Error Descriptions Addendum for 1st Quarter 2013 Technical Release

Error 2251:

Title: Programs with Select *

DBA sent a list of programs which include either "SELECT *" or INSERTs without column specification.

Programs with this condition should be cleaned up.

Error 2270:

Title: References to UCOP table/view name instead of SYSIBM.SYSDUMMY1 - SELECT CURRENT TIMESTAMP

Some programs also show SQL statements coded SELECT CURRENT TIMESTAMP INTO :H FROM UCOP-table-name

These types of statements should not reference a UCOP table/view. They must use SYSIBM.SYSDUMMY1 instead of a UCOP table/view name.

There are several reasons for correcting this.

The most important is that the UCOP view/table may contain more than one row. When this is the case an SQLCODE -811 occurs.

Problems caused by this condition:

1) DB2 takes extra steps to handle this error.
2) The -811 SQLCODE is a ‘false positive’ (red herring) that confuses problem resolution for no good reason.
3) The program code probably doesn’t properly test the SQLCODE after the statement. This is a bad habit others may emulate.
4) Coders may not discover the -811 error if only 1 row exists in a referenced UCOP table/view, until another row is inserted. By then it is in Production, and causes an error.

Another reason is that the table/tablespace may be unavailable for some odd reason, causing the app error to mislead an investigator.

This type of SQL should be revised when identified.

Error 2436:

Title: Conflicting USING lists when calling program UCPPPEDB

Calls to program UCPPPEDB are made from programs:
This makes it impossible to compile UCPPPEDB once for use in all runtime environments, i.e. it must be compiled with the CICS translator to correctly process calls from PS001, PS028, and UCARSM; it must be compiled without the CICS translator to correctly process calls from UCSWP03. This means that it is impossible to compile UCPPPEDB using the compile process currently in use for production PPS compiles.

The correction required a change to programs PS001, PS028, and UCARSM to call UCPPPEDB USING ARSM-INTERFACE and recompile program UCPPPEDB using compile type "DUAL-ONLINE ONLY, SPAS".

**Error 2442:**

**Title:** PS005 SQL change

DBA posted the following request, which should be made in program PS005 to reduce CPU cost.

Change SQL:

```
SELECT EMPLOYEE_ID
FROM PPPPER
WHERE EMP_NAME LIKE UCASE (:H)
```

To:

1. Use COBOL to convert the Host variable to Upper Case.
2. Change the SQL, eliminating use of the UCASE function, to

```
SELECT EMPLOYEE_ID
FROM PPPPER
WHERE EMP_NAME LIKE (:H)
```

**Error 2445:**

**Title:** PPP465 Fails on insert to PPPEUD

The EDB contains a record of the latest priority gross control number for each employee. Every time there is PAR activity, the number – functionally like a generation number – is bumped up. It’s updated in the EDB and written to the PAR, where it should be unique for the employee.

However, when an employee has a break in service where the employee is removed from the EDB these records are not removed. When an employee is then rehired the employee is assigned a new PGCN and that count restarts at 0001. When the rows are already present from the previous employment the batch process fails.

Current support response is to delete the old information so that the process can continue.
There exists a local change in PPP465 for UCLA (only used for their informational loads) which addresses this issue by replacing the row with the new one. An alternate, possibly more correct, option would be to determine the correct new PGCN and insert the row using that number.

**Error 2448:**

**Title:** PPP465 SQL revision - payroll batch performance issue.

**DBA reports:**

1. Program PPP465 contains this SQL statement:

   ```sql
   DELETE FROM PPPVZPIR_PIR WHERE NOT PAY_CYCLE_END_DATE >= :H
   ```

   This statement cannot use an existing index because of the word 'NOT' in the WHERE clause. As a result, a DB tablespace scan occurs resulting in, for instance, over 5 million 'DB2 getpages' for San Diego. Job PPP465 runs at critical processing 'bottleneck' periods when several campuses may be processing concurrently. The excessive I/O that results from this DELETE statement contributes to noticeable processing delays.

   Please change this WHERE clause to state the condition in a positive declaration rather than a negative one. It might be something like:

   ```sql
   DELETE FROM PPPVZPIR_PIR WHERE PAY_CYCLE_END_DATE < :H
   ```

**Error 2450:**

**Title:** PS001 select from PPPDET should only be done when needed

```sql
SELECT DET_DEC_PLACE INTO : H
FROM PPPVZDET_DET
WHERE DET_CARD_TYPE = '1' AND DET_ELEM_NO = : H
```

The above Select statement occurs at a very high volume in CICS. That would be hundreds of thousands of times an hour.

Please review the design for invoking this SQL and revise it to be invoked only as needed