

Clearance Reset Pricing POS Extract [pccrdnld]

Design Overview

The Clearance Reset Pricing Information (pccrdnld) module is used to send clearance reset pricing information to the point of sale system. When a clearance event is within a pre-determined number of days of its reset date, this program will gather current clearance pricing information and the pricing information to which an item will be reset. This information will be written out to a table that will be used as an interface point with the point of sale. Clearance records will be updated to indicate that they have had reset prices downloaded to the point of sale by setting a reset downloaded date with the current date.

If the Batch with Online Users indicator is set to 'Y', it will bulk lock the clear_reset_calc records first to check for record locking for a given set of records before performing the necessary updates. If the records are locked by another application, they will not be processed, thus an update to the clear_reset_calc table for the set of records will not happen. Otherwise, the records will be processed and an update to the clear_reset_calc table for the set of records will happen.

Tables	Index	Select	Insert	Update	Delete
PERIOD	No	Yes	No	No	No
UNIT_OPTIONS	No	Yes	No	No	No
CLEAR_RESET_CALC	No	Yes	No	Yes	No
CLEAR_SUSP_HEAD	No	Yes	No	No	No
CLEAR_SUSP_DETAIL	No	Yes	No	No	No
PRICE_ZONE_GROUP_STORE	Yes	Yes	No	No	No
POS_MODS	No	No	Yes	No	No
ITEM_MASTER	Yes	Yes	No	No	No
ITEM_LOC	Yes	Yes	No	No	No
ITEM_ZONE_PRICE	Yes	Yes	No	No	No
SYSTEM_OPTIONS	No	Yes	No	No	No
BATCH_LOCK_LOG	No	No	Yes	No	Yes

Scheduling Constraints

Processing Cycle: Phase 1

Scheduling Diagram: N/A

Pre-Processing: pcdnld should complete processing before this module begins

Post-Processing: N/A

Threading Scheme: Clearance

Restart Recovery

The driving cursor was changed because of changes to the tables and also the system indicator is no longer used.

```
EXEC SQL DECLARE c_get_reset CURSOR FOR  
SELECT crc.clearance,
```

```
        crc.item,
        TO_CHAR(crc.reset_date, 'DDMMYYYY'),
        pzgs.store,
        ROWIDTOCHAR(crc.rowid),
        im.item_level,
        ';' || TO_CHAR(crc.clearance) ||
        ';' || crc.item ||
        ';' || TO_CHAR(crc.zone_group_id) ||
        ';' || TO_CHAR(crc.zone_id)
FROM clear_reset_calc crc,
     clear_susp_head csh,
     price_zone_group_store pzgs,
     item_master im,
     v_restart_clearance rv
WHERE crc.reset_date is NOT NULL
AND (
    (csh.status = 'C'
    AND EXISTS (SELECT 'x'
                FROM clear_susp_detail csd
                WHERE csd.clearance = csh.clearance
                AND csd.item = crc.item
                AND csd.zone_group_id = crc.zone_group_id
                AND csd.zone_id = crc.zone_id
                AND csd.downloaded_date is NOT NULL))
    OR
    (csh.status = 'A'
    AND :pd_pos_extract_days >= (crc.reset_date -
to_date(:ps_vdate, 'DDMMYYYY'))
    )
)
AND csh.clearance = crc.clearance
AND crc.zone_group_id = pzgs.zone_group_id
AND crc.zone_id = pzgs.zone_id
AND crc.reset_downloaded_date is NULL
AND crc.item = im.item
AND im.status = 'A'
AND im.pack_ind = 'N'
AND im.item_level = im.tran_level
AND rv.driver_value = csh.clearance
AND rv.driver_name = :ps_restart_driver_name
AND rv.num_threads = :pi_restart_num_threads
AND rv.thread_val = :pi_restart_thread_val
ORDER BY crc.clearance,
        crc.item,
        crc.zone_group_id,
        crc.zone_id;
```

Program Flow

N/A

Shared Modules

N/A

Function Level Description

N/A

I/O Specification

N/A

Technical Issues

N/A