FA120R.MBR

Path: NXCLOUD/rpgsrc/FA120R.MBR **Generated**: 2026-01-08 12:26:11 **Processing Time**: 12901ms

Business Logic for Job Register Status Maintenance

This document outlines the business rules that govern the Job Register Status Maintenance process, based on an analysis of the RPG program FA120R. The primary focus is on how job records are retrieved, displayed, and updated within the system.

The core logic for job status maintenance is contained within the main processing logic of the program, which handles user interactions and updates job records based on user input.

Job Status and Record Handling Rules

Job Register Status Maintenance: fjobl1, fjoblu, fausrl1, ffa120d

1. Retrieve Job Information

•**Logic:** The program retrieves job details from the job register file (fjobl1) based on a specific key. If the record is found, it populates job-related fields; if not, it initializes these fields to zero.

•**File:** fjobl1 (Job Register)

•**Field:** fajb01 to fajb10

•**Condition:** The process will not populate job fields if the record is not found (*in90 is on).

2. Display Job Information

•**Logic:** The program displays the job information on the screen using the a1bld format.

•**File:** ffa120d (Display File)

•**Field:** Various job fields

•**Condition:** The display is triggered after job information is retrieved.

User Interaction and Query Rules

1. User Query on Job Start

•**Logic:** If the user presses F1, the program checks which job field to query based on user input and retrieves the corresponding user information from the user register.

•**Files:**

•fausrl1 (User Register)

•**Fields:**

•ausrl1_user (User ID)

•**Condition:** The query is executed if the user presses F1 and the current action is valid.

2. Field-Specific User Lookup

•**Logic:** The program checks which field is being queried (e.g., A1JB01, A1JB02, etc.) and retrieves the corresponding user information and timestamps.

•**File:** fausrl1 (User Register)

•**Field:** ausrl1_user

•**Condition:** The lookup occurs based on the specific field selected by the user.

Job Update Rules

1. Update Job Register

- **Logic:** The program updates the job register with the current values from the display fields. If the record exists, it updates the existing record; otherwise, it creates a new record.
- **File:** fjoblu (Job Update)
- **Fields:**
- fajb01 to fajb10 (Job fields)
- **Condition:** The update occurs if the job record is found (*in90 is off).

2. Set Timestamps on Update

- **Logic:** When updating a job record, the program sets the timestamps for when the job was created and last updated.
- **File:** fjoblu (Job Update)
- **Condition:** Timestamps are set based on the current time when the record is updated.

Special Conditions (Program-Specific)

1. Job Record Not Found Handling (FA120R)

- **Logic:** If a job record is not found during the retrieval process, all job fields are initialized to zero.
- **File:** fjobl1 (Job Register)
- **Field:** fajb01 to fajb10
- **Condition:** This occurs when the chain operation to fjobl1 fails (*in90 is on).

2. User Lookup Subroutine (FA120R)

- **Logic:** A subroutine is called to look up the user information from the user register based on the user ID associated with the job.
- **File:** fausrl1 (User Register)
- **Field:** ausrl1_user
- **Condition:** This subroutine is called whenever user information needs to be retrieved for a job.

Subprogram Calls Affecting Logic

Beyond direct file checks, several external subprograms are called that play a significant role in the workflow.

1. x_user (User Lookup Subroutine)

- **Trigger:** Called when user information is needed based on job fields.
- **Logic:** Retrieves user details from the user register and populates the user-related fields.
- **Impact:** This call ensures that the correct user information is displayed alongside job details.

2. *inzsr (Initialization Subroutine)

- **Trigger:** Called at the start of the program to initialize keys for file access.
- **Logic:** Sets up keys for job and user registers based on the firm number.
- **Impact:** This is crucial for ensuring that the program can access the correct records based on the firm context.

3. exfmt (Display Format Subroutine)

- **Trigger:** Called to display the job information on the screen.

- **Logic:** Formats and presents job details to the user.
- **Impact:** This represents the user interface component of the job maintenance process, allowing user interaction with job records.