

What's New in System APIs

Bruce Vining
bvining@brucevining.com

In This Session ...

- This session, which is intended for application programmers, covers new and changed key system APIs that were introduced to the i operating system in V5R4 through 7.1
 - 7.1: @ 50 new, @ 80 changed
 - V6R1: @170 new, @ 250 changed
 - V5R4: @ 90 new, @ 170 changed
- Examples shown will be using free form RPG

What We'll Cover ...



- Major New V5R4 APIs
- Major New V6R1 APIs
- Major New 7.1 APIs
- Overview of Additional New and Changed
- Wrap-up

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Message Watching



- Watch technology provides the ability to be notified in real time when a problem occurs
- Identify what the problem is
 - The symptom of a potential problem
 - 'Problem' can be
 - A message and where it is logged
 - A LIC Log
- Dynamically define (add or remove) the watch points
- When the problem occurs get notified via an exit point

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Message Watching



- STRWCH, ENDWCH, WRKWCH command
- QSCSWCH, QSCEWCH APIs
- Watch for messages (V5R4) on:
 - QSYSOPR
 - QHST
 - *MSGQ
 - Job
 - Specific job
 - Generic job (VIN*)
 - *ALL
 - Conditioned on replacement data (V5R4), Msg type and/or severity (7.1)
- Watch for LIC Log entry (V5R4) on:
 - *ALL major and minor codes (but not *ALL *ALL)
 - Wildcard major and minor codes
 - Specific major and minor codes
 - Conditioned on replacement data
- Watch for PAL entry (V6R1) on:
 - *ALL System Reference Codes (SRCs)
 - Wildcard and Generic SRCs
 - Conditioned on replacement data – Resource name, type, or model

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Message Watch Exit Program



- Reason for call (Input)
 - *MSGID (V5R4)
 - *LICLOG (V5R4)
 - *PAL (V6R1)
 - *STRWCH (V6R1)
 - *ENDWCH (V6R1)
- Session ID (Input)
- Return code (Output)
 - Error
- Problem data (Input):
 - MSGID
 - Message queue
 - Job
 - Sending program/procedure
 - Receiving program/procedure
 - Replacement data
 - etc

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Message Watching Asynchronous Notification



- Problem detection/isolation outside of executing job
 - Runs concurrent to job experiencing problem (good and bad)
 - No need to poll watched job (good performance)
 - Job Interrupt API (discussed next)
- Sample scenario
 - CPF2456 - Log version &1 in &2 closed and should be saved.
 - Automate saving of log to *SAVF

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Message Watch Getting Started



- Start the Watch
 - STRWCH SSNID(HISTORY) WCHPGM(SAVELOG)
WCHMSG(CPF2456) WCHMSGQ(*HSTLOG)
- Work with Watches (WRKWCH *ALL)

Work with Watches

Type options, press Enter.
1=Start 2=End

Opt	Type/Session	Origin	User	Status
	STRWCH			
	HISTORY	STRWCH	VINING	ACTIVE

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Exit Program Data Definitions



```

dSaveLog      pr
d Type        10
d Session     10
d Error       10
d Data        likeds(DataDS)

dSaveLog      pi
d Type        10
d Session     10
d Error       10
d Data        likeds(DataDS)

dExcCmd       pr
d Cmd         100
d CmdLen      15 5
               const

dDataDS       ds      1000
d MsgID       5      11
d OffRplDta   441    444i 0

dMsgDtaPtr    s      *
dMsgDta       ds      75  based(MsgDtaPtr)
d File        1      10
d Lib         11     20

dCmd          s      80

```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Message Watch Exit Program



```

/free

if Type = '*MSGID';                                // Make sure message watch

    if Data.MsgID = 'CPF2456';                      // Make sure for CPF2456
        MsgDtaPtr = %addr(Data) + Data.OffRplDta;
        Cmd = 'SAVOBJ OBJ(' + File + ') LIB(' + Lib +
            ') DEV(*SAVF) SAVF(QGPL/MYSAVES)';
        ExcCmd(Cmd :%len(%trimr(Cmd)));             // Save the log
    else;                                           // Not CPF2456
        dsply ('Wrong Msg:' + Data.MsgID);
    endif;

else;                                           // Not a *MSGID watch
    dsply ('Wrong Type:' + Type);
endif;

Error = *blanks;
return;
*inlr = *on;

/end-free

```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Call Job Interrupt Program (QWCJBITP)



- Allows you to run an exit program in the initial thread of another job
 - See what's in the QTEMP library
 - Update objects such as a *DTAARA in QTEMP
 - Basically run whatever the current caller of the API is authorized to
 - Can be very dangerous when in the wrong hands....
- New system value QALWJOBITP
 - Never allow
 - Default new jobs to not allow
 - Default new jobs to allow
- New system exit point QIBM_QWC_JOBITPPGM
- New Change Job Interrupt Status (QWCCJITP) API
- Sample scenario – run commands in another job

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Call Job Interrupt Sample Command



```

CMD          PROMPT('Call Job Interrupt Program')
PARM         KWD(JOB) TYPE(Q1) PROMPT('Job name')
PARM         KWD(CMD) TYPE(*CMDSTR) LEN(512) MIN(1) +
              PROMPT('Command')
Q1:          QUAL TYPE(*NAME) LEN(10) MIN(1)
              QUAL TYPE(*NAME) LEN(10) MIN(1) PROMPT('User')
              QUAL TYPE(*CHAR) LEN(6) RANGE(000000 999999) +
                  MIN(1) FULL(*YES) PROMPT('Number')
  
```

Call Job Interrupt Program (CALLJOB)

Type choices, press Enter.

Job name	_____	Name
User	_____	Name
Number	_____	000000-999999
Command	_____	

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Call Job Interrupt Sample CPP



```

dJobInt      pr      extpgm('JOBINT')
d Job        26      const
d Cmd        512     const

dJobInt      pi
d Job        26      const
d Cmd        512     const

dInterruptJob pr      extpgm('QWCJBITP')
d Controls   likeds(Controls) const
d CtlFormat  8        const
d ErrorCode  10i 0    const

dControls    ds
d PgmName    10      inz('RUNCMD')
d PgmLib     10      inz('VINING')
d TgtJob     26
d Reserved   2        inz(*ALLx'00')
d OffPgmDta  10i 0
d LenPgmDta  10i 0
d PgmDta     512

```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Call Job Interrupt Sample CPP



```

/free

TgtJob = Job; // Set target Job ID

LenPgmDta = %len(%trim(Cmd)); // Set length of command
// to execute

PgmDta = %subst(Cmd :1 :LenPgmDta); // Set command to execute
OffPgmDta = (%addr(PgmDta) - %addr(Controls)); // and offset to cmd

InterruptJob( Controls : 'JITP0100' : 0); // Call the program in
// the target job

*inlr = *on;
return;

/end-free

```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Code related to calling a command



```

dRunCmd      pr      extpgm('RUNCMD')
d PgmDta      512      const
d LenPgmDta    10i 0    const

dRunCmd      pi
d PgmDta      512      const
d LenPgmDta    10i 0    const

dProcessCmd   pr      extpgm('QCAPCMD')
d PgmDta      512      const
d LenPgmDta    10i 0    const
d Options      const liked(Options)
d LenOptions    10i 0    const
d Format        8      const
d NewCmd        1
d LenAvlNewCmd  10i 0    const
d LenNewCmd     10i 0
d ErrorCode     10i 0    const

dOptions      ds
d CmdType      10i 0    inz(0)
d DBCS         1      inz('0')
d Prompter     1      inz('0')
d Syntax       1      inz('0')
d MsgKey       4      inz(' ')
d CCSID        10i 0    inz(0)
d              5      inz(*ALLx'00')
```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Exit Program Main Function



```

dNewCmd      s      1
dLenNewCmd    s      10i 0

/free

ProcessCmd( PgmDta :LenPgmDta :Options :%size(Options) :'CPOP0100'
           :NewCmd :0 :LenNewCmd :0);

*inlr = *on;
return;

/end-free
```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Call Job Interrupt Sample Results



- Target job
 - DSPDTAARA QTEMP/TEST
 - Returns a value of 'A'
- Source job
 - CALLJOB JOB(223306/VINING/QPADEV000C)
CMD(CHGDTAARA DTAARA(QTEMP/TEST)
VALUE('S'))
- Target job
 - DSPDTAARA QTEMP/TEST
 - Returns a value of 'S'

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Enhanced Integration at the User Interface



- User Customized Options for WRKOUTQ, WRKSPLF, D P

```

Session A - [24 x 80]
File Edit View Communication Actions Window Help
Work with Output Queue

Queue:  QPRINT      Library:  QGPL      Status:  RLS

Type options, press Enter.
1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
8=Attributes   9=Work with printing status

Opt  File      User      User Data  Sts  Pages  Copies  Form Type  Pty
--  -
-   TST00      SUNYCDL  CRTPGM    R0Y   146    1    *STD      5
-   QPSECPWD  DHQB     ANZDFTPW  R0Y    1     1    *STD      5
P   SPLEXIT    VINING   R0Y     16    1    *STD      5
-   QPRINT     QPMGR    OPN      0     1    *STD      5

Parameters for options 1, 2, 3 or command
==>
F3=Exit  F11=View 2  F12=Cancel  F20=Writers  F22=Printers
F24=More keys

MB a MW 13/008
1902 - Session successfully started
  
```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

User Defined Options on Selected IBM Panels



- Options A through Z can be defined
 - 'A' for Archive
 - 'P' for Priority or Print Next
 - Pick your own...
- Exit programs registered using Registration Facility
 - ADDEXITPGM EXITPNT(QIBM_QSP_SPLF_LSTACT)
FORMAT(LASP0100) PGMNBR(*LOW)
PGM(VINING/SPLEXIT) PGMDTA(*JOB 1 P)
 - One exit program per exit point option (P in this example)
 - 26 exit programs per exit point (A to Z)
- Function is actually within UIM
 - Parameters to pass are defined in UIM source
 - Only current user is QIBM_QSP_SPLF_LSTACT (Spool List Action)
 - Let IBM know if you believe this should be extended to other areas



Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Parameters Passed IBM_QSP_SPLF_LSTACT



- Exit Point Name – QIBM_QSP_SPLF_LSTACT
- Exit Point Format – LASP0100
- Action – for example option P (Priority and/or Print Next)
- Spool Identifiers – the exit program parameter that will vary across exits
- Length of spool identifiers
- Spool Identifiers:
 - Job name
 - Job user
 - Job number
 - Spool file name
 - Spool file number
 - Job system name
 - Date and time created
 - Qualified output queue name

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

SPLEXIT Program in RPG: Support Option 'P' for Prioritize Spooled Output



■ The Prototypes:

```

dSplExit          pr                      ExtPgm('SPLEXIT')
d ExitPointName    20                    const
d Format           8                      const
d Action           1                      const
d SpoolID          likeds(espp0100)
d SpoolIDLen       10i 0 const

dSplExit          pi
d ExitPointName    20                    const
d Format           8                      const
d Action           1                      const
d SpoolID          likeds(espp0100)
d SpoolIDLen       10i 0 const

dMovSplf          pr                      ExtPgm('QSPMOVSP')
d MoveInfo         65535                 const options(*varsize)
d MoveInfoLen      10i 0 const
d MoveInfoFormat   8                      const
d QUSEC            likeds(qusec)

```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Support Option 'P' for Prioritize Spooled Output



■ Prototype continued, QSYSINC Includes, Variables

```

dSndPgmMsg        pr                      extpgm('QMHSNDPM')
d MsgID           7                      const
d QualMsgF        20                    const
d MsgDta          65535                 const options(*varsize)
d LenMsgDta       10i 0 const
d MsgType         10                    const
d CallStackEntry  65535                 const options(*varsize)
d CallStackCntr   10i 0 const
d MsgKey          4                      const
d QUSEC            likeds(qusec)
d LenCSE          10i 0 const options(*nopass)
d CSEQual         20                    const options(*nopass)
d DSPWaitTime     10i 0 const options(*nopass)
d CSEType         10                    const options(*nopass)
d CCSID           10i 0 const options(*nopass)

d/copy qsysinc/qrpglesrc,espxpts
d/copy qsysinc/qrpglesrc,qspmovsp
d/copy qsysinc/qrpglesrc,qusec

dMsgText          s                      80
dMsgKey           s                      4
dWait             s                      1

```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

QSYSINC Definition Format LASP0100



DESPP0100	DS			Qsp LASP0100 t
D*				
D ESPJN00		1	10	Job Name
D*				
D ESPUN00		11	20	User Name
D*				
D ESPJNBR00		21	26	Job Number
D*				
D ESPSN04		27	36	Spfl Name
D*				
D ESPSNBR02		37	40B 0	Spfl Number
D*				
D ESPJSN03		41	48	Job System Name
D*				
D ESPSCD03		49	55	Date in CYMMDD format
D*				
D ESPSCT03		56	61	Time in HHMMSS format
D*				
D ESPOQN02		62	71	Output Queue Name
D*				
D ESPOQL02		72	81	Output Queue Lib
D*				

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

QSYSINC Definition QSPMOVSP Format MSPF0100



DQSPF0100	DS			Qsp MSPF0100
D*				
D QSPSJN01		1	10	Source Job Name
D*				
D QSPSJUN01		11	20	Source Job User Name
D*				
D QSPSJNBR01		21	26	Source Job Number
D*				
D QSPSIJID01		27	42	Source Internal Job ID
D*				
D QSPISFID		43	58	Source Internal Spool File ID
D*				
D QSPSSFN		59	68	Source Spooled File Name
D*				
D QSPSSFN00		69	72B 0	Source Spooled File Number
D*				
D QSPTOQN		73	82	Target Output Queue Name
D*				
D QSPTOQL		83	92	Target Output Queue Library
D*				
D QSPSJSN		93	100	Source Job System Name
D*				
D QSPSSCD		101	107	Source Spfl Create Date
D*				
D QSPERVED06		108	108	Reserved
D*				
D QSPSSCT		109	114	Source Spfl Create Time
D*				

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

SPLEXIT Program The Code



■ Getting The Job Done

```
/free
if ExitPointName = 'QIRM_QSP_SPLF_LSTACT';
  if Format = 'LASP0100';
    if Action = 'P';
      // Priority request?
      QSPSJN01 = SpoolID.ESPJN00; // Job Name
      QSPSJUN01 = SpoolID.ESPUN00; // User Name
      QSPSJNBR01 = SpoolID.ESPJNBR00; // Job Number
      QSPSIJID01 = *Blanks; // Internal Job ID
      QSPISFID = *Blanks; // Internal Spool ID
      QSPSSFN = SpoolID.ESPSPN04; // Spool File Name
      QSPSSFN00 = SpoolID.ESPSPNBR02; // Spool File Number
      QSPTOQN = 'SAME'; // Target Output Queue
      QSPTOQL = *Blanks; // Target Output Queue Lib
      QSPSJSN = SpoolID.ESPJSN03; // Job System Name
      QSPSSCD = SpoolID.ESPSCD03; // Spool File Create Date
      QSPSSCT = SpoolID.ESPSCD03; // Spool File Create Time
      QUSBRPV = 0; // Error code Bytes Provided
      MovSplf( QSPF0100 :%size(QSPF0100) :MSPF0100 :QUSEC);
      MsgText = 'Spool file ' + %trimr(QSPSSFN) +
        ' moved to top of output queue';
      SndPgmMsg( 'CPF9898' :QCPFMSG *LIBL'
        :MsgText :%len(%trimr(MsgText))
        :'*COMP' :'*PGMBDY' :1 :MsgKey :QUSEC);
    endif;
  endif;
endif;
```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

More of the code



■ Error-Related Logic

```
else;
  dsply ('Called with unexpected option:' + Action) ' ' Wait;
endif;
else;
  dsply ('Called with unexpected format:' + Format) ' ' Wait;
endif;
else;
  dsply 'Unexpected caller' ' ' Wait;
endif;
*inlr = *on;
return;
/end-free
```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

After the Exit Program Runs Completion Message Shown on Panel



```

Session A - [24 x 80]
File Edit View Communication Actions Window Help
Work with Output Queue

Queue: QPRINT      Library: QGPL      Status: RLS

Type options, press Enter.
1=Send 2=Change 3=Hold 4=Delete 5=Display 6=Release 7=Messages
8=Attributes 9=Work with printing status

Opt File      User      User Data  Sts  Pages  Copies  Form  Type  Pty
--  ---      -
--  TST00     SUNVCDL  CRTPGM    RDY   146     1     *STD   5
--  QPSECWD    DH0B     ANZDFTPWD RDY    1     1     *STD   5
--  SPLEXIT     VINING   RDY    16     1     *STD   5
--  QPRINT     QPGMR    OPN     0     1     *STD   5

Parameters for options 1, 2, 3 or command
====>
F3=Exit  F11=View 2  F12=Cancel  F20=Writers  F22=Printers
F24=More keys
Spool file SPLEXIT moved to top of output queue.

MA a MW 12/003
1902 - Session successfully started
  
```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Set User Function: More Meaningful Operator Information

- Just What is PGM - P18A54R?



```

Session B - [24 x 80]
File Edit View Communication Actions Window Help
Work with Active Jobs

CPU %: .6 Elapsed time: 00:00:03 Active jobs: 191 LP12UT12 08/26/07 08:18:59

Type options, press Enter.
2=Change 3=Hold 4=End 5=Work with 6=Release 7=Display message
8=Work with spooled files 13=Disconnect ...

Opt Subsystem/Job User      Type CPU % Function      Status
--  ---
--  QINTER      QSYS      SBS .0 DEQW
--  QPADEV000N  VINING    INT .0 PGM-P18A54R  DSPW
--  QPADEV000P  VINING    INT .3 CMD-WRKACTJOB RUN
--  QSERVER      QSYS      SBS .0 DEQW
--  QPWFSERVSD  QUSER     BCH .0 SELW
--  QSERVER      QPGMR     ASJ .0 EVTW
--  QZDASRVSD   QUSER     BCH .0 SELW
--  QZLSFILET    QUSER     PJ .0 TIMW
--  QZLSSERVER  QPGMR     BCH .0 EVTW

Parameters or command
====>
F3=Exit  F5=Refresh  F7=Find  F10=Restart statistics
F11=Display elapsed data  F12=Cancel  F23=More options  F24=More keys

MA b MW 15/031
1902 - Session successfully started
  
```

Set User Function: More Meaningful Operator Information (cont.)

- Just What is PGM – P18A54R? Change to say 'Payroll'

```

Session B - [24 x 80]
File Edit View Communication Actions Window Help

Work with Active Jobs                                08/26/07  LP12UT12
CPU %: .5 Elapsed time: 00:08:39 Active jobs: 191 08:27:35

Type options, press Enter.
2=Change 3=Hold 4=End 5=Work with 6=Release 7=Display message
8=Work with spooled files 13=Disconnect ...

Current
Opt Subsystem/Job User Type CPU % Function Status
---
0INTER QSYS SBS .0 DEOW
0PADEV000N VINING INT .0 USR-Payroll DSPW
0PADEV000P VINING INT .0 CMD-WRKACTJOB RUN
QSERVER QSYS SBS .0 DEOW
QPWFSESRVSD QUSER BCH .0 SELW
QSERVER QPGMR ASJ .0 EVTW
QZDASRVSD QUSER BCH .0 SELW
QZLSFILET QUSER PJ .0 TIMW
QZLSSERVER QPGMR BCH .0 EVTW

Parameters or command
==>
F3=Exit F5=Refresh F7=Find F10=Restart statistics
F11=Display elapsed data F12=Cancel F23=More options F24=More keys

More...

MF b MW 10/002
3902 - Session successfully started

```

Set User Function: More Meaningful Operator Information with Change Current Job API QWCCCJOB

- Prototype and Data

```

dChgCurJob      pr               extpgm('QWCCCJOB')
d ChgInf         65535            const options(*varsize)
d QUSEC          liked(QUSEC)

d/copy qsysinc/qrpglesrc,qusec

dChgInf         ds
d NbrAtrs        10i 0
d KeyAtr         10i 0
d LenKeyDta      10i 0
d KeyDta         10

dWait           s                1

```



Set User Function: More Meaningful Operator Information with Change Current Job API

▪ Making the Change

```
/free
QUSBPRV = 0;           // API to send escapes
NbrAtrs = 1;           // Change one job value
KeyAtr = 3;            // Change function name
LenKeyDta = 10;         // Function name is 10 bytes
KeyDta = 'Payroll';     // Current application is Payroll
ChgCurJob( ChgInf :QUSEC); // Make the change
dsply 'I am off running Payroll' ' ' Wait;

*inlr = *on;
return;
/end-free
```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved



Set User Function: Change Current Job (QWCCCJOB)

- Two New Keys Introduced
 - 3 – Set Function Name
 - Function type will be USR
 - Function name will be determined by caller
 - 4 – Reset Function Type and Name
 - Function type and name will be set to blanks
- Function is Updated When Calling System Functions
- Function is Retained When Calling Application Functions

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Message Handler: New Inquiry Message Exit Point



- **Eliminate** operator involvement
- Exit point QIBM_QMH_HDL_INQEXT
- Works with *EXT job message queue
- Runs after job INQMSGRPY processing
 - *DFT – use default message reply
 - *SYSRPYL – use system reply list
- Intercepts normal display of Display Program Messages screen
 - Any number of exit programs can be registered
 - First exit program to respond to message ends calling of exit programs
 - If no exit program responds, system displays Display Program Messages screen
 - Send Reply API (QMHSNDRM) enhanced to support *EXT message queue
- Exit program(s) registered to QIBM_QMH_REPLY_INQ continue to run
 - Reject response
 - Replace response
 - Etc.

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Inquiry Message Exit Point – Conditional Responses



Prototypes

```

dCatchINQ      pr                      ExtPgm ('CATCHINQ')
d CallType      10i 0 const
d QualMsgQName  20  const
d MsgKey        4  const
d MsgID         7  const

dCatchINQ      pi
d CallType      10i 0 const
d QualMsgQName  20  const
d MsgKey        4  const
d MsgID         7  const

dSndRply       pr                      extpgm ('QMHSNDRM')
d MsgKey        4  const
d QualMsgQName  20  const
d ReplyText     65535 const options(*varsize)
d LenReplyText  10i 0 const
d RmvMsg        10  const
d QUSEC         10  const
d CCSID         10i 0 const options(*nopass)
d AllowReject   10  const options(*nopass)
  
```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Inquiry Message Exit Point – Conditional Responses (cont.)



▪ Data Definitions

```
d/copy qsysinc/qrpglesrc,qusec
```

```
d          sds
d CurUsr          358    367

dWait          s          1

dCancel        c          const('C')

dDump          c          const('D')
```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Inquiry Message Exit Point – Conditional Responses (cont.)



▪ The Code

```
/free
if CallType = 1;                                // Inquiry msg needing reply
  if MsgID = 'RNQ0102';                          // Attempt to divide by zero
    QUSBPRV = 0;                                  // API Error Code set to 0
    Select;
      When CurUsr = 'JOE_USER'; // RPG dump for user
        SndRply( MsgKey :QualMsgQName :Dump :1 :'*NO' :QUSEC);
      When CurUsr = 'VINING'; // Cancel for Vining
        SndRply( MsgKey :QualMsgQName :Cancel :1 :'*NO' :QUSEC);
      Other; // Leave for next exit program
    EndSl;
  endif; // Leave for next exit program
else;
  dsply ('Unexpected call type of ' + %char(CallType)) ' ' Wait;
endif;

*inlr = *on;
return;
/end-free
```

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Some of the new V5R4 Security APIs



- Check Password Exit Point (QIBM_QSY_CHK_PASSWRD)
 - Allows for audit record to record non-compliance of password
 - Does not allow password change to be stopped
 - Does get called for CRTUSRPRF, CHGUSRPRF, CHGPWD, QSYCHGPW
- Cryptographic Key Management

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Some the new V5R4 Service APIs



- Control PTF Order (QESCPTFO)
 - Order Individual PTFs, Groups, Cum packages, PSP information
 - Electronic - *SAVF or *IMAGE
 - Media
 - Resume orders at later time
 - Cancel orders
 - Optional status updates with *DTAQ
- Set User Policy (QPDETPOL)
 - Maximum PTF order size for electronic distribution for LAN and modem
 - Length of time to retain problem related information
 - Level of problem documentation to send
- Retrieve Policy Data (QPDETRTV)
- Collect Hung Job Service Documentation (QPDETHNG)
 - Collects information related to job that appears to be hung
 - Used by watch policy monitor when ENDJOBABN used
- Change Contact Information (QEDCHGIN)
- Retrieve Contact Information (QEDRTVCI)
- Watches – End (QSCEWCH), Start (QSCSWCH), Exit Program (QPDETWCH)
- Several other APIs related to collecting data related to software problems

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Some of the new V6R1 APIs



- Change *DTAQ API (QMHQCDQ)
 - Automatic reclaim of storage when *DTAQ is empty
 - i5/OS locking of *DTAQ – recognize use of ALCOBJ/WRKOBJLCK by users
 - API only works on local *DTAQs
- Exit Point for Change Journal Receiver
 - QIBM_QJO_CHG_JRNRCV
 - Exit program receives control when journal receiver is detached from journal
- Open List of History Log (QMHOLHST)
- Retrieve Printer File Attributes (QDFRPRTA)
- Change Storage Pool API (QWCCHGPL) to manage storage pool size, activity level, and tuning values
- Cryptographic Exit Points
 - Clear Master Key
 - Delete Key Store Record
 - Set Master Key
 - Translate Key Store

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

More of the new V6R1 APIs



- Verify Name API (QCAVFYNM)
- Delete Object API (QLIDLTO)
- Change Program Data API (QBNCHGPD)
 - Update *Module, *Pgm, and *Srvpgm
 - Source file location (including debug view locations)
 - Bound *Module library names
 - Bound *Srvpgm library names
 - Prevent update of bound *SRVPGM library (6.1/7.1 PTF)
- Create and Send MIME E-mail (QtmsCreateSendEmail)
 - Optionally sign and encrypt MIME E-mail using Secure MIME (S/MIME)
 - Attachment support from IFS stream file
 - What if? Watch for abnormal job termination, get joblog, send it!

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Some of the V6R1 Changed Interfaces



- Command Analyzer Retrieve Exit support for after command has run (5.4, 6.1, 7.1 PTF)
- Retrieve Job Information (QUSRJOBI) and Open List of Jobs (QGYOLJOB)
 - Message key and qualified message queue of the message that an active job is waiting on
 - *A real assist for automating of system operations*
- Send Reply Message (QMHSNDRM)
 - Support for *EXT for external message queue
- Convert Edit Code (QECCVTEC) and Edit (QECEDT)
 - Support for up to 63 digits
- Convert Date and Time (QWCCVTDT)
 - Year offsets

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Some of the new 7.1 APIs



- Spool File Security exit point
 - Function specific (Chg, Cpy, Dlt, Dsp, Hld, Mov, Rst, Save, Snd, etc) authorization control to a spool file
- Retrieve Watch Information (QSCRWCHI)
- Retrieve Watch List (QSCRWCHL)
- Open List of Output Queues (QSPOLOTQ)
- Retrieve Spool Information (QSPSPLI)
 - Number of spool files for a user or all users
 - Number of spool files on an ASP
- Socket exit points to allow/disallow:
 - accept()
 - connect()
 - listen()

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Key Points to Take Home



- System APIs continue to grow in both callable interfaces and exit points
- Most System APIs are created due to user requirements
- Improvements target end user, development, and support
 - UIM customization exits
 - Message Handler exits
 - Job Interrupt
 - Message Watches

Copyright Bruce Vining Services LLC, 2010, 2012
All Rights Reserved

Last Chance Before the Break



How to contact me:
Bruce Vining
bvining@brucevining.com

Copyright Bruce Vining Services LLC, 2010, 2011
All Rights Reserved