2TERM (web)

PASE CGI trick httpd.conf (any instance) ...

* IBM i Apache HTTP server has been trained to run PASE CGI(s) directories starting with /QOpenSys. This is the only trick you have to learn to get this whole thing on the web.

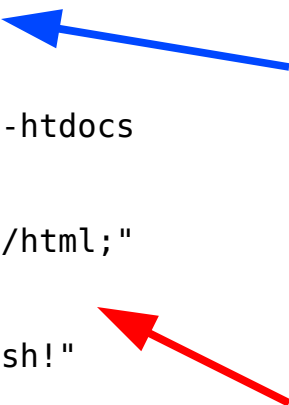
```
> mkdir -p /QopenSys/sh-bin/sh-htdocs /myasp2/www/zend2/conf/httpd.conf

> cd /QopenSys/sh-bin
> cat sh-cgi
#!/QopenSys/usr/bin/sh
exec $PATH_TRANSLATED

> cd /QopenSys/sh-bin/sh-htdocs
> cat hello.sh
#!/QopenSys/usr/bin/sh
echo "Content-Type: text/html;"
echo ""
echo ""
echo "Hello, world from sh!"

> chmod -R 0755 /QopenSys/sh-bin
```

```
# PASE<>ILE string conversion requires CCSID
# invalid QCCSID 65535 (hex, binary, evil)
DefaultFsCCSID 37
CGIJobCCSID 37
# sh cgi wrapper program
ScriptAlias /sh-bin/ /QopenSys/sh-bin/
AddType application/x-httpd-sh .sh
Action application/x-httpd-sh /sh-bin/sh-cgi
<Directory /QopenSys/sh-bin>
Options +ExecCGI
order allow,deny
allow from all
</Directory>
# sh applications (.sh)
Alias /sh-htdocs /QopenSys/sh-bin/sh-htdocs
<Location /sh-htdocs>
Order deny,allow
Allow from all
</Location>
```



References

Full examples at ...

<http://www.youngiprofessionals.com/wiki/index.php/PASE/SHELL>