

Batch Detail Design

I. Functional Area

Complex Deals Management

II. Module Affected

Precostcalc1.pc

III. Design Overview

This batch module is responsible for data maintenance tasks that are necessary before running precostcalc2.

A new indicator has been added to the system_options table called Batch with Online Users (btch_w_usr_ind). If the Batch with Online Users indicator is set to 'Y' this means that users can be online at any given time. Table views were created such that the program will determine the current COST_CHANGE_TRIGGER_TEMP and RECLASS_TRIGGER_TEMP table view, and then process records from the non-online application views. This program will then:

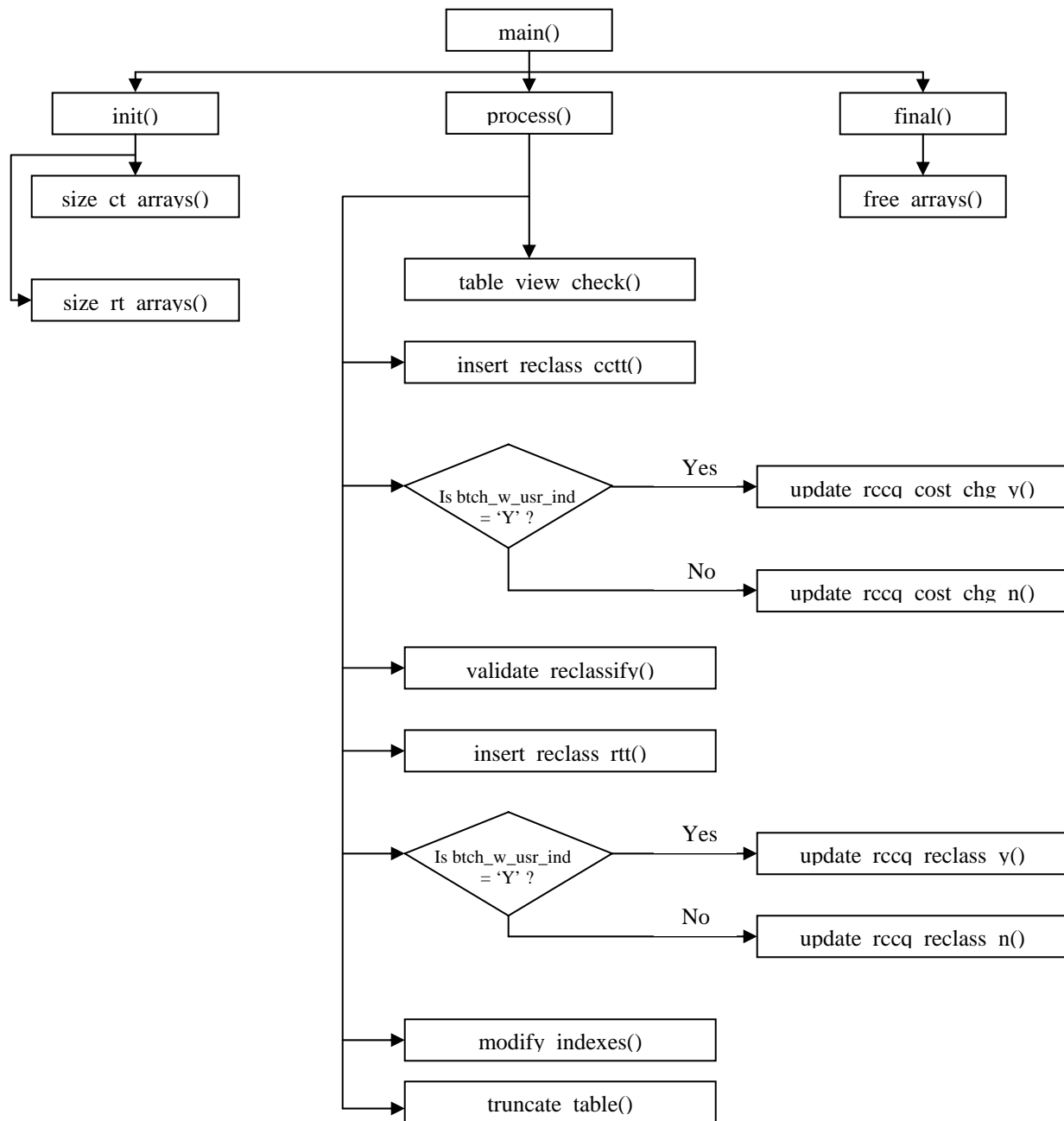
- Process records from the COST_CHANGE_TRIGGER_TEMP and RECLASS_TRIGGER_TEMP tables. (RECLASS_TRIGGER_TEMP is populated only by database trigger and COST_CHANGE_TRIGGER_TEMP is populated by database trigger and edi_cost_change_sql.create_cost_chg.)
- Insert new records or update existing ones on reclass_cost_chg_queue.
- Truncate and rebuilt indexes on both tables, COST_CHANGE_TRIGGER_TEMP and RECLASS_TRIGGER_TEMP, after all processing is done. (The user running this program for this function must have been granted the 'drop any table' and 'alter any index' system privilege, or be the owning schema user.)

IV. Stored Procedures / Shared Modules (Maintainability)

N/A



V. Program Flow



VI. Function Level Description

Declare a fetch array structs to hold array-fetched records from driving cursors.

Declare driving cursors Cursors will be declared to fetch records from the COST_CHANGE_TRIGGER_TEMP table view and RECLASS_TRIGGER_TEMP table view to update existing records in the RECLASS_COST_CHG_QUEUE table.

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

Init(): Standard Retek init function. Calls retek_init(), size_ct_arrays(), and size_rt_arrays(). If this is a new start, set the restart cursor variable to 1. Declare a cursor that will fetch btch_w_usr_ind from SYSTEM_OPTIONS and vdate from PERIOD.

Process(): This function will process records from the COST_CHANGE_TRIGGER_TEMP and RECLASS_TRIGGER_TEMP tables. The function will either insert into or update existing records on RECLASS_COST_CHG_QUEUE table. These inserts/updates need to be done since approved cost changes affect an item's future cost and reclassifications can affect future cost because supplier deals can be set up at merchandise hierarchy levels (e.g., department-level deals).

- Call table_view_check to return the current on-line application view for both COST_CHANGE_TRIGGER_TEMP and RECLASS_TRIGGER_TEMP tables.
- Call insert_reclass_cctt() if the program is not a rerun.
- If the btch_w_usr_ind is set to 'Y', call update_cost_chg_y(). Otherwise, call update_cost_chg_n().
- Call validate_reclassify().
- Call insert_reclass_rtt() if the program is not a rerun.
- If the btch_w_usr_ind is set to 'Y', call update_reclass_y(). Otherwise, call update_reclass_n().
- If the btch_w_usr_ind is set to 'Y', delete from records from the COST_CHANGE_TRIGGER_TEMP table view and RECLASS_TRIGGER_TEMP table view that does not exist in the BATCH_LOCK_LOG table. Otherwise, truncate the tables.

Modify_indexes(): Allows indexes to be disabled or rebuilt before and/or after the action that affects them. The individual program passes in the table name and mode (what action to take "disable" or "rebuild") and performs that action. The owner of the index is determined using the synonym_trace function in the library oracle.pc.

Modify_partition_indexes(): Called by the modify_indexes function to determine if the indexes that need modified are partitioned indexes. If so, then the statement is modified to take that into account to accomplish the action. Index_owner, index_name and mode is passed to this function. Nothing is passed back out.

Truncate_table(): The table_name is passed to this function so that it can be truncated. The owner of the table is determined by using the synonym_trace function in the library oracle.pc.

Check_lock(): This function will declare a cursor that will lock records on RECLASS_COST_CHG_QUEUE table based on rowid.

Validate_reclassify(): This function will perform a reclassification validation. If the reclassification is rejected, data will be deleted from the RECLASS_TRIGGER_TEMP table. Otherwise, data will be inserted into the RECLASS_COST_CHG_QUEUE table.

Delete_reclass_trigger_temp(): This function will delete a record from the RECLASS_TRIGGER_TEMP table based on the item passed as an argument to the function.



Insert_reclass_cttt(): This function will insert records from the COST_CHANGE_TRIGGER_TEMP table into the RECLASS_COST_CHG_QUEUE table.

Insert_reclass_rtt(): This function will insert records from the RECLASS_TRIGGER_TEMP table into the RECLASS_COST_CHG_QUEUE table.

Update_rccq_cost_chg_y(): If btch_w_usr_ind is set to 'Y', this function will lock records in the RECLASS_COST_CHG_QUEUE table and then update the unit_cost information or revert item/supplier/country/loc/start_date's future cost back to the pre-cost change value. If a lock is encountered, skip processing and fetch the next set of records.

Update_rccq_cost_chg_n(): If btch_w_usr_ind is set to 'N', this function will update the unit_cost information or revert item/supplier/country/loc/start_date's future cost back to the pre-cost change value from the RECLASS_COST_CHG_QUEUE table.

Update_rccq_reclass_y(): If btch_w_usr_ind is set to 'Y', this function will lock records in the RECLASS_COST_CHG_QUEUE table and then revert item/supplier/country/loc/start_date's future cost back to the pre-reclassification value. If a lock is encountered, skip processing and fetch the next set of records.

Update_rccq_reclass_n(): If btch_w_usr_ind is set to 'N', this function will revert item/supplier/country/loc/start_date's future cost back to the pre-reclassification value.

Table_view_check(): This function returns the permanent table that is set as the current table view.

Size_ct_arrays(): Sizes the fetch array table to the commit size for records fetched from the cost_change_trigger_temp.

Size_rt_arrays(): Sizes the fetch array table to the commit size for records fetched from the reclass_trigger_temp.

Free_arrays(): Frees fetch array.

Final(): Standard Retek final function. Calls free_arrays() and retek_close().

VII. Input Specifications

'Table-To-Table'

Select data from:

Table Name	Column Name	Column Type	Transformation
SYSTEM_OPTIONS	BTCH_W_USR_IND	VARCHAR2(1)	NONE
PERIOD	VDATE	DATE	NONE
USER_OBJECTS	OBJECT_NAME	VARCHAR2(128)	NONE
USER_OBJECTS	OBJECT_ID	NUMBER	NONE
PUBLIC_DEPENDENCY	REFERENCE_OBJECT_ID	NUMBER	NONE
DBA_INDEXES	INDEX_NAME	VARCHAR2(30)	NONE
DBA_INDEXES	OWNER	VARCHAR2(30)	NONE
DBA_INDEXES	PARTITIONED	VARCHAR2(3)	NONE
RECLASS_ITEM	RECLASS_NO	NUMBER(4)	NONE
RECLASS_ITEM	ITEM	VARCHAR2(25)	NONE
RECLASS_HEAD	TO_DEPT	NUMBER(4)	NONE
RECLASS_HEAD	TO_CLASS	NUMBER(4)	NONE
RECLASS_HEAD	TO_SUBCLASS	NUMBER(4)	NONE
ITEM_MASTER	ITEM	VARCHAR2(25)	NONE
ITEM_MASTER	ITEM_LEVEL	NUMBER(1)	NONE



ITEM_MASTER	TRAN_LEVEL	NUMBER(1)	NONE
ITEM_MASTER	DEPT	NUMBER(4)	NONE
ITEM_MASTER	CLASS	NUMBER(4)	NONE
ITEM_MASTER	SUBCLASS	NUMBER(4)	NONE
RECLASS_COST_CHG_QUEUE	ITEM	VARCHAR2(25)	NONE
RECLASS_COST_CHG_QUEUE	SUPPLIER	NUMBER(10)	NONE
RECLASS_COST_CHG_QUEUE	ORIGIN_COUNTRY_ID	VARCHAR2(3)	NONE
RECLASS_COST_CHG_QUEUE	START_DATE	DATE	NONE
RECLASS_COST_CHG_QUEUE	LOCATION	NUMBER(10)	NONE
RECLASS_COST_CHG_QUEUE	REC_TYPE	VARCHAR2(1)	NONE
RECLASS_COST_CHG_QUEUE	ROWID	ROWID	NONE
GROUPS	DIVISION	NUMBER(4)	NONE
DEPS	GROUP_NO	NUMBER(4)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	ITEM	VARCHAR2(25)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	SUPPLIER	NUMBER(10)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	ORIGIN_COUNTRY_ID	VARCHAR2(3)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	START_DATE	DATE	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	LOCATION	NUMBER(10)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	ROW_ID	ROWID	NONE
ITEM_SUPP_COUNTRY_LOC	ITEM	VARCHAR2(25)	NONE
ITEM_SUPP_COUNTRY_LOC	SUPPLIER	NUMBER(10)	NONE
ITEM_SUPP_COUNTRY_LOC	ORIGIN_COUNTRY_ID	VARCHAR2(3)	NONE
ITEM_SUPP_COUNTRY_LOC	LOC	NUMBER(10)	NONE
ITEM_SUPP_COUNTRY_LOC	LOC_TYPE	VARCHAR2(1)	NONE
COST_CHANGE_TRIGGER_TEMP	ITEM	VARCHAR2(25)	NONE
COST_CHANGE_TRIGGER_TEMP	SUPPLIER	NUMBER(10)	NONE
COST_CHANGE_TRIGGER_TEMP	ORIGIN_COUNTRY_ID	VARCHAR2(3)	NONE
COST_CHANGE_TRIGGER_TEMP	ACTIVE_DATE	DATE	NONE
COST_CHANGE_TRIGGER_TEMP	UNIT_COST	NUMBER(20,4)	NONE
COST_CHANGE_TRIGGER_TEMP	ROWID	ROWID	NONE
RECLASS_TRIGGER_TEMP	DEPT	NUMBER(4)	NONE
RECLASS_TRIGGER_TEMP	CLASS	NUMBER(4)	NONE
RECLASS_TRIGGER_TEMP	SUBCLASS	NUMBER(4)	NONE
RECLASS_TRIGGER_TEMP	RECLASS_DATE	DATE	NONE

VIII. Output Specifications

'Table-To-Table'

Delete from: RECLASS_COST_CHG_QUEUE, RECLASS_CC_QUEUE_LOCK_TEMP, BATCH_LOCK_LOG, COST_CHANGE_TRIGGER_TEMP_A/B, and RECLASS_TRIGGER_TEMP_A/B.

Update data on:

Table Name	Column Name	Column Type	Transformation
RECLASS_COST_CHG_QUEUE	DIVISION	NUMBER(4)	Set to NULL.
RECLASS_COST_CHG_QUEUE	GROUP_NO	NUMBER(4)	Set to NULL.
RECLASS_COST_CHG_QUEUE	DEPT	NUMBER(4)	Set to NULL.
RECLASS_COST_CHG_QUEUE	CLASS	NUMBER(4)	Set to NULL.
RECLASS_COST_CHG_QUEUE	SUBCLASS	NUMBER(4)	Set to NULL.
RECLASS_COST_CHG_QUEUE	REC_TYPE	NUMBER(4)	Set to G.



RECLASS_COST_CHG_QUEUE	UNIT_COST	NUMBER(20,4)	Set to NULL or to fetched unit_cost value.
RECLASS_COST_CHG_QUEUE	PROCESS_FLAG	VARCHAR2(1)	Set to N.

Insert into:

Table Name	Column Name	Column Type	Transformation
RECLASS_CC_QUEUE_LOCK_TEMP	ITEM	VARCHAR2(25)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	SUPPLIER	NUMBER(10)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	ORIGIN_COUNTRY_ID	VARCHAR2(3)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	START_DATE	DATE	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	UNIT_COST	NUMBER(20,4)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	LOCATION	NUMBER(10)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	REC_TYPE	VARCHAR2(1)	NONE
RECLASS_CC_QUEUE_LOCK_TEMP	ROWID	ROWID	
BATCH_LOCK_LOG	PROGRAM_NAME	VARCHAR2(25)	NONE
BATCH_LOCK_LOG	TABLE_NAME	VARCHAR2(32)	NONE
BATCH_LOCK_LOG	KEY_VALUE1	VARCHAR2(25)	NONE
BATCH_LOCK_LOG	KEY_VALUE2	VARCHAR2(25)	NONE
BATCH_LOCK_LOG	KEY_VALUE3	VARCHAR2(25)	NONE
BATCH_LOCK_LOG	KEY_VALUE4	VARCHAR2(25)	NONE
BATCH_LOCK_LOG	KEY_VALUE5	VARCHAR2(25)	NONE
BATCH_LOCK_LOG	KEY_ROWID	ROWID	NONE
BATCH_LOCK_LOG	LOCKED_DATE	DATE	NONE
BATCH_LOCK_LOG	THREAD_VAL	NUMBER(10)	NONE
ITEM_LOC_TEMP	ITEM	VARCHAR2(25)	NONE
ITEM_LOC_TEMP	LOC	NUMBER(10)	NONE
ITEM_LOC_TEMP	PRIMARY_COST_PACK	VARCHAR2(25)	NONE
RECLASS_COST_CHG_QUEUE	ITEM	VARCHAR2(25)	NONE
RECLASS_COST_CHG_QUEUE	SUPPLIER	NUMBER(10)	NONE
RECLASS_COST_CHG_QUEUE	ORIGIN_COUNTRY_ID	VARCHAR2(3)	NONE
RECLASS_COST_CHG_QUEUE	START_DATE	DATE	NONE
RECLASS_COST_CHG_QUEUE	UNIT_COST	NUMBER(20,4)	NONE
RECLASS_COST_CHG_QUEUE	LOCATION	NUMBER(10)	NONE
RECLASS_COST_CHG_QUEUE	DIVISION	ROWID	NONE
RECLASS_COST_CHG_QUEUE	GROUP_NO	NUMBER(4)	NONE
RECLASS_COST_CHG_QUEUE	DEPT	NUMBER(4)	NONE
RECLASS_COST_CHG_QUEUE	CLASS	NUMBER(4)	NONE
RECLASS_COST_CHG_QUEUE	SUBCLASS	NUMBER(4)	NONE
RECLASS_COST_CHG_QUEUE	LOC_TYPE	VARCHAR2(1)	NONE
RECLASS_COST_CHG_QUEUE	REC_TYPE	VARCHAR2(1)	NONE
RECLASS_COST_CHG_QUEUE	PROCESS_FLAG	VARCHAR2(1)	NONE

IX. Scheduling Considerations

This module must be run before precostcalc2.

X. Locking Strategy



XI. Restart/Recovery

This program has restart recovery based on item/supplier/origin country/location/start date.

XII. Performance Considerations

N/A

XIII. Security Considerations

N/A

XIV. Unit Test Considerations

- Run the program with btch_w_usr_ind set to 'N' and there are no locked records.
- Run the program with btch_w_usr_ind set to 'N' and there are locked records.
- Run the program with btch_w_usr_ind set to 'Y' and there are no locked records.
- Run the program with btch_w_usr_ind set to 'Y' and there are some locked records.
- Run the program with btch_w_usr_ind set to 'Y' and some of the locks on the records have been released.
- Run the program with btch_w_usr_ind set to 'Y' and all of the locks on the records have been released.

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

