

Working Knowledge Series

The Life Cycle of a Job

Tom Davidson

Cleindori Consulting, LLC

Tom.Davidson@CleindoriConsulting.com

Ground Rules

- Please set cell phones to 'stun'
- Feel free to ask questions as we go along, I may defer to later in the session or off-line.
- Not all slides in handout may be included, they are included for your reference only.

Cleindori Consulting LLC

- Specialties:
 - IBM i
 - Performance
 - Database Administration
 - Application design and implementation
 - Contract Programming
 - Administration
 - JDE EnterpriseOne and OneWorld CNC
 - Application integration
- www.CleindoriConsulting.com

Speaker Bio

- 26+ years on the i, developer the whole time
- IBM ILE Certification
- 10+ years of performance tuning
- Now consulting, specialties include JD Edwards, performance tuning, sizing, and contract programming

Agenda

- Overview
- Details
- Permanent job structures
- Temporary Job Structures

Overview

- At a high level the life cycle of a job is:
 - Job enters system
 - Job starts
 - Job cycles between active and wait states
 - Job ends
 - Job leaves system

How jobs enter the system

- User requested
 - SBMJOB
 - Job scheduler
- ‘Automatically’
 - Prestart job entry
 - Autostart job Entry
 - Communications Entry
 - Workstation Entry

What happens when jobs enter the system

- Assigned a permanent job structure
 - A QTEMP library is assigned
 - A job message queue is created
 - If a command or request is on SBMJOB or JOBD a request message is put on job message queue
- Depending on how the job entered the system is either placed in a job queue or directly into a subsystem.

What happens when a job starts

- Job starts when it enters a subsystem.
- Message CPF1124 placed in history log
- Job assigned a temporary job structure
 - Stack
 - Open data path area
 - Library list

What happens when a job starts (Enters from a job queue)

- Routing data is retrieved
 - How routing data is determined
 - SBMJOB, RRTJOB, TFRJOB all have a place to enter routing data, do support special values
 - Workstation entries, workstation types, communication entries, and remote location names entries all have a job description attached
 - Prestart jobs do not use routing entries.

What happens when a job starts (Prestart job)

- All job attributes are in the PJE

```
*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
*****
01 *                Display Prestart Job Entry Detail                * 01
02 *                                                    System:  ISERIES1 * 02
03 * Subsystem description:  QUSRWRK          Status:  ACTIVE          * 03
04 *                                                                * 04
05 * Program . . . . . :  QZDASOINIT          * 05
06 *   Library . . . . . :           QSYS          * 06
07 * User profile . . . . . :  QUSER          * 07
08 * Job . . . . . :  QZDASOINIT          * 08
09 * Job description . . . . . :  *USRPRF          * 09
10 *   Library . . . . . :                               * 10
18 * Pool identifier . . . . . :  1          * 18
05 * Class . . . . . :  QPWFSEVER          * 05
06 *   Library . . . . . :           QSYS          * 06
07 *   Number of jobs to use class . . . . . :  *CALC          * 07
24 *                                                                * 24
*****
*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
```

Example of Determining Routing Data

- Batch Example

```
*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
*****
01 *          Display Command String          * 01
02 *                                          * 02
03 * SBMJOB CMD(CALL PGM(X))                * 03
04 *          JOB(*USRPRF)                   * 04
05 *          RTGDTA(*JOB)                   * 05
19 *                                          * 19
20 * Press ENTER to continue.                * 20
21 *                                          * 21
22 * F3=Exit   F5=Refresh   F12=Cancel   F13=How to use this display   F24=More keys* 22
23 *                                          * 23
24 *                                          * 24
*****
*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
```

Example of Determining Routing Data

User profile shows what job description to use

```
*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
*****
01 *                Display User Profile - Basic                * 01
02 *                                                         * 02
03 * User profile . . . . . :      DAVIDSONT                    * 03
04 *                                                         * 04
05 * Job description . . . . . :      DAVIDSONTJ                * 05
06 *   Library . . . . . :      DAVIDSONT                      * 06
07 * Accounting code . . . . . :                                     * 07
08 * Message queue . . . . . :      DAVIDSONTQ                 * 08
09 *   Library . . . . . :      DAVIDSONT                      * 09
20 *                                                         * 20
21 * Press Enter to continue.                                     * 21
22 *                                                         * 22
23 * F3=Exit   F12=Cancel                                       * 23
24 *                                                         * 24
*****
*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
```


What happens when a job starts (Enters from a job queue)

- Routing data is processed
 - First match in routing entries is picked
 - Class is assigned
 - Memory pool is assigned
 - Initial program is set from routing entry

Example of Routing Entries

```
*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
*****
01 *                               Display Routing Entries                               * 01
02 *                               System:    ISERIES1                               * 02
03 * Subsystem description:    QINTER           Status:    ACTIVE                   * 03
04 *                               * 04
05 * Type options, press Enter.                               * 05
06 *     5=Display details                                     * 06
07 *                               * 07
08 *                               Start * 08
09 * Opt      Seq Nbr      Program      Library      Compare Value      Pos      * 09
10 *          10          QCMD          QSYS          'QCMDI '           1      * 10
11 *          15          QCMD          QSYS          'QIGC '           1      * 11
12 *          20          QCMD          QSYS          'QS36MRT '        1      * 12
13 *          40          QARDRIVE     QSYS          '525XTEST '       1      * 13
14 *          700         QCL           QSYS          'QCMD38 '         1      * 14
15 *          9999        QCMD          QSYS          *ANY               * 15
```


What happens when a job starts (No matter how it enters)

- Class is used to assign the following:
 - Run priority
 - Time slice
 - Default wait time
 - Max CPU, Temp Storage, and threads
- Initial program is called
 - If QCMD request message is received and executed
 - Otherwise program is executed normally

States while job is running

- Job runs until one of 4 states is reached
 - Time slice end
 - Job is transferred or rerouted
 - A wait
 - Job ends
- At time slice end
 - Lowers it's priority slightly
 - If no higher priority job, and an activity level is available continue running
 - Otherwise go into a wait

States while job is running

- Transferred (TFRJOB)
 - Job is sent to requested job queue
 - Temporary job structure is released
- Job is Rerouted (RRTJOB)
 - Job is sent to requested subsystem and follows path for subsystem entry
 - Temporary job structure released

States while job is running

- Waiting in an activity level
 - There are two types of wait states
 - Waiting in an activity level
 - This is normally a very short time.
 - Purpose is to prevent a job from being paged out when it would be more efficient to keep itself in memory.
 - WRKACTJOB status normally ends in an 'A'
 - A normal wait
 - Job becomes eligible to be paged out.
 - A special case - Seizes

States while job is running

- Wait
 - 2 basic types of waits
 - System wait
 - User wait
- System Waits
 - Waiting for system resource to become available
 - Includes page faults and CPU resources
 - Seizes and lock waits

Waits while job is running

- User waits
 - Waits initiated by the user
 - Includes (not limited to)
 - Job is held
 - Wait for entry on data queue (DEQW)
 - Wait for user controlled resources (MTXW/SEMW)

What happens when job ends

- Message CPF1164 placed in history log
- Temporary job structure is released
- QTEMP is cleared
- Job Log is created (if required)
- Job Message Queue is deleted (if possible)

What happens when job exits the system

- This happens when last spool file has been deleted (or if none are created)
- Permanent job structures cleared and released (recycled)
- QTEMP cleared and released

Permanent Job Structure

How and when created

- Created initially at IPL
 - Number controlled by QTOTJOB system value
 - They are re-cycled
 - After they are consumed, QADLTOTJ more will be created (can occur many times)
- Hold information about job that is persistent
 - Can cross IPL's

Job Table

Session A - [24 x 80]

File Edit View Communication Actions Window Help

Display Job Tables ISERIES1
03/27/09 09:15:36

Permanent job structures:

Initial	1000
Additional	10
Available	5746
Total	9419
Maximum	163520

Temporary job structures:

Initial	350
Additional	10
Available	124

Table	Size	-----Entries-----			
		Total	Available	In-use	Other
1	9674496	9419	5746	3673	0

Press Enter to continue.

F3=Exit F5=Refresh F11=In-use entries F12=Cancel

Bottom

MA a 01/001

128 | 1902 - Session successfully started

Permanent Job Structure

Contents

- I'll cover 3 main parts
 - Job Information
 - User Information
 - Output Information
- Also contains other information
 - Spooled input
 - Environment variables

Permanent Job Structure

Job Information

- Definition Information
 - When it entered system
 - When it ended
 - Who submitted it
 - Job description
 - Print text/Print key info
 - Default CCSID
 - Can it be multithreaded

Permanent Job Structure

Job Information

- Execution information
 - Job type
 - Special Environment
 - Job switches
 - Current job status

Job message queue (joblog)

Permanent Job Structure Printout Information

- Name of spool file
- Output queue it went to
- Does it still exist

Temporary Job Structures

- Created initially at IPL
 - Number controlled by QACTJOB
 - They are recycled
 - When they are all in use, QADLACTJ more will be created. This process can occur many times.

Temporary Job Structures

Contents

- Job information
- Runtime information
- Also contains other information
 - Whole series of pointers to objects
 - Stack space
 - Static storage space

Temporary Job Structures

Contents

- Job information
 - Library list
 - Job date/time
 - Current user

Temporary Job Structures

Contents

- Runtime information
 - ODP's
 - Override information
 - Activation groups
 - Mutes/semaphore list
 - SQL info (commitment control information)
 - Runtime information (WRKJOB #3)

Summary

- The job life cycle while overall very simple. Start, run/wait, end has many things going on in the background.
- Knowledge of the permanent and temporary job structures can help explain why you see some of the things you do.

For More Information