FX785R.MBR

Path: NXCLOUD/rpgsrc/FX785R.MBR **Generated**: 2026-01-08 12:58:16 **Processing Time**: 9869ms

Business Logic for ASOFAK Konvertering

This document outlines the business rules that govern the conversion of routine registers within the ASOFAK system, based on an analysis of the RPG program FX785R. The primary focus is on the logic that processes and updates routine records.

The core logic for converting routine registers is contained within the main program logic of FX785R. The program reads records from the aforlur file, applies specific transformation rules based on the value of the afruti field, and updates the records accordingly.

Routine Conversion Rules

ASOFAK: aforlur

1. Set Fixed Value for Specific Routines

- **Logic:** If the afruti field matches any of the specified routine types (e.g., 'FTILB', 'FOBEK', etc.), the afacpi field is set to a fixed value of 15.
- **File:** aforlur (Routine Register)
- **Field:** afruti, afacpi
- **Condition:** The process will set afacpi to 15 if afruti is equal to any of the following: 'FTILB', 'FOBEK', 'FPLUKK', 'FPAKK', 'FPAKKNEG', 'FBEST', 'BPLUKK', 'BPAKK', 'BPAKKNEG'.

2. Transform Routine Codes

- **Logic:** The program transforms certain afruti values to new codes. For example, 'FP651' is changed to 'FH601', and 'FO202' is changed to 'FR612'.
- **File:** aforlur (Routine Register)
- **Field:** afruti
- **Condition:** The process will change afruti from specific old codes to new codes based on a predefined mapping.

Update and Read Logic

1. Update Routine Records

- **Logic:** After processing the transformations, the program updates the records in the aforlur file with the new values.
- **File:** aforlur (Routine Register)
- **Field:** afruti, afacpi
- **Condition:** The update occurs after the transformations are applied to the current record being processed.

Program Flow and Termination

1. Program Termination

- **Logic:** The program sets the last record indicator (*inlr) to on and returns control to the calling program.

•**File:** None
•**Condition:** This occurs at the end of the program after all records have been processed and updated.

2. Initialization Subroutine

•**Logic:** The program includes an initialization subroutine that is called at the beginning of execution.
•**File:** None
•**Condition:** This subroutine is executed to set up any necessary program state before processing begins.

_

Subprogram Calls Affecting Logic

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

1. None (No external subprograms identified)

•**Trigger:** The main program executes without calling any external subprograms.
•**Logic:** The program operates independently, processing records directly from the aforlur file.
•**Impact:** The absence of external calls simplifies the logic, focusing solely on the transformations and updates defined within the program.

This documentation provides a clear overview of the business logic implemented in the FX785R program, detailing the rules and processes that govern the conversion of routine registers in the ASOFAK system.