

Pre-Processing:	pccext and pccext must complete before this module begins
Post-Processing:	N/A
Threading Scheme:	Clearance

Restart Recovery

```
EXEC SQL DECLARE c_reset CURSOR FOR
SELECT csh.clearance,
       csh.status,
       csh.reason,
       crc.item,
       crc.zone_group_id,
       crc.zone_id,
       NVL(TO_CHAR(crc.reset_date,'DDMMYYYY'), :today),
       pzgs.store,
       csh.rowid,
       '||TO_CHAR(csh.clearance)||'
       '||crc.item||'
       '||TO_CHAR(pzgs.store)
FROM v_restart_clearance rv,
     price_zone_group_store pzgs,
     clear_reset_calc crc,
     clear_susp_head csh
WHERE ((csh.status = 'A'
       AND crc.reset_date = to_date(:tomorrow, 'DDMMYYYY'))
      OR (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)))
       AND csh.clearance = crc.clearance
       AND crc.zone_group_id = pzgs.zone_group_id
       AND crc.zone_id = pzgs.zone_id
       AND rv.driver_value = csh.clearance
       AND rv.driver_name = :ps_restart_driver_name
       AND rv.num_threads = TO_NUMBER(:ps_restart_num_threads)
       AND rv.thread_val = TO_NUMBER(:ps_restart_thread_val)
       AND (csh.clearance > NVL(:ps_restart_clearance, -999) OR
          (csh.clearance = :ps_restart_clearance AND
           (crc.item > :ps_restart_item OR
            (crc.item = :ps_restart_item AND
             (pzgs.store > :ps_restart_store)))))
ORDER BY csh.clearance, item, store;
```

Program Flow

N/A

Shared Modules

N/A

Function Level Description

N/A

I/O Specification

N/A

Technical Issues

N/A