RECEIVED
DEC 17 4 45 PM '01
POSTAL RATE COMMISSION
OFFICE OF THE SECRETARY

Docket No. R2001-1

USPS-LR-J-198

Selected TY Costs by Shape & Density Level, Provided in Response to ADVO/USP-T43-1

Category 4 Library Reference

Includes 1 Compact Disk

Table of Contents

Introduction

This library reference provides the supporting analyses used to estimate test year volume-variable costs by weight increment for Standard Enhanced Carrier Route (ECR) mail by shape, density level, and weight increment for mail processing (cost segment 3.1), window service (3.2), and city carrier in-office (6.1). This is a Category 4 library reference provided in response to interrogatory ADVO/USPS-T43-1.

The methodology used to develop the costs by weight increment distributions for Standard ECR mail presented here is very similar to that used in USPS-LR-J-58. The supporting program documentation and program lists can be found in USPS-LR-J-58.

Other testimony and library references used in this analysis include:

- USPS-T-11 for BY00 CRA costs
- USPS-LR-J-10 for the IOCS data set

Organization

The final cost by weight increment estimates by shape and density level for Standard ECR mail are provided in the Excel workbook ADVO-T43-1.xls. The volume-variable costs for mail processing are presented in worksheet 'Mail Processing'. The worksheet 'Window Service' presents the test year volume-variable window service costs, and the worksheet 'City Carriers' contains the test year city carrier in-office volume-variable costs.

The underlying methodology used to develop these costs is discussed in USPS-LR-J-58, page 5. The FORTRAN programs used to develop the underlying base year costs for each cost segment are almost identical to those described in USPS-LR-J-58. Those FORTRAN programs that are different are presented in Appendix A of this document.

Appendix A: Program Documentation

I. Computer Hardware and Software

The IOCS data processing is performed on a Data General AViiON minicomputer with four Pentium Pro microprocessors and one gigabyte of RAM, running the DGUX version of UNIX operating system. Source programs ending with an ".f" file extension are FORTRAN programs and programs ending in an ".sm" file extension are SORT/MERGE programs. The remaining data processing is performed in Excel workbooks (.xls file extension) on PCs running the Microsoft Windows NT 4 and Windows 2000 operating systems and Microsoft Office.

II. Test Year Standard ECR Mail Volume-Variable Cost Estimations

The volume-variable cost distribution FORTRAN programs used to estimate volume-variable costs for mail processing (cost segment 3.1), window service (3.2), and city carriers in-office (6.1) are the same as those described in USPS-LR-J-58. The programs described in this appendix summarize the cost estimates generated in USPS-LR-J-58 for each cost segment by weight increment, shape, and density level for Standard Enhanced Carrier Route (ECR) mail.

A. Mail Processing (Cost Segment 3.1)

Program: **sumclass_mod_ecr_wgt.f** - Rolls up the output from the FORTRAN program modsproc00_wgt.f (USPS-LR-J-58) from activity code to subclass by cost pool, shape, density level, and weight increment

Input: **mods002by.data** – Estimated mail processing volume-variable costs by cost pool, activity code, and weight increment for MODS 1&2 offices (USPS-LR-J-58)
costpools.00.619 - List of cost pools for MODS 1&2 offices (USPS-LR-J-55)
activity00.ecr.cra – List of the direct and class specific mixed activity codes
classes_ecr.old - List of new CRA subclasses
classmap_ecr.old - Maps IOCS activity codes to the appropriate CRA subclass

Output: **mod00cra_ecr_wgt.csv** – Estimated mail processing volume-variable costs for selected Standard ECR mail by cost pool, shape, density level, and weight category for MODS 1&2 offices

Program: **sumclass_bmc_ecr_wgt.f** - Rolls up the output from the FORTRAN program bmcproc00_wgt.f (USPS-LR-J-58) from activity code to subclass by cost pool, shape, density level, and weight increment

Input: **bmc002by_wgt.data** – Estimated mail processing volume-variable costs by cost pool, activity code, and weight increment for BMCs (USPS-LR-J-58)
costpools.00.bmc.619 – List of cost pools for BMCs (USPS-LR-J-55)
activity00.ecr.cra – List of the direct and class specific mixed activity codes
classes_ecr.old - List of new CRA subclasses
classmap_ecr.old - Maps IOCS activity codes to the appropriate CRA subclass

Output: **bmc00cra_ecr_wgt.csv** – Estimated mail processing volume-variable costs for Standard ECR mail by cost pool, shape, density level, and weight category for BMCs

Program: **sumclass_nmod_ecr_wgt.f** - Rolls up the output from the FORTRAN program nmodproc00_wgt.f (USPS-LR-J-58) from activity code to subclass by cost pool, shape, density level, and weight increment

Input: **nmod00by_wgt.data** – Estimated mail processing volume-variable costs by cost pool, activity code, and weight increment (USPS-LR-J-58)
costpools.00.nmod.619 – List of cost pools for Non-MODS offices (USPS-LR-J-55)
activity00.ecr.cra – List of the direct and class specific mixed activity codes
classes_ecr.old - List of new CRA subclasses
classmap_ecr.old - Maps IOCS activity codes to the appropriate CRA subclasses

Output: **nmod00cra_ecr_wgt.csv** – Estimated mail processing volume-variable costs for Standard ECR mail by cost pool, shape, density level, and weight category for Non-MODs offices

Workbook: **Mail Proc ECR Wgt.xls** – Summarizes the BY00 Standard ECR mail processing volume-variable cost estimates for all offices by cost pool, shape, density level, and weight increment. Applies test year piggyback factors, cost ratios, and reconciliation factors to convert base year mail processing costs to test year mail processing cost to be used in the weight increment analysis

Input: **mod00cra_ecr_wgt.csv** – Estimated mail processing volume-variable costs for selected Standard ECR mail by cost pool, shape, density level, and weight category for MODS 1&2 offices
bmc00cra_ecr_wgt.csv – Estimated mail processing volume-variable costs for Standard ECR mail by cost pool, shape, density level, and weight category for BMCs
nmod00cra_ecr_wgt.csv – Estimated mail processing volume-variable costs for Standard ECR mail by cost pool, shape, density level, and weight category for Non-MODs offices
Test Year 03 Piggyback Factors, Cost Ratios, and Reconciliation Factors –USPS-LR-J-53, workbook 'SHP03U~1.xls'

B. Window Service (Cost Segment 3.2)

Workbook: **Window ECR Wgt.xls** – Summarizes the direct labor and stamp sales/meter setting window service volume-variable costs into total window service costs by shape, density level, and weight increment for Standard ECR mail

Input: **win cost by oz 00 new.xls** – Window service direct labor volume-variable costs by subclass, shape, and weight increment (USPS-LR-J-58)
sm cost by oz 00 new.xls – Window service stamp sales/meter setting volume-variable costs by subclass, shape, and weight increment (USPS-LR-J-58)
LR58AECR_revised.xls – worksheet 'TY Window', estimated test year window service costs for Standard ECR mail (USPS-LR-J-58)

C. City Carrier In-Office (Cost Segment 6.1)

Program: **rpt_city_ecr_wgt.f** – Summarized LIOCATT cost distribution results by shape, density level, and weight increment for city carrier in-office Standard ECR mail costs

Input: **activity00.ecr.all** - List of activity codes and corresponding subclass codes
classes_ecr.old – List of old CRA subclasses
level1b – Level 1 distributed direct costs (USPS-LR-J-58)
level2b - Level 2 distributed direct costs (USPS-LR-J-58)
level3a – Level 3 direct costs (USPS-LR-J-58)
level3b – Level 3 distributed direct costs (USPS-LR-J-58)

Output: **car_ecr_wgt00.csv** – Estimated City Carrier In-Office costs by shape, density level, and weight increment for Standard ECR mail

Workbook: **City Carrier ECR Wgt.xls** – Reports City Carrier In-Office costs for Standard ECR mail by shape, density level, and weight increment. Adjusts the FORTRAN replication of LIOCATT to match the BY00 CRA Cost Segment 6.1 costs

Input: **car_ecr_wgt00.csv** – Estimated City Carrier In-Office costs by shape, density level, and weight increment for Standard ECR mail
BY00 CRA Cost Segment 6.1 Costs – CRA City Carrier In-Office costs from CS06&7.xls, worksheet 'Outputs to CRA' (USPS-LR-J-57)
LR58AECR_revised.xls – worksheet 'TY City', estimated test year city carrier in-office costs for Standard ECR mail (USPS-LR-J-58)

D. Summary

Workbook: **ADVO-T43-1.xls** – Summary of TY Standard ECR mail volume-variable cost estimates by shape, density level, and weight increment for mail processing, window service, and city carrier in-office

Input: **Mail Proc ECR Wgt.xls**
Window ECR Wgt.xls
City Carrier ECR Wgt.xls

Appendix B: Program Lists

```
      program sumclass_mod_ecr_wgt

c     Purpose:  Sum distributed volume-variable mail processing costs for MODS 1&2 offices to subclass
c               Costs are calculated in the Fortran program modsproc00_wgt.f (USPS-LR-J-58)
                For Standard ECR Mail by shape and density (ADVO/USPS-T43-1)

      implicit none

      integer*4 nact, ncl, nmod, nshp, nmat, nshp2, nw

      parameter (nmod = 42)      !  Number of cost pools
      parameter (nact = 255)     !  Number of activity codes
      parameter (ncl = 80)       !  Number of subclasses
      parameter (nshp = 3)       !  Number of shapes
      parameter (nmat = 1)       !  Number of cost categories
      parameter (nshp2 = 5)      !  Number of shapes (class map)
      parameter (nw = 22)        !  Number of weight increments

      real*8    dollars(nmat,nw,nmod,nact)
      real*8   cdols(nmat,nmod,ncl,nshp,nw)

      integer*4 imod, iact, icl, i, j, k, shape, is
      integer*4 ier, shp(nact), iw, imod2
      integer*4 clmap(nact), mod(nmod), ldcl(nmod)

      character*16 grp(nmod)
      character*9  class(ncl), clcode
      character*9  class2(ncl)
      character*4  acodes(nact), temp, acin(nshp2)
      character*5  shapetype(nshp)/'1Ltr ','2Flt ','3Pcl '/

      ier = 0

c     Map of cost pools
      open(30,file='costpools.00.619')
32    format(i2,a16,i2)

      do i = 1, nmod
         read(30,32) mod(i), grp(i), ldcl(i)
      end do
      print *, 'Mod groups read'

c     Map of activity codes
      open(20,file='activity00.ecr.cra')
21    format(a4)

      do i = 1, nact
         read (20,21) acodes(i)
         is = shape(acodes(i))
         shp(i) = is
      end do
      print*, 'Read in activity codes '

c     Map of subclasses
      open(33,file='classes_ecr.old')
34    format(a9)
      do i = 1, ncl
         read(33,34) class(i)
         class2(i) = class(i)
      end do
      print*, 'Read in classes '

c     Maps activity codes to subclass
      open(35,file='classmap_ecr.old')
36    format(a9,3x,a4,4(4x,a4))
      do i = 1, nact
         clmap(i) =  0
      end do
      do while (ier.eq.0)
         read(35,36,iostat=ier,end=101) clcode, acin
         do i = 1, nshp2
            j = 0
            if (acin(i).ne.'    ') then
               do iact = 1,nact
                  if (acodes(iact).eq.acin(i)) then
                     j = iact
                  end if
               end do
               if (j.gt.0) then
                  temp = acin(i)
```

```
                if ((temp(2:2).eq.'6').or.(temp(2:2).eq.'7').or.
     &             (temp(2:2).eq.'8').or.(temp(1:2).eq.'54')) then
                   clmap(j) = 37
                else
                   k = 0
                   do icl = 1,ncl
                      if (class2(icl).eq.clcode) then
                         k=icl
                      end if
                   end do
                   if (k.gt.0) then
                      clmap(j) = k
                   else
                      print *,' bad class code = ',clcode,' ',clcode
                   end if
                end if
             else
                print *,' activity code not found ',acin(i)
             end if
          end if
       end do
    end do
101 print *,' read exit of classmap = ',ier
    ier = 0

C  Initialize matrices

    do imod = 1, nmod
       do icl = 1, ncl
          do j = 1, nmat
             do is = 1, nshp
                do iw = 1, nw
                   cdols(j,imod,icl,is,iw) = 0.
                end do
             end do
          end do
       end do
    end do

c     Read in distributed cost data
    open(40,file='mods002by.data')
41  format(10x,f18.9)

    do imod = 1, nmod
       do iact = 1, nact
          do iw = 1, nw
             read (40,41) (dollars(j,iw,imod,iact),j=1,nmat)
          end do
       end do
    end do

C  Sum data to classes

    do j = 1, nmat
       do imod = 1, nmod
          do iact = 1, nact
             do iw = 1, nw
                icl = clmap(iact) !  Subclass for corresponding activity code
                is = shp(iact) !  Assign shape
                if (icl.eq.24) icl = 18 !  Combine ECR LOT (Regular and Nonprofit)
                if (icl.eq.25) icl = 19 !  Combine ECR WSS  (Regular and Nonprofit)
                if (icl.eq.26) icl = 20 !  Combine ECR WSH  (Regular and Nonprofit)
                if (icl.eq.27) icl = 21 !  Combine ECR Auto  (Regular and Nonprofit)
                imod2 = imod
                if (icl.gt.0) then
                   cdols(j,imod2,icl,is,iw) = cdols(j,imod2,icl,is,iw)
     &                + dollars(j,iw,imod2,iact)
                else
                   print *,' activity ',acodes(iact),' not in class map ', iact
                end if
             end do
          end do
       end do
    end do

c Combine pools to create Fcn 1 & Fcn 4 Support
    do j = 1, nmat
       do imod = 1, nmod
          do icl = 1, ncl
             do is = 1, nshp
```

```
                  do iw = 1, nw
                     if (imod.eq.31) then !  1Support
                        cdols(j,30,icl,is,iw) = cdols(j,30,icl,is,iw) + cdols(j,imod,icl,is,iw) !  1Misc
                     else if (imod.eq.39) then !  LD48 Adm
                        cdols(j,38,icl,is,iw) = cdols(j,38,icl,is,iw) + cdols(j,imod,icl,is,iw) !  LD48 Oth
                     end if
                  end do
               end do
            end do
         end do
      end do

C     Write out costs by subclass, cost pool, shape, and weight increment

      open(50,file='mod00cra_ecr_wgt.csv',recl=1000)
51    format(i2,',',i2,',',a16,',',a10,',',i2,',',a5,22(', ',f18.9))

      do imod = 1, nmod
         do icl = 1, ncl
            do is = 1, nshp
               if ((imod.ne.31).and.(imod.ne.39)) then
                  if ((icl.ge.18).and.(icl.le.21)) then
                     write (50,51) imod,ldcl(imod), grp(imod), class(icl), icl, shapetype(is),
     &                  (cdols(1,imod,icl,is,iw), iw = 1, nw)
                  end if
               end if
            end do
         end do
      end do

      end

C-------------------------------------------------------------------------------
C     Assign shape

      function shape(act)

      integer*4    shape
      character*4  act

      if (act(1:1).eq.'1') then
         shape = 1  !  Letters
      else if (act(1:1).eq.'2') then
         shape = 2  !  Flats
      else if ((act(1:1).eq.'3').or.(act(1:1).eq.'4')) then
         shape = 3  !  IPPs/Parcels
      else
         shape = 3  !  Other (Special Service)
         if (act.gt.'1000') then
            print*, 'No shape for actv ', act
         end if
      end if

      return
      end
```

```
      program sumclass_bmc_ecr_wgt

c     Purpose:  Sum distributed volume-variable mail processing costs for BMCs to subclass
c               Costs are calculated in the Fortran program bmcproc00_wgt.f (USPS-LR-J-58)
c               For Standard ECR Mail by shape and density (ADVO/USPS-T43-1)

      implicit none

      integer*4 nact, ncl, nmod, nshp, nmat, nshp2, nw

      parameter (nmod = 6)       !  Number of cost pools
      parameter (nact = 255)     !  Number of activity codes
      parameter (ncl = 80)       !  Number of subclasses
      parameter (nshp = 3)       !  Number of shapes
      parameter (nmat = 8)       !  Number of cost categories
      parameter (nshp2 = 5)      !  Number of shapes (class map)
      parameter (nw = 22)        !  Number of weight increments

      real*8    dollars(nmat,nw,nmod,nact)
      real*8   cdols(nmat,nmod,ncl,nshp,nw)

      integer*4 imod, iact, icl, i, j, k, shape, is
      integer*4 ier, shp(nact), iw
      integer*4 clmap(nact), mod(nmod), ldcl(nmod)

      character*14  grp(nmod)
      character*9 class(ncl), clcode, class2(ncl)
      character*4 temp
      character*4 acodes(nact), acin(nshp2)
      character*5 shapetype(nshp)/'1Ltr ','2Flt ','3Pcl '/


      ier = 0

c     Map of cost pools
      open(30,file='costpools.00.bmc.619')
32    format(i4,a14,i5)

      do i = 1, nmod
         read(30,32) mod(i), grp(i), ldcl(i)
      end do
      print *, 'BMC groups read'

c     Map of activity codes
      open(20,file='activity00.ecr.cra')
21    format(a4)

      do i = 1, nact
         read (20,21) acodes(i)
         is = shape(acodes(i))
         shp(i) = is
      end do

c     Map of subclasses
      open(33,file='classes_ecr.old')
34    format(a9)
      do i = 1, ncl
         read(33,34) class(i)
         class2(i) = class(i)
      end do

c     Maps activity codes to subclass
      open(35,file='classmap_ecr.old')
36    format(a9,3x,a4,4(4x,a4))
      do i = 1, nact
         clmap(i) =  0
      end do
      do while (ier.eq.0)
         read(35,36,iostat=ier,end=101) clcode, acin
         do i = 1, nshp2
            j = 0
            if (acin(i).ne.'    ') then
               do iact = 1,nact
                  if (acodes(iact).eq.acin(i)) then
                     j = iact
                  end if
               end do
               if (j.gt.0) then
                  temp = acin(i)
                  if (((temp(2:2).eq.'6').or.(temp(2:2).eq.'7').or.
```

```
     &               (temp(2:2).eq.'8').or.(temp(1:2).eq.'54'))) then
                     clmap(j) = 37
                  else
                     k = 0
                     do icl = 1,ncl
                        if (class2(icl).eq.clcode) then
                           k=icl
                        end if
                     end do
                     if (k.gt.0) then
                        clmap(j) = k
                     else
                        print *,' bad class code = ',clcode,' ',clcode
                     end if
                  end if
               else
                  print *,' activity code not found ',acin(i)
               end if
            end if
         end do
      end do
101 print *,' read exit of classmap = ',ier

      ier = 0

C  Initialize matrices

      do imod = 1, nmod
         do icl = 1, ncl
            do j = 1, nmat
               do is = 1, nshp
                  do iw = 1, nw
                     cdols(j,imod,icl,is,iw) = 0.
                  end do
               end do
            end do
         end do
      end do

c     Read in distributed cost data
      open(40,file='bmc002by_wgt.data')
41    format(10x,8f18.9)

      do imod = 1, nmod
         do iact = 1, nact
            do iw = 1, nw
               read (40,41) (dollars(j,iw,imod,iact),j=1,nmat)
            end do
         end do
      end do

C  Sum data to classes

      do j = 1, nmat
         do imod = 1, nmod
            do iact = 1, nact
               do iw = 1, nw
                  icl = clmap(iact) !  Subclass for corresponding activity code
                  is = shp(iact) !  Assign shape
                  if (icl.eq.24) icl = 18 !  Combine ECR LOT  (Regular and Nonprofit)
                  if (icl.eq.25) icl = 19 !  Combine ECR WSS  (Regular and Nonprofit)
                  if (icl.eq.26) icl = 20 !  Combine ECR WSH  (Regular and Nonprofit)
                  if (icl.eq.27) icl = 21 !  Combine ECR Auto  (Regular and Nonprofit)
                  if (icl.gt.0) then
                     cdols(j,imod,icl,is,iw) = cdols(j,imod,icl,is,iw)
     &                  + dollars(j,iw,imod,iact)
                  else
                     print *,' activity ',acodes(iact),' not in class map '
                  end if
               end do
            end do
         end do
      end do

C     Write out costs by subclass, cost pool, shape, and weight increment

      open(50,file='bmc00cra_ecr_wgt.csv')
51    format(i2,',',i2,',',a14,',',a10,',',i2,',',a5,22(',',f18.9))

      do imod = 1, nmod
```

```
         do icl = 1, ncl
            do is = 1, nshp
               if ((icl.ge.18).and.(icl.le.21)) then
                  write (50,51) imod, ldc1(imod), grp(imod), class(icl), icl, shapetype(is),
     &               (cdols(1,imod,icl,is,iw), iw = 1, nw)
               end if
            end do
         end do
      end do

      end

c-------------------------------------------------------------------------------
      function shape(act)

      integer*4    shape
      character*4  act

      if (act(1:1).eq.'1') then
         shape = 1  !  Letters
      else if (act(1:1).eq.'2') then
         shape = 2  !  Flats
      else if ((act(1:1).eq.'3').or.(act(1:1).eq.'4')) then
         shape = 3  !  IPPs/Parcels
      else
         shape = 3  !  Other (Special Service)
         if (act.gt.'1000') then
            print*, 'No shape for actv ', act
         end if
      end if

      return
      end
```

```
      program sumclass_nmod_ecr_wgt

c     Purpose:  Sum distributed volume-variable mail processing costs for MODS 1&2 offices to subclass
c               Costs are calculated in the Fortran program nmodproc00_wgt.f (USPS-LR-J-58)
                For Standard ECR Mail by shape and density (ADVO/USPS-T43-1)

      implicit none

      integer*4 nact, ncl, nmod, nshp, nmat, nshp2, nw

      parameter (nmod = 8)       !  Number of cost pools
      parameter (nact = 255)     !  Number of activity codes
      parameter (ncl = 80)       !  Number of subclasses
      parameter (nshp = 3)       !  Number of shapes
      parameter (nmat = 8)       !  Number of cost categories
      parameter (nshp2 = 5)      !  Number of shapes (class map)
      parameter (nw = 22)        !  Number of weight increments

      real*8    dollars(nmat,nw,nmod,nact)
      real*8   cdols(nmat,nmod,ncl,nshp,nw)

      integer*4 imod, iact, icl, i, j, k, shape, is
      integer*4 ier, shp(nact), iw
      integer*4 clmap(nact), mod(nmod), ldc1(nmod)

      character*14  grp(nmod)
      character*9 class(ncl), clcode
      character*9 class2(ncl)
      character*4 acodes(nact), temp,acin(nshp2)
      character*5 shapetype(nshp)/'1Ltr ','2Flt ','3Pcl '/


      ier = 0

c     Map of cost pools
      open(30,file='costpools.00.nmod.619')
32    format(i4,a14,i5)

      do i = 1, nmod
         read(30,32) mod(i), grp(i), ldc1(i)
      end do

      print *, 'Mod groups read'

c     Map of activity codes
      open(20,file='activity00.ecr.cra')
21    format(a4)

      do i = 1, nact
         read (20,21) acodes(i)
         is = shape(acodes(i))
         shp(i) = is
      end do
      print*, 'Read in activity codes '

c     Map of subclasses
      open(33,file='classes_ecr.old')
34    format(a9)
      do i = 1, ncl
         read(33,34) class(i)
         class2(i) = class(i)
      end do
      print*, 'Read in classes '

c     Maps activity codes to subclass
      open(35,file='classmap_ecr.old')
36    format(a9,3x,a4,4(4x,a4))
      do i = 1, nact
         clmap(i) =  0
      end do
      do while (ier.eq.0)
         read(35,36,iostat=ier,end=101) clcode, acin
         do i = 1, nshp2
            j = 0
            if (acin(i).ne.'    ') then
               do iact = 1,nact
                  if (acodes(iact).eq.acin(i)) then
                     j = iact
                  end if
               end do
```

```
            if (j.gt.0) then
               temp = acin(i)
               if (((temp(2:2).eq.'6').or.(temp(2:2).eq.'7').or.
     &             (temp(2:2).eq.'8').or.(temp(1:2).eq.'54'))) then
                  clmap(j) = 37
               else
                  k = 0
                  do icl = 1,ncl
                     if (class2(icl).eq.clcode) then
                        k=icl
                     end if
                  end do
                  if (k.gt.0) then
                     clmap(j) = k
                  else
                     print *,' bad class code = ',clcode,' ',clcode
                  end if
               end if
            else
               print *,' activity code not found ',acin(i)
            end if
         end if
      end do
   end do
101 print *,' read exit of classmap = ',ier

   ier = 0

C  Initialize matrices

   do imod = 1, nmod
      do icl = 1, ncl
         do j = 1, nmat
            do is = 1, nshp
               do iw = 1, nw
                  cdols(j,imod,icl,is,iw) = 0.
               end do
            end do
         end do
      end do
   end do

c     Read in distributed cost data
   open(40,file='nmod00by_wgt.data')
41 format(10x,8f18.9)

   do imod = 1, nmod
      do iact = 1, nact
         do iw = 1, nw
            read (40,41) (dollars(j,iw,imod,iact),j=1,nmat)
         end do
      end do
   end do

C  Sum data to classes

   do j = 1, nmat
      do imod = 1, nmod
         do iact = 1, nact
            do iw = 1, nw
               icl = clmap(iact) !  Subclass for corresponding activity code
               is = shp(iact) !  Assign shape
               if (icl.eq.24) icl = 18 !  Combine ECR LOT  (Regular and Nonprofit)
               if (icl.eq.25) icl = 19 !  Combine ECR WSS  (Regular and Nonprofit)
               if (icl.eq.26) icl = 20 !  Combine ECR WSH  (Regular and Nonprofit)
               if (icl.eq.27) icl = 21 !  Combine ECR Auto  (Regular and Nonprofit)
               if (icl.gt.0) then
                  cdols(j,imod,icl,is,iw) = cdols(j,imod,icl,is,iw)
     &               + dollars(j,iw,imod,iact)
               else
                  print *,' activity ',acodes(iact),' not in class map ', iact
               end if
            end do
         end do
      end do
   end do

C     Write out costs by subclass, cost pool, shape, and weight increment
   open(50,file='nmod00cra_ecr_wgt.csv')
51 format(i2,',',i2,',',a14,',',a10,',',i2,',',a5,22(',',f18.9)) !
```

```
      do imod = 1, nmod
         do icl = 1, ncl
            do is = 1, nshp
               if ((icl.ge.18).and.(icl.le.21)) then
                  write (50,51) imod, ldcl(imod), grp(imod), class(icl), icl, shapetype(is),
     &               (cdols(1,imod,icl,is,iw), iw = 1, nw)
               end if
            end do
         end do
      end do

      end


c-----------------------------------------------------------------------------
      function shape(act)

      integer*4    shape
      character*4  act

      if (act(1:1).eq.'1') then
         shape = 1                !  Letters
      else if (act(1:1).eq.'2') then
         shape = 2                !  Flats
      else if ((act(1:1).eq.'3').or.(act(1:1).eq.'4')) then
         shape = 3                !  IPPs/Parcels
      else
         shape = 3                !  Other (Special Service)
         if (act.gt.'1000') then
            print*, 'No shape for actv ', act
         end if
      end if

      return
      end
```

```
      PROGRAM rpt_city_ecr_wgt

c     Purpose:  To summarize city carrier in-office costs by weight increment, shape,
c               and CRA subclass for Standard ECR Mail by density category (ADVO/USPS-T43-1)

      IMPLICIT NONE

      integer*4   numfun, numact, w, nopr, nshp, numfun2, ncl

      parameter   (numfun = 22) ! number of weight increments (including no weights)
      parameter   (numfun2 = 21) ! number of weight increments (without no weights)
      parameter   (numact = 501) ! number of activity codes
      parameter   (nopr = 12)    !  number of operations
      parameter   (ncl = 243)    !  number of subclasses
      parameter   (nshp = 4)     !  number of shapes

      integer*4   is, shape, ishp, shp(numact), icl

      real*8      carrier(ncl,nshp,numfun)
      real*8      indata, cardist(ncl,nshp,numfun), sum
      real*8      actwgt(numfun)

      integer*4   pay, opr, bf, act
      integer*4   unit, i, ier, iw
      integer*4   class(numact)

      character*4 acodes(numact)
      character*9 classes(ncl)
      character*5 shapetype(nshp)

      ier = 0

c     Map of activity codes and codes to corresponding subclass
      open(16,file='activity00.ecr.all')
17    format(a4,i6)

      do i = 1,numact
         read(16,17) acodes(i), class(i)
         is = shape(acodes(i))
         shp(i) = is
      end do
      close(16)

c     Map of subclasses
      open(16,file='classes_ecr.old')
18    format(a9)
      do i = 1, ncl
         read(16,18) classes(i)
      end do
      close(16)

c     Initialize matrices
      do icl = 1, ncl
         do ishp = 1, nshp
            do iw = 1, numfun
               carrier(icl,ishp,iw) = 0.0
               cardist(icl,ishp,iw) = 0.0
            end do
         end do
      end do

c     Open files of LIOCATT results
      open(20,file='level1b')
      open(21,file='level2b')
25    format(2x,i1,i2.2,i1,2i3.3,2x,f13.1)
      open(30,file='level3a')
      open(31,file='level3b')
35    format(i1,i2.2,i1,2i3.3,2x,f13.1)

C       Assemble data for report


      do unit = 20,21
         do while (ier.eq.0)
            read (unit,25,iostat=ier,end=100) pay,opr,bf,act,w,indata
            icl = class(act)     !  Assign subclass
            ishp = shp(act)      !  Assign shape
            if (icl.eq.24) icl = 18 !  Combine ECR LOT  (Regular and Nonprofit)
            if (icl.eq.25) icl = 19 !  Combine ECR WSS  (Regular and Nonprofit)
            if (icl.eq.26) icl = 20 !  Combine ECR WSH  (Regular and Nonprofit)
```

```
            if (icl.eq.27) icl = 21 !  Combine ECR Auto  (Regular and Nonprofit)
            if (pay.eq.3) then  !  city carriers
               if (icl.gt.0) then
                  if (ishp.gt.0) then
                     carrier(icl,ishp,w) = carrier(icl,ishp,w) + indata/1000.
                  else
                     print*, 'Invalid shape ', acodes(act), shp(act)
                  end if
               else
                  print*, 'Invalid class assignment ', acodes(act), ' ', class(act)
               end if
            end if
         end do
100      print *,' Read exit of unit ',unit,' = ',ier
         ier = 0
      end do

      do unit = 30,31
         do while (ier.eq.0)
            read (unit,35,iostat=ier,end=101) pay,opr,bf,act,w,indata
            icl = class(act)     !  Assigns subclass
            ishp = shp(act)      !  Assigns shape
            if (icl.eq.24) icl = 18 !  Combine ECR LOT  (Regular and Nonprofit)
            if (icl.eq.25) icl = 19 !  Combine ECR WSS  (Regular and Nonprofit)
            if (icl.eq.26) icl = 20 !  Combine ECR WSH  (Regular and Nonprofit)
            if (icl.eq.27) icl = 21 !  Combine ECR Auto  (Regular and Nonprofit)
            if (pay.eq.3) then  !  city carriers
               if (icl.gt.0) then
                  if (ishp.gt.0) then
                     carrier(icl,ishp,w) = carrier(icl,ishp,w) + indata/1000.
                  else
                     print*, 'Invalid shape ', acodes(act), shp(act)
                  end if
               else
                  print*, 'Invalid class assignment ', acodes(act), ' ', class(act)
               end if
            end if
         end do
101      print *,' Read exit of unit ',unit,' = ',ier
         ier = 0
      end do

c     Redistribute no weight city carrier costs
      do icl = 1, 37            !  Skip SSV Subclasses
         do ishp = 1, nshp
            sum = 0.0
            if (carrier(icl,ishp,numfun).gt.0.0) then
               do i = 1, numfun2 !  Distribute over all weight increments
                  sum = sum + carrier(icl,ishp,i)
               end do
               if (sum.gt.0.0) then
                  do i = 1, numfun2
                     cardist(icl,ishp,i) = cardist(icl,ishp,i) +
     &                  (carrier(icl,ishp,numfun)*(carrier(icl,ishp,i)/sum))
                  end do
                  carrier(icl,ishp,numfun) = 0.0
               else
                  sum = 0.0
                  do i = 1, numfun2
                     actwgt(i) = 0.0
                  end do
                  if (carrier(icl,ishp,numfun).gt.0.0) then
                     do i = 1, numfun2 !  Redistribute over all weight increments
                        actwgt(i) = actwgt(i) + carrier(icl,ishp,i)
                        sum = sum + carrier(icl,ishp,i)
                     end do
                     if (sum.gt.0.0) then
                        do i = 1, numfun2
                           cardist(icl,ishp,i) = cardist(icl,ishp,i) +
     &                        (carrier(icl,ishp,numfun)*(actwgt(i)/sum))
                        end do
                        carrier(icl,ishp,numfun) = 0.0
                     else
                        print*, 'Unable to distribute no weight costs for class ', classes(icl)
                     end if
                  end if
               end if
            end if
         end do
      end do
```

```
      print*, 'Costs redistributed '

      Add in redistributed no weight costs
      do icl = 1, ncl
         do ishp = 1, nshp
            do i = 1, numfun
               carrier(icl,ishp,i) = carrier(icl,ishp,i) + cardist(icl,ishp,i)
            end do
         end do
      end do

      print*, 'Redistributed costs added in '

      shapetype(1) = '1Ltr '
      shapetype(2) = '2Flt '
      shapetype(3) = '3Pcl '
      shapetype(4) = '4None'

c     Write out results
      open(45,file='car_ecr_wgt00.csv')
41    format(a9,',',i3,',',a5,22(',',f15.5))
      do icl = 1, ncl
         do ishp = 1, 3
            if ((icl.ge.18).and.(icl.le.21)) then
               write (45,41) classes(icl), icl, shapetype(ishp), (carrier(icl,ishp,iw), iw = 1, numfun) !
            end if
         end do
      end do

      end

c-------------------------------------------------------------------------------
      function shape(act)

      integer*4    shape
      character*4  act

      if (act(1:1).eq.'1') then
         shape = 1  !  Letters
      else if (act(1:1).eq.'2') then
         shape = 2  !  Flats
      else if ((act(1:1).ge.'3').and.(act(1:1).le.'4')) then
         shape = 3  !  IPPs/Parcels
      else
         shape = 4  !  Other (special services)
      end if

      return
      end
```