

# Batch Detail Design

## I. Functional Area

Promotion

## II. Module Affected

postprmext.pc

## III. Design Overview

This batch module is responsible for data maintenance tasks that are necessary before running prmext.pc.

This program is responsible for updating the promotion item status on the PROMSKU table, setting it to NULL if the promotion item has been extracted successfully or to 'Delete Processed' (DP) if the item was marked for deletion before extraction. It also resets all promotion status on the PRMHEAD table to 'E' after the item has been extracted, or 'M' when the promotion has been completed.

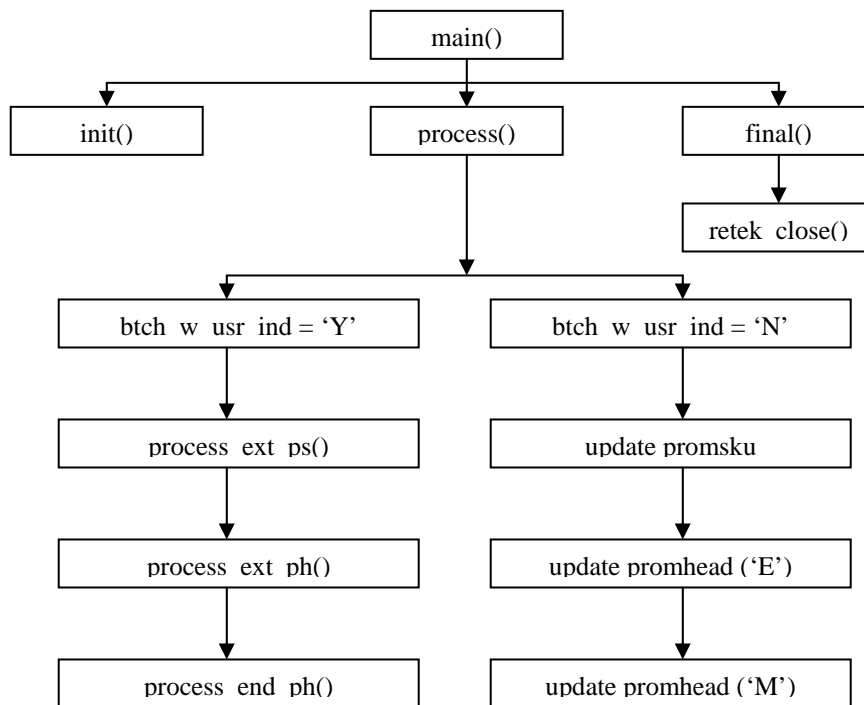
The LUW of this module is promotion.

## IV. Stored Procedures / Shared Modules (Maintainability)

N/A



## V. Program Flow



## VI. Function Level Description

**Declare** global variables, a fetch array structure to hold array-fetched records from driving cursors, and function prototypes.

**Main():** Standard Retek main function. Validates input parameters, calls init, process and final functions. Logs appropriate message.

**Init():** Standard Retek init function. Calls `retex_init()`. Get the `unit_options.pos_extract_days`, `period.vdate` and `system_options.btch_w_usr_ind`. Calculate the `pos_extract_date`. If `system_options.btch_w_usr_ind = 'Y'` and running for the first time, delete any records from the `batch_lock_log` for `postprmxext`. If it is a restart, check the `batch_lock_log` table if any records were not processed by `postprmxext.pc` when it was last ran. This is to inform the user after `postprmxext.pc` runs successfully that there are unprocessed records due to locking.

**Process():** Main processing is divided in two scenarios, `system_options.btch_w_usr_ind = 'Y'` and `system_options.btch_w_usr_ind = 'N'`:

- If `btch_w_usr_ind = 'N'`
  - Update PROMSKU status, set it to NULL if the promotion item has been extracted successfully or to 'Delete Processed' (DP) if the item was marked for deletion before extraction.
  - Update PROMHEAD status to 'E' of all promotions that were extracted.
  - Update PROMHEAD status to 'M' of all promotions that were already extracted and end date is less than or equal to the extract date.
- If `btch_w_usr_ind = 'Y'`
  - In a switch, determine which function to call.



- If ps\_restart\_cursor = 1, update the status of extracted PROMSKU records, and then proceed to the next 2 functions.
- If ps\_restart\_cursor = 2, update the status of extracted PROMHEAD records, and then proceed to the last function.
- If ps\_restart\_cursor = 3, update the status of PROMHEAD records that are ending.

**Size\_array():** Sizes the fetch array to the commit size.

**Free\_array():** Frees fetch array.

**Process\_ext\_ps():** In a while loop, fetch the PROMSKU records to be updated to an array. Check if any of the records is locked by another process. If any records are locked, write the information about these records in the batch\_lock\_log table and then continue with the next batch of PROMSKU records. If no locks were found, update the status of the PROMSKU records to NULL if the promotion item has been extracted successfully or to 'Delete Processed' (DP) if the item was marked for deletion before extraction

**Check\_lock\_ext\_ps():** Checks if any of the PROMSKU records in the array are locked.

**Process\_ext\_ph():** In a while loop, fetch the PROMHEAD records to be updated to an array. Check if any of the records is locked by another process. If any records are locked, write the information about these records in the batch\_lock\_log table and then continue with the next batch of PROMHEAD records. If no locks were found, update the status of the PROHEAD records to 'E'.

**Process\_end\_ph():** In a while loop, fetch the PROMHEAD records to be updated to an array. Check if any of the records is locked by another process. If any records are locked, write the information about these records in the batch\_lock\_log table and then continue with the next batch of PROMHEAD records. If no locks were found, update the status of the PROMHEAD records to 'M'

**Check\_lock\_ph():** Checks if any of the PROMHEAD records in the array are locked.

**Final():** Standard Retek final function. Calls retek\_close().

## VII. Input Specifications

### 'Table-To-Table'

Select data from:

Table Name	Column Name	Column Type	Transformation
UNIT_OPTIONS	POS_EXTRACT_DAYS	NUMBER(2)	NONE
PERIOD	VDATE	DATE	NONE
SYSTEM_OPTIONS	BTCH_W_USR_IND	VARCHAR2(1)	NONE
PROMHEAD	PROMOTION	NUMBER(10)	NONE
PROMHEAD	ROWID	ROWID	NONE
PROMSKU	PROMOTION	NUMBER(10)	NONE
PROMSKU	ROWID	ROWID	NONE
PROMHEAD_LOCK_TEMP	PROMOTION	NUMBER(10)	NONE
PROMHEAD_LOCK_TEMP	ROW_ID	ROWID	NONE
PROMSKU_LOCK_TEMP	PROMOTION	NUMBER(10)	NONE
PROMSKU_LOCK_TEMP	ROW_ID	ROWID	NONE



## VIII. Output Specifications

### 'Table-To-Table'

**Delete from:** BATCH\_LOCK\_LOG  
PROMHEAD\_LOCK\_TEMP  
PROMSKU\_LOCK\_TEMP

### Update data on:

Table Name	Column Name	Column Type	Transformation
PROMHEAD	STATUS	VARCHAR2(1)	N/A
PROMSKU	STATUS	VARCHAR2(2)	N/A

### Insert into:

Table Name	Column Name	Column Type	Transformation
PROMHEAD_LOCK_TEMP	PROMOTION	NUMBER(10)	N/A
PROMHEAD_LOCK_TEMP	ROW_ID	ROWID	N/A
PROMSKU_LOCK_TEMP	PROMOTION	NUMBER(10)	N/A
PROMSKU_LOCK_TEMP	ROW_ID	ROWID	N/A

## IX. Scheduling Considerations

This module must be run after prmext.pc.

## X. Locking Strategy

Before processing a group of records, it will be locked first with the no wait clause. If this group of records includes a row that has already been locked by another application, the whole group will be skipped, a flag will be set, information about these records will be written to the batch\_lock\_log table, and a non-fatal error will be written in the log file. The batch will then continue processing the next group of records.

## XI. Restart/Recovery

This program has restart recovery based on promotion/cursor number combination.

## XII. Performance Considerations

N/A

## XIII. Security Considerations

N/A

## XIV. Unit Test Considerations



N/A

## **XV. Design Assumptions**

N/A

## **XVI. Outstanding Design Issues**

N/A

Issue Description	Priority	Resolution

## **XVII. Approval and Distribution**

The detailed design should be approved by:

Title	Name
Design Lead	

The detailed design should be distributed to:

Title	Name
Quality Control Lead	

## **XVIII. Appendix**

