

# VA540R.MBR

**Type:** Single File Analysis **Path:** NXCLOUD/rpgsrc/VA540R.MBR **Generated:** 2026-01-29 12:15:41  
**Processing Time:** 31556ms

# Business Logic for VA540R

This document outlines the business rules that govern the subfile management and user interaction for the program `VA540R`. The primary focus is on the conditions that will **block** an order or record from being displayed or processed in the subfile.

The core logic for subfile handling and user interaction is contained within the `subfile` subroutine in `VA540R`. A record must pass all the following checks to be displayed or processed in the subfile.

## Order Status and Header Rules

- \*\*Firm Validation\*\***
  - \*\*Logic:\*\*** The record's firm (`VAIFIR`) must match the firm passed as a parameter (`P\_FIRM`) to the program. If the firm does not match, the record is skipped.
  - \*\*File:\*\*** `VINFL1` (Information Type Master) and `VINFL2` (Information Description Master)
  - \*\*Field:\*\*** `VAIFIR`
  - \*\*Condition:\*\*** The process will not include a record in the subfile if `VAIFIR` is not equal to `W\_FIRM`. This ensures that only records belonging to the specified firm are processed.
- \*\*End of File Check\*\***
  - \*\*Logic:\*\*** The program stops processing records when the end of the file (`%EOF`) is reached.
  - \*\*File:\*\*** `VINFL1` and `VINFL2`
  - \*\*Field:\*\*** N/A (System EOF indicator)
  - \*\*Condition:\*\*** The process will terminate reading records if `%EOF` is true. This prevents attempts to process beyond the available data.

## Configuration and Authorization Rules

- \*\*Subfile Page Size\*\***
  - \*\*Logic:\*\*** The subfile is limited to a maximum number of records per page (`C\_SFIL`), and additional records are not displayed until the user navigates to the next page.
  - \*\*File:\*\*** N/A (Subfile control logic)
  - \*\*Field:\*\*** `W\_SRRN`, `W\_SPGE`
  - \*\*Condition:\*\*** The subfile will not display more records than the defined page size (`C\_SFIL`). This ensures proper subfile pagination.
- \*\*Cursor Positioning\*\***
  - \*\*Logic:\*\*** The program positions the cursor based on user input or the current record number (`W\_FCRN`).
  - \*\*File:\*\*** N/A (Subfile control logic)
  - \*\*Field:\*\*** `W\_FCRN`
  - \*\*Condition:\*\*** If `W\_FCRN` is not zero, the program positions the cursor to the corresponding record in the subfile. This ensures accurate navigation within the subfile.

## Financial and Transactional Rules

- \*\*Selection Validation\*\***
  - \*\*Logic:\*\*** The user must select a valid record (`BlVALG = 1`) for further processing.
  - \*\*File:\*\*** `B1SFL` (Subfile Data)
  - \*\*Field:\*\*** `BlVALG`

\* \*\*Condition:\*\* The program will not proceed with processing unless `B1VALG` is set to `1`. This ensures that only user-selected records are acted upon.

## ## Special Conditions (Program-Specific)

### 6. \*\*Positioning by Information Type\*\*

\* \*\*Logic:\*\* If the user specifies an information type (`B2ITYP`), the program positions the database to the corresponding record.

\* \*\*File:\*\* `VINFL1`

\* \*\*Field:\*\* `B2ITYP`

\* \*\*Condition:\*\* The program will position to the record if `B2ITYP` is not equal to `\*BLANK`. This allows filtering by information type.

### 7. \*\*Positioning by Description\*\*

\* \*\*Logic:\*\* If the user specifies a description (`B2ITXT`), the program positions the database to the corresponding record.

\* \*\*File:\*\* `VINFL2`

\* \*\*Field:\*\* `B2ITXT`

\* \*\*Condition:\*\* The program will position to the record if `B2ITXT` is not equal to `\*BLANK`. This allows filtering by description.

### 8. \*\*Subfile Clearing\*\*

\* \*\*Logic:\*\* The subfile is cleared before being repopulated to ensure no stale data is displayed.

\* \*\*File:\*\* N/A (Subfile control logic)

\* \*\*Field:\*\* `W\_SRRN`, `W\_SFRN`, `W\_SSRN`, `W\_SPGE`

\* \*\*Condition:\*\* The subfile is cleared by setting `W\_SRRN`, `W\_SFRN`, `W\_SSRN`, and `W\_SPGE` to zero. This ensures that only current data is displayed.

## ## Subprogram Calls Affecting Logic

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

### 1. \*\*`FORNY` (Refresh Subfile)\*\*

\* \*\*Trigger:\*\* Called when the user presses a specific function key (`IN10`).

\* \*\*Logic:\*\* Clears the subfile and repopulates it based on the current positioning.

\* \*\*Impact:\*\* Ensures the subfile reflects the most recent data.

### 2. \*\*`POSISJONER` (Positioning)\*\*

\* \*\*Trigger:\*\* Called when the user specifies an information type (`B2ITYP`) or description (`B2ITXT`).

\* \*\*Logic:\*\* Positions the database to the specified record and repopulates the subfile.

\* \*\*Impact:\*\* Allows the user to filter and navigate records efficiently.

### 3. \*\*`CLR\_SUBFILE` (Clear Subfile)\*\*

\* \*\*Trigger:\*\* Called before repopulating the subfile.

\* \*\*Logic:\*\* Clears all subfile records and resets control variables.

\* \*\*Impact:\*\* Prevents stale data from being displayed.

### 4. \*\*`CRT\_SUBFILE` (Create Subfile)\*\*

\* \*\*Trigger:\*\* Called during initial loading or after clearing the subfile.

\* \*\*Logic:\*\* Reads records from the database and populates the subfile until the page size is reached or the end of the file is encountered.

\* \*\*Impact:\*\* Ensures the subfile is populated with the correct data.

### 5. \*\*`BCK\_SUBFILE` (Back Page)\*\*

\* \*\*Trigger:\*\* Called when the user navigates back in the subfile (`IN11`).

\* \*\*Logic:\*\* Reads records in reverse to populate the previous page of the subfile.

\* \*\*Impact:\*\* Enables backward navigation within the subfile.

```
6.  **`DSP_SUBFILE` (Display Subfile)**
    *  **Trigger:** Called to display the subfile on the screen.
    *  **Logic:** Displays the subfile and positions the cursor as needed.
    *  **Impact:** Ensures the user interface reflects the current state of the subfile.

7.  ***INZSR` (Initialization Subroutine)**
    *  **Trigger:** Automatically executed when the program starts.
    *  **Logic:** Initializes program variables, positions the database, and populates the
subfile.
    *  **Impact:** Ensures the program starts in a consistent state.
```