

# FO410R.MBR

Type: Single File Analysis Path: NXCLOUD/rpgsrc/FO410R.MBR Generated: 2026-01-29 12:15:41  
Processing Time: 18242ms

---

```
# Business Logic for FO410R
```

This document outlines the business rules that govern the deletion of an order (entirely), based on an analysis of the RPG program `FO410R`. The primary focus is on the conditions that will \*\*block\*\* an order from being deleted.

The core logic for the deletion process is contained within the main program logic and its subroutines in `FO410R`. An order must pass all the following checks to be successfully deleted.

```
## Order Status and Header Rules
```

1. \*\*Order Header Chain Check\*\*

```
*   **Logic:** The program attempts to retrieve the order header record using the key `fohelu_key`. If the record is not found (`*IN92 = *ON`), the deletion process cannot proceed.  
*   **File:** `FOHELU` (Order Header Register)  
*   **Field:** `FOHELU_NUMM`, `FOHELU_SUFF`  
*   **Condition:** The process will not delete the order if the order header record cannot be retrieved using the provided order number (`FOHELU_NUMM`) and suffix (`FOHELU_SUFF`).
```

2. \*\*Order Header Unlock\*\*

```
*   **Logic:** If the order header record is successfully retrieved, the program unlocks the record before proceeding. If the unlock fails, the deletion process is blocked.  
*   **File:** `FOHELU` (Order Header Register)  
*   **Field:** `FOHELU_NUMM`, `FOHELU_SUFF`  
*   **Condition:** The order cannot be deleted if the header record remains locked.
```

```
## Configuration and Authorization Rules
```

3. \*\*Deposit File Check\*\*

```
*   **Logic:** If the deposit file (`SDEPLU`) exists for the order, it must be deleted. However, this logic is temporarily disabled due to a locking issue.  
*   **File:** `SDEPLU` (Deposit Register)  
*   **Field:** `SDEPLU_NUMM`, `SDEPLU_KUND`  
*   **Condition:** The process will not delete the order if the deposit file exists and cannot be deleted. (Currently commented out in the program.)
```

4. \*\*Transport Logging Check\*\*

```
*   **Logic:** If transport logging entries exist for the order in the `FTRPI2` file, they must be deleted before the order can be removed.  
*   **File:** `FTRPI2` (Transport Logging)  
*   **Field:** `FTRPI2_RNUM`, `FTRPI2_RSUF`  
*   **Condition:** The process will not delete the order if transport logging entries exist and cannot be deleted.
```

```
## Financial and Transactional Rules
```

5. \*\*Memo Balance Update\*\*

```
*   **Logic:** The program calculates the memo balance (`W_SALD`) and calls the subprogram `FA922R` to update the customer register. If this update fails, the deletion process is blocked.  
*   **File:** `FOHELU` (Order Header Register)  
*   **Field:** `FOTOTR`, `FOTOTS`  
*   **Condition:** The process will not delete the order if the memo balance update fails.
```

```

6. **Santander Payment Check**
  *   **Logic:** If the order was paid using a Santander card, the corresponding record in the `RKSJLU` file must be deleted.
    *   **File:** `RKSJLU` (Santander Payment Status)
    *   **Field:** `RKSJLU_JNUM`, `RKSJLU_JSUF`
    *   **Condition:** The process will not delete the order if the Santander payment record exists and cannot be deleted.

## Special Conditions (Program-Specific)

7. **Shadow Table Deletion**
  *   **Logic:** If the order line exists in the shadow table (`FODMIU`), it must be deleted before the order can be removed.
    *   **File:** `FODMIU` (Order Line Shadow Table)
    *   **Field:** `FODMIU_NUMM`, `FODMIU_SUFF`, `FODMIU_LINE`
    *   **Condition:** The process will not delete the order if the shadow table entry exists and cannot be deleted.

8. **Length Specification Deletion**
  *   **Logic:** If length specifications exist in the `FVLSLU` file for the order, they must be deleted before the order can be removed.
    *   **File:** `FVLSLU` (Length Specification Register)
    *   **Field:** `FVLSLU_NUMM`, `FVLSLU_SUFF`, `FVLSLU_LINE`, `FVLSLU_LPNR`
    *   **Condition:** The process will not delete the order if length specifications exist and cannot be deleted.

```

#### ## Subprogram Calls Affecting Logic

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

1. \*\*`FS711R` (Order Line Deletion Logging)\*\*
  - \* \*\*Trigger:\*\* Called during the deletion of each order line to log the deletion process.
  - \* \*\*Logic:\*\* Logs the deletion of order lines for traceability.
  - \* \*\*Impact:\*\* Ensures that all deleted order lines are logged, blocking the deletion process if logging fails.
  
2. \*\*`VL001R` (Inventory Update)\*\*
  - \* \*\*Trigger:\*\* Called after deleting an order line if inventory updates are required.
  - \* \*\*Logic:\*\* Updates inventory levels based on the deleted order line.
  - \* \*\*Impact:\*\* Blocks the deletion process if inventory updates fail.
  
3. \*\*`FA922R` (Customer Memo Balance Update)\*\*
  - \* \*\*Trigger:\*\* Called after calculating the memo balance for the order.
  - \* \*\*Logic:\*\* Updates the customer register with the new memo balance.
  - \* \*\*Impact:\*\* Blocks the deletion process if the memo balance update fails.
  
4. \*\*`VA750R` (Processing Code Validation)\*\*
  - \* \*\*Trigger:\*\* Called during the inventory update process to validate processing codes.
  - \* \*\*Logic:\*\* Ensures that processing codes are valid before proceeding with inventory updates.
  - \* \*\*Impact:\*\* Blocks the deletion process if processing code validation fails.