APPENDIX

2 Example Data

Introduction **255**
Access Descriptors **256**
ACCTDBD Database Access Descriptor **256**
EMPLINF2 Database Access Descriptor **258**
WIRETRAN Database Access Descriptor **259**
View Descriptors **260**
ACCTDBD Database View Descriptors **260**
EMPLINF2 Database View Descriptors **262**
WIRETRAN Database View Descriptor **262**
Creating SAS Data Sets **262**
MYDATA.BIRTHDAY **263**
MYDATA.CHECKS **263**
MYDATA.CHGDATA **264**
MYDATA.CHKCRED **264**
MYDATA.CHKDEBD **266**
MYDATA.EMPLDATA **270**
MYDATA.INITSEG **273**
MYDATA.PHONENUM **274**
MYDATA.SAVCRED **274**
MYDATA.SAVDEBD **276**
MYDATA.CUSTOMER **278**
MYDATA.NEWADDR **279**
VER6.SSNUMS **279**
SAS Statements for Loading DB2 Table BANKCHRG **280**
Creating SAS Data Set MYDATA.BANK **280**
Loading DB2 Table BANKCHRG from MYDATA.BANK **280**
DB2 View Descriptor for BANKCHRG **281**

Introduction

This appendix lists the data in the sample IMS-DL/I databases ACCTDBD and EMPLINF2, and the DB2 table BANKCHRG used in the examples in this book. It also includes the data in the descriptor files and SAS data files used in the examples in Chapter 4, "Using IMS-DL/I Data in SAS Programs," on page 47 and Chapter 5, "Browsing and Updating IMS-DL/I Data," on page 69. See Chapter 3, "Defining SAS/ACCESS Descriptor Files," on page 39 for complete information on the WIRETRN database.

Sample JCL for allocating the IMS databases, creating DBDs, creating PSBs, and creating needed flat files is provided in the SAS Sample Library files. If you want to run these examples, see "About the Example Data in This Book" on page 8 or contact

your SAS Software Representative for information about how to access the files in the SAS Sample Library provided with this release.

Access Descriptors

ACCTDBD Database Access Descriptor

This section describes the MYLIB.ACCOUNT access descriptor for the ACCTDBD database that is used in the examples. This section provides the statements used to create the ACCOUNT access descriptor in batch, interactive line, or noninteractive mode. The ACCTDBD database is described in detail in Chapter 2, "Understanding IMS-DL/I Essentials," on page 11.

```
JCL statements;
proc access dbms=ims;
  create mylib.account.access;
    dbd=acctdbd dbtype=hdam;
    record='customer_record' sg=customer sl=225;
      item=soc_sec_number lv=2  dbf=$11.
                          key=u
                          se=ssnumber;
      item=customer_name  lv=2  dbf=$40.
                          se=custname;
      item='address info' lv=2;
      item=addr_line_1    lv=3  dbf=$30.
                          se=custadd1;
      item=addr_line_2    lv=3  dbf=$30.
                          se=custadd2;
      item=city           lv=3  dbf=$28.
                          se=custcity;
      item=state          lv=3  dbf=$2.
                          se=custstat;
      item=country        lv=3  dbf=$20.
                          se=custland;
      item=zip_code       lv=3  dbf=$10.
                          se=custzip;
      item=home_phone     lv=2  dbf=$12.
                          se=custhphn;
      item=office_phone   lv=2  dbf=$12.
                          se=custophn;

    record='checking_account_record' sg=chckacct
         sl=40;
      item=check_account_number lv=2  dbf=12.
                                key=u
                                se=acnumber;
      item=check_amount         lv=2  dbf=pd5.2
                                se=stmtamt
                                dbc=l;
      item=check_date           lv=2  dbf=6.0
                                fmt=date7.
```

```
                              se=stmtdate
                              dbc=mmddyy6.;
  item=filler1                lv=2  dbf=$2.;
  item=check_balance          lv=2  dbf=pd5.2
                              se=stmtbal
                              dbc=l;

record='checking_debit_record' sg=chckdebt sl=80;
  item=check_debit_amount     lv=2  dbf=pd5.2
                              key=y
                              se=debtamt
                              dbc=l;
  item=check_debit_date       lv=2  dbf=6.0
                              fmt=date7.
                              se=debtdatd
                              dbc=mmddyy6.;
  item=filler2                lv=2  dbf=$2.;
  item=check_debit_time       lv=2  dbf=$8.
                              se=debttime;
  item=check_debit_desc       lv=2  dbf=$59.
                              se=debtdesc;

record='checking_credit_record' sg=chckcrdt sl=80;
  item=check_credit_amount    lv=2  dbf=pd5.2
                              key=y
                              se=crdtamt
                              dbc=l;
  item=check_credit_date      lv=2  dbf=6.0
                              fmt=date7.
                              se=crdtdate
                              dbc=mmddyy6.;
  item=filler3                lv=2  dbf=$2.;
  item=check_credit_time      lv=2  dbf=$8.
                              se=crdttime;
  item=check_credit_desc      lv=2  dbf=$59.
                              se=crdtdesc;

record='savings_account_record' sg=saveacct sl=40;
  item=savings_account_number lv=2 dbf=12.
                              key=y
                              se=acnumber;
  item=savings_amount         lv=2  dbf=pd5.2
                              se=stmtamt
                              dbc=l;
  item=savings_date           lv=2  dbf=6.0
                              fmt=date7.
                              se=stmtdate
                              dbc=mmddyy6.;
  item=filler4                lv=2  dbf=$2.;
  item=savings_balance        lv=2  dbf=pd5.2
                              se=stmtbal
                              dbc=l;

record='savings_debit_record' sg=savedebt sl=80;
```

```
      item=savings_debit_amount lv=2  dbf=pd5.2
                                key=y
                                se=debtamt
                                dbc=1;
      item=savings_debit_date   lv=2  dbf=6.0
                                fmt=date7.
                                se=debtdate
                                dbc=mmddyy6.;
      item=filler5              lv=2  dbf=$2.;
      item=savings_debit_time   lv=2  dbf=$8.
                                se=debttime;
      item=savings_debit_desc   lv=2  dbf=$59.
                                se=debtdesc;

    record='savings_credit_record' sg=savecrdt sl=80;
      item=savings_credit_amount lv=2 dbf=pd5.2
                                 key=y
                                 se=crdtamt
                                 dbc=1;
      item=savings_credit_date   lv=2  dbf=6.0
                                 fmt=date7.
                                 se=crdtdate
                                 dbc=mmddyy6.;
      item=filler6               lv=2  dbf=$2.;
      item=savings_credit_time   lv=2  dbf=$8.
                                 se=crdttime;
      item=savings_credit_desc   lv=2  dbf=$59.
                                 se=crdtdesc;
  list all;
run;
```

EMPLINF2 Database Access Descriptor

This section describes the MYLIB.EMPLOYEE access descriptor for the EMPLINF2 database used in the examples and provides the statements that are used to create the EMPLOYEE access descriptor in batch, interactive line, or noninteractive mode.

```
proc access dbms=ims;
  create mylib.employee.access;
    database=emplinf2 dbtype=hidam;
    record='employee record' segment=employee
                             seglng=150;
      item=employee_id      lv=2  dbf=pd3.0
                            key=u
                            se=empid;
      item=last_name        lv=2  dbf=$10.
                            se=lastname;
      item=first_name       lv=2  dbf=$20.
                            se=frstname;
      item=hire_date        lv=2  dbf=6.0
                            fmt=date7.
                            se=hiredate
                            dbc=mmddyy6.;
```

```
      item=birthday          lv=2  dbf=7.0
                             fmt=date7.
                             se=birthday
                             dbc=mmddyy6.;
      item=ssn               lv=2  dbf=$11.
                             se=ssn;
      item=gender            lv=2  dbf=$6.
                             se=gender;
      item=status            lv=2  dbf=$9.
                             se=status;
      item=phone_extension   lv=2  dbf=$9.
                             se=phone;
      item=vacation          lv=2  dbf=ib4.
                             se=vacation
                             dbc=1;
      item=department        lv=2  dbf=zd6.0
                             se=deptment;
      item=zip_code          lv=2  dbf=$5.
                             se=zipcode;
      item=city_and_state    lv=2  dbf=$15.
                             se=citystat;
      item=street            lv=2  dbf=$20.
                             se=street;
      item=security          lv=2  dbf=rb4.
                             fmt=10.0
                             se=security
                             dbc=1;
      item=sick_leave        lv=2  dbf=6.2
                             se=sicklv
                             dbc=1;
  list all;
```

WIRETRAN Database Access Descriptor

This section describes the MYLIB.WIRETRAN access descriptor for the WIRETRAN database used in examples and provides the statements used to create the WIRETRAN access descriptor in batch, interactive line, or noninteractive mode.

```
proc access dbms=ims;
  create mylib.wiretrn.access;
    database=wiretrn dbtype=hdam;
    record='wire transaction' segment=wiretran
                              seglng=100;
      item='ssn - account'    lv=2  dbf=$23.
                              se=ssnacc
                              key=y;
      item='account type'     lv=2  dbf=$1.
                              se=accttype;
      item='wire date'        lv=2  dbf=$8.
                              se=wiredate;
      item='wire time'        lv=2  dbf=$8.
                              se=wiretime;
      item='wire amount'      lv=2  dbf=pd5.2
                              se=wireammt
```

```
                                dbc=1;
      item='wire description' lv=2  dbf=$40.
                                se=wiredesc;
   an=y;
  list all;

run;
```

View Descriptors

ACCTDBD Database View Descriptors

This section shows SAS statements used to create the view descriptors for the ACCTDBD database used in the examples in this book. The ACCTDBD database is described in Chapter 2, "Understanding IMS-DL/I Essentials," on page 11. The view descriptors are presented here in alphabetical order for easy reference.

You can create all the view descriptors used in the book by using PROC ACCESS statements. The view descriptors are based on the MYLIB.ACCOUNT access descriptor shown earlier in this appendix.

```
proc access dbms=ims ad=mylib.account;
  create vlib.account.view psb=accupsb;
    select  soc_sec_number
            customer_name
            city
            state
            zip_code;
  list view;

  create vlib.cdbtdate.view psb=accupsb;
    select  check_account_number
            check_date;
  list view;

  create vlib.chckacct.view psb=accupsb;
    select  soc_sec_number
            customer_name
            check_account_number
            check_date
            check_balance;
  list view;

  create vlib.chkcrd.view psb=accupsb pcb=2;
    select  customer_record
            checking_account_record
            checking_credit_record;
    reset 17 28;
  list view;

  create vlib.chkdeb.view psb=accupsb pcb=3;
    select  customer_record
```

```
          checking_account_record
          checking_debit_record;
  reset 17 22;
list view;

create vlib.chktrans.view psb=accupsb;
  select  customer_name
          check_account_number
          check_date
          check_balance;
list view;

create vlib.credits.view psb=accupsb;
  select  soc_sec_number
          check_account_number
          check_credit_amount
          check_credit_date
          check_credit_time
          check_credit_desc;
list view;

create vlib.custacct.view psb=accupsb;
  select  soc_sec_number
          customer_name
          check_account number;
list view;

create vlib.custinfo.view psb=accupsb;
  select  2 3 5 6 7 8 9 10 11 12;
list view;

create vlib.custphon.view psb=accupsb;
  select  soc_sec_number
          customer_name
          home_phone
          office_phone;
list view;

create vlib.savebal.view psb=accupsb;
  select  soc_sec_number
          customer_name
          city
          32 36;
list view;

create vlib.ssname.view psb=accupsb;
  select  soc_sec_number
          customer_name;
list view;

create vlib.trans.view psb=accupsb;
  select  soc_sec_number
          check_account_number
          check_debit_amount;
```

```
  list view;

run;
```

EMPLINF2 Database View Descriptors

This section shows SAS statements used to create the view descriptors for the EMPLINF2 database used in the examples in this book. The view descriptors are presented here in alphabetical order for easy reference. You can create all the view descriptors used in the book by using PROC ACCESS statements. The view descriptors are based on the MYLIB.EMPLOYEE access descriptor shown earlier in this appendix.

```
proc access dbms=ims accdesc=mylib.employee;
  create vlib.emplload.view psbname=empilpsb;
    select  employee_record;
  list view;

  create vlib.emplview.view psbname=empiupsb;
    select  employee_record;
  list view;

  create vlib.empbday.view psbname=empiupsb;
    select  employee_id
            last_name
            first_name
            birthday
            phone_extension;
  list view;
run;
```

WIRETRAN Database View Descriptor

This section shows SAS statements used to create the VLIB.WIREDATA view descriptor for the WIRETRAN database used in the examples in this book. The view descriptor is based on the MYLIB.WIRETRAN access descriptor shown earlier in this appendix.

```
proc access dbms=ims ad=mylib.wiretran;
  create vlib.wiredata.view psbname=acctsam
     pcbindex=5;
    select 'wire transaction';
  list view;
run;
```

Creating SAS Data Sets

This section shows the SAS statements that create each data file used in the book's examples.

MYDATA.BIRTHDAY

The SAS data set MYDATA.BIRTHDAY is updated with data from the EMPLINF2 database.

```
data mydata.birthday;
  input @01 employee_id   6.
        @08 last_name     $10.
        @19 birthday      date7.;
  format employee_id   6.
         last_name     $10.
         birthday      date7.;
  datalines;
  1247 Garcia     04APR54
  1078 Gibson     23APR36
  1005 Knapp      06OCT38
  1024 Mueller    17JUN53
;

proc print data=mydata.birthday;
  title2 'SAS Data Set MYDATA.BIRTHDAY';
run;
```

MYDATA.CHECKS

The SAS data set MYDATA.CHECKS is used to update the ACCTDBD database.

```
data mydata.checks;
  length customer_name  $40.;
  input  customer_name  & $
         soc_sec_number  $11.
         check_account_number
         check_balance
         check_date  date7.;
  format check_account_number  12.
         check_balance  12.2
         check_date  date7.;
  datalines;
COWPER, KEITH  241-98-4542 183352795865
862.31 25MAR95
OLSZEWSKI, STUART  309-22-4573 382654397566
486.00 02APR95
NAPOLITANO, BARBARA  250-36-8831 284522378774
104.20 10APR95
MCCALL, ROBERT  367-34-1543 644721295973
571.92 05APR95
;

proc print data=mydata.checks;
  title2 'SAS Data Set MYDATA.CHECKS';
run;
```

MYDATA.CHGDATA

The SAS data set MYDATA.CHGDATA is used to update the ACCTDBD database.

```
data mydata.chgdata;
  input account  12.
        charge;
  format account 14.
         charge  dollar7.;
  datalines;
345620135872 10
345620134522 7
345620123456 12
382957492811 3
345620134663 8
345620131455 6
345620104732 9
;
run;

proc print data=mydata.chgdata;
  title2 'SAS Data Set MYDATA.CHGDATA';
run;
```

MYDATA.CHKCRED

The SAS data set MYDATA.CHKCRED is used to add the checking credit path to the ACCTDBD database.

```
data mydata.chkcred;
        /****  CUSTOMER data    ***/
  input @1   soc_sec_number   $11.
        @13  customer_name    $40.  /
        @1   addr_line_1      $30.
        @32  addr_line_2      $30.  /
        @1   city             $28.
        @30  state            $2.
        @33  country          $20.
        @54  zip_code         $10.  /
        @1   home_phone       $12.
        @14  office_phone     $12.
        /****  CHCKACCT data    ***/
        @27  check_account_number 12.0
        @40  check_amount         12.2
        @53  check_date           date7. /
        @1   filler1              $2.
        @4   check_balance        12.2
        /****  CHCKCRDT data    ***/
        @17  check_credit_amount  12.2
        @30  check_credit_date    date7.
        @38  filler3              $2.
        @41  check_credit_time    $8.   /
        @1   check_credit_desc    $59.;
    format check_credit_date    date7.;
```

```
  datalines;
667-73-8275 WALLS, HOOPER J.


                345620145345
         1563.23 31MAR95    15:42:43
MAIN ST BRANCH DEPOSIT
667-73-8275 WALLS, HOOPER J.


              345620154633
       1563.23 31MAR95    15:42:43
BAD ACCT_NUM
434-62-1234 SUMMERS, MARY T.


              345620104732
       400.00 02APR95    10:23:46
ACH DEPOSIT
436-42-6394 BOOKER, APRIL M.


              345620135872
       50.00 02APR95    12:16:34
ACH DEPOSIT
434-62-1224 SMITH, JAMES MARTIN


              345620134564
       1342.42 22MAR95    23:23:52
ACH DEPOSIT
434-62-1224 SMITH, JAMES MARTIN


             345620134663
      120.00 28MAR95    10:26:45
ACH DEPOSIT
178-42-6534 PATTILLO, RODRIGUES


             745920057114
    1300.00 12JUN95    14:34:12
ACH DEPOSIT
156-45-5672 O'CONNOR, JOSEPH


            345620123456
    100.00 01APR95    12:24:34
ATM DEPOSIT
657-34-3245 BARNHARDT, PAMELA S.


           345620131455
    230.00 04APR95    14:24:11
```

```
ACH DEPOSIT
667-82-8275 COHEN, ABRAHAM


           382957492811
   100.00 16APR95    09:21:14
ACH DEPOSIT
456-45-3462 LITTLE, NANCY M.


          345620134522
  50.00 05APR95    12:14:52
ACH DEPOSIT
234-74-4612 WIKOWSKI, JONATHAN S.


          345620113263
         672.32 31MAR95
ATM DEPOSIT
;
```

MYDATA.CHKDEBD

The SAS data set MYDATA.CHKDEBD is used to add the checking debit path to the ACCTDBD database.

```
data mydata.chkdebd;
        /****  CUSTOMER data    ***/
  input @1   soc_sec_number    $11.
        @13  customer_name     $40.  /
        @1   addr_line_1       $30.
        @32  addr_line_2       $30.  /
        @1   city              $28.
        @30  state             $2.
        @33  country           $20.
        @54  zip_code          $10.  /
        @1   home_phone        $12.
        @14  office_phone      $12.
        /****  CHCKACCT data    ***/
        @27  check_account_number  12.0
        @40  check_amount          12.2
        @53  check_date            date7. /
        @1   filler1               $2.
        @4   check_balance         12.2
        /****  CHCKDEBT data    ***/
        @17  check_debit_amount    12.2
        @30  check_debit_date      date7.
        @38  filler2               $2.
        @41  check_debit_time      $8.    /
        @1   check_debit_desc      $59.;
    format check_date    date7.;
    format check_debit_date   date7.;
  datalines;
667-73-8275 WALLS, HOOPER J.
```

```
              345620145345
   1266.34       820.00 23MAR95    23:54:53
CHECK 2958
667-73-8275 WALLS, HOOPER J.


              345620145345
   1266.34        52.00 23MAR95    23:54:53
CHECK 2948
667-73-8275 WALLS, HOOPER J.


              345620145345
   1266.34       193.00 28MAR95    22:51:43
CHECK 2951
667-73-8275 WALLS, HOOPER J.


              345620154633  1303.41 28MAR95
  1298.04          .   .

434-62-1234 SUMMERS, MARY T.
                    4322 LEON ST.
GORDONSVILLE       VA USA         26001-0670
803-657-1687    345620104732  826.05 27MAR95
  825.45           .   .

436-42-6394 BOOKER, APRIL M.
                    9712 WALLINGFORD PL.
GORDONSVILLE       VA USA          26001-0670
803-657-1346    345620135872   220.11 26MAR95
   234.89          .   30MAR94    22:34:45
CHECK 103
434-62-1224 SMITH, JAMES MARTIN
                    133 TOWNSEND ST.
GORDONSVILLE       VA USA          26001-0670
803-657-3437    345620134564   2392.93 16MAR95
   2645.34       432.87 18MAR95    22:13:48
CHECK 1826
434-62-1224 SMITH, JAMES MARTIN

              345620134564
   2645.34      19.23 18MAR95    22:13:48
CHECK 1821
434-62-1224 SMITH, JAMES MARTIN


              345620134564
  2645.34     723.23 22MAR95    21:48:12
CHECK 1828
434-62-1224 SMITH, JAMES MARTIN
```

```
              345620134564
  2645.34      82.32 22MAR95    21:48:12
CHECK 1829
434-62-1224 SMITH, JAMES MARTIN


              345620134564
  2645.34      73.62 26MAR95    21:22:24
CHECK 1830
434-62-1224 SMITH, JAMES MARTIN


              345620134564
  2645.34      31.23 26MAR95    21:22:24
CHECK 1831
434-62-1224 SMITH, JAMES MARTIN


              345620134564
   2645.34     162.87 29MAR94    22:51:12
CHECK 1835
434-62-1224 SMITH, JAMES MARTIN


              345620134564
   2645.34       7.12 29MAR95    22:51:12
CHECK 1836
434-62-1224 SMITH, JAMES MARTIN


              345620134564
   2645.34      62.34 31MAR95    23:02:12
CHECK 1833
434-62-1224 SMITH, JAMES MARTIN


              345620134663      0.00 24MAR95
      143.78      25.00 28MAR95    15:53:29
ATM MAIN ST.
178-42-6534 PATTILLO, RODRIGUES
                           9712 COOK RD.
ORANGE         VA USA           26042-1650
803-657-1346 803-657-1345 745920057114 1404.90
10JUN95     1502.78   25.89 10JUN95   11:45:25
CHECK 412
156-45-5672 O'CONNOR, JOSEPH
                           235 MAIN ST.
ORANGE         VA USA           26042-1650
803-657-5656 803-623-4257 345620123456  353.65
27MAR95    463.23     13.29 28MAR95   22:23:53
CHECK 934
156-45-5672 O'CONNOR, JOESPH
```

```
803-657-5656 803-623-4257 345620123456
         463.23   32.87 31MAR95    23:35:53
CHECK 931
156-45-5672 O'CONNOR, JOSEPH


               345620123456
    463.23      50.00 02APR95    10:23:41
ATM GREEN ST
156-45-5672 O'CONNOR, JOESPH


              345620123456
    463.23      13.42 31MAR95    23:35:53
CHECK 935
657-34-3245 BARNHARDT, PAMELA S.
                            RT 2 BOX 324
CHARLOTTESVILLE     VA USA      25804-0997
803-345-4346 803-355-2543 345620131455
1243.25  29MAR95       1243.25       .   .

667-82-8275 COHEN, ABRAHAM
                            2345 DUKE ST.
CHARLOTTESVILLE     VA USA      25804-0997
803-657-7435 803-645-4234 382957492811 7462.51
03APR95       7302.06       .   .

456-45-3462 LITTLE, NANCY M.
                            4543 ELGIN AVE.
RICHMOND          VA USA       26502-3317
803-657-3566   345620134522     608.24 25MAR95
     831.65      42.73 29MAR95    23:12:34
CHECK 296
456-45-3462 LITTLE, NANCY M.


              345620134522
    831.65     172.45 29MAR95    23:12:34
CHECK 301
456-45-3462 LITTLE, NANCY M.


             345620134522
    831.65      38.23 30MAR95    22:51:34
CHECK 297
456-45-3462 LITTLE, NANCY M.


              345620134522
    831.65      10.00 02APR95    21:51:34
CHECK 298
234-74-4612 WIKOWSKI, JONATHAN S.
                         4356 CAMPUS DRIVE
```

```
RICHMOND          VA USA           26502-5317
803-467-4587 803-654-7238 345620113263 672.32
28MAR95       13.28       .   .

;
```

MYDATA.EMPLDATA

The SAS data set MYDATA.EMPLDATA is used to load the EMPLINF2 database.

```
data mydata.empldata;
  input @01 employee_id     6.
        @08 last_name       $10.
        @19 first_name      $20.
        @40 hire_date       yymmdd6.
        @47 birthday        yymmdd6.
        @54 ssn             $11. /
        @01 gender          $6.
        @08 status          $9.
        @18 phone_extension $9.
        @28 vacation        8.2
        @37 department      8.
        @46 zip_code        $5.
        @52 city_and_state  $15. /
        @01 street          $20.
        @21 security        5.
        @27 sick_leave      8.2;
  format hire_date   yymmdd6.
         birthday   yymmdd6.;
  datalines;
  1001 Waterhouse Clifton P.781231 480101 254-43-6089
Male  Full Time X5109  8.00   200 78752 Austin,TX
505 Cat Mountain Tr.  310     8.00
  1002 Bowman    Hugh E.   801230 310714 329-88-6729
Male  Full Time X5901  40.00 1000 78741 Austin,TX
47 Cypress Point Cir  310    80.00
  1003 Salazar    Yolanda   821230 401212 166-88-7516
Female Full Time X5169  80.00 200 78641 Leander,TX
6811 Picket Fence Dr  310    56.00
  1004 Knight     Althea    841229 500409 942-62-3354
Female Full Time X5218 300 78664 Round Rock,TX
8222 Whitewing Way    110    16.00
  1005 Knapp     Patrice R. 811230 371004 353-43-1272
Female Full Time X5012 8.00   100 78748 Austin,TX
19 Pack Saddle Pass   110    44.00
  1006 Garrett   Olan M.   781231 350123 776-94-3545
Male  Full Time X5208 80.00  300 78731 Austin,TX
67 Running Doe Ln.    110    60.00
  1007 Brown     Virgina P.801230 460524 675-29-9081
Female Full Time X5258 48.00  300 78610 Buda,TX
2713 Nutty Brown Mil  110    32.00
  1008 Hernandez Jesse L.   821230 330326 123-12-0987
Male  Full Time X5448 56.00  500 78664 Round Rock,TX
4319 Red Stone Lane     .     8.00
```

```
  1009 Jones     Michael Y. 850330 310521 543-87-1934
Male  Full Time X5713 80.00  800 78748 Austin,TX
23 Moonlight Bend La    .    80.00
  1010 Smith     Janet F.   811230 470807 105-32-9011
Female Full Time X5621 16.00  700 78737 Austin,TX
523 Rim Rock Road       .     8.00
  1011 Van Hotten Gwendolyn 790201 420913 766-30-9237
Female Full Time X5311  .     400 78641 Leander,TX
623 Fauntleroy Trail    .    32.00
  1012 Quintero   Pedro     810214 480221 339-94-2674
Male  Full Time X5348 32.00  400 78741 Austin,TX
77 Button Quail Cove    .    40.00
  1015 Scholl    Madison A. 830304 450319 765-43-0581
Male  Full Time X5419 40.00  500 78741 Austin,TX
3910 Covered Wagon      .    80.00
  1017 Waggonner Merrilee D.850330 360427 586-54-8967
Female Full Time X5914 56.00 1000 78722 Austin,TX
941 Bridgewater Dr.     .    40.00
  1020 Rudd      Fred       601230      . 145-67-6532
Male  Part Time        .     100
                       .      .
  1024 Mueller    Patsy     790403 520617 857-51-1838
Female Full Time X5822 40.00 900 78620 Dripping Spring
6935 Cherry Creek Rd  110    40.00
  1031 Chan       Tai       810502 460704 843-09-7123
Male  Full Time X5331 40.00  400 78755 Austin,TX
1412 Arapahoe Trail     .    80.00
  1049 Fernandez  Sophia    830516 440911 764-91-0193
Female Full Time X5847 96.00  900 78744 Austin,TX
4700 Old Stage Trail    .    40.00
  1050 Ameer      David     850530 511010 456-34-6543
Male  Full Time X5495 56.00  500 78735 Austin,TX
231 Little Hill Cir.    .      .
  1062 Littlejohn Fannie    850429 540517 978-63-3930
Female Full Time X5653 8.00  700 78660 Pflugerville,TX
813 Lime Rock Dr.     110    48.00
  1067 Cahill     Jacob     790105 401225 102-78-8765
Male  Full Time X5042 60.00  100 78748 Austin,TX
121 Hidden Hollow       .    36.00
  1071 Canady     Frank A.  810331 411119 345-91-4321
Male  Full Time X5406  8.00  500 78756 Austin,TX
741 Canyonwood Lane     .     8.00
  1074 Millsap    Joel B.   830831 360612 675-23-8027
Male  Full Time X5224 24.00  300 78755 Austin,TX
1201 Broken Bow Pass  110    48.00
  1077 Gibson     Teddy B.  850929 460423 567-89-2345
Male             X5703 80.00  800 78753 Austin,TX
4441 Hansford           .    80.00
  1078 Gibson    George J.  820930 460423 567-89-2346
Male  Full Time X5703 80.00  800 78753 Austin,TX
2311 Hansford           .    80.00
  1083 Savage    William D. 791001 530120 211-95-9608
Male  Full Time X5505 80.00  600 78737 Austin,TX
97 Cimarron Circle      .    48.00
```

```
  1086 Schmidt   Penny      811017 270219 901-45-4567
Female Full Time X5822 80.00  900 78735 Austin,TX
6419 Wild Rose Road     .    80.00
  1092 Polanski  Ivan L.    831130 471011 497-36-7845
Male  Full Time X5621 56.00 700 78620 Dripping Spring
2501 Timberline Tr.     .      .
  1101 Nathaniel  Darryl    860101 440321 584-86-6945
Male  Full Time X5544 40.00  600 78735 Austin,TX
1892 Red River Road   210     8.00
  1105 Faulkner  Carrie Ann 830102 510817 987-76-7469
Female Full Time X5417 48.00  500 78756 Austin,TX
5649 Foothill Park    110    16.00
  1112 Jones     Rita M.    790202 481224 890-98-6789
Female Full Time X5271 24.00  300 78735 Austin,TX
907 Hickory Stick       .     8.00
  1119 Goodson   Alan F.    820116 500621 234-67-8901
Male  Full Time X5512 48.00  600 78626 Georgetown,TX
11410 Smokey Hill Rd    .    16.00
  1120 Reid      David G.   830214 450815 442-04-0121
Male  Full Time X5369 80.00  400 78752 Austin,TX
1322 Lazy Lane        224    80.00
  1123 Freeman   Leopold    861030 350209 828-26-7282
Male  Part Time X5604   .    700 78757 Austin,TX
13 Timber Hills Tr.   106      .
  1133 Williamson Janice L. 831103 520519 131-41-9129
Female Full Time X5802 40.00  900 78610 Buda,TX
2706 Frontier Valley    .     8.00
  1139 Seaton     Gary     800403 561003 286-04-6279
Male  Full Time X5545 80.00  600 78757 Austin,TX
2111 Wind Ridge Road    .    80.00
  1145 Juarez    Armando   820501 470528 876-19-0378
Male  Full Time X5987 48.00 1000 78626 Georgetown,TX
1017 Woodstone Sq.      .    16.00
  1156 Reed      Kenneth D. 840830 550105 875-15-1388
Male  Full Time X5307 64.00  400 78641 Leander,TX
1349 Begonia Terrace    .    40.00
  1161 Richardson Travis Z. 860913 371130 654-54-8127
Male  Full Time X5325 88.00  400 78752 Austin,TX
2009 Mountain Lake    110    96.00
  1213 Johnson    Bradford  840131 540415 321-32-9446
Male  Full Time X5446 40.00  500 78724 Austin,TX
678 Buffalo Gap Road    .    40.00
  1217 Rodriguez Romualdo R. 810131 290209 493-77-4863
Male  Full Time X5874 32.00  900 78746 Austin,TX
804 Lazy Brook Lane     .    48.00
  1219 Kaatz      Freddie   830131 570621 181-49-4592
Male  Full Time X5387 80.00  400 78753 Austin,TX
4713 Cedar Tree Lane    .    80.00
  1234 Shropshire Leland G. 850415 490904 555-21-4173
Male  Full Time X5616 32.00  700 78752 Austin,TX
606 Bull Creek Trail    .    40.00
  1238 Throckmort Stewart Q.850516 310804 109-07-5098
Male  Full Time X5391 40.00  400 78756 Austin,TX
479 Roundup Circle      .    40.00
```

```
  1247 Garcia    Francisco  840730 550505 678-23-0123
Male  Full Time X5348 80.00  400 78756 Austin,TX
479 Whispering Wind     .    72.00
  1261 Collins    Lillian   810824 510501 302-59-2781
Female Full Time X5616 80.00  700 78664 Round Rock,TX
9117 Beaver Creek Rd    .    48.00
  1265 Slye      Leonard R. 840331 601218 434-21-1300
Male  Half Time X5123   .    200 78742 Austin,TX
4106 Main St.           .     .
  1266 Redfox    Richard B. 850902 440404 210-65-2786
Male  Full Time X5386 48.00  400 78660 Pflugerville,TX
9807 Three Oaks Tr.     .    48.00
  1272 Smith     Garland P. 850413 540405 397-80-8491
Male  Full Time X5415  8.00  500 78602 Bastrop,TX
7594 Red Cliff Rd.      .    48.00
  1313 Smith     Jerry Lee  850130 420913 823-10-0951
Male  Full Time X5169  .     200 78745 Austin,TX
8203 Friar Tuck Ln.     .    16.00
  1327 Brooks    Ruben R.   820430 520225 789-56-2109
Male  Full Time X5347 80.00  400 78744 Austin,TX
2509 Loganberry Dr.     .    80.00
  1900 Smith      John        .      .      .
                         .     100
                         .      .
;
```

MYDATA.INITSEG

The SAS data set MYDATA.INITSEG is used to initially load the ACCTDBD database.

```
data mydata.initseg;
        /****  CUSTOMER data    ***/
  input @1   soc_sec_number    $11.
        @13  customer_name     $40.  /
        @1   addr_line_1       $30.
        @32  addr_line_2       $30.  /
        @1   city              $28.
        @30  state             $2.
        @33  country           $20.
        @54  zip_code          $10.  /
        @1   home_phone        $12.
        @14  office_phone      $12.
        /****  CHCKACCT data    ***/
        @27  check_account_number  12.0
        @40  check_amount          12.2
        @53  check_date            date7. /
        @1   filler1               $2.
        @4   check_balance         12.2
        /****  CHCKDEBT data    ***/
        @17  check_debit_amount    12.2
        @30  check_debit_date      date7.
        @38  filler2               $2.
        @41  check_debit_time      $8.    /
```

```
        @1   check_debit_desc      $59.;
    format check_date  date7.;
    format check_debit_date  date7.;
  datalines;
667-73-8275 WALLS, HOOPER J.
                    4525 CLARENDON RD
RAPIDAN          VA USA           22215-5600
803-657-3098 803-645-4418 345620145345 1702.19 15MAR95
        1266.34          .   19MAR94    21:22:53
CHECK 2947;
```

MYDATA.PHONENUM

The SAS data set MYDATA.PHONENUM is used to update the ACCTDBD database.

```
data mydata.phonenum;
  soc_sec_number = '667-73-8275';
  home_phone = '703-657-3098';
  office_phone = '703-645-4418';
  output;
  soc_sec_number = '434-62-1234';
  home_phone = '703-645-441 ';
  office_phone = '            ';
  output;
  soc_sec_number = '178-42-6534';
  home_phone = '703-657-1346';
  office_phone = '703-657-1345';
  output;
  soc_sec_number = '156-45-5672';
  home_phone = '703-657-5656';
  office_phone = '703-623-4257';
  output;
  soc_sec_number = '657-34-3245';
  home_phone = '703-345-4346';
  office_phone = '703-355-5438';
  output;
  soc_sec_number = '456-45-3462';
  home_phone = '703-657-3566';
  office_phone = '703-645-1212';
  output;
  soc_sec_number = '416-41-3162';
  home_phone = '703-657-3166';
  office_phone = '703-615-1212';
  output;
run;

proc print data=mydata.phonenum;
  title2 'SAS Data Set MYDATA.PHONENUM';
run;
```

MYDATA.SAVCRED

The SAS data set MYDATA.SAVCRED is used to add the savings credit path to the ACCTDBD database.

```
data mydata.savcred;
        /****  CUSTOMER data    ***/
  input @1   soc_sec_number    $11.
        @13  customer_name     $40.  /
        @1   addr_line_1       $30.
        @32  addr_line_2       $30.  /
        @1   city              $28.
        @30  state             $2.
        @33  country           $20.
        @54  zip_code          $10.  /
        @1   home_phone        $12.
        @14  office_phone      $12.
        /****  SAVEACCT data    ***/
        @27  savings_account_number 12.0
        @40  savings_amount         12.2
        @53  savings_date           date7. /
        @1   filler4                $2.
        @4   savings_balance        12.2
       /****  SAVECRDT data    ***/
        @17  savings_credit_amount  12.2
        @30  savings_credit_date    date7.
        @38  filler6                $2.
        @41  savings_credit_time    $8.   /
        @1   savings_credit_desc    $59.;
    format savings_credit_date  date7.;
  datalines;
667-73-8275 WALLS, HOOPER J.


            459923888253      784.29  28MAR95
   672.63         8.45 30MAR95    09:34:18
INTEREST
434-62-1234 SUMMERS, MARY T.
                              4322 LEON ST.
GORDONSVILLE        VA USA         26001-0670
            345689404732      8406.0  27MAR95
 8364.24        41.82 30MAR95    23:46:03
INTEREST
436-42-6394 BOOKER, APRIL M.
                           9712 WALLINGFORD PL.
GORDONSVILLE        VA USA         26001-0670
           144256844728       809.45 21MAR95
 1032.23        50.00 26MAR95    12:26:15
INTEREST
434-62-1224 SMITH, JAMES MARTIN
                              133 TOWNSEND ST.
GORDONSVILLE       VA USA         26001-0670
           345689473762       130.64 15MAR95
  261.64         1.31 30MAR95    23:45:53
INTEREST
434-62-1224 SMITH, JAMES MARTIN
                              133 TOWNSEND ST.
GORDONSVILLE       VA USA         26001-0670
           345689498217      9421.79 16MAR95
```

```
 9374.92        46.07 30MAR95    23:45:32
INTEREST
178-42-6534 PATTILLO, RODRIGUES
                                 9712 COOK RD.
ORANGE             VA USA         26042-1650
           345689462413       950.96 15MAR95
  946.23         4.73 30MAR95    23:44:25
INTEREST
156-45-5672 O'CONNOR, JOESPH
                                 235 MAIN ST.
ORANGE             VA USA         26042-1650
           345689435776       136.40 27MAR95
  284.97         1.43 30MAR95    23:48:56
INTEREST
657-34-3245 BARNHARDT, PAMELA S.
                                 RT 2 BOX 324
CHARLOTTESVILLE    VA USA         25804-0997
           859993641223       845.35 18MAR95
 2553.45        71.44 26MAR95    08:41:28
INTEREST
667-82-8275 COHEN, ABRAHAM
                                 2345 DUKE ST.
CHARLOTTESVILLE    VA USA         25804-0997
           884672297126       945.25 26MAR95
  793.25        52.33 28MAR95    11:45:26
INTEREST
456-45-3462 LITTLE, NANCY M.


          345689463822       929.24 25MAR95
  924.62         4.62 30MAR95    23:46:01
INTEREST
234-74-4612 WIKOWSKI, JONATHAN S.
                            4356 CAMPUS DRIVE
RICHMOND           VA USA         26502-3317
               .                        .   .
             .           .   .

;
```

MYDATA.SAVDEBD

The SAS data set MYDATA.SAVDEBD is used to add the savings debit path to the ACCTDBD database.

```
data mydata.savdebd;
        /****  CUSTOMER data    ***/
  input @1   soc_sec_number      $11.
        @13  customer_name       $40.  /
        @1   addr_line_1         $30.
        @32  addr_line_2         $30.  /
        @1   city                $28.
        @30  state               $2.
        @33  country             $20.
```

```
        @54  zip_code            $10.  /
        @1   home_phone          $12.
        @14  office_phone        $12.
        /****  SAVEACCT data    ***/
        @27  savings_account_number 12.0
        @40  savings_amount         12.2
        @53  savings_date           date7. /
        @1   filler4                $2.
        @4   savings_balance        12.2
        /****  SAVEDEBT data    ***/
        @17  savings_debit_amount   12.2
        @30  savings_debit_date     date7.
        @38  filler5                $2.
        @41  savings_debit_time     $8.    /
        @1   savings_debit_desc     $59.;
    format savings_date  date7.;
    format savings_debit_date  date7.;
  datalines;
667-73-8275 WALLS, HOOPER J.


          459923888253       784.29 28MAR95
        672.63        .   .

434-62-1234 SUMMERS, MARY T.
                               4322 LEON ST.
GORDONSVILLE       VA USA        26001-0670
          345689404732       8406.00 27MAR95
        8364.24        .   .

436-42-6394 BOOKER, APRIL M.
                          9712 WALLINGFORD PL.
GORDONSVILLE       VA USA        26001-0670
          144256844728       809.45 21MAR95
        1032.23        .   .

434-62-1224 SMITH, JAMES MARTIN
                              133 TOWNSEND ST.
GORDONSVILLE       VA USA        26001-0670
          345689473762       130.64 15MAR95
         261.64     132.31 03APR94    14:42:43
MAIN ST BRANCH WITHDRAWAL
434-62-1224 SMITH, JAMES MARTIN
                              133 TOWNSEND ST.
GORDONSVILLE       VA USA        26001-0670
          345689498217      9421.79 16MAR95
        9374.92        .   .

178-42-6534 PATTILLO, RODRIGUES
                               9712 COOK RD.
ORANGE             VA USA        26042-1650
          345689462413       950.96 15MAR95
         946.23        .   .
```

```
156-45-5672 O'CONNOR, JOESPH
                                235 MAIN ST.
ORANGE             VA USA         26042-1650
          345689435776       136.40 27MAR95
         284.97     150.00 31MAR94    12:23:42
ATM GREEN ST
657-34-3245 BARNHARDT, PAMELA S.
                                RT 2 BOX 324
CHARLOTTESVILLE    VA USA         25804-0997
          859993641223       845.35 18MAR95
        2553.45        .   .

667-82-8275 COHEN, ABRAHAM
                                2345 DUKE ST.
CHARLOTTESVILLE    VA USA         25804-0997
         884672297126       945.25 26MAR95
         793.25        .   .

456-45-3462 LITTLE, NANCY M.


         345689463822       929.24 25MAR95
         924.62        .   .

234-74-4612 WIKOWSKI, JONATHAN S.
                                4356 CAMPUS DRIVE
RICHMOND           VA USA         26502-3317
                  .                      .   .
            .          .   .

;
```

MYDATA.CUSTOMER

The SAS data set MYDATA.CUSTOMER is a SAS data set used to update the ACCTDBD database.

```
data mydata.customer;
        /****  CUSTOMER data    ***/
  input @1   soc_sec_number    $11.
        @13  customer_name     $40. /
        @1   addr_line_1       $30.
        @32  addr_line_2       $30. /
        @1   city              $28.
        @30  state             $2.
        @33  country           $20.
        @54  zip_code          $10. /
        @1   home_phone        $12.
        @14  office_phone      $12.;
  datalines;
131-73-2785 HUTTLINGER, HORTENSE H.
                               2785 HILLARY PL
RAPIDAN          VA USA           22215-5600
803-657-4097 803-645-4419
```

```
232-62-2432 MANNERLY, MAYNARD M.
                              6525 MORGAN ST
RAPIDAN          VA USA           22215-5600
803-657-9066 803-645-4420
;
```

MYDATA.NEWADDR

The SAS data set MYDATA.NEWADDR is in Version 6 format and is used to update the ACCTDBD database.

```
data mydata.newaddr;
  /****  CUSTOMER data    ***/
  input @1   ssn         $11.
        /* social security number */
        @13  newaddr1    $30.
        /* first line of address */
        @44  newaddr2    $30. /
        /* second line of address */
        @1   newcity     $28.
        /* customer city */
        @30  newstate    $2.
        /* customer state */
        @33  newzip      $10.;
        /* customer zip code */
  datalines;
178-42-6534              1111 PAUL PLACE
RAPIDAN       VA 22215-5600
156-45-5672              2222 OSCAR DR.
ORANGE        VA 26042-1650
;
```

VER6.SSNUMS

The SAS data set VER6.SSNUMS is in Version 6 format and is used to update the ACCTDBD database.

```
data ver6.ssnums;
  input @1  ssnumb $11.
        @13 name   $40.;
  datalines;
267-83-2241 GORDIEVSKY, OLEG
276-44-6885 MIFUNE, YUKIO
352-44-2151 SHIEKELESLAM, SHALA
436-46-1931 NISHIMATSU-LYNCH, CAROL
;

proc print data=mydata.ssnums;
  title2 'SAS Data Set VER6.SSNUMS';
run;
```

SAS Statements for Loading DB2 Table BANKCHRG

The following are the executable SAS DATA step, PROC APPEND, and PROC ACCESS statements to load DB2 table BANKCHRG.

Creating SAS Data Set MYDATA.BANK

The SAS data set MYDATA.BANK is used to load the DB2 table BANKCHRG.

Note: If you do not have DB2 at your site, change MYDATA.BANK to MYDATA.BANKCHRG and execute only the following program: △

```
data mydata.bank;
  input @1  ssn      $11.
        @13 accountn 12.
        @26 chckchrg 5.2
        @32 atmfee   5.2
        @38 loanchrg 6.2;
  format accountn 14.
         chckchrg 5.2
         atmfee   5.2
         loanchrg 6.2;
  datalines;
667-73-8275 345620145345 3.75  5.00  2.00
434-62-1234 345620104732 15.00 25.00 552.23
436-42-6394 345620135872 1.50  7.50  332.15
434-62-1224 345620134564 9.50  0.00  0.00
178-42-6534 .            0.50  15.00 223.77
156-45-5672 345620123456 0.00  0.00  0.00
657-34-3245 345620132455 10.25 10.00 100.00
667-82-8275 .            7.50  7.50  175.75
456-45-3462 345620134522 23.00 30.00 673.23
234-74-4612 345620113262 4.50  7.00  0.00
;

proc print data=mydata.bank;
  title2 'SAS Data Set MYDATA.BANK';
run;
```

Loading DB2 Table BANKCHRG from MYDATA.BANK

The following program loads DB2 table BANKCHRG from the SAS data set MYDATA.BANK. You must have DB2 installed at your site to run this program.

```
proc dbload dbms=db2 data=mydata.bank;
  accdesc=mylibdb2.bankchrg;
  table=<owner>.bankchrg;
  load;
run;
```

DB2 View Descriptor for BANKCHRG

The following program creates a DB2 view descriptor for the DB2 table BANKCHRG. You must have DB2 installed at your site to run this program.

```
proc access dbms=db2 ad=mylibdb2.bankchrg;
  create vlibdb2.bankchrg.view;
  select all;
  list view;
run;

proc print data=vlibdb2.bankchrg;
  title2 'DB2 Table BANKS.BANKCHRG';
run;
```

Glossary

ACB
See Application Control Block (ACB).

ACBLIB
the data set containing the Application Control Blocks.

access descriptor
a SAS/ACCESS file that describes data to the SAS System that is managed by an external software vendor's DBMS product. You use an access descriptor as a master file to create view descriptors. See also view descriptor.

Application Control Block (ACB)
a DL/I control block that contains the combined information from the Database Descriptions (DBDs) and Program Specification Blocks (PSBs).

attach parameter list
a set of parameters passed to DL/I when the IMS-DL/I engine or the IMS-DL/I DATA step interface is executed in a DL/I environment. The parameters vary for each region type. Most parameters can be modified with the SAS system options specified for the SAS/ACCESS interface for IMS-DL/I.

batch mode
a method of executing SAS programs in which you prepare a file containing SAS statements and any necessary operating system commands, and submit the program to the computer's batch queue. While the program executes, control of the SAS System returns to your terminal or workstation where you can perform other tasks. Batch mode is sometimes referred to as running in the background. The job output can be written to files or printed on an output device.

batch region
a DL/I processing environment for accessing DL/I databases. Database data sets must be allocated to this region. A batch region is supervised by the DL/I batch control program.

BMP region
an abbreviation for Batch Message Processing region. The BMP region is a DL/I processing environment in IMS/ESA DB/DC subsystems and in CICS for running programs that access active online DL/I databases and message queues, as well as non-DL/I data sets. Database data sets are allocated to an online control region, not to the BMP region.

browsing data
the process of viewing the observations in a file. Depending on how the file is accessed, observations may be viewed one at a time or as a group in a tabular format. You cannot update data that you are browsing.

call (DL/I)
a request made by the IMS-DL/I engine to DL/I or by the IMS-DL/I DATA step interface to access one or more segments of a database or message queue, or to perform some system function.

checkpoint
the result of a CHKP call. A checkpoint establishes a synchronization point in the execution of the program. A synchronization point is used by DL/I backout services to determine which updates to back out (cancel) in the event of an abend or system crash. The program must be restarted at the synchronization point.

checkpoint ID
an 8-byte value written to the DL/I log record to identify the program checkpoint.

child
a segment in a DL/I database that is the direct dependent of another segment, which is called its parent. The data in a dependent segment rely on the parent segment and all higher segments for complete identification and qualification.

command code
a special indicator used in a Segment Search Argument (SSA) to modify the type of call being issued. The most commonly used command code is the D code, which is used to issue a path call.

control block
a storage area created by the system containing information for controlling system operations. The control block is often loaded from a control block library; for example, Database Descriptions (DBDs) may be stored in DBDLIBs.

DATA step
a group of statements in a SAS program that begins with a DATA statement and ends with either a RUN statement, another DATA statement, a PROC statement, the end of the job, or the semicolon that immediately follows instream data lines. The DATA step enables you to read raw data or other SAS data sets and use programming logic to create a SAS data set, write a report, or write to an external file.

DATA step view
a DATA step program that generates a SAS data view. Like other SAS data views, a DATA step view contains a definition of data stored elsewhere; it does not contain the physical data. DATA step views can only function as input data sets in SAS System releases beyond Release 6.08.

An input DATA step view is generated from a DATA step program. The view's input data can come from one or more sources, including external files and other SAS data sets. Because a DATA step view only reads (opens for input) other files, you cannot update this view's underlying data.

data type
an attribute of every item in a database. The data type tells the operating system how much physical storage to set aside for the item and the type of data the item will contain. It is similar to the type attribute of SAS variables.

data value
an element in a collection of data values that are organized and presented to the SAS System in a rectangular structure of columns and rows. A data value represents the intersection of a row and a column.

database
an organized collection of related data. In IMS-DL/I, a database is a collection of interrelated data elements that are organized according to a particular data model that can be processed by multiple applications.

Database Administrator (DBA)
the person responsible for developing and maintaining database management systems at a computer site.

Database Description (DBD)
a DL/I control block that defines the hierarchical data structure and the physical characteristics of a database to DL/I.

database management system (DBMS)
an integrated software package that enables you to create and manipulate data in the form of databases. See also relational database management system.

Database Recovery Control (DBRC)
an IMS facility that controls restoration of databases after a system failure. DBRC also supports data sharing among IMS/ESA subsystems.

DBB region
a DL/I batch processing environment for running programs that can access DL/I databases as well as non-DL/I data sets. In a DBB region, DL/I accesses the ACBLIB for control block information.

DBD
See Database Description (DBD).

DBDGEN
the utility procedure that generates a Database Description (DBD).

DBDLIB
a data set that contains the Database Descriptions (DBDs).

DBRC
Database Recovery Control (DBRC).

dependent segment
a segment that has a parent segment. The data in a dependent segment rely on the parent segment and all higher segments for complete identification and qualification.

DL/I
an abbreviation for Data Language/I. DL/I is IBM's database language for IMS/ESA, CICS/OS/ESA, CICS/DOS/VS, and DL/I DOS/VS systems.

DLI region
a DL/I batch processing environment for running programs that can access DL/I databases, as well as non-DL/I data sets. No access to message queues is possible. In a DLI region, DL/I accesses the DBDLIB and PSBLIB for control block information.

editing data
the process of viewing a file with the ability to change its data. You might see the data one observation at a time or in a tabular format.

engine
a part of the SAS System that reads from or writes to a file. Each engine allows the SAS System to access files with a particular format. There are several types of engines. See also interface view engine.

feedback data
the data returned from IMS to the IMS-DL/I engine (usually in the PCB mask) after a DL/I call has been issued.

field
the smallest unit of data storage in an IMS-DL/I database.

file
a collection of related records treated as a unit. SAS files are processed and controlled through the SAS System and are stored in a SAS data library.

format
an instruction the SAS System uses to display or write each value of a variable. Some formats are supplied by SAS software. Other formats can be written by the user with the FORMAT procedure in base SAS software or with SAS/TOOLKIT software.

Get call
a DL/I call that retrieves one or more segments so that the contents of the segments can be processed or mapped by the IMS-DL/I engine.

hierarchical database
a database organized as a tree structure of segments. A DL/I database has a hierarchical data structure.

hierarchical sequence
the standard processing sequence for segments of a database record; the sequence is basically top-to-bottom, front-to-back, and left-to-right. When segments in a DL/I database are retrieved one after the other in sequential fashion, DL/I presents them in hierarchical sequence.

hierarchical structure
an arrangement of data in which records occur at distinct levels with different types of information at each level. Records are related to other records as ancestors, descendants, siblings, and so on.

IMS/ESA
an abbreviation for Information Management System/Enterprise System Architecture. IMS/ESA is an IBM database management system that uses the DL/I language.

IMS/ESA Resource Lock Manager (IRLM)
a facility for ensuring database integrity among multiple DL/I subsystems.

index
in SAS Software, a part of a SAS data file that stores both the values of a variable (in a SAS data file) and a set of directions that enable the SAS System, under certain circumstances, to locate observations in a data file more quickly and efficiently. Indexing variables usually makes data set processing faster, although the SAS System determines the most efficient way to process data maintained by the SAS System.

informat
an instruction the SAS System uses to read raw data values to create variable values. Some informats are supplied by SAS software. Other informats can be written by the user with the FORMAT procedure in base SAS software or with SAS/TOOLKIT software.

interactive line mode
a method of running SAS programs in which you enter one line of a SAS program at a time at the SAS session prompt. The SAS System processes each line immediately after you press the ENTER key. Procedure output and informative messages are returned directly to the display monitor.

interface view engine
a SAS System engine that retrieves data directly from files formatted by other software vendors and presents the data to the SAS System in the form of a SAS data

set. Interface view engines are transparent to the user and are not specified in the LIBNAME statement.

I/O area
a data structure into which a retrieved segment is placed or from which a segment being written is taken.

I/O PCB
an abbreviation for Input/Output Program Communication Block. The I/O PCB communicates information on non-database access requests.

IRLM
See IMS/ESA Resource Lock Manager (IRLM).

key field
a field that identifies and provides access to an occurrence of a segment. A key field is also called a sequence field.

libref
(1) the name temporarily associated with a SAS data library. You assign a libref with a LIBNAME statement or with operating-system control language. (2) the first part of a multilevel SAS filename indicating the SAS data library in which the file is stored. For example, in the name SASUSER.ACCOUNTS, the name SASUSER is the libref.

logical database
a collection of database segments from one or more physical databases. It enables the SAS/ACCESS IMS-DL/I interface to view a database structure that is different from the physical structure.

member
a file in a SAS data library.

member name
a name given to a file in a SAS data library. A member name can reference a SAS data set, catalog, access descriptor, or stored program.

member type
a name assigned by the SAS System that identifies the type of information stored in a SAS file. Member types include ACCESS, DATA, CATALOG, PROGRAM, and VIEW.

missing value
a value in the SAS System indicating that no data are stored for the variable in the current observation. By default, the SAS System prints a missing numeric value as a single period (.) and a missing character value as a blank space.

noninteractive mode
a method of running SAS programs in which you prepare a file of SAS statements and submit the program to the operating system. The program runs immediately and occupies your current session.

observation
the horizontal component of a SAS data set. An observation is a collection of data values associated with a single entity, such as a customer or state. Each observation contains one data value for each variable in the data file. An observation is analogous to a record in an IMS database. Unlike rows in a DBMS table or file, observations in a SAS data file have an inherent order.

online access region
a DL/I processing environment for running batch-type programs that can access active online DL/I databases. In the SAS/ACCESS interface to IMS-DL/I, the only online access region type supported is BMP region.

online control region
a DL/I region that controls databases and terminals and schedules activities using these resources for online processing.

parent
a segment in a DL/I database that has one or more dependent segments, which are called its children. The data in a dependent segment rely on the parent segment and all higher segments for complete identification and qualification.

path
a single route through a database following the hierarchical sequence of segments from a higher-level segment to a lower-level segment.

path call
a DL/I call to a database that returns multiple segments from the hierarchical path.

PCB
See Program Communication Block (PCB).

PCB mask
a data structure to which DL/I returns information about the DL/I calls issued by an application.

physical database
a collection of database segments in a specified hierarchical structure. These segments are organized according to a particular DL/I access method.

PROC SQL view
a SAS data view defined by the SQL procedure that reads data from its underlying SAS data files, other PROC SQL views, SAS/ACCESS views, or DATA step views. Its output table can be a subset or a superset of one or multiple underlying structures. In the current release of the SAS System, you cannot reference a PROC SQL view to update its underlying data.

PROC step
a group of SAS statements that call and execute a procedure, usually with a SAS data set as input.

Program Communication Block (PCB)
a DL/I control block that defines a message queue or the part of a database that can be accessed by the IMS-DL/I engine or by the IMS-DL/I DATA step interface. A PCB is part of a Program Specification Block (PSB).

Program Specification Block (PSB)
a DL/I control block that defines the DL/I resources used by the IMS-DL/I engine or by the IMS-DL/I DATA step interface. Each database used by the IMS-DL/I engine is defined by a separate Program Communication Block (PCB) within the PSB.

program view
the part of a database that can be accessed by the IMS-DL/I engine or by the IMS-DL/I DATA step interface. The program view is established by the Program Communication Block (PCB).

PSB
See Program Specification Block (PSB).

PSBGEN
the utility procedure that generates a Program Specification Block (PSB).

PSBLIB
a data set that contains the Program Specification Blocks (PSBs).

qualified call
a DL/I call that specifies at least one Segment Search Argument (SSA).

qualified SSA
a Segment Search Argument that contains one or more qualification statements to specify search criteria for locating particular segment occurrences.

random access
the access mode used by the IMS-DL/I engine or by the IMS-DL/I DATA step interface when a WHERE statement is specified from which the engine can generate qualified Segment Search Arguments. In the SAS/ACCESS interface to IMS-DL/I, the distinction between sequential and random access differs from that of some other programming languages.

read integrity
the sharing of database access so that two programs cannot access a record simultaneously if one of the programs intends to update the record.

region type
the kind of DL/I processing environment. The IMS-DL/I engine uses two categories of region types: batch regions (DLI or DBB) and online access regions (BMP).

relational database management system
a database management system that organizes and accesses data according to relationships between data items. The main characteristic of a relational database management system is the two-dimensional table.

restart
the process of resuming an interrupted program without repeating completed transactions.

restricted option
a SAS option that has been installed at your site so that its default setting cannot be overridden by the applications programmers.

return code
a code passed to the operating system that reports the results of executing a command or job step.

root segment
the highest-level segment in a database.

SAS data file
a SAS data set that contains both the data values and the descriptor information.

SAS data library
a collection of one or more SAS files that are recognized by the SAS System. Each file is a member of the library. See also libref and member.

SAS data set
a collection of information stored as a unit under the SAS System. A SAS data set is arranged in a rectangular, two-dimensional format. Each item in a SAS data set is called a data value. Data values in a row comprise an observation, and those in a column comprise a variable. See also SAS data file and SAS data view.

SAS data view
a SAS data set in which the descriptor information and observations are obtained from other files. SAS data views store only the information required to retrieve data values or descriptor information.

SAS/ACCESS views
See SAS data view and view descriptor.

search field
a field defined to DL/I in the Database Description (DBD) that can be used to search for particular segments. A search field does not provide unique identification of the segment.

segment
in a DL/I database, a grouping of related items of data in a database structure. The segment is the unit of data that can be accessed by the IMS-DL/I engine or by the IMS-DL/I DATA step interface.

segment level
the relative distance of a given segment from the root segment along the hierarchical path. This is usually represented numerically, with the root segment at level 1 and its immediate dependents at level 2.

segment occurrence
in a DL/I database, a specific instance in a set of segments of one type.

Segment Search Argument (SSA)
the formatted search criteria passed to DL/I to identify a particular segment or group of segments to be processed. Multiple SSAs may be specified on one DL/I call.

segment type
in a DL/I database, a category of related data elements. There may be multiple segment occurrences for a given segment type.

sensitive segment
a segment in a DL/I database that the IMS-DL/I engine or the IMS-DL/I DATA step interface can access. A segment is defined as *sensitive* for a given program in the Program Specification Block (PSB).

sequence field
See key field.

sequential access
the default access mode used by the IMS-DL/I engine to retrieve all segments down one path of a database. Sequential access is a method of file access in which the records are read or written one after the other from the beginning of the file to the end. In the SAS/ACCESS interface to IMS-DL/I, the distinction between sequential and random access differs somewhat from that of other programming languages.

siblings
segments that share a common parent segment.

SSA
See Segment Search Argument (SSA).

status code
a 2-byte indicator field returned to an application program by DL/I to indicate the relative success of an attempted call.

Structured Query Language (SQL)
the standardized, high-level query language used in relational database management systems to create and manipulate database management system objects. The SAS System implements SQL through the SQL procedure.

subsystem
a complete DL/I configuration, including the DL/I region controller and service modules, the DL/I databases.

synchronization point
a commitment of completed updates and a time at which all DL/I resources held since the last synchronization point are released. Synchronization points are

established explicitly by CHKP calls, which the IMS-DL/I engine issues (by default) at the beginning and end of update processing. Synchronization points can be used to resume processing of an interrupted job.

twins
segments that represent multiple occurrences of the same segment type under a single parent.

undefined field
a field that is not defined to DL/I in a Database Description (DBD). An undefined field is neither a sequence field nor a search field. A segment cannot be accessed by specifying this field to DL/I.

unqualified call
a DL/I call that contains no Segment Search Argument (SSA).

unqualified SSA
a Segment Search Argument that specifies a segment type and, optionally, a command code.

Update call
a DL/I call that signals the intent to alter (modify, delete, or add) information in the database.

update integrity
the sharing of database access so that two programs cannot access a record simultaneously if both of the programs intend to update the record.

variable
a column in a SAS data set. A variable is a set of data values that describe a given characteristic across all observations.

view descriptor
a SAS/ACCESS file that defines all or a subset of interface product data described by an access descriptor. The access descriptor describes the data in a single product table or view, when views are allowed in the product. See also access descriptor.

windowing procedure
a procedure that uses windows and menus to accomplish a SAS System task. For example, ACCESS, FSVIEW, and DBLOAD are windowing procedures.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS/ACCESS® Interface to IMS-DL/I Software: Reference, Version 8*, Cary, NC: SAS Institute Inc., 1999. 316 pp.

SAS/ACCESS® Interface to IMS-DL/I Software: Reference, Version 8

ISBN 1–58025–548–5

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, October 1999