Service Request 83280
End Benefits Separation Month

Technical Specification
Version History

<table>
<thead>
<tr>
<th>Version #</th>
<th>Date</th>
<th>Revised By</th>
<th>Reason for Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>09/6/2011</td>
<td>Caroline Rider / Maxine Gerber</td>
<td>Initial version</td>
</tr>
<tr>
<td>1.1</td>
<td>09/06/2011</td>
<td>Maxine Gerber</td>
<td>Made the following changes:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- For checking whether PPBUTSEP was successful or not, it is enough to check only the return status – no need to check the return value as well.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- PPRCNET was updated so that the current/future decision matches PPP400</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- PPOT2033 was updated to include a REPORT/UPDATE option.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- PPOT2033 was updated to include a check for enrollment in a benefit prior to setting the end date.</td>
</tr>
<tr>
<td>1.2</td>
<td>09/07/2011</td>
<td>Maxine Gerber</td>
<td>fixed a typo in 2.1.2</td>
</tr>
<tr>
<td>1.3</td>
<td>03/02/2012</td>
<td>Caroline Rider</td>
<td>Added modifications to include separation reason in tests for coverage end dates, and dependent coverage for deaths.</td>
</tr>
<tr>
<td>1.4</td>
<td>03/15/2012</td>
<td>Caroline Rider</td>
<td>Marked the Rush Check modifications as optional, other fixes</td>
</tr>
<tr>
<td>1.5</td>
<td>6/1/2012</td>
<td>Sreelekha Sen</td>
<td>Removed changes to copybook CPWSW88S and changed related code</td>
</tr>
</tbody>
</table>
Table of Contents

Version History .................................................................................................................................................................1

1 Introduction........................................................................................................................................................................3
  1.1 Service Request 83280..................................................................................................................................................3
  1.2 Error Report(s).........................................................................................................................................................3

2 Overview of System Modifications .................................................................................................................................3
  2.1 CTL Maintenance ........................................................................................................................................................3
  2.2 EDB Maintenance Process ........................................................................................................................................3
  2.3 Compute .....................................................................................................................................................................3
  2.4 One-Time Process ....................................................................................................................................................4
  2.5 Rush Checks (Optional) ..........................................................................................................................................4

3 Design Considerations ..................................................................................................................................................4
  3.1 Assumptions and Dependencies.................................................................................................................................4

4 Mainframe Design ..........................................................................................................................................................4
  4.1 CTL Maintenance ........................................................................................................................................................4
    4.1.1 Database Changes ...............................................................................................................................................4
    4.1.2 Copy Members ....................................................................................................................................................5
    4.1.3 COBOL Programs ............................................................................................................................................5
    4.1.4 UPAY Forms ....................................................................................................................................................6
    4.1.5 Control Table Updates .......................................................................................................................................6
  4.2 EDB Maintenance ......................................................................................................................................................6
    4.2.1 COBOL Programs ............................................................................................................................................6
    4.2.2 Copy Members ....................................................................................................................................................9
    4.2.3 Bind Members ....................................................................................................................................................9
    4.2.4 Control Table Updates ......................................................................................................................................9
  4.3 Compute ....................................................................................................................................................................9
    4.3.1 COBOL Programs ...............................................................................................................................................9
    4.3.2 Bind Members ...................................................................................................................................................10
    4.3.3 Control Table Updates ....................................................................................................................................10
  4.4 One-Time Process ....................................................................................................................................................11
    4.4.1 Cobol Programs ...............................................................................................................................................11
    4.4.2 Bind Members ...................................................................................................................................................12
    4.4.3 JCL Changes ....................................................................................................................................................12
  4.5 Rush Checks (Optional) .........................................................................................................................................12
    4.5.1 COBOL Programs ............................................................................................................................................12
    4.5.2 Copy Members ..................................................................................................................................................15
    4.5.3 Control Table Updates ..................................................................................................................................16

5 Unit Testing Requirements ..........................................................................................................................................16
  5.1 CTL Maintenance ....................................................................................................................................................16
  5.2 One-Time Process ...................................................................................................................................................16
  5.3 EDB Maintenance Process .....................................................................................................................................16
  5.4 Compute ..................................................................................................................................................................17
  5.5 Rush Checks (Optional) .......................................................................................................................................17
1 Introduction

1.1.1 Service Request 83280
SR83280 asks that coverage for benefits that are paid in advance (medical, dental, vision, legal, Accidental Death and Dismemberment (AD&D), and employee-paid life insurance and dependent life insurance) stop on the last day of the separation month. Deductions (and matching contributions) should not be taken during the separation month. If the separation reason is retirement, coverage should always continue for one month beyond the separation month. If the separation reason is death, coverage for each of the medical, dental, vision, and legal benefits will depend on whether the employee has dependent coverage for the benefit (if the employee has dependents who would normally be covered in the month beyond the separation month, coverage should continue for one month beyond the separation month; otherwise, coverage should end with the separation month); AD&D, life insurance, and dependent life insurance will always continue for one month beyond the separation month.

1.1.2 Error Report(s)
None

2 Overview of System Modifications
Currently, when a separation is recorded on the EDB, coverage end dates for medical, dental, vision, and legal are set to the last day of the month following the separation month. This request asks that the coverage end dates for those benefits are set to the last day of the separation month, and that no deduction or contribution is taken during the separation month. In addition, no deductions should be taken for AD&D or employee-paid life or dependent life insurance during the separation month. The new rule is covered by collective bargaining, so whether the new rule or the old rule applies depends on the employee’s benefit eligibility codes (EBEUC and EBERC), and there are exceptions for separation reasons of retirement (coverage is always allowed for one month beyond the separation date) and death (coverage is always allowed for one month beyond the separation date for benefits with covered dependents).

2.1.1 CTL Maintenance
A new indicator will be added to the Bargaining Unit Table (BUT) that will indicate whether the new rule applies to the bargaining unit.

2.1.2 EDB Maintenance Process
The process that sets the employee coverage end dates for medical, dental, vision, and legal insurances will be modified to use the old rule (last day of the month after the separation month) or the new rule (last day of the separation month) depending on the employee’s EBEUC and EBERC, and on the employee’s separation reason and, for separation reason of death, the employee’s dependent coverage.

2.1.3 Compute
The deduction process currently compares the coverage end dates for medical, dental, vision, and legal insurances to the pay cycle dates to determine whether to take the deduction or contribution; it should require no further modification for those benefits, because the coverage dates will be set by the EDB maintenance process. For AD&D, life, and dependent life, the process will be modified to construct a working coverage end date from the separation month and year, depending on whether the employee’s EBEUC and EBERC are following the old or new rule for separations, as well as the employee’s
separation reason. The date will be compared to the pay cycle month and year to determine whether to take the AD&D, life, and/or dependent life insurance deductions.

2.1.4 One-Time Process
A new one-time process will be developed to modify coverage end dates for employees with separations after a spec-card defined date. The process will need to consider the employee’s separation reason and dependent coverage as well.

2.1.5 Rush Checks (Optional)
At this point, rush check modifications may not be necessary. Rush Checks function OPT1, which calculates and generates rush check transactions, does not automatically calculate the health and welfare deductions; Rush Checks function OPT2, which can be used for modeling the effects of pay and deductions, is not likely to be used for employees who are separating.

3 Design Considerations

3.1 Assumptions and Dependencies

- The compute process currently compares the coverage end dates for medical, dental, vision, and legal insurances to the pay cycle to determine whether those deductions (and any matching contributions) are taken; if we modify EDB maintenance to set the coverage end dates according to the new rule, no further modifications are required for those benefits.
- Dependent coverage for separation reasons of death will be evaluated based on the dependent coverage for the month following the separation month.
- Rush Check modifications may not be necessary.

4 Mainframe Design

4.1 CTL Maintenance

4.1.1 Database Changes

4.1.1.1 PPPBUT Table Change
The PPPBUT table contains the basic data for a bargaining unit of the Bargaining Unit Table. It will be modified to include a 1-character flag to indicate whether the new rule applies to the bargaining unit.

4.1.1.1.1 DDL Members
- TBBUT00C is the DDL to create the PPPBUT table. It will be modified to include the new indicator.
- TBBUT03A (new) will be the DDL to alter the PPPBUT table by adding the new indicator.
- PPPVZBUT is the DDL to create the full-table view PPPVZBUT_BUT of the PPPBUT table. It will be modified to include the new indicator.
4.1.1.2 Include Members

- PPPVZBUT is the layout for the full table view PPPVZBUT_BUT of the PPPBUT table. It will be modified to include the new indicator.

4.1.2 Copy Members

4.1.2.1 CPCTBUTI

CPCTBUTI defines the working input for the update of the PPPBUT table. It will be modified to include the new indicator in column 40, as defined on the modified UPAY565A.

4.1.3 COBOL Programs

4.1.3.1 PPCTT17T

PPCTT17T is the transaction manager for updates to the basic data of the Bargaining Unit Table (control table 17). It will be modified to populate the new indicator in the CPCTBUTI copylib member with the data from the input transaction.

WORKING-STORAGE
Add a one-byte field to BUTT-DATA-5 and reduce the filler.

PROCEDURE DIVISION
In section 2200-MOVE-TRANSACTION-DATA, add code to move the new field from BUTT-DATA-5 to the new field in BARG-UNIT-BUT-TABLE-INPUT (described by copylib member CPCTBUTI).

4.1.3.2 PPCTBUTE

PPCTBUTE edits the transactions that will update the PPPBUT (Bargaining Unit Table -- Unit Definition) control table. It will be modified to allow values of “Y” or “N” for the new indicator.

WORKING-STORAGE
Add any new messages

PROCEDURE DIVISION
In sections 2000-ADD-BUT and 2100-CHANGE-BUT, add code to allow only “Y” or “N” for the new indicator. Reject the transaction with an error message if a different value is input.

4.1.3.3 PPCTBUTU

PPCTBUTU updates the PPPBUT (Bargaining Unit Table -- Unit Definition) control table. It will be modified to update the new indicator from the input transactions.

PROCEDURE DIVISION
Modify sections 2000-INSERT-BUT and 2100-UPDATE-BUT to insert/update the new indicator.

4.1.3.4 PPCTR17

PPCTR17 prints the Bargaining Unit Table (control table 17). It will be modified to include the new indicator on the report.

WORKING-STORAGE
Add a new line following “UDUE RETENTION” for displaying the new indicator. The field label should be “BEN SEP MONTH”.

PROCEDURE DIVISION
In section 2100-PRINT-BUT-TABLE, add code to print “YES” or “NO” based on the value of the new indicator in PPPVZBUT and print the new line.

4.1.3.5 Programs Requiring Recompile

PPBUTFET

4.1.4 UPAY Forms

4.1.4.1 UPAY565A

UPAY565A contains the layouts for selected transactions to update the Bargaining Unit Table in the CTL database. It will be modified to include the new BEN SEP MONTH indicator in transaction U5, column 40.

4.1.5 Control Table Updates

4.1.5.1 CTL Database

UPAY forms are included with the requirements document and will contain detailed CTL update transactions. These will be keyed by the programmer and applied to the CTL database by running program PPP004.

4.1.5.1.1 System Messages Table (MSG)

A new message will be required for program PPCTBUTE when an invalid value is entered on the input transaction for the BEN SEP MONTH. The severity should be 5.

4.1.5.1.2 Bargaining Unit Table (BUT)

The new BEN SEP MONTH indicator in the BUT must be initialized for all bargaining units. Information will be provided by the Business Partners prior to release.

4.2 EDB Maintenance

4.2.1 COBOL Programs

4.2.1.1 PPBUTSEP (new)

PPBUTSEP will be a new called program which will determine whether an employee’s coverage end dates should be derived with the new rule (last day of separation month) or with the old rule (last day of the month following the separation month). This program will be called by

- PPEI330 for derivation of the coverage end dates for the EDB
- PPP400 for derivation of a “working” coverage end date to determine whether to take deductions for AD&D, life, and dependent life.
- (Optional) PPRCNET for derivation of a “working” coverage end date to determine whether to take deductions for AD&D, life, and dependent life.

Inputs and Outputs

Input to this sub-program will be via linkage parameters. The input parameters to this program will be as follows:
Output from this sub-program will be via linkage parameters. The output parameter from this program will be as follows:

- **Return Flag**: Flag to indicate whether or not to use the new rule for coverage end date. Values are
  - "Y" to use the new rule
  - "N" to use the old rule

- **Status Code** to indicate the overall status of the program. Values are
  - "0" for no errors
  - "1" for DB2 errors
  - "9" for other errors

**WORKING-STORAGE**

- Include (new) copylib member CPLNKBSP which will define the linkage area layout.
- Include PPPVZBUT and all DB2 artifacts.

**PROCEDURE DIVISION**

Logic will be as follows:

- Initialize the output fields
- If the input parameter EBERC code = “C”, then do the following:
  - Using the input parameter EBEUC code as the key, obtain the new Ben Sep Month field from PPPVZBUT. If the EBEUC code is not found on the BUT table, obtain the field from the default record for the unrepresented population (key = "<<").
  - If the Ben Sep Month indicator is “N” or blank, then set the return flag to “N” indicating that the old rule is in effect; otherwise set it to “Y”, indicating the new rule.
- If the input parameter EBERC code is NOT = “C”, then set the return flag to “Y” indicating that the new rule is in effect.
- If there are any errors, set the Status Code appropriately.

**4.2.1.2 PPEI330**

PPEI330 performs maintenance on the insurance enrollments if there is a separation action. It will be modified to call new subprogram PPBUTSEP with the employee’s EBEUC and EBERC. If the returned value is ‘Y’, the coverage end dates for medical, dental, vision, and legal will use the new rule; otherwise, they will use the old rule.

**WORKING-STORAGE**

- Add copylib CPLNKBSP which contains the linkage to program PPBUTSEP.
- New error message (see Control Table Updates below).
- Add CPLNKCVC2 (PPCOVEDB-INTERFACE)
- Add CPWSW88S (COLUMN-88-WORK)

PROCEDURE DIVISION

At the beginning of section 2000-SEP-DERIVE-COV-END-DATE, replace the code that sets WS-ISO-LAST-DAY-COVERAGE with the following:

- Set WS-ISO-SAME-MONTH-END to the last day of the separation month:
  - move SEPARATE-DATE to WS-ISO-LAST-DAY-COVERAGE
  - move 1 to WS-LAST-DAY-DD
  - move WS-ISO-LAST-DAY-COVERAGE to XDC3-ISO-DATE
  - perform XDC3-GET-END-OF-MONTH-DATE
  - move XDC3-ISO-DATE to WS-ISO-SAME-MONTH-END

- Set WS-ISO-NEXT-MONTH-END to the last day of the month after the separation month:
  - move SEPARATE-DATE to WS-ISO-LAST-DAY-COVERAGE
  - move 1 to WS-LAST-DAY-DD
  - move WS-ISO-LAST-DAY-COVERAGE to XDC3-ISO-DATE
  - move 1 to XDC3-NUMBER-OF-MONTHS
  - perform XDC3-OFFSET-MONTHS
  - perform XDC3-GET-END-OF-MONTH-DATE
  - move XDC3-ISO-DATE to WS-ISO-NEXT-MONTH-END

- Add an error handling routine similar to the one in program PPE1199, section 6710-PPDUEUTL-ERROR. SET KRMI-RETURN-WITH-HALT TO TRUE. A new message should be added to the message table for this purpose (see Control Table Updates below).

- Call PPBUTSEP with the employee’s BEN-ELIG-UNIT-CD (EBEUC) and BEN-ELIG-REP-CD (EBERC).

- If the return status is not “0”, it’s an error. Set a message number and perform the error routine.

- If the returned value is “Y”, move WS-ISO-LAST-DAY-COVERAGE.

- If the returned value is “N”, move WS-ISO-LAST-DAY-COVERAGE.

- Move SEPARATE-REASON to W88-SEPARATE-REASON.

- If SEPARATE-REASON-RETIRE-REG or SEPARATE-REASON-RETIRE-VOL or SEPARATE-REASON-RETIRE-EARLY or SEPARATE-REASON-RETIRE-COMPULS or SEPARATE-REASON = 'RF', move WS-ISO-LAST-DAY-COVERAGE.

- Prior to each of the tests for medical, dental, vision, and legal insurances, call PPCOVEDB to derive dependent coverage for the benefit for the month after the separation month if the separation reason is death – see PPEC722 for an example of medical coverage derivation.

  - if SEPARATE-REASON-DEATH
    - move ‘M’/’D’/’V’/’J’ to KCV2-BENEFIT-TYPE
    - move WS-ISO-LAST-MONTH-END to KCV2-PLAN-END-DATE
    - move SEPARATE-DATE to KCV2-EFFECTIVE-DATE
    - call PPCOVEDB using PPCOVEDB-INTERFACE, PPPDEP-ARRAY
    - if KCV2-NORMAL
4.2.2 Copy Members

4.2.2.1 CPLNKBSP (new)
CPLNKBSP will define the linkage area for subroutine PPBUTSEP. See program description for input and output parameters.

4.2.3 Bind Members

4.2.3.1 PPBUTSEP (new)
New Package bind member for PPBUTSEP.

4.2.3.2 PPP080
Plan bind member for PPP080 will be modified to include PPBUTSEP.

4.2.3.3 PPP120
Plan bind member for PPP120 will be modified to include PPBUTSEP.

4.2.4 Control Table Updates

4.2.4.1 System Messages Table (MSG)
New messages will be added to the System Messages table to handle invalid return from PPBUTSEP. These messages should all have severity = 5.

4.3 Compute

4.3.1 COBOL Programs

4.3.1.1 PPP400
PPP400 performs Gross-to-Net processing within the compute and creates a PAR file from the EDB and the preliminary PAR file with gross-to-net activity. It will be modified to call new subprogram PPBUTSEP with the employee’s EBEUC and EBERC. If the returned value is ‘Y’ (new rule) and the employee’s separation date is in the current month, deductions for Employee-Paid Life, Dependent Life and AD&D are not taken; otherwise, they are. The deductions are always taken if the separation date is in the current month and the separation reason is retirement or death.

WORKING-UNITS
- Add copylib CPLNKBSP which contains the linkage to program PPBUTSEP.
- Add a field to hold the “working” coverage end date for Employee-Paid Life, Dependent Life and AD&D.
- New error message (see Control Table Updates below).

PROCEDURE DIVISION
In section CURRENT-ACTVTY-PROCESSING-0800, if the employee has a separation date in the current or future month, derive a working coverage end date for Employee-Paid Life, Dependent Life and AD&D as follows:
- Call PPBUTSEP with the employee’s BEN-ELIG-UNIT-CD (EBEUC) and BEN-ELIG-REP-CD (EBERC).
If the return status is not “0”, it’s an error. Set a message number and perform CALL-MESSAGE-HANDLER-6080.

If the return value is “Y”, then derive the working coverage end date as the last day of the separation month, otherwise it is the last day of the month after the separation month.

If SEPARATE-REASON is 'KA' OR 'RA' OR 'RB' OR 'RC' OR 'RD' OR 'RF' or SEPARATE-REASON is ‘KA’, set the working coverage end date to the last day of the month after the separation month.

Note that this has the following effect, as per the requirements:
If the separation date is in a future month the deductions should be taken.
If the separation date is in the current month, the employee’s EBERC is ‘C’ and the Ben Sep Month for the employee’s EBEUC is ‘N’ or blank, take the deductions.
If the separation date is in the current month and the separation reason is retirement or death, take the deductions.
For any other case, (i.e., BEN SEP MONTH is not “N” or blank or separation date is in a prior month or the reason is not retirement or death) do not take the deductions.

In section CREATE-DEDUCTION-TO-TAKE-1220,

The following code snippet looks at deductions which are Insurance premiums (XGTN-GROUP = ‘I’), and determines whether the coverage end date has been reached for each XGTN-BENEFIT-TYPE. If it has, it sets EFDT-DONT-TAKE-DEDUCTION to TRUE.

```
IF XGTN-GROUP = 'I'
   EVALUATE TRUE
   WHEN (XGTN-BENEFIT-TYPE = 'H')
      IF HEALTH-COVEND-DATE OF BEN-ROW-DATA
         NOT = XDC3-LOW-ISO-DATE
         AND HEALTH-COVEND-DATE OF BEN-ROW-DATA
         < ISO-PREM-COV-MONTH-DATE
         SET EFDT-DONT-TAKE-DEDUCTION TO TRUE
   END-IF
   ...
```

Currently, the EVALUATE only looks for types H, D, V, J. Similar code will be added for types L (Employee-Paid Life), Y (Dependent Life) and A (AD & D) as follows:

- If the working coverage end date is prior to ISO-PREM-COV-MONTH-DATE, set EFDT-DONT-TAKE-DEDUCTION to TRUE.

In section SQL-SELECT-PER-9700, add SEPARATE-REASON to the items returned from the SELECT.

4.3.2 Bind Members

4.3.2.1 PPP400

Plan bind member for PPP400 will be modified to include PPBUTSEP.

4.3.1 Control Table Updates

4.3.1.1 System Messages Table (MSG)

New messages will be added to the System Messages table to handle invalid return from PPBUTSEP. These messages should all have severity = 5.
4.4 One-Time Process

4.4.1 Cobol Programs

4.4.1.1 PPOT2033

This new one-time program will modify coverage end dates for employees with separations after a spec-card defined date, and will also consider the separation reason and dependent coverage for the separation reason of death. The program will have a “report only” run mode so that output files can be produced without updating the EDB.

Inputs and Outputs

Input

- Spec card with cut-off date for future separations. Only employees with separation dates after the spec card date will have their benefits coverage end date adjusted.
- PPPBEN table.
- PPPPER table.
- PPPSCR table.
- PPPDEP array.

Output

- Updates to PPPBEN table.
- Employee Change File.
- Tab-delimited file listing original and updated coverage end dates. See Attachment A of BRD for layout.
- Totals report

WORKING-STORAGE

- Copylib CPLNKBSP which contains the linkage to program PPBUTSEP.
- Record layout for spec card, which will include the cutoff date and run mode “REPORT” or “UPDATE”.
- Record layouts (INCLUDE) for PPPBEN and PPPPER tables.
- CPWSEDEP (PPPDEP-ARRAY).
- CPWSW88S (COLUMN-88-WORK) Record layout for the output tab-delimited file.
- Copylib CPWSXCHG which contains the record layout for the employee change file (ECF).
- Copylib CPWSXSCR which contains the record layout for the System Control Record, which will be used to populate the ECF header.
- Appropriate totals for display on the totals report.
- Cursor definition on PPPPER table to obtain employee IDs, separation date, and separation reason for any employee with a separation date later than the spec card date.

PROCEDURE DIVISION

Initialization

- Validate spec card date and run mode
- Open the cursor on the PPPPER. Fetch the first row. If none found, abort.
- Read the PPPSCR record.
- Populate and write the ECF header record
Loop through the cursor. For each employee ID found:

- If the separation reason is death, call PPDEPUTL for the employee (to load PPPDEP-ARRAY).
- Follow the logic in PPEI330 for determining the coverage end dates for medical, dental, vision, and legal plans; don’t use the KRMI/KMTA error-handling mechanism, but do issue warning/abend messages for the same error conditions in the totals report
- For each enrolled plan type, check the COVEND_DATE field in the PPPBEN row and if it is not equal to the derived coverage end date:
  - If UPDATE mode, update the EDB field with the derived coverage end date
  - Write an ECF record with the change.
  - Write a record to the tab-delimited file
  - Add to the totals in working storage

Wrap-up

- Write the totals report

4.4.2 Bind Members

4.4.2.1 PPOT2033

Plan bind member for PPOT2033. This will include PPBUTSEPD.

4.4.3 JCL Changes

4.4.3.1 PPOT2033

Sample JCL will be provided for running the one-time program PPOT2033.

4.5 Rush Checks (Optional)

4.5.1 COBOL Programs

4.5.1.1 PPRCNET (Optional)

PPRCNET calculates deductions for online Rush Check processing. It will be modified to call new subprogram PPBUTSEPD with the employee’s EBEUC and EBERC. If the returned value is ‘Y’ (new rule) and the employee’s separation date is in the current month, deductions for Employee-Paid Life, Dependent Life and AD&D are not taken; otherwise, they are. The deductions are always taken if the separation date is in the current month and the separation reason is retirement or death.

WORKING-STORAGE

- Add copylib CPLNKBSP which contains the linkage to program PPBUTSEPD.
- Add a field to hold the “working” coverage end date for Employee-Paid Life, Dependent Life and AD&D.
- New error message (see Control Table Updates below).
- Add CPWSW88S (COLUMN-88-WORK)

PROCEDURE DIVISION
In section 0500-INITIALIZATION, if the employee has a separation date in the current or future month, derive a working coverage end date for Employee-Paid Life, Dependent Life and AD&D as follows:

- Call PPBUTSEP with the employee’s BEN-ELIG-UNIT-CD (EBEUC) and BEN-ELIG-REP-CD (EBERC).
- If the return status is not “0”, it’s an error. Set a message number and perform CALL-MESSAGE-HANDLER-6080.
- If the return value is “Y”, then derive the working coverage end date as the last day of the separation month, otherwise it is the last day of the month after the separation month.
- Move SEPARATE-REASON to W88-SEPARATE-REASON
- If SEPARATE-REASON-RETIRE-REG or SEPARATE-REASON-RETIRE-VOL or SEPARATE-REASON-RETIRE-EARLY or SEPARATE-REASON-RETIRE-COMPULS or SEPARATE-REASON = 'RF' or SEPARATE-REASON-DEATH, set the working coverage end date to the last day of the month after the separation month.

Note that this has the following effect, as per the requirements:
- If the separation date is in a future month the deductions should be taken.
- If the separation date is in the current month, the employee’s EBERC is ‘C’ and the Ben Sep Month for the employee’s EBEUC is ‘N’ or blank, take the deductions.
- If the separation date is in the current month and the separation reason is retirement or death, take the deductions.
- For any other case, (i.e., BEN SEP MONTH is not “N” or blank or separation date is in a prior month or the reason is not retirement or death) do not take the deductions.

In section 3800-CREATE-DEDUCTION,
- The following code snippet looks at deductions which are Insurance premiums (XGTN-GROUP = ‘I’), and determines whether the coverage end date has been reached for each XGTN-BENEFIT-TYPE. If it has, it sets EFDT-DONT-TAKE-DEDUCTION to TRUE.

```
IF XGTN-GROUP = 'I'
AND RCPW-TRANS-ID = 'RC10'
EVALUATE TRUE
WHEN (XGTN-BENEFIT-TYPE = 'H')
IF HEALTH-COVEND-DATE OF BEN-ROW-DATA NOT = XDC3-LOW-ISO-DATE
AND HEALTH-COVEND-DATE OF BEN-ROW-DATA < ISO-PREM-COV-MONTH-DATE
SET EFDT-DONT-TAKE-DEDUCTION TO TRUE
END-IF
```

Currently, the EVALUATE only looks for types H, D, V, J. Similar code will be added for types L (Employee-Paid Life), Y (Dependent Life) and A (AD & D) as follows:

- If the working coverage end date is prior to ISO-PREM-COV-MONTH-DATE, set EFDT-DONT-TAKE-DEDUCTION to TRUE.

4.5.1.2 PPRCGRSS (Optional)

PPRCGRSS calculates gross pay for online Rush Check processing. It will be modified to move the separation reason from CPWSRCTS to CPWSRCPW.

PROCEDURE DIVISION
In section 6000-LOAD-WRITE-RCPW, move RCTS-SEPARATE-REASON to RCPW-SEPARATE-REASON.
4.5.1.3 PPWRC10 (Optional)

PPWRC10 is the screen processor for the RC10 function (Rush Check Opt1 Employee Data). It will be modified to move the separation reason into CPWSRCTS.

PROCEDURE DIVISION

In para 8100-GET-PER-DATA., include SEPARATE_REASON and :SEPARATE-REASON in the SELECT, and move it to RCTS-SEPARATE-REASON.

4.5.1.4 PPWRC11 (Optional)

PPWRC11 is the screen processor for the RC11 function (Rush Check Opt1 Earns & Deds). It will be modified to move the separation reason into CPWSRCTS.

PROCEDURE DIVISION

In para 8100-GET-PER-DATA., include SEPARATE_REASON and :SEPARATE-REASON in the SELECT, and move it to RCTS-SEPARATE-REASON.

4.5.1.5 PPWRC20 (Optional)

PPWRC20 is the screen processor for the RC20 function (Rush Check Opt2 Employee Data). It will be modified to move the separation reason into CPWSRCTS.

PROCEDURE DIVISION

In para 8100-GET-PER-DATA., include SEPARATE_REASON and :SEPARATE-REASON in the SELECT, and move it to RCTS-SEPARATE-REASON.

4.5.1.6 PPWRC21 (Optional)

PPWRC21 is the screen processor for the RC21 function (Rush Check Opt2 Earns & Deds). It will be modified to move the separation reason into CPWSRCTS.

PROCEDURE DIVISION

In para 8100-GET-PER-DATA., include SEPARATE_REASON and :SEPARATE-REASON in the SELECT, and move it to RCTS-SEPARATE-REASON.
4.5.1.7 PPAPORFH (recompile only: CPWSRCTS) (Optional)

4.5.1.8 PPAPORKS (recompile only: CPWSRCPW, CPWSRCTS) (Optional)

4.5.1.9 PPAPORSR (recompile only: CPWSRCPW, CPWSRCTS) (Optional)

4.5.1.10 PPAPORUP (recompile only: CPWSRCPW, CPWSRCTS) (Optional)

4.5.1.11 PPRCADVC (recompile only: CPWSRCPW) (Optional)

4.5.1.12 PPRCAUDT (recompile only: CPWSRCPW) (Optional)

4.5.1.13 PPRCMDL2 (recompile only: CPWSRCPW) (Optional)

4.5.1.14 PPRCOPT1 (recompile only: CPWSRCPW, CPWSRCTS) (Optional)

4.5.1.15 PPRCOPT2 (recompile only: CPWSRCPW, CPWSRCTS) (Optional)

4.5.1.16 PPRCOPT3 (recompile only: CPWSRCPW, CPWSRCTS) (Optional)

4.5.1.17 PPRCVEND (recompile only: CPWSRCPW) (Optional)

4.5.1.18 PPWRCAD (recompile only: CPWSRCTS) (Optional)

4.5.1.19 PPWRCAI (recompile only: CPWSRCTS) (Optional)

4.5.1.20 PPWRCOV (recompile only: CPWSRCTS) (Optional)

4.5.1.21 PPWRCO2 (recompile only: CPWSRCPW, CPWSRCTS) (Optional)

4.5.1.22 PPWRC12 (recompile only: CPWSRCTS) (Optional)

4.5.1.23 PPWRC30 (recompile only: CPWSRCTS) (Optional)

4.5.2 Copy Members

4.5.2.1 CPWSRCPW (Optional)
CPWSRCPW defines an EXTERNAL area used by Rush Check processing. It will be modified to include RCPW-SEPARATE-REASON X(2).

4.5.2.2 CPWSRCTS (Optional)
CPWSRCTS defines an EXTERNAL area used by Rush Check processing. It will be modified to include RCTS-SEPARATE-REASON X(2).
4.5.3 Control Table Updates

4.5.3.1 System Messages Table (MSG) (Optional)
New messages will be added to the System Messages table to handle invalid return from PPBUTSEP. These messages should all have severity = 5.

5 Unit Testing Requirements

5.1.1 CTL Maintenance
- Modify one of the BUT transactions so that it has a value of something other than “Y” or “N” for the new flag. Run PPP004 and confirm that the transaction is rejected.
- Run PPP004 with the BUT update transactions and confirm the changes on the BUT report.

5.1.2 One-Time Process
- Determine a separation date cut-off.
- Develop test cases in the EDB as follows:
  - Separation dates before, equal to and after the cut-off date.
  - Some EBERCs of “C” and some not “C”.
  - EBEUCs with Ben Sep Month of “Y”, “N” and blank
  - Some coverage end dates already set to the separation month
- Run PPOT2033 with the spec card set to the cut-off date.
- Run PPP180 with the ECF. Verify the following:
  - Only employees with the following conditions should have had their coverage dates (medical, dental, vision and legal) re-derived to the last day of the separation month:
    - The employee’s separation date is after the spec card date
    - The employee’s EBERC is not = “C”
    - The Ben Sep Month for the employee’s EBEUC is “Y”
    - The coverage end dates are not already equal to the new derived coverage end date
  - Confirm that the PPP180 report matches the information on the output tab-delimited file as well as the totals report.
- Run the history update process with the ECF and validate results.

5.1.3 EDB Maintenance Process
- Derive a variety of test cases in the EDB as follows:
  - Some EBERCs of “C” and some not “C”.
  - EBEUCs with Ben Sep Month of “Y”, “N” and blank
- Using the ESEP screen and/or batch transactions, enter separation dates and reasons and update.
- Using the EINS screen, verify that the coverage end dates (medical, dental, vision and legal) were derived as follows:
  - To the last day of the month following the separation month if any of these are true:
    - the Ben Sep Month for the employee’s EBEUC is ‘N’ or blank and the employee’s EBERC is ‘C’
    - the employee’s separation reason is retirement
the employee’s separation reason is death and the employee has dependents who are covered in the month following the separation month
  o to the last day of the separation month for all other conditions

5.1.4 Compute

• Set up test cases as follows:
  o Employees with separation date in the current month, prior month and future month, as well as employees without a separation date.
  o Employees with separation reasons of retirement, death, and other
  o Some EBERCs of “C” and some not “C”.
  o EBEUCs with Ben Sep Month of “Y”, “N” and blank
  o Employees with Employee-Paid Life, Dependent Life and AD&D insurance premiums.

• Run the compute and verify the results as follows:
  o The Employee-Paid Life, Dependent Life and AD&D insurance premiums should be taken only if all of the following conditions are met:
    ▪ The employee’s separation date is in the current month
    ▪ The employee’s EBERC is “C”
    ▪ The Ben Sep Month for the employee’s EBEUC is “N” or blank.

5.1.5 Rush Checks (Optional)

• Create rush checks for the test employees and use the same criteria as for the compute for verification.