Service Request SR83505
AFSCME SX Lump Sum

Technical Specification

January 20, 2012
Prepared by Peter Parker

Information Technology Services
Office of the President
University of California
### 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>11/10/2011</td>
<td>Peter Parker</td>
<td>1st draft</td>
</tr>
<tr>
<td>1.2</td>
<td>01/20/2012</td>
<td>Peter Parker</td>
<td>Revise – BRD revised on 01/20/2012</td>
</tr>
</tbody>
</table>
Table of Contents
Version History ........................................................................................................................................... 1
1 Introduction .............................................................................................................................................. 3
  1.1.1 Service Request SR83505 ............................................................................................................. 3
2 Background ............................................................................................................................................ 3
3 Overview of System Modifications ..................................................................................................... 3
  3.1.1 Compute ........................................................................................................................................ 3
4 Design Considerations ....................................................................................................................... 3
  4.1 Assumptions and Dependencies ..................................................................................................... 3
5 Mainframe Design .............................................................................................................................. 4
  5.1 AFSCME SX Lump Sum .................................................................................................................. 8
    5.1.1 Cobol Programs ......................................................................................................................... 8
    5.1.2 Bind Members PPOTRRRR ...................................................................................................... 11
6 Unit Testing Requirements ................................................................................................................ 11
1 Introduction

SR83505 - Issue a lump sum payment in lieu of retro to AFSCME SX members starting October 2011.

The October 10, 2010 agreement includes the following paragraph:
The University shall pay employees a one-time non base-building lump sum based on actual earnings for the pay period commencing October 1 and concluding on the last day of the pay period prior to the implementation of the salary range increase. This lump sum shall be retirement eligible. All appropriate taxes and UCRP contributions shall be deducted from the lump sum payment. For employees at the minimum of the salary range, this lump sum payment shall also include the difference between their September rate of pay and the new $13.70 salary range minimum, calculated from October 1 through the last day of the pay period prior to the implementation of the new minimum rate.

1.1.1 Service Request SR83505

A onetime program should calculate a lump sum payment in lieu of a retro based on 3% of eligible earnings for October 2011. The payment is retirement eligible, non-base building.

2 Background

This onetime process is very similar to the process developed for SR83209, release 1969.

3 Overview of System Modifications

Using PAY.ONET.COBOL(PPOT1969) as a starting place, write a onetime program with the same file outputs.

3.1.1 Compute

Generated an FT transaction will be processed by the Compute to create LUMP SUM payments.

4 Design Considerations

4.1 Assumptions and Dependencies

The existing code in PPOT1969 will be reused as much as possible. It is an assumption that most of the code required to produce the output files required can be reused. It is expected that the code required to select the eligible populace as well as the code that calculates the LUMP sum amount must be rewritten.
### Mainframe Design

The requirements matrix from the Business Requirements is reproduced here:

<table>
<thead>
<tr>
<th>Req. ID</th>
<th>Requirement Type/Category</th>
<th>Requirement Description</th>
</tr>
</thead>
</table>
| R0001B  | One-time Process          | **Select Eligible Employees for processing based on PAR data**  
Have earnings during the pay periods listed below with PAR title unit ‘SX’ and PAR representation code ‘C’  
- a. MO or MA beginning January 1 and ending January 31 2012  
- b. BW beginning January 8 and ending January 21 2012  
- c. SM beginning January 16 and ending January 31 2012 |
| R0002   | One-time Process          | **Select Eligible Earnings** of Eligible Employees for processing based on PAR data.  
- Earnings transaction end date  
  - BW October 15, October 29 or November 12, 2011  
  - MO, MA October 31, 2011  
  - SM October 15 or October 31  
- All PAR data available beginning October to the day the one-time program is run should be processed so as to include transactions processed after October for October pay (i.e., Late Pay).  
- Skip expense transfer transactions.  
- PAR Title Unit Code “SX” and PAR Appointment Representation Code “C”.  
- DOS Code has RNG ADJ indicator “Y” and either  
  - Pay Category “N” with Type of Hours Code “R”, “O”, or “P” or  
  - Pay Category “A” with Type of Hours Code “O” or “P” |
| R0003   | One-time Process          | **Create 7 Output Files**  
- Lump Sum Payment Transactions (one for each pay cycle MO, MA, SM, BW)  
- Warnings and Controls Report |
<table>
<thead>
<tr>
<th>Req. ID</th>
<th>Requirement Type/Category</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>R0004</td>
<td>One-time Process</td>
<td>Calculate a Lump Sum amount for each earnings record</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Determine the underlying straight time <em>hourly</em> rate (DIST_PAYRATE)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Monthly rates are per 174 hours</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Semi-Monthly rates are per 87 hours</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Biweekly rates are per 80 hours</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- If the underlying straight time hourly rate is $13.31/hour or greater, multiply the earnings by 3%.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- If the underlying straight time hourly rate is less than $13.31, multiply the earnings by 13.70 divided by the underlying straight time hourly rate minus 1 (or minus 100%).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- For example, if the straight time hourly rate is 13.00 multiply by (13.70/13.00) – 1 which equals .0538</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- For ERIT distributions (DOS code ERT) the lump sum amount will be negative</td>
</tr>
<tr>
<td>R0005</td>
<td>One-time Process</td>
<td>Calculate the total lump sum payment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sum the amounts calculated from each earnings record</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Multiply the sum by 1.0058</strong></td>
</tr>
<tr>
<td>R0006</td>
<td>One-time Process</td>
<td>Prorate the payment across FAU based on EDB data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- For each distinct FAU associated with a current regular distribution in appointments having Appointment Title Unit Code “SX” and Appointment Representation Code “C”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Current distribution has Pay End Date greater or equal to program run date</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Regular distribution has DOS Type Hours Code equal to “R”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sum the distribution percents by FAU</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sum the distribution percents overall</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- For example, FAU 1 45%, FAU 2 45%, FAU 3 35% Overall 125%</td>
</tr>
<tr>
<td>Req. ID</td>
<td>Requirement Type/Category</td>
<td>Requirement Description</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o The proration percent is the percent associated with each FAU divided by the overall percent.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o For example, FAU 1 45/125 is 36%, FAU 2 36%, FAU 3 35/125 is 28%.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o The lump sum payment for this example is allocated 36% to FAU 1, 36% to FAU 2 and the remainder to FAU 3. Note that FAU 3 is not calculated as a percent in order to avoid round off errors.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• If there are no current regular distributions satisfying the above, the program should attempt to prorate across expired regular distributions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For employees having no current or expired in-unit distributions issue message “*&lt;ERROR&gt; *ELIG APPT; NO ELIG DIST; PAYMENT BYPASSED “ to the warnings and control report and enter the calculated lump sum in the tab delimited file.</td>
</tr>
<tr>
<td>R0007</td>
<td>One-time Program</td>
<td><strong>Lump Sum Transactions</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Generate an FT transaction for each FAU having DOS code LSP and pay period end date coinciding with the next regular pay cycle for the employee.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o MA 02/29/12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o MO 02/29/12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o SM 2/29/12 (SM2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>o BW 03/03/12 (B1)</td>
</tr>
<tr>
<td>R0008</td>
<td>One-time Program</td>
<td><strong>Output</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For employees with Employment Status Code “A” or “P” output to transactions file, tab delimited file, costing file.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Transactions are placed in one of four files depending on the employees primary pay cycle.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• For employees with Employment Status Code “N” output to tab delimited file only. Issue message “&lt;ERROR&gt; *UNPAID LEAVE OF ABSENCE; PAYMENT BYPASSED “ to warnings and control report.</td>
</tr>
<tr>
<td>R0009</td>
<td>One-time Program</td>
<td><strong>Tab Delimited File</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Columns of the tab delimited file should consist of:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Employee Name (EDB0105)</td>
</tr>
<tr>
<td>Req. ID</td>
<td>Requirement Type/Category</td>
<td>Requirement Description</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Employee ID</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Employee Home Department (EDB0114)</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Home Department Code Description from the Home Department Table</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Employment Status Code (EDB0144)</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Deduction Pay Schedule Code (EDB0152)</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>For each FT transaction:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Full Accounting Unit (FAU)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Transaction Amount</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The column headers should be</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>EMPLOYEE NAME</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>EMP ID</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>DEPT</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>DEPT DESCRIPTION</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>STATUS</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>SCHED</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>FAU</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>AMOUNT</strong></td>
</tr>
<tr>
<td>R0010</td>
<td>One-time Program</td>
<td><strong>Warnings and Controls Report</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>● The report should show totals for all payments processed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● Totals should be grouped by primary pay schedule</td>
</tr>
<tr>
<td></td>
<td></td>
<td>● For each pay schedule, the report should display the number of transactions and the total amount of the lump sum payments.</td>
</tr>
</tbody>
</table>
5.1 **AFSCME SX Lump Sum**

5.1.1 Cobol Programs

5.1.1.1 **PPOTRRRR** (RRRR = release number, determined at release time)

PPOTRRRR is a one-time program that will generate FT transaction.

**Inputs and Outputs**

Seven output files, the same as PPOT1969:

*---> WARNING AND CONTROL REPORT
    SELECT PRINT-FILE    COPY CPSLXPRRT.

*---> FT TRANS FILES FOR MO / BW / SM / MA PAY SCHEDULES
    SELECT FT-TRANS-MO   ASSIGN TO UT-S-FTPAYMO.
    SELECT FT-TRANS-BW   ASSIGN TO UT-S-FTPAYBW.
    SELECT FT-TRANS-SM   ASSIGN TO UT-S-FTPAYSM.
    SELECT FT-TRANS-MA   ASSIGN TO UT-S-FTPAYMA.

*---> FT COSTING FILE FOR PROCESSING IN PPP960
    SELECT FT-COST-FILE  ASSIGN TO UT-S-FTCOSTFL.

*---> ELECTRONIC FILE (TAB DELIMITED)
    SELECT ELEC-FILE     ASSIGN TO UT-S-ELECFILE.

**WORKING-STORAGE**

Start with working storage the same as PPOT1969. Customize as needed.

Add this cursor (see R0002). Open and fetch in initialization. Populate a working storage table with INDEX, use COBOL SEARCH verb to lookup DOS codes retrieved from the PAR.

*---> DOS_ROW CURSOR
    EXEC SQL
    DECLARE DOS_ROW CURSOR FOR
    SELECT
        DOS_EARNINGS_TYPE, DOS_DESCRIPTION, DOS_PAY_CATEGORY, DOS_HOURS_CODE
    FROM PPPVZDOS_DOS WHERE ((DOS_RANGE_ADJ_IND = 'Y') AND
        (DOS_PAY_CATEGORY = 'A' AND DOS_HOURS_CODE IN ('O','P'))) OR
        ((DOS_RANGE_ADJ_IND = 'Y') AND
        (DOS_PAY_CATEGORY = 'N' AND DOS_HOURS_CODE IN ('R','O','P')))
    ORDER BY DOS_EARNINGS_TYPE
END-EXEC

PROCEDURE DIVISION

R0001B, R0002 – Process each record read from the sequential merged PAR

- The sequential PAR file input should include all PAR data created by payroll Computes with earnings transaction end dates listed in R0002 and ending with the most recent PAR data created up to the day this one-time is run.

- Process all PAR earnings segments in XPAR-EARNING-DISTRIBUTION OCCURS 0 TO 500 TIMES DEPENDING ON XPAR-NO-ACCTS
  - Reject earnings if XPAR-DOS() is not found in SEARCH of working storage table loaded in initialization from DOS_ROW cursor
Reject earnings unless one of these conditions is true.

- XPAR-PAYCY-CODE() = ‘B’ AND (‘111002’ <= XPAR-PERIOD-END-DATE() <= ‘111112’)
- XPAR-PAYCY-CODE() = ‘M’ AND (‘111001’ <= XPAR-PERIOD-END-DATE() <= ‘111031’)
- XPAR-PAYCY-CODE() = ‘S’ AND (‘111001’ <= XPAR-PERIOD-END-DATE() <= ‘111031’)

Reject earnings if XPAR-EXPENSE-TRANSFER ( XPAR-TRANS-SEQ-CODE = ‘65’ )

Reject earnings unless

- XPAR-TITLE-UNIT-CODE() = ‘SX’ AND XPAR-EARN-COVERAGE-IND() = ‘C’

PERFORM 9000-SEL-PER-AND-PCM-EMP-DTLS for each employee id selected from the par

After all PAR data for each employee has been processed, if any earnings are found that meet these criteria then the employee is eligible for a lump sum payment, otherwise the employee is not eligible.

- As PAR data is processed check for earnings during the pay periods listed below with PAR title unit ‘SX’ and PAR representation code ‘C’.
  - XPAR-PAYCY-CODE() = ‘B’ AND (‘120108’ <= XPAR-PERIOD-END-DATE() <= ‘120121’)
  - XPAR-PAYCY-CODE() = ‘M’ AND (‘120101’ <= XPAR-PERIOD-END-DATE() <= ‘120131’)
  - XPAR-PAYCY-CODE() = ‘S’ AND (‘120116’ <= XPAR-PERIOD-END-DATE() <= ‘120131’)

R0003 – Code in PPOT1969 should provide the basic framework for creation the 7 output files.

R0004, R0005 - Calculate a Lump Sum amount for each earnings record; add calculated amount to accumulated LUMP SUM amount for each eligible employee.

- MOVE ZERO TO LUMP-SUM-WRK

- IF XPAR-RATE-TYP-SAL()
  IF XPAR-SEMI-MNTHLY-PAYCY() COMPUTE WRK-MO-AMT = XPAR-PAYRATE-SAL() /87 END-IF
  IF XPAR-BI-WKLY-PAYCY() COMPUTE WRK-MO-AMT = XPAR-PAYRATE-SAL() /80 END-IF
  IF XPAR-MONTHLY-PAYCY() COMPUTE WRK-MO-AMT = XPAR-PAYRATE-SAL() /174 END-IF

- IF XPAR-RATE-HRS()
  MOVE XPAR-PAYRATE-HR() TO WRK-MO-AMT

- IF WRK-MO-AMT NOT < 13.31
  MOVE .03 TO WRK-PERCENT
ELSE
  COMPUTE WRK-PERCENT = (13.70 / WRK-MO-AMT) – 1
END-IF

- COMPUTE LUMP-SUM-WRK = WRK-PERCENT * XPAR-EARN-AMT()
R0006, R0007 Prorate the payment across FAU based on EDB data, Create Lump Sum Transactions; if EDB data is not available, prorate using PAR data

Program PPOT1969 contains the required code.

R0008 Output

**EVALUATE PRI-PAY-SCHED**

```
WHEN 'BW'
  WRITE FT-TRN-REC-BW FROM FT-TRANS-RECORD
  ADD 1 TO EOJ-FT-REC-BW
  ADD FT-AMOUNT TO EOJ-FT-AMT-BW

WHEN 'SM'
  WRITE FT-TRN-REC-SM FROM FT-TRANS-RECORD
  ADD 1 TO EOJ-FT-REC-SM
  ADD FT-AMOUNT TO EOJ-FT-AMT-SM

WHEN 'MA'
  WRITE FT-TRN-REC-MA FROM FT-TRANS-RECORD
  ADD 1 TO EOJ-FT-REC-MA
  ADD FT-AMOUNT TO EOJ-FT-AMT-MA

WHEN OTHER
  WRITE FT-TRN-REC-MO FROM FT-TRANS-RECORD
  ADD 1 TO EOJ-FT-REC-MO
  ADD FT-AMOUNT TO EOJ-FT-AMT-MO
```

END-EVALUATE.

R0009 Tab Delimited File

Program PPOT1969 contains this record layout for the Tab Delimited file:

```
01 WS-ELEC-HDR-REC.
  05 FILLER PIC X(26) VALUE 'EMPLOYEE NAME'.
  05 FILLER PIC X(01) VALUE X'05'.
  05 FILLER PIC X(09) VALUE 'EMP ID'.
  05 FILLER PIC X(01) VALUE X'05'.
  05 FILLER PIC X(06) VALUE 'DEPT'.
  05 FILLER PIC X(01) VALUE X'05'.
  05 FILLER PIC X(30) VALUE 'DEPT DESCRIPTION'.
  05 FILLER PIC X(01) VALUE X'05'.
  05 FILLER PIC X(06) VALUE 'STATUS'.
  05 FILLER PIC X(01) VALUE X'05'.
  05 FILLER PIC X(10) VALUE 'SEP DATE'.
  05 FILLER PIC X(01) VALUE X'05'.
  05 FILLER PIC X(05) VALUE 'SCHED'.
  05 FILLER PIC X(01) VALUE X'05'.
  05 FILLER PIC X(30) VALUE 'FAU'.
  05 FILLER PIC X(01) VALUE X'05'.
  05 FILLER PIC X(08) VALUE 'AMOUNT'.
  05 FILLER PIC X(01) VALUE X'05'.
```

This contains one field that is not requested for SR83505: 'SEP DATE'.
It should be removed.

R0010 Warnings and Controls Report

Report produced by PPOT1969 fulfills this requirement.
5.1.2  Bind Members PPOTRRRR

********************************************************************
*  BIND MEMBER: PPOTRRRR                                           *
*  RELEASE: ____RRR______ SERVICE REQUEST(S): ____8SSSS____        *
*  NAME:_________________ CREATION DATE:      ___??/??/??__        *
*  DESCRIPTION:                                                   *
*                                                              *
*  - BIND FOR A NEW ONE-TIME PROGRAM.                           *
*  -                                                 *
********************************************************************

BIND
PLAN(PPOTRRRR) -
MEMBER(PPOTRRRR) -
OWNER(PAYADM) -
ACTION(REPLACE) -
RETAIN -
VALIDATE(RUN) -
ISOLATION(CS) -
FLAG(I) -
ACQUIRE(USE) -
RELEASE(COMMIT) -
EXPLAIN(NO) -

6  Unit Testing Requirements

Unit testing will require either finding or entering employees on the test EDB that will have eligible earnings who are paid on BW, MO, and SM pay schedules. Once these employees are identified test Computes must be run to created PAR data. Program PPP460 can be run to create a sequential PAR file. Both salaried and hourly employees should be included, as well as employees whose underlying straight time hourly rate is less than $13.31 and greater than 13.31.