FA120R.MBR

Path: NXCLOUD/rpgsrc/FA120R.MBR **Generated**: 2026-01-08 13:32:08 **Processing Time**: 13387ms

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 status records are retrieved, displayed, and updated within the system.

The core logic for maintaining job register status is contained within the *inzsr subroutine in FA120R. The program retrieves job details from the job register, allows for user interaction, and updates the job status based on user input.

Job Status and Header Rules

Job_Register: fjobl1, fjoblu, fausrl1, ffa120d

1. Retrieve Job Details

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

•**File:** fjobl1 (Job Register)

•**Field:** fajb01, fajb02, ..., fajb10

•**Condition:** The process will not select a record if the key lookup (*in90) indicates that no record was found.

2. Display Job Information

•**Logic:** The program displays the job information on the screen for user interaction.

•**File:** ffa120d (Display File)

•**Field:** Various job fields

•**Condition:** The display is executed after retrieving job details.

Configuration and Authorization Rules

1. User Lookup

•**Logic:** The program looks up the user information from the user register (ausrl1) based on the user ID associated with the job. It retrieves the user's name for display purposes.

•**Files:**

•ausrl1 (User Register)

•**Fields:**

•ausrl1_user (User ID)

•**Condition:** The user information is retrieved only if the user ID exists in the user register.

2. Job Update Logic

•**Logic:** The program updates the job register with the current job status and user information when changes are made.

•**File:** fjoblu (Job Register Update)

•**Field:** fajb01, fajb02, ..., fajb10

•**Condition:** The update occurs if the job record is found (*in90 is off).

Financial and Transactional Rules

1. Job Status Update Timing

•**Logic:** The program records timestamps for when the job status is updated, capturing the start and end times of the job.
•**File:** fjoblu (Job Register Update)
•**Fields:**
•fajeda (End Date)
•fajeti (End Time)
•**Condition:** Timestamps are recorded only if the job record is being updated (i.e., the record exists).

2. Initial Job Record Creation

•**Logic:** If a job record does not exist, a new record is created with the current firm and user information.
•**File:** fjoblu (Job Register Update)
•**Condition:** A new record is written if the job record is not found (*in90 is on).

Special Conditions (Program-Specific)

1. Exit Logic (FA120R)

•**Logic:** The program allows users to exit via a specific key (F3), ensuring that any unsaved changes are not lost.
•**File:** ffa120d (Display File)
•**Field:** None
•**Condition:** The exit is triggered by the F3 key being pressed.

2. User Interaction Handling (FA120R)

•**Logic:** The program handles user interactions, such as querying who started the job and displaying relevant information based on user input.
•**File:** ffa120d (Display File)
•**Fields:** Various user-related fields
•**Condition:** User queries are processed only if the corresponding function key (F1) is pressed.

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:** This subroutine is called to retrieve user information based on the user ID associated with the job.
•**Logic:** It chains to the user register and retrieves the user's name.
•**Impact:** This call ensures that user-related information is accurately displayed and updated in the job register.

2. *inzsr (Initialization Subroutine)

•**Trigger:** This subroutine initializes the program and sets up keys for file access.
•**Logic:** It establishes the keys for the job register and user register files.
•**Impact:** This initialization is crucial for ensuring that the program can correctly access and manipulate job and user records.

3. exfmt (Display Format Subroutine)

•**Trigger:** This subroutine is called to display the job information on the screen.
•**Logic:** It formats and presents the data to the user.
•**Impact:** This is a key interaction point for users, allowing them to view and interact with job records effectively.