


[print](#) | [close](#)

APIs by Example: UIM & Work Management APIs, Part 3

[System iNetwork Programming Tips Newsletter](#)

[Carsten Flensburg](#)

Carsten Flensburg

Thu, 07/13/2006 (All day)

Here's the final part of the new work management command suite, which I introduced in the previous installments of this article (links to previous installments are at the end of this article): The Work with Job Queue Jobs (WRKJOBQJOB) command. The Work with Subsystem Activity (WRKSBSACT), Work with Subsystem Job Queues (WRKSBSJOBQ), and WRKJOBQJOB commands together offer an alternate facility to monitor and work with subsystems, job queues, and batch jobs. The commands are all based on a number of work management APIs as well as User Interface Manager (UIM) APIs and UIM List Panel Groups.

So far, I have explained core UIM concepts, list panel group UIM tags, and how to actually code a UIM list panel group. Today, I focus on the UIM API dialog with the panel group, as well as the UIM exit program called by the UIM to perform certain list management tasks.

As usual, let me begin with the command prompt. Here's what the WRJOBQJOB command looks like when prompted:

```

Work with Job Queue Jobs (WRKJOBQJOB)

Type choices, press Enter.
Job queue . . . . . Name
Library . . . . . *LIBL Name, *LIBL, *CURLIB
  
```

Running the WRKJOBQJOB command or selecting option 5 from the list panel of the WRKSBSJOBQ command that I presented last time leads to the display of the following list panel:

Work with Job Queue Jobs						WYNDHAMW	
						06-07-06 17:59:17	
Job queue :		QMQM		Job queue status :		RLS	
Library :		QMQM		Subsystem :		QMQM	
Type options, press Enter.							
2=Change 3=Hold 4=End 5=Work with 6=Release 8=Spooled files							
		Current		Job		Entered Submitted	
Opt	Job	User	Type	Pty	----Status--	Time	By
	AMQZXMA0	QMQM	BCH	4	ACTIVE CNDW	02:23:39	QMQM
	AMQZFUMA	QMQM	BCH	5	ACTIVE CNDW	02:23:46	QMQM
	AMQALMPX	QMQM	BCH	4	JOBQ	02:24:40	QMQM
	AMQRRMFA	QMQM	BCH	4	JOBQ	02:25:17	QMQM

AMQZDMAA	QMQM	BCH	4	JOBQ		02:25:19	QMQM
RUNMQCHI	QMQM	BCH	5	JOBQ		02:25:19	QMQM
AMQZLAA0	QMQM	BCH	5	JOBQ		02:25:21	QMQM
RUNMQCHL	QMQM	BCH	5	JOBQ	HLD	02:25:24	QMQM
							More...
Parameters or command							
===>							
F3=Exit F5=Refresh F6=Hold job queue F10=Work with subsys jobs							
F11=Display job origin F12=Cancel F21=Print list F24=More keys							

The jobs are presented in job status order followed by job queue priority and job number order, so active jobs appear at the top of the list, and the jobs waiting on the job queue are listed in the order that they will become active, for each individual job queue priority.

You can perform a number of actions against the jobs in the list, and a set of command function keys is available to manage the list and run commands such as Work with Subsystem Jobs (WRKSBSJOB), Work with Job Schedule Entries (WRKJOBSCDE), and Work with Active Jobs (WRKACTJOB). All options, function keys, and list panel fields are documented with the online cursor-sensitive help facility included.

The steps involved in activating and running the list panel group are performed by the UIM APIs in the CBX158 CPP:

1. The Open Display Application (QUIOPNDA) API is first called to initialize the panel group session, and it then returns an application handle to be used in the continued UIM dialog.

```
OpnDspApp( UIM.AppHdl
           : PNLGRP_Q
           : SCP_AUT_RCL
           : PRM_IFC_0
           : HLP_WDW
           : ERRRC0100
           );
```

2. Using the preceding application handle, the Put Dialog Variable (QUIPUTV) API is called to add the name and library of the exit program, which UIM should call to perform certain list management tasks, to the variable pool. You specify the exit program variable name to the UIM on the appropriate UIM tag keywords, and the UIM then retrieves this information from the variable pool before performing the exit program call.

```
**-- UIM Panel exit program record:
D Exprcd          Ds          Qualified
D ExitPg          20a      Inz ( 'CBX158E  *LIBL' )
.
.
PutDlgVar( UIM.AppHdl
           : Exprcd
           : %Size( Exprcd )
           : 'EXPRCD'
           : ERRRC0100
           );
```

```

.*-- UIM Variable record definitions:
:VARRCD    NAME=EXPRCD
           VARS='EXITPG'.

```

- Next, the QUIPUTV API is called again to initialize the list header information, which includes date and time, job queue name and library, as well as job queue status and subsystem name. This is done from a subroutine that is then later called again, whenever this information needs refreshing.

```

**-- UIM Panel header record:
D HdrRcd          Ds          Qualified
D  SysDat          7a
D  SysTim          6a
D  TimZon          10a
D  JobQue          10a
D  JobQueLib       10a
D  JobQueSts       1a
D  ActSbs          10a
.
.
PutDlgVar( UIM.AppHdl
           : HdrRcd
           : %Size( HdrRcd )
           : 'HDRRCD'
           : ERRC0100
           );

```

```

.*-- UIM Variable record definition
:VARRCD    NAME=HDRRCD
           VARS='DATE TIMZON JBQNAM JBQLIB JBQSTS ACTSBS'.

```

- Subsequently, the job list is built by the Add List Entry (QUIADDLE) API. For each job entry returned by the Open List of Jobs (QGYOLJOB) API, a list entry is added. The QGYOLJOB API also handles sorting the list in the correct order.

```

**-- UIM List entry:
D LstEnt          Ds          Qualified
D  Option          5i 0
D  JobId           26a
D  JobNam          10a
D  JobUsr          10a
D  JobNbr          6s 0
D  JobTyp          1a
D  JobSts1         7a
D  JobSts2         7a
D  JobPty          2a
D  EntDat          7a
D  EntTim          6a
D  CurUsr          10a
D  SbmJob          10a
D  SbmUsr          10a

```

```

D   SbmNbr                                6s  0
.
.
AddLstEnt( UIM.AppHdl
           : LstEnt
           : %Size( LstEnt )
           : 'DTLRCD'
           : 'DTLLST'
           : UIM.EntLocOpt
           : UIM.LstHdl
           : ERRC0100
           );

```

```

.*-- UIM Variable record definition
:VARRCD   NAME=DTLRCD
          VARS='OPTION JOBID JOBNAM JOBUSR JOBNBR JOBTYP
          JOBST1 JOBST2'
          VARS='JOBPTY ENTDAT ENTTIM CURUSR SBMJOB SBMUSR
          SBMNBR'.
.*-- UIM List definition:
:LISTDEF  NAME=DTLLST
          VARS='OPTION JOBID JOBNAM JOBUSR JOBNBR JOBTYP
          JOBST1 JOBST2'
          VARS='JOBPTY ENTDAT ENTTIM CURUSR SBMUSR SBMJOB
          SBMNBR'
          MSGID=CPI0944
          MSGF='QCPFMSG'.

```

5. When the list has been entirely built, the Set List Attributes (QUISETLA) API is called to mark the list complete and set the display position of the list to the first (top) record.

```

SetLstAtr( UIM.AppHdl
           : 'DTLLST'
           : LIST_COMP
           : EXIT_SAME
           : POS_TOP
           : TRIM_SAME
           : ERRC0100
           );

```

6. By calling the Display Panel (QUIDSPP) API, the list panel is then displayed and, until the QUIDSPP API returns to its caller, all interaction with the user is performed by the UIM.

```

DspPnl( UIM.AppHdl
        : UIM.FncRqs
        : PNLGRP
        : RDS_OPT_INZ
        : ERRC0100
        );
Select;
When   UIM.FncRqs = RTN_ENTER;
      Leave;

```

```

When   UIM.FncRqs = KEY_F17;
      ExSr   PosLstTop;

When   UIM.FncRqs = KEY_F18;
      ExSr   PosLstBot;

EndSl;

```

7. Pressing one of the following command function keys in the list panel causes the QUIDSPP API to return to its caller:

- a. If F5 is pressed, the return value 5 is returned from the Display Panel API and causes the application to rebuild and reposition the list.
- b. If F17 or F18 is pressed, the QUISETLA API is called to set the list's display position to top or bottom, respectively.
- c. If either F3 or F12 is pressed, the command ends, but first the Close Application (QUICLOA) API is called to free all allocated resources.

The CPP is compiled with activation group attribute *NEW to allow the command to be called repeatedly from the WRKJOBQJOB command's command line.

The CBX158E List Exit Program (LEP) is called by the UIM whenever one of the command function keys F6, F21, or F22 is pressed, to perform the actual function key action.

1. If F6 is pressed, the LEP executes the Hold Job Queue (HLDJOBQ) command or the Release Job Queue (RLSJOBQ) command against the job queue specified on the WRKJOBQJOB command, depending on the current status of the job queue.
2. If F21 is pressed, the LEP in turn calls:
 - a. The Add Print Application (QUIADDPA) API to enable the UIM to print the list.
 - b. The Print Panel (QUIPRTP) API, once for the list header to be printed, and once for the job list to be printed.
 - c. The Remove Print Application (QUIRMVPA) API to end the printing session and close the printer file.
3. If F22 is pressed, the QUIGETV API is called to retrieve the internal variables Cursor Entry Handle and Cursor Field Name. If a valid list entry is selected, the Get List Entry (QUIGETLE) API is called to retrieve the job name of that entry, and the WRKACTJOB command is run for that specific job name.

The preceding exit program calls are controlled by the UIM KEYI tag.s ACTION='CALL EXITPG' keyword specified for the command function keys in question:

```

:KEYI      KEY=F6
           HELP='HLPF6/JBQHLD'
           COND=JBQRLS
           ACTION='CALL EXITPG'

```

```
PRIORITY=20
.F6=Hold job queue
```

Likewise, the CBX158E is called by the UIM every time one of the following list options is run for a list entry:

1. If option 2, 3, or 6 was successfully run, the selected list entry is retrieved by the QUIGETLE API, and the entry.s status field is updated to .Chg., .Hld., or .Rls. respectively. The Update List Entry (QUIUPDLE) API is then called to have the list panel reflect the changed entry status value.
2. If option 4 was successfully run, the Remove List Entry (QUIRMVLE) API is called to remove the list entry for the job that was ended. Another way to handle this situation would be to simply change the entry.s status to .End. and wait for the job to terminate, and then leave it to the user to press F5 to refresh the list.

The call of the exit program is achieved by specifying the UIM LISTACT tag.s USREXIT='CALL EXITPG' keyword for the preceding list options:

```
:LISTACT  ENTER='CMD CHGJOB JOB (&JOBNBR/&JOBUSR/&JOBNAM)
           &CMDPRM '
          PROMPT='CMD ?CHGJOB ?*JOB (&JOBNBR/&JOBUSR/&JOBNAM)
           &CMDPRM '
          HELP='WRKJOBQJOB/OPTCHG '
          OPTION=2
          NOCMD=PROMPT
          USREXIT='CALL EXITPG '
          .2=Change
```

I offer this short article series about the UIM APIs and UIM list panel groups to encourage you to consider UIM as a viable and quick alternative to writing all the UI and dialog yourself. After you have a few different types of UIM applications and code snippets in your toolbox, creating a new, fully functional utility is simple.

This APIs by Example includes the following sources:

```
CBX158  -- Work with Job Queue Jobs - CCP
CBX158E -- Work with Job Queue Jobs - UIM Exit Program
CBX158H -- Work with Job Queue Jobs - Help
CBX158P -- Work with Job Queue Jobs - Panel Group
CBX158V -- Work with Job Queue Jobs - VCP
CBX158X -- Work with Job Queue Jobs

CBX158M -- Work with Job Queue Jobs - Build command
```

To create all these objects, compile and run CBX158M. Compilation instructions are in the source headers, as usual.

Previously published UIM-related APIs by Example articles:

<http://www2.systeminetwork.com/article.cfm?id=52457>

<http://www2.systeminetwork.com/article.cfm?id=52574>

<http://www2.systeminetwork.com/article.cfm?id=52665>

<http://www2.systeminetwork.com/article.cfm?id=52739>

This article demonstrates the following APIs:

Open List of Job Queues (QSPOLJBQ) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/qspoljbq.htm>

Open List of Jobs (QGYOLJOB) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/qgyoljob.htm>

Get List Entries (QGYGTLE) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/qgygtle.htm>

Close List (QGYCLST) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/qgyclst.htm>

Open Display Application (QUIOPNDA) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quiopnda.htm>

Close Application (QUICLOA) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quicloa.htm>

Display Panel (QUIDSPP) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quidspp.htm>

Display Long Text (QUILNGTX) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quilngtx.htm>

Put Dialog Variable (QUIPUTV) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quiputv.htm>

Get Dialog Variable (QUIGETV) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quigetv.htm>

Add List Entry (QUIADDLE) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quiaddle.htm>

Get List Entry (QUIGETLE) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quigetle.htm>

Update List Entry (QUIUPDLE) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quiupdle.htm>

Remove List Entry (QUIRMVLE) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quirmvle.htm>

Retrieve List Attributes (QUIRTVLA) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quirtvla.htm>

Set List Attributes (QUISETLA) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quisetla.htm>

Delete List (QUIDLTL) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r4/topic/apis/quidltl.htm>

Print Panel (QUIPRTP) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/quiprtp.htm>

Add Print Application (QUIADDPA) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/quiaddpa.htm>

Remove Print Application (QUIRMVPA) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/quirmvpa.htm>

Process Commands (QCAPCMD) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/qcapcmd.htm>

Retrieve Message (QMHRTVM) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/QMHRTVM.htm>

Send Program Message (QMHSNDPM) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/QMHSNDPM.htm>

Retrieve Object Description (QUSROBJD) API:

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/apis/qusrobjd.htm>

MI built-in function:

Copy memory (_MEMMOVE):

<http://publib.boulder.ibm.com/infocenter/iserics/v5r3/topic/rzatk/MEMMOVE.htm>

You can retrieve the source code for this API example from the following link:

http://www.pentontech.com/IBMContent/Documents/article/52825_79_UimWrkMgt3.zip

Source URL: <http://iprodeveloper.com/rpg-programming/apis-example-uim-work-management-apis-part-3>