International Classification of Diseases (ICD)

Technical Manual

Version 18.0
October 2000
Revised Apr 21, 2014

Department of Veterans Affairs (VA)
Office of Information and Technology (OI&T)
Office of Enterprise Development (OED)

Revision History

Date	Description of Change	Author
06/26/2003	Initial Version	Crosskey, Johnny
09/04/2003	LEX*2.0*25 – Code Set Versioning	Crosskey, Johnny
04/21/2014	ICD*18.0*57 - Update with ICD-10	Kimball Rowe
04/24/2014	Technical Writer Review	Kimberlee Mann

Table of Contents

1. Introduction

The International Classification of Diseases (ICD), Clinical Modification is a system of codes and terminology that arranges diseases and injuries into groups according to established criteria. It is based on the design for the classification of morbidity and mortality information for statistical purposes and published by the World Health Organization (WHO). These codes provide an effective means of communication between physicians, patients, and third parties.

ICD V. 18.0 exported the International Classification of Diseases, Ninth Revision, Clinical Modification (ICD-9-CM) classification system containing both diagnostic and procedural codes. Since its initial release in October 2000, the ICD package has had (2) two major changes:

The Code Set Versioning project of 2003-2004 modified the data dictionary and routines so the ICD package can provide time sensitive information based on the date and time service was provided to the patient or the date and time the code and ICD terminology was used. Users can select codes and terminology that were appropriate on a date that an event occurred. (Patches ICD*18.0*7 and ICD*18.0*12)

The release of ICD-10-CM diagnostic codes and terms and ICD-10-PCS procedural coded and terms included the International Classification of Diseases, Tenth Revision, Clinical Modification (ICD-10-CM) diagnostic related codes and terms and the International Classification of Diseases, Tenth Revision, Procedural Classification System (ICD-10-PCS) procedural related codes and terms. (Patch ICD*18.0*57)

1.1 Code Formats

ICD-9-CM Diagnoses Codes
- 3 – 5 digits
- First digit is alpha (E or V) or numeric
- Digits 2 – 5 are numeric
- Contains a decimal point
- Examples:
  - 496
  - 511.9
  - V02.61
  - E891.8

ICD-10-CM Diagnoses Codes
- 3 – 7 digits
- Digit 1 is alpha
- Digit 2 is numeric
- Digits 3 – 7 are alpha or numeric
- Not case sensitive
- Contains a decimal point
- Examples:
  - A78
  - A69.21
  - S52.131a

ICD-9-CM Procedure Codes
- 3 – 4 digits
- All digits numeric
- Contains a decimal point
- Examples:
  - 43.5
  - 44.42

ICD-10-PCS Procedure Codes
- 7 digits
- Either alpha or numeric
- Letters O and I not used
- No decimal point
- Examples:
  - 0FB03ZX
  - 0DQ10ZZ

2. Implementation and Maintenance (Post ICD-10)

There are no site-configurable features connected with the ICD package.

Total disk space requirements for the ICD globals are as follows:

Global	Blocks	Bytes	MB
^ICD9	16,337	103,010,968	100.59
^ICD0	9,431	63,468,660	61.98
^ICDS	1	324	0.00
^ICD	327	1,786,772	1.74
^ICM	9	64,268	0.06
^ICDID	4	21,584	0.02
^ICDIP	4	21,508	0.02
Total	**26,113**	**168,374,084**	**164.42**

3. Special Lookup Routine – ICDEXLK

A special lookup program was written for the ICD DIAGNOSIS file #80 and ICD OPERATION/PROCEDURE file #80.1 to navigate through the versioned (date sensitive) data stored in these files. The name of the special lookup is stored in the data dictionary for these files:

```
^DD(80,0,"DIC")="ICDEXLK"
^DD(80.1,0,"DIC")="ICDEXLK"
```

Each time an application makes a ^DIC call to either file 80 or 80.1, the special lookup routine is invoked, provided the FileMan variable DIC(0) does not contain an "I" for "ignore the special lookup."

NOTE: Only the ^DIC call honors the special lookup routine. Those calls that allow the user to specify the indexes (IX^DIC and MIX^DIC1), and the Data Base Server calls (FIND^DIC, $$FIND1^DIC, and UPDATE^DIE) all ignore the Special Lookup Program. As a result, the FileMan calls that ignore the Special Lookup Program will not be able to conduct versioned searches or return versioned data so use IX^DIC, MIX^DIC1 FIND^DIC, and $$FIND1^DIC with a great deal of care. Never use any FileMan entry point that alters the data in these files (i.e., ^DIE, EN^DIB, ^DIK FILE^DIE, UPDATE^DIE and FILE^DICN)

3.1 Package Special Lookup Variables

The following local variables in the ICD namespace should be NEWed or KILLed by the calling application

3.1.1 ICDVDT Versioning Date (Fileman format)

If this variable is supplied, then only active codes on that date will be included in the selection list.

```
   1.  V74.6    SCREENING FOR YAWS
   2.  V77.5    SCREENING FOR GOUT
   3.  V76.9    SCREEN-NEOPLASM NOS
   4.  V76.43   SCREEN MAL NEOP-SKIN
   5.  V78.8    SCREEN-BLOOD DIS NEC
```

If not supplied, the date will default to TODAY and all codes may be selected, active and inactive.

```
  1.  V74.6   SCREENING FOR YAWS
  2.  V77.5   SCREENING FOR GOUT
  3.  V76.8   SCREEN-NEOPLASM NEC (Inactive)
  4.  V76.9   SCREEN-NEOPLASM NOS
  5.  V76.43  SCREEN MAL NEOP-SKIN
```

3.1.2 ICDSYS Coding System (from file 80.4)

```
 1    ICD   ICD-9-CM
 2    ICP   ICD-9 Proc
30    10D   ICD-10-CM
31    10P   ICD-10-PCS
```

If supplied only codes belonging to the coding system will be included in the selection list.

```
S ICDSYS=1,X="DIABETES MELLITUS KETOACIDOSIS"

2 matches found

  1.  249.11  SEC DM KETOACD UNCNTRLD (Major CC)
  2.  249.10  SEC DM KETO NT ST UNCNTR (Major CC)

S ICDSYS=30,X="DIABETES MELLITUS KETOACIDOSIS"

8 matches found

  1.  E09.11  Drug/chem diabetes mellitus w
              ketoacidosis w coma
  2.  E13.11  Oth diabetes mellitus with
              ketoacidosis with coma
  3.  E09.10  Drug/chem diabetes mellitus w
              ketoacidosis w/o coma
  4.  E10.11  Type 1 diabetes mellitus with
              ketoacidosis with coma
  5.  E13.10  Oth diabetes mellitus with Ketoacidosis
              without coma
```

If not supplied codes from any coding system will be included in the selection list.

```
S X="DIABETES MELLITUS KETOACIDOSIS"

10 matches found

  1.  249.11   SEC DM KETOACD UNCNTRLD (Major CC)
  2.  249.10   SEC DM KETO NT ST UNCNTR (Major CC)
  3.  E09.11   Drug/chem diabetes mellitus w
               ketoacidosis w coma
  4.  E13.11   Oth diabetes mellitus with Ketoacidosis
               with coma
  5.  E09.10   Drug/chem diabetes mellitus w
               ketoacidosis w/o coma
```

3.1.3 ICDFMT Display Format

```
Controls the format of the terms and code presented
for selection on the selection list, 1-4, default = 1
```

```
1   Fileman format, code and short text (default)

      250.00    DMII WO CMP NT ST UNCNTR

2   Fileman format, code and description

      250.00   DIABETES MELLITUS WITHOUT MENTION OF
               COMPLICATION, TYPE II OR UNSPECIFIED
               TYPE, NOT STATED AS UNCONTROLLED

3   Lexicon format, short text followed by code

      DMII WO CMP NT ST UNCNTR (250.00)

4   Lexicon format, description followed by code

      DIABETES MELLITUS WITHOUT MENTION OF
      COMPLICATION, TYPE II OR UNSPECIFIED TYPE, NOT
      STATED AS UNCONTROLLED (250.00)
```

3.1.4 Fileman Variables used

The following are FileMan local variables used by the Special Lookup and should be NEWed or KILLed by the calling application

Input

X	**(Optional) User's input. If it exists, DIC(0) should not contain "A" for "Ask"**
DIC	**(Required) The file number or an explicit global root in the form ^GLOBAL(or ^GLOBAL(X,Y,**
DIC(0)	**(Optional) A string of alphabetic characters which alter how DIC responds. At a minimum this string must be set to null. (Required) Default value for ICD files "AEM"**

The following characters are applicable to a versioned file:

```
A   Ask the entry; if erroneous, ask again
B   Only the B index is used
E   Echo information
F   Forget the lookup value
I   Ignore the special lookup program
M   Multiple-index lookup allowed
O   Only find one entry if it matches exactly
S   Suppresses display of .01
T   Search until user selects or enters ^^
X   EXact match required
Z   Zero node in Y(0), external form in Y(0,0)
```

The following characters are NOT applicable to a versioned file (not used):

C Versioned cross-references not turned off
K Primary Key not established
L Learning a new entry LAYGO not allowed
N Uppercase, IEN lookup allowed (not forced)
n ICD has no pure numeric entries
Q Input is pre-processed, ?? not necessary
U All values are external
V Verification is not optional

DIC("A") **(Optional) A prompt that is displayed prior to the reading of the X input. If DIC("A") is not defined, a prompt will be supplied by the special lookup routine.**

DIC("B") **(Optional) The default answer which is presented to the user when the lookup prompt is issued. If a terminal user simply presses the Enter/Return key, the DIC("B") default value will be used, and returned in X. DIC("B") will only be used if it is non-null.**

DIC("S") **(Optional) DIC("S") is a string of M code that DIC executes to screen an entry from selection. DIC("S") must contain an IF statement to set the value of $T. Those entries that the IF sets as $T=0 will not be displayed or selectable. When the DIC("S") code is executed, the local variable Y is the internal number of the entry being screened and the M naked indicator is at the global level @(DIC_"Y,0)")**

DIC("W") **(Optional) An M command string which is executed when DIC displays each of the entries that match the user's input. The condition of the variable Y and of the naked indicator is the same as for DIC("S"). WARNING: If DIC("W") is defined, it overrides the display of the versioned identifiers for the file. Thus, if DIC("W") is set it will suppress the display of versioned data and there is a risk of displaying unversioned data.**

DIC("?N",<file>)=n **(Optional) The number "n" should be an integer set to the number of entries to be displayed on the screen at one time when using "?" help in a lookup.**

3.1.5 FileMan Variables not used

DIC("DR")
DIC("PTRIX",<from>,<to>,<file>)
DIC("T")
DIC("V")
DIC("?PARAM",<file>,"INDEX")
DIC("?PARAM",<file>,"FROM",<subscript>)
DIC("?PARAM",<file>,"PART",<subscript>)

3.1.6 FileMan Variables KILLed

```
DLAYGO
DINUM
```

3.1.7 FileMan Variables Modified

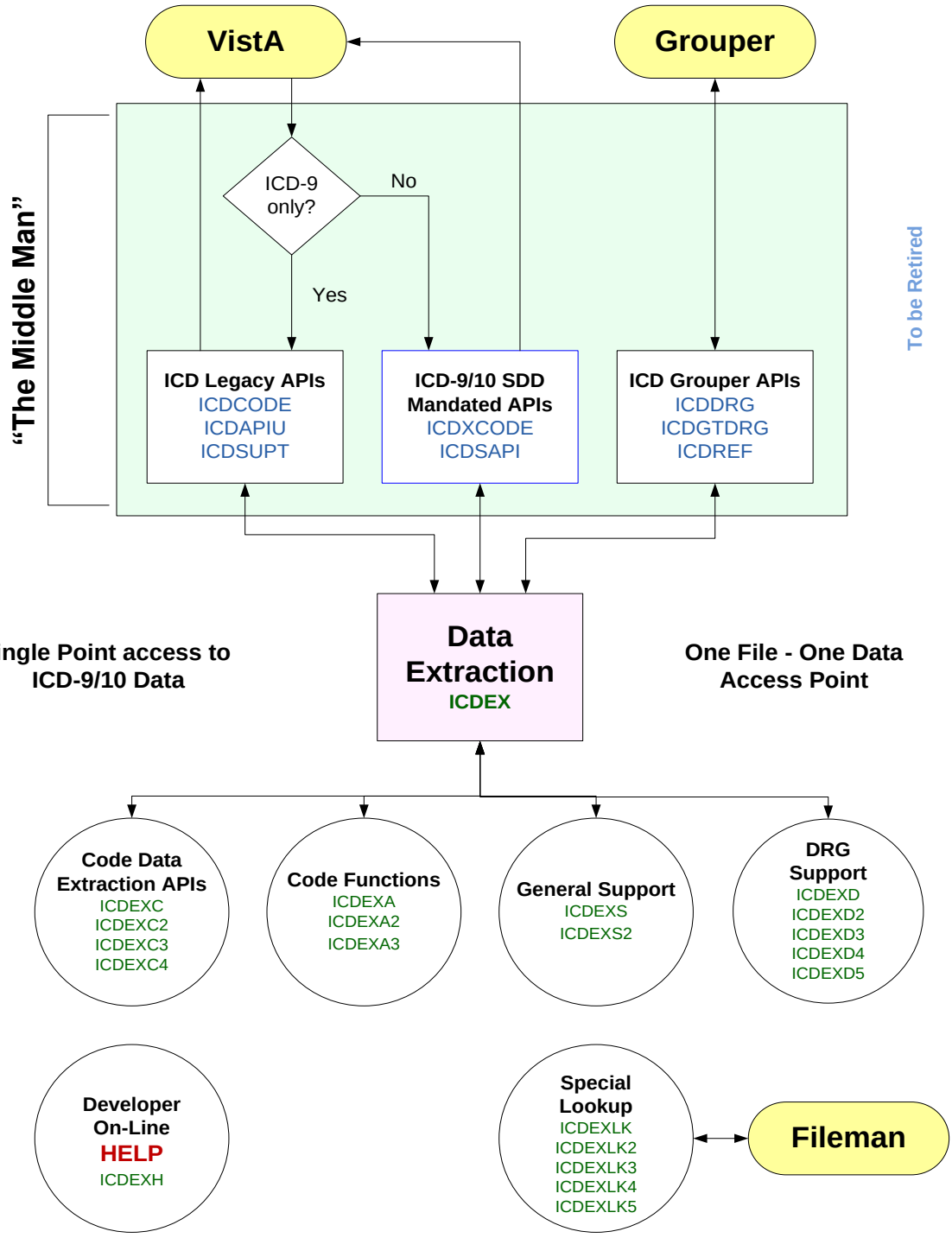
```
If DIC(0) contains an "L" it will be removed
```

3.1.8 Output Variables

```
Always Returned

      Y           IEN ^ Code              FileMan

If DIC(0) contains "Z"

      Y(0)        0 Node                  FileMan
      Y(0,0)      Code                    FileMan
      Y(0,1)      $$ICDDX or $$ICDOP      Non-FileMan
      Y(0,2)      Long Description        Non-FileMan
```

4. Applications Programmer Interfaces (APIs)

4.1 Overview (Data Extraction)



4.2 Legacy APIs (ICD-9-CM)

The following APIs are supported under the ICD-9-CM coding system and will continue to be supported throughout the transition to ICD-10. These APIs will be retired once ICD-10 is fully operational. It is suggested that applications use the supported ICD-10 API (which also work with ICD-9 codes) or subscribe to the equivalent data extraction API in the routine ICDEX.

4.2.1 ICDCODE, ICR 3990 (scheduled for retirement)

This Integration Control Registration (ICR) shall be retired 18 months after the ICD-10 implementation date established by the Department of Health and Human Services (HHS).. See equivalent APIs in ICR 5747.

ICD-9 Diagnosis Data

$$ICDDX^ICDCODE(CODE,CDT,DFN,SRC) ICR 3990

Input:

```
CODE  Code/IEN (required)
CDT   Date (default = TODAY)
DFN   Not in use
SRC   Source
        0 = exclude local codes
        1 = include local codes
```

Output:

```
Returns a 19 piece string delimited by ^

 1  IEN of code in file 80
 2  ICD-9 Dx Code (#.01)
 3  Identifier String ID;ID;ID
 4  Versioned Dx (67 multiple)
 5  Unacceptable as Principal Dx (#101)
 6  Major Dx Cat (#5)
 7  MDC13 (5.5)
 8  Compl/Comorb (#70)
 9  ICD Expanded (#8) 1:Yes 0:No
10  Status (66 multiple)
11  Sex (#9.5)
12  Inactive Date (66 multiple)
13  MDC24 (#5.7)
14  MDC25 (#5.9)
15  Age Low (#14)
16  Age High (#15)
17  Activation Date (.01 of 66 multiple)
18  Message
19  Versioned Complication/Comorbidity (#103)

  or
```

```
-1^Error Description
```

Recommended Replacement API:
$$ICDDX^ICDEX(CODE,CDT,SYS,FMT)
Subscribe to ICR 5747

ICD-9 Procedure Data

$$ICDOP^ICDCODE(CODE,CDT,DFN,SRC) ICR 3990

Input:

```
CODE  ICD code or IEN format, (required)
 CDT  Date (default = TODAY)
 DFN  Not in use
 SRC  Source
        0 = exclude local codes
        1 = include local codes
```

Output:

```
Returns a 14 piece string delimited by ^

 1  IEN of code in file 80.1
 2  ICD-9 code (#.01)
 3  Id (#2)
 4  MDC24 (#5)
 5  Versioned Oper/Proc (67 multiple)
 6  <null>
 7  <null>
 8  <null>
 9  ICD Expanded (#8) 1:Yes 0:No
10  Status (66 multiple)
11  Use with Sex (#9.5)
12  Inactive Date (66 multiple)
13  Activation Date (66 multiple)
14  Message

  or

-1^Error Description
```

Recommended Replacement API:
$$ICDOP^ICDEX(CODE,CDT,SYS,FMT)
Subscribe to ICR 5747

ICD-9 Description

$$ICDD^ICDCODE(CODE,'OUTARR',CDT) ICR 3990

```
Input:

    CODE   ICD Code or IEN (required)
    ARY    Array Name for description
```

```
                e.g. "ABC" or "ABC("TEST")"
                Default = ^TMP("ICDD",$J)
          CDT   Date (default = TODAY)

     Output:

          #   Number of lines in array

          @ARY(1:n) - Versioned Description (68 multiple)
          @ARY(n+1) - blank
          @ARY(n+1) - message: CODE TEXT MAY BE INACCURATE

            or

          -1^Error Description

          ** NOTE - USER MUST INITIALIZE ^TMP("ICDD",$J), IF USED
**
```

Recommended Replacement API:
$$ICDD^ICDEX(CODE,ARY,CDT,SYS,LEN)
Subscribe to ICR 5747

ICD-9 Internal Entry Number from Code

$$CODEN^ICDCODE(CODE,FILE) ICR 3990

```
     Input:

          CODE   ICD code (required)
          FILE   File Number to search for code
                    80 = ICD Dx file
                    80.1 = ICD Oper/Proc file

     Output:

          IEN~global root
            or
          -1~error message
```

Recommended Replacement API:
$$CODEN^ICDEX(CODE,FILE)
Subscribe to ICR 5747

ICD-9 Code from IEN
$$CODEC^ICDCODE(IEN,FILE) ICR 3990

```
Input:
  IEN    IEN of ICD code    REQUIRED
  FILE   File Number to search for code
         80 = ICD Dx file
         80.1 = ICD Oper/Proc file

Output: ICD code, -1 if not found
```

Recommended Replacement API:
$$CODEC^ICDEX(FILE,IEN)
Subscribe to ICR 5747

4.2.2 ICDAPIU, ICR 3991 (scheduled for retirement)

This Integration Control Registration (ICR) shall be retired 18 months after the ICD-10 implementation date established by the HHS. See equivalent API in ICR 5747.

Status of an ICD-9 Code

$$STATCHK^ICDAPIU(CODE,CDT) ICR 3991

```
Input:

    CODE  ICD Code
    CDT   Date to screen against

Output:

    2-Piece String containing Status and IEN
```

Recommended Replacement API:
$$STATCHK^ICDEX(CODE,FILE)
Subscribe to ICR 5747

Next ICD-9 Code in a Sequence

$$NEXT^ICDAPIU(CODE) ICR 3991

```
Input:

    CODE  ICD Code   REQUIRED

Output:

    The Next ICD Code, Null if none
```

Recommended Replacement API:
$$NEXT^ICDEX(CODE,SYS,CDT)
Subscribe to ICR 5747

Previous ICD-9 Code in a Sequence

$$PREV^ICDAPIU(CODE) ICR 3991

```
Input:

    CODE   ICD Code   REQUIRED

Output:

    The Previous ICD Code, Null if none
```

Recommended Replacement API:
$$PREV^ICDEX(CODE,SYS,CDT)
Subscribe to ICR 5747

Activation History of an ICD-9 Code

$$HIST^ICDAPIU(CODE,ARY) ICR 3991

```
Input:

    CODE   ICD Code                       REQUIRED
   .ARY    Array, passed by Reference     REQUIRED

Output:

    Mirrors ARY(0) (or, -1 on error)
    ARY(0) = Number of Activation History Entries
    ARY(<date>) = status    where: 1 is Active
    ARY("IEN") = <ien>
```

Recommended Replacement API:
$$HIST^ICDEX(CODE,.ARY,SYS)
Subscribe to ICR 5747

Date Business Rules for ICD-9

$$DTBR^ICDAPIU(CDT,CS) ICR 3991

```
Input:

    CDT   Code Date to check default TODAY
    CS    Code System (Default 0 = ICD)

Output:

    If CDT < ICD-9 Date and CS=0, use ICD-9 Date
    If CDT < 2890101 and CS=1, use 2890101
    If CDT < 2821001 and CS=2, use 2821001
    If CDT is year only, use first of the year
    If CDT is year and month only, use first of the month
```

Recommended Replacement API:
$$DBTR^ICDEX(CDT,STD,SYS)
Subscribe to ICR 5747

Warning Message – Text may be inaccurate for date

$$MSG^ICDAPIU(CDT,CS) ICR 3991

```
Input:

    CDT   Code Date to check
          FileMan format, Default = TODAY
    CS    Code System
          0:ICD, 1:CPT/HCPCS, 2:DRG, 3:LEX
          Default = 0

Output:

    User Alert
```

Recommended Replacement API: $$MSG^ICDEX(CDT,STD,SYS)
Subscribe to ICR 5747

Activation Periods (active-inactive) for ICD-9 Code

PERIOD^ICDAPIU(CODE,ARY) ICR 3991

```
Input:

    CODE  ICD Code (required)
    ARY   Array, passed by Reference (required)

Output:

    ARY(0) = IEN ^ Selectable ^ Error Message

       Where IEN = -1 if error
       Selectable = 0 for VA Only codes
       Error Message if applicable

    ARY(Activation Date) = Inactivation Date ^ Short Name

       Where the Short Name is versioned as follows:

       Period is active    Short Description for the date
                           the period became active

       Period is inactive  Short Description for the date
                           the period became inactive
```

Recommended Replacement API:
$$PERIOD^ICDEX(CODE,.ARY,SYS)
Subscribe to ICR 5747

4.3 Supported ICD-9/10 APIs (wrapper APIs)

The following APIs are supported for both the ICD-9 and ICD-10 coding systems and will continue to be supported throughout the transition to ICD-10:

4.3.1 ICDXCODE, ICR 5699 (scheduled for retirement)

This Integration Control Registration (ICR) contains interim APIs mandated by the ICD-10 project (formerly referred to as the "ICD wrapper APIs"). All of them call into ICDEX to return data. Applications should replace these APIs with the equivalent APIs in routine ICDEX (ICR 5747) as soon as possible. This ICR shall be retired 36 months after the ICD-10 implementation date established by HHS.

ICD Code Data

$$ICDDATA^ICDXCODE(CSYS,CODE,DATE,FRMT) ICR 5699

```
Input:

  CSYS   Coding system, Required
  CODE   Code/IEN/variable pointer, Required
  DATE   Code Set Date (default = TODAY)
  FRMT   Code format "E" external (default)
                     "I" internal (IEN)

Output:

 Diagnosis returns a 20 piece string delimited by "^"

   1  IEN of code in file 80
   2  ICD-9 Dx Code                (#.01)
   3  Identifier                   (#1.2)
   4  Versioned Dx                 (67 multiple)
   5  Unacceptable as Principal Dx (#1.3)
   6  Major Dx Cat                 (72 multiple)
   7  MDC13                        (#1.4)
   8  Compl/Comorb                 (103 multiple)
   9  ICD Expanded                 (#1.7)
   10 Status                       (66 multiple)
   11 Sex                          (10 multiple)
   12 Inactive Date                (66 multiple)
   13 MDC24                        (#1.5)
   14 MDC25                        (#1.6)
   15 Age Low                      (11 multiple)
   16 Age High                     (12 multiple)
   17 Activation Date              (66 multiple)
   18 Message
   19 Complication/Comorbidity     (103 multiple)
   20 Coding System                (#1.1)

  Procedures returns A 14 piece string delimited by "^"
```

```
 1  IEN of code in file 80.1
 2  ICD procedure code           (#.01)
 3  Identifier                   (#1.2)
 4  MDC24                        (#1.5)
 5  Versioned Oper/Proc          (67 multiple)
 6  <null>
 7  <null>
 8  <null>
 9  ICD Expanded                 (#1.7)
10 Status                       (66 multiple)
11 Use with Sex                 (10 multiple)
12 Inactive Date                (66 multiple)
13 Activation Date              (66 multiple)
14 Message
15 Coding System                (#1.1)

 or

 -1^Error Description
```

ICD Code Description

$$ICDDESC^ICDXCODE(CSYS,CODE,DATE,.ARY) ICR 5699

```
Input:

  CSYS     Coding system
  CODE     ICD Code (required)
  CDT      Date (default = TODAY)
 .ARY      Array Name passed by reference

Output:

  $$ICDDESC  Number of lines in array

         @ARY(1) - Versioned Description (68 multiple)
         @ARY(2) - blank
         @ARY(3) - message: CODE TEXT MAY BE INACCURATE
                   (ICD-9 ONLY)
```

Status of an ICD Code

$$STATCHK^ICDXCODE(CSYS,CODE,DATE) ICR 5699

```
Input:

  CSYS    Coding system
  CODE    Code (IEN not allowed)
  DATE    Date (default = TODAY)

Output:

   2-Piece String containing the code's status
   and the IEN if the code exists, else -1.
   The following are possible outputs:

          1^IEN    Active Code
          0^IEN    Inactive Code
          0^-1     Code not Found
```

Next ICD Code in a Sequence

$$NEXT^ICDXCODE(CSYS,CODE) ICR 5699

```
Input:

    CSYS    Coding system                    Required
    CODE    ICD-10 Code (IEN not allowed)    Required

Output:

   $$NEXT  The Next ICD Code, Null if none
```

Previous ICD Code in a Sequence

$$PREV^ICDXCODE(CSYS,CODE) ICR 5699

```
Input:

    CSYS    Coding system                    Required
    CODE    ICD-10 Code (IEN not allowed)    Required

Output:

   $$PREV  The Previous ICD Code, Null if none
```

Activation History of an ICD Code

$$HIST^ICDXCODE(CSYS,CODE,ARY) ICR 5699

```
Input:

   CSYS     Coding system                      Required
   CODE     ICD Code (IEN not allowed)         Required
  .ARY      Array, passed by Reference         Required

Output:

   $$HIST   Mirrors ARRAY(0) or, -1 on error
   ARY(0)       =  Number of Activation History Entries
   ARY(<date>)  =  Status    where: 1 is Active
   ARY("IEN")   =  <ien>
```

Activation Periods (active-inactive) for ICD-9 Code

$$PERIOD^ICDXCODE(CSYS,CODE,ARY) ICR 5699

```
Input:

   CSYS     Coding system                      Required
   CODE     ICD Code (IEN not allowed)         Required
  .ARY      Array, passed by Reference         Required

Output:

   ARY(0) = IEN ^ Selectable ^ Error Message

       Where IEN = -1 if error
       Selectable = 0 for VA Only codes
       Error Message if applicable

   ARY(Activation Date) = Inactivation Date ^ Short Name

       Where the Short Name is versioned as follows:

       Period is active    Short Description for the date
                           the period became active

       Period is inactive  Short Description for the date
                           the period became inactive

   or -1^0 (no period or error)
```

4.3.2 ICDSAPI, ICR 5757 (scheduled for retirement)

This Integration Control Registration (ICR) contains an interim API mandated by the ICD-10 project (formerly referred to as the "ICD wrapper APIs"). It calls DIC and the ICD Special Lookup ICDEXLK. Applications should replace this API with a call to FileMan (DIC) as soon as possible. This ICR shall be retired 36 months after the ICD-10 implementation date established by HHS.

Search for an ICD Code (DIC)

$$SEARCH^ICDSAPI(FILE,SCR,DI,VDT,FMT) ICR 5757

Input:

FILEID This can be either a file number, a file root, a file identifier, a coding system or a source abbreviation that can be resolved to a file number.

Number	Root	ID	Coding System	Source Abbreviation
80	^ICD9(	DIAG	1 or 30	ICD or 10D
80.1	^ICD0(	PROC	2 or 31	ICP or 10P

SCREEN This is a string of MUMPS code that is executed to screen an entry from selection. It must contain an IF statement to set the value of $T. Those entries that the IF statement sets $T to 0 (false) will not be displayed or selectable.

DISFIL A string of alphabetic characters which alter how the lookup responds. Default value "AEMQZ". DIC(0) will be set to the contents of this parameter.

Parameters applicable to a versioned file

- A Ask the entry; if erroneous, ask again
- B Only the B index is used
- E Echo information
- F Forget the lookup value
- I Ignore the special lookup program
- M Multiple-index lookup allowed
- O Only find one entry if it matches exactly
- S Suppresses display of .01
- T Search until user selects or enters ^^
- X EXact match required
- Z Zero node in Y(0), external form in Y(0,0)

Parameters not applicable to a versioned file and ignored by this lookup

- C Versioned cross-references not turned off

```
                K   Primary Key not established
                L   Learning a new entry LAYGO not allowed
                N   IEN lookup allowed (not forced)
                n   ICD has no pure numeric entries
                Q   Input is pre-processed, ?? not necessary
                U   All values are external
                V   Verification is not optional

   DATE        Versioning Date (Fileman format)

                 If supplied only active codes on that date
                 will be included in the selection list.

                 If not supplied, the date will default to
                 TODAY and all codes may be selected, active
                 and inactive.

                 In both cases the display will be altered
                 based on the date.

   FORMAT     Output Format

                1  Fileman, Code and Short Text (default)

                   250.00    DMII WO CMP NT ST UNCNTR

                2  Fileman, Code and Description

                   250.00    DIABETES MELLITUS WITHOUT
                             MENTION OF COMPLICATION, TYPE
                             II OR UNSPECIFIED TYPE, NOT
                             STATED AS UNCONTROLLED

                3  Lexicon, Short Text and Code

                   DMII WO CMP NT ST UNCNTR (250.00)

                4  Lexicon, Description and Code

                   DIABETES MELLITUS WITHOUT MENTION OF
                   COMPLICATION, TYPE II OR UNSPECIFIED TYPE,
                   NOT STATED AS UNCONTROLLED (250.00)

Output:

   $$SEARCH   This is the value of Y (below)

   Y          IEN ^ Code           Fileman

              or

              -1 if not found

   If DISFIL/DIC(0) containing the character "Z"

     Y(0)     0 Node               Fileman
     Y(0,0)   Code                 Fileman
```

```
Y(0,1)   $$ICDDX or $$ICDOP   Non-Fileman
Y(0,2)   Long Description     Non-Fileman
```

4.4 Data Extraction APIs by Subscription

4.4.1 ICDEX, ICR 5747

The following APIs were developed to replace all direct global reads to ICD files 80 and 80.1. To track which applications are extracting data, these APIs are available by subscription only. If there are future changes to the data dictionaries or APIs, the ICD package developers can quickly contact the affected applications to coordinate the changes.

4.4.2 ICD Code APIs (formerly ICDCODE)

ICD Diagnosis Code Data

$$ICDDX^ICDEX(CODE,CDT,SYS,FMT,LOC) ICR 5747

```
Input:

  CODE  Code/IEN (required)
   CDT  Date (default = TODAY)
   SYS  Coding System (taken from file 80.4)
          1 = ICD-9 Diagnosis
         30 = ICD-10 Diagnosis
   FMT  Format
          E = External (default)
          I = Internal Entry Number
   LOC  Use Local codes
          1 = Yes
                0 = No (default)

Output:

 Returns a 20 piece string delimited by "^"

   1  IEN of code in ^ICD9(
   2  ICD-9 Dx Code                 (#.01)
   3  Identifier String ID;ID;ID    File 82
   4  Versioned Dx                  (67 multiple)
   5  Unacceptable as Principal Dx  (#1.3)
   6  Major Dx Cat                  (72 multiple)
   7  MDC13                         (#1.4)
   8  Compl/Comorb                  (103 multiple)
   9  ICD Expanded                  (#1.7)
   10 Status                        (66 multiple)
   11 Sex                           (10 multiple)
   12 Inactive Date                 (66 multiple)
   13 MDC24                         (#1.5)
   14 MDC25                         (#1.6)
   15 Age Low                       (11 multiple)
   16 Age High                      (12 multiple)
```

```
17 Activation Date              (66 multiple)
18 Message
19 Complication/Comorbidity     (103 multiple)
20 Coding System                (#1.1)
21 Primary CC Flag              (103 multiple)
22 PDX Exclusion Code           (#1.11)

or

-1^Error Description
```

ICD Procedure Code Data

$$ICDOP^ICDEX(CODE,CDT,SYS,FMT) ICR 5747

```
Input:

  CODE  Code/IEN (required)
   CDT  Date (default = TODAY)
   SYS  Coding System (taken from file 757.03)
          2 = ICD-9 Procedure
         31 = ICD-10 Procedure
   FMT  Format
          E = External (default)
          I = Internal Entry Number
   LOC  Use Local codes
          1 = Yes
                0 = No (default)


Output:

 Returns a 14 piece string delimited by "^"

   1  IEN of code in ^ICD0(
   2  ICD procedure code           (#.01)
   3  Identifier                   (#1.2)
   4  MDC24                        (#1.5)
   5  Versioned Oper/Proc          (67 multiple)
   6  <null>
   7  <null>
   8  <null>
   9  ICD Expanded                 (#1.7)
   10 Status                       (66 multiple)
   11 Use with Sex                 (10 multiple)
   12 Inactive Date                (66 multiple)
   13 Activation Date              (66 multiple)
   14 Message
   15 Coding System                (#1.1)

   or

   -1^Error Description
```

ICD Code Description

$$ICDD^ICDEX(CODE,ARY,CDT,SYS,LEN) ICR 5747

```
Input:

  CODE  Code, external format (required)
  ARY   Array Name passed by reference (required)
  CDT   Date (optional, default = TODAY)
  SYS   Coding System (optional)
  LEN   Sting Length (optional, > 27, default 245)

Output:

  #   Number of lines in array

  ARY(1) - Versioned Description (68 multiple)

  If there is a warning message (ICD-9 only):

    ARY(n+1) - blank
    ARY(n+2) - warning message: CODE TEXT MAY BE INACCURATE

  Or  -1^Error Description
```

Internal Entry Number (IEN) from Code

$$CODEN^ICDEX(CODE,FILE) ICR 5747

```
Input:

  CODE  ICD code (required)
  FILE  File Number to search for code
           80 = ICD Dx file
           80.1 = ICD Oper/Proc file
Output:

  IEN~Global Root    or    -1~error message
```

ICD Code from Internal Entry Number (IEN)

$$CODEC^ICDEX(FILE,IEN) ICR 5747

```
Input:

   IEN     Internal Entry Number (required)
   FILE    File Number 80 or 80.1 (required)

Output:

   $$CODE  An ICD Diagnosis or Procedure code

          or -1 ^ message on error

 Retire IA 280, 365, 582, 5388, 5404
```

Code IEN from Code (BA cross-reference)

$$CODEBA^ICDEX(CODE,ROOT) ICR 5747

```
Input:

  CODE  ICD Code, either ICD-9 or ICD-10 (required)
  ROOT  File Root or Number (required)
           ^ICD9( or 80
           ^ICD0( or 80.1

Output:

  IEN   IEN for CODE in ROOT or -1 if not found
```

Code IEN from Code and Coding System (ABA cross-reference)

$$CODEABA^ICDEX(CODE,ROOT,SYS) ICR 5747

```
Input:

  CODE  ICD Code, either ICD-9 or ICD-10 (required)
  ROOT  File Root or Number (required)
           ^ICD9( or 80
           ^ICD0( or 80.1
  SYS   File Root or Number (required)
          1 = ICD-9 Diagnosis
          2 = ICD-9 Procedure
         30 = ICD-10 Diagnosis
         31 = ICD-10 Procedure

Output:

  IEN   IEN for CODE in ROOT for SYS or -1 if not found
```

File for Code

$$CODEFI^ICDEX(CODE) ICR 5747

```
Input:

  CODE  ICD code (required)

Output:

  FILE  File Number
          80   = ICD Dx file
          80.1 = ICD Oper/Proc file
          Null
```

Coding System for Code and File

$$CODECS^ICDEX(CODE,FILE,CDT) ICR 5747

```
Input:

  CODE  ICD code/IEN (required)
  FILE  File Number (required)
           80   = ICD Dx file
           80.1 = ICD Oper/Proc file
  CDT   Date used to determine Coding
        System (optional, default TODAY)

Output:

  SYS   2 piece "^" delimited string
           1   Coding System
           2   Coding Nomenclature

                1  ^  ICD-9-CM
                2  ^  ICD-9 Proc
               30  ^  ICD-10-CM
               31  ^  ICD-10-PCS

        or null if not found
```

Coding System for IEN and File

$$CSI^ICDEX(FILE,IEN) ICR 5747

```
Input:

  FILE   File Number (required)
  IEN    IEN in file 80 (required)

Output:

  $$CSI  Coding System

           1  ^  ICD-9-CM
           2  ^  ICD-9 Proc
          30  ^  ICD-10-CM
          31  ^  ICD-10-PCS

        or null if not found
```

Versioned Major Diagnostic Category

$$VMDC^ICDEX(IEN,CDT,FMT) ICR 5747

```
Input:

  IEN   IEN in file 80 (required)
  CDT   Date to use to Extract MDC (default TODAY)
  FMT   Output Format
          0 = MDC only (default)
          1 = MDC ^ Effective Date

Output:

  MDC   Major Diagnostic Category
```

Versioned Age Low

$$VAGEL^ICDEX(IEN,CDT,FMT) ICR 5747

```
Input:

  IEN   IEN in file 80 (required)
  CDT   Date to use to Extract Age Low (default TODAY)
  FMT   Output Format
          0 = Age Low only (default)
          1 = Age Low ^ Effective Date

Output:

  AGEL  Age Low
```

Versioned Age High

$$VAGEH^ICDEX(IEN,CDT,FMT) ICR 5747

```
Input:

  IEN   IEN in file 80 (required)
  CDT   Date to use to Extract Age High (default TODAY)
  FMT   Output Format
          0 = Age High only (default)
          1 = Age High ^ Effective Date

Output:

  AGEH  Age High
```

Versioned Complication/Comorbidity

$$VCC^ICDEX(IEN,CDT,FMT) ICR 5747

```
Input:

  IEN   IEN in file 80 (required)
  CDT   Date to use to Extract CC (default TODAY)
  FMT   Output Format
          0 = CC only (default)
          1 = CC ^ Effective Date

Output:

  $$VCC  Complication/Comorbidity (FMT=0)
         Complication/Comorbidity^Effective Date (FMT=1)
```

Versioned Complication/Comorbidity Primary Flag

$$VCCP^ICDEX(IEN,CDT,FMT) ICR 5747

```
Input:

  IEN     IEN in file 80 (required)
  CDT     Date to use to Extract CC Primary Flag (default TODAY)
  FMT     Output Format
            0 = CC Primary Flag only (default)
            1 = CC Primary Flag ^ Effective Date ^ External Value

Output:

  $$VCCP  Complication/Comorbidity (FMT=0)
          Complication/Comorbidity ^ Effective Date ^ Value (FMT=1)
```

Versioned Sex

$$VSEX^ICDEX(FILE,IEN,CDT,FMT) ICR 5747

```
Input:

  FILE  File
          80    ICD Diagnosis file
          80.1  ICD Operation/Procedure file
  IEN   IEN (required)
  CDT   Date to use to Extract Sex (default TODAY)
  FMT   Output Format
          0 = Sex only (default)
          1 = Sex ^ Effective Date

Output:

  SEX   Sex
          M   Male
          F   Female
          Null
```

Status/Activation Date/Inactivation Date

$$SAI^ICDEX(FILE,IEN,CDT) ICR 5747

```
Input:

  FILE  File
          80    ICD Diagnosis file
          80.1  ICD Operation/Procedure file
  IEN   IEN or code (required)
  CDT   Date to use to Extract Status (default TODAY)

Output:

  5 piece "^" delimited string

     1  Status
     2  Inactivation Date
     3  Activation Date
     4  IEN
     5  Short Text in use on Activation Date (piece 3)
```

Versioned Short Text

$$VST^ICDEX(FILE,IEN,CDT) ICR 5747

```
Input:

  FILE  Global Root/File #/Coding System/SAB
  IEN   IEN  (required)
  CDT   Date to use to Extract Text (default TODAY)

Output:

  VST   Short Text from either file 80 or 80.1
```

Versioned Long Text

$$VLT^ICDEX(FILE,IEN,CDT) ICR 5747

```
Input:

  FILE  Global Root/File #/Coding System/SAB
  IEN   IEN  (required)
  CDT   Date to use to Extract Text (default TODAY)

Output:

  VLT   Long Text (description) from either file 80 or 80.1
```

Versioned Short Text Diagnosis

$$VSTD^ICDEX(IEN,CDT) ICR 5747

```
Input:

  IEN   IEN  (required)
  CDT   Date to use to Extract Text (default TODAY)

Output:

  VST   Short Text from file 80
```

Versioned Short Text Procedures

$$VSTP^ICDEX(IEN,CDT) ICR 5747

```
Input:

  IEN   IEN  (required)
  CDT   Date to use to Extract Text (default TODAY)

Output:

  VST   Short Text from file 80.1
```

Versioned Long Text Diagnosis

$$VLTD^ICDEX(IEN,CDT) ICR 5747

```
Input:

  IEN   IEN  (required)
  CDT   Date to use to Extract Text (default TODAY)

Output:

  VLT   Long Text from file 80
```

Versioned Long Text Procedures

$$VLTP^ICDEX(IEN,CDT) ICR 5747

```
Input:

  IEN   IEN  (required)
  CDT   Date to use to Extract Text (default TODAY)

Output:

  VLT   Long Text from file 80.1
```

Short Description (Formatted)

$$SD^ICDEX(FILE,IEN,CDT,ARY,LEN) ICR 5747

```
Input:

  IEN   Internal Entry Number (Required)
  FILE  File Number (Required)
  CDT   Date, Default TODAY (Optional)
 .ARY   Array Passed by Reference (Optional)
  LEN   Text Length (15-79, default 60) (Optional)

Output:

  $$SD  Short Description OR -1 ^ Error Message
  ARY   Description in segment lengths specified
```

Long Description (Formatted)

$$LD^ICDEX(FILE,IEN,CDT,ARY,LEN) ICR 5747

```
Input:

  IEN   Internal Entry Number (Required)
  FILE  File Number (Required)
  CDT   Date, Default TODAY (Optional)
 .ARY   Array Passed by Reference (Optional)
  LEN   Text Length (15-79, default 245) (Optional)

Output:

  $$LD  Long Description OR -1 ^ Error Message
  ARY   Description in lengths specified
```

Short Description History

$$SDH^ICDEX(FILE,IEN,ARY) ICR 5747

```
Input:

  IEN    Internal Entry Number (Required)
  FILE   File Number (Required)
 .ARY    Array Passed by Reference (Optional)

Output:

  $$SDH  This is a three piece "^" delimited
         string containing:

           1   The number of short descriptions found
           2   The earliest date found
           3   The latest date found

         OR -1 ^ Error Message

  ARY    This is a local array containing a history
```

```
              of short descriptions by date:

                ARY(0)= # ^ Earliest Date ^ Latest Date
                ARY(DATE)=Short Description
```

Long Description History

$$LDH^ICDEX(FILE,IEN,ARY) ICR 5747

```
          Input:

            IEN    Internal Entry Number (Required)
            FILE   File Number (Required)
           .ARY    Array Passed by Reference (Optional)

          Output:

            $$LDH  This is a three piece "^" delimited
                   string containing:

                     1   The number of long descriptions found
                     2   The earliest date found
                     3   The latest date found

                   OR -1 ^ Error Message

            ARY    This is a local array containing a history
                   of long descriptions by date:

                     ARY(0)= # ^ Earliest Date ^ Latest Date
                     ARY(DATE)=Long Description
```

4.4.3 ICD API Utilities (formerly ICDAPIU)

Status of an ICD Code

$$STATCHK^ICDEX(CODE,CDT,SYS) ICR 5747

```
          Input:

             CODE   ICD Code  REQUIRED
             CDT    Date to screen against (default = TODAY)
             SYS    Numeric Coding System (optional, however, if
                    specified it must be correct)

          Output:

             2-Piece String containing the code's status
             and the IEN if the code exists, else -1.
             The following are possible outputs:
                  1^IEN            Active Code
                  0^IEN            Inactive Code
                  0^-1^Message     Code not Found or Error

          This API requires the ACT Cross-Reference
```

```
^ICD9("ACT",<code>,<status>,<date>,<ien>)
^ICD0("ACT",<code>,<status>,<date>,<ien>)
```

Date Business Rules (ICD-9/ICD-10)

$$DTBR^ICDEX(CDT,STD,SYS) ICR 5747

```
Input:

  CDT   Code Date to check (FileMan format, default=Today)
  STD   Standard

          0 = ICD (Default)
          1 = CPT/HCPCS
          2 = DRG

  SYS   Coding System

          1 = ICD-9-CM
          2 = ICD-9-PCS
         30 = ICD-10-CM
         31 = ICD-10-PCS

Output:

  If CDT < ICD-9 Date and STD=0, use ICD-9 Date
  If CDT < ICD-10 Date and STD=0 and SYS=30, use ICD-10 Date
  If CDT < ICD-10 Date and STD=0 and SYS=31, use ICD-10 Date
  If CDT < 2890101 and STD=1, use 2890101
  If CDT < 2821001 and STD=2, use 2821001
  If CDT is year only, use first of the year
  If CDT is year and month only, use first of the month
```

Implementation Date

$$IMP^ICDEX(SYS,CDT) ICR 5747

```
Input:

  SYS   Coding System

          1 = ICD-9-CM
          2 = ICD-9-PCS
         30 = ICD-10-CM
         31 = ICD-10-PCS

Output:

  $$IMP  Date the Coding System was Implemented
```

Warning Message – Text may be inaccurate for date

$$MSG^ICDEX(CDT,STD,SYS) ICR 5747

```
Input:
```

```
CDT   Code Date to check (FileMan format, Default = today)
STD   Code System

         0   ICD (default)
         1   CPT/HCPCS
         2   DRG
         3   LEX

SYS   Coding System

        1 = ICD-9-CM
        2 = ICD-9-PCS
       30 = ICD-10-CM
       31 = ICD-10-PCS

Output:

  User Alert Message
```

Code is Selectable

$$SEL^ICDEX(FILE,IEN) ICR 5747

```
Input:

   FILE   File number 80 or 80.1 (required)
   IEN    Internal Entry Number (required)

Output:

   $$SEL  Boolean value

            1  Selectable
            0  Not Selectable

           -1  on error
```

Next Code in a Sequence

$$NEXT^ICDEX(CODE,SYS,CDT) ICR 5747

```
Input:

   CODE   ICD Code or Null for the first code
   SYS    Coding System - see ^ICDS

            1 = ICD-9-CM
            2 = ICD-9-PCS
           30 = ICD-10-CM
           31 = ICD-10-PCS

  CDT   Code Date to check
        If CDT is passed, then the code
        returned is the next active code
        based on date.  If it is not
```

```
            passed then the next code is
            returned regardless of status.

Output:

   The Next ICD Code, Null if none
```

Previous Code in a Sequence

$$PREV^ICDEX(CODE,SYS,CDT) ICR 5747

```
Input:

   CODE   ICD Code or Null for the last code
   SYS    Coding System - see ^ICDS

             1 = ICD-9-CM
             2 = ICD-9-PCS
            30 = ICD-10-CM
            31 = ICD-10-PCS

  CDT   Code Date to check
        If CDT is passed, then the code
        returned is the previous active
        code based on date.  If it is
        not passed then the previous
        code is returned regardless of
        status.

Output:

   The Previous ICD Code, Null if none
```

Activation History of an ICD Code

$$HIST^ICDEX(CODE,ARY,SYS) ICR 5747

```
Input:

   CODE   ICD Code                    (required)
   .ARY   Array, passed by Reference (required)
   SYS    Coding System - see ^ICDS

             1 = ICD-9-CM
             2 = ICD-9-PCS
            30 = ICD-10-CM
            31 = ICD-10-PCS

Output:    Mirrors ARY(0) (or, -1 on error)

   ARY(0) = Number of Activation History Entries
   ARY(<date>) = status    where: 1 is Active
   ARY("IEN") = <ien>
```

Activation Periods (active-inactive) for ICD-9 Code

$$PERIOD^ICDEX(CODE,ARY,SYS) ICR 5747

```
Input:

   CODE   ICD Code (required)
   ARY    Array, passed by Reference (required)
   SYS    Coding System - see ^ICDS

             1 = ICD-9-CM
             2 = ICD-9-PCS
            30 = ICD-10-CM
            31 = ICD-10-PCS

Output:

  $$PERIOD  Number of activation periods found

  ARY(0) = IEN ^ Selectable ^ Error Message

       Where IEN = -1 if error
       Selectable = 0 for VA Only codes
       Error Message if applicable

  ARY(Activation Date) = Inactivation Date ^ Short Name

       Where the Short Name is versioned as follows:

       Period is active    Short Description for the date
                           the period became active

       Period is inactive  Short Description for the date
                     the period became inactive
```

$ORDER BA or ABA Cross-Reference

$$OBA^ICDEX(FILE,CODE,SYS,REV) ICR 5747

```
Input:

   CODE = ICD Code, can be null
   FILE   File Number 80 or 80.1
   SYS    Coding System (internal) from file 80.4
   REV    Reverse $Order if set to 1

Output:

   $$OBA  Next or Previous Code

 This API replaces the need to access the BA cross-reference
 in a FOR loop.

    $$OBA(<file>,<code>,<system>) replaces:
```

```
   $O(^ICD9("BA",(<code>_" ")) and
   $O(^ICD0("BA",(<code>_" "))

 F  S CODE=$$OBA^ICDEX(80,CODE,1) Q:'$L(CODE)  D
 F  S CODE=$$OBA^ICDEX(80,CODE,30) Q:'$L(CODE)  D
 F  S CODE=$$OBA^ICDEX(80.1,CODE,2) Q:'$L(CODE)  D
 F  S CODE=$$OBA^ICDEX(80.1,CODE,31) Q:'$L(CODE)  D

Retire IA 5388, 5404
```

$ORDER D or AD Cross-Reference

$$OD^ICDEX(FILE,WORD,SYS,REV) ICR 5747

```
Input:

   FILE   File Number 80 or 80.1
   WORD   Word, can be null or a 2 piece string
          containing Word and IEN where the word
          is stored
   SYS    Coding System (internal)
            Acceptable values can be found on the ASYS
            cross-reference.  At the time of this
            writing, it includes:

            File 80
               1   ICD-9-CM
              30   ICD-10-CM

            File 80.1
               2   ICD-9 Proc
              31   ICD-10-PCS

   REV    Reverse $Order if set to 1

Output:

   2 Piece "^" delimited string

        1   WORD   Next or Previous word in D Index
        2   IEN    Internal Entry Number where WORD is found

Retire IA 5388, 5404
```

Date Last Modified

$$DLM^ICDEX(FILE,IEN,FIELD,CDT) ICR 5747

```
Input


   FILE     File Number 80 or 80.1          (required)
   IEN      Internal Entry Number           (required)
   FIELD    Field Number of Versioned Data  (optional)

              File 80
```

```
                    10   Sex                          5;0
                    11   Age Low                      6;0
                    12   Age High                     7;0
                    66   Status                      66;0
                    67   Diagnosis                   67;0
                    68   Description                 68;0
                    71   DRG Grouper                  3;0
                    72   Major Diagnostic Category    4;0
                   103   Complication/Comorbidity    69;0

                 File 80.1

                    10   Sex                          3;0
                    66   Status                      66;0
                    67   Operation/Procedure         67;0
                    68   Description                 68;0
                    71   DRG Grouper                  2;0

             If the field is passed, then the date last
             modified (based on date) for the field is
             returned.  If the field is not passed, then
             the date last modified (based on date) for
             the record at IEN is returned.

   CDT     Date to base output on (default is today)
           Business rules apply

Output:

   $$DLM   Date Last Modified

    or -1 ^ message on error
```

Select Coding System (Interactive)

$$CS^ICDEX(FILE,FMT) ICR 5747

```
Input

   FILE   File Number 80 or 80.1 (optional)
          If not provided, you will be prompted
          for the ICD File, there is no default
          value.

   FMT    Format

            E  Display External only (default)
            I  Display Internal with External

Output

   $$CS   2 piece "^" delimited string

             1  Coding System (internal)
```

```
                2  Coding System (external)

              or -1 on error or non-selection
                 ^^ double up-arrows
                  ^ timeout or single up-arrow
```

4.4.4 ICD Support (formerly ICDSUPT)

Effective Date and Status

$$EFF^ICDEX(FILE,IEN,CDT) ICR 5747

```
Input:

   FILE   File number 80/80.1 (required)
   IEN    ICD IEN (required)
   EDT    Date to check (FileMan format) (required)

Output:

  A 3 piece "^" delimited string

        1   Status
              1 - Active
              0 - Inactive
        2   Inactivation Date
        3   Activation Date
    -or-
        -1^error message
```

Initial Activation Date

$$IA^ICDEX(FILE,IEN) ICR 5747

```
Input:

  FILE   Global Root/File Number (Required)
  IEN    Internal Entry Number (Required)

Output:

  $$IA   Initial Activation Date OR -1 ^ Error Message
```

Last Activation Date

$$LA^ICDEX(FILE,IEN,CDT) ICR 5747

```
Input:

  IEN    Internal Entry Number (Required)
  FILE   Global Root/File Number (Required)
  CDT    Date (default = TODAY) (Optional)
```

```
Output:

  $$LA   Last Current Activation Date OR -1 ^ Error Message
```

Last Inactivation Date

$$LI^ICDEX(FILE,IEN,CDT) ICR 5747

```
Input:

  IEN    Internal Entry Number (Required)
  FILE   Global Root/File Number (Required)
  CDT    Date (default = TODAY) (Optional)

Output:

  $$LI   Last Current Inactivation Date OR -1 ^ Error Message
```

Last Status

$$LS^ICDEX(FILE,IEN,CDT) ICR 5747

```
Input:

  IEN    Internal Entry Number (Required)
  FILE   Global Root/File Number (Required)
  CDT    Date (default = TODAY) (Optional)

Output:

  $$LS   Last Status (1/0) OR -1 ^ Error Message
```

Convert Code to a Numeric Value

$$NUM^ICDEX(CODE) ICR 5747

```
Input:

   CODE   ICD CODE (required)

Output:

   NUM    Numerical representation of CODE

          or

          -1 on error
```

Convert Numeric Value to a Code

$$COD^ICDEX(NUM) ICR 5747

```
Input:
```

```
   NUM    Numerical representation of an ICD Code (required)

Output:

   CODE   ICD Code

          or

          null on error
```

Internal or External Format

$$IE^ICDEX(CODE) ICR 5747

```
Input:

  CODE   ICD code or IEN

Output:

  $$IE  Set of Codes

          I  X is in an internal format (IEN)
          E  X is in an external format (Code)

          Null on error
```

Resolve File Number

$$FILE^ICDEX(SYS) ICR 5747

```
Input:

  X     File/Identifier/Coding System/Code (required)

Output:

  FILE  File Number or -1 on error
```

Resolve Global Root

$$ROOT^ICDEX(SYS) ICR 5747

```
Input:

  X     File Number, File Name, Root, Identifier
        or Coding System (required)

Output:

  ROOT  Global Root for File or null
```

Diagnosis/Procedure file Header Node

$$HDR(FILE) ICR 5747

```
Input:

   X      File Number or Global Root
             80    or  ^ICD9(
             80.1  or  ^ICD0(

Output:

  $$HDR   Diagnosis/Procedure File Header Node

Replaces ICR 2435 and 2436
```

Resolved Coding System Version (uses file 80.4)

$$VER(SYS,REL) ICR 5747

```
Input:

   SYS   Pointer to the coding system file 80.4
   REL   Indicates the relationship of the output coding
         system to the input coding system (Optional)

            0   N/A - Return the current version (default)
            1   Return the next version
           -1   Return the previous version

   Output

   $$VER  This is a 5 piece string containing:

            1   Coding System (pointer to file 80.4)
            2   Coding System Nomenclature
            3   Coding System Abbreviation
            4   File Number containing the Coding System
            5   Date Coding System was Implemented

            or

            -1   on error
```

Resolved Coding System (uses file 80.4)

$$SYS^ICDEX(SYS,CDT,FMT) ICR 5747

```
Input:

  SYS     System/Source Abbreviation/System Identifier/Code
  CDT     Date (optional)
  FMT     Output Format (optional)
```

```
            I  Internal (default)
            E  External
            B  Both Internal ^ External

Output:

  $$SYS System (numeric or alpha)

            Internal  External
               1      ICD-9-CM
               2      ICD-9 Proc
              30      ICD-10-CM
              31      ICD-10-PCS

         or
            -1   on error
```

Coding System Information (uses file 80.4)

$$SINFO^ICDEX(SYS,CDT) ICR 5747

```
Input:

  SYS      System/Source Abbreviation/System Identifier/Code
  CDT      Date (optional)

Output:

  $$SINFO System Info (numeric or alpha)

            Internal  External
               1      IEN to file 80.4
               2      Coding System
               3      Coding System Nomenclature
               4      Coding system Abbreviation
               5      File where the Coding System is stored
               6      Implementation Date

         or
            -1   on error
```

Coding System Name

$$SNAM^ICDEX(SYS) ICR 5747

```
Input:

  SYS    Numeric System Identifier (field 1.1)

Output:
```

```
$$SYS  Character System Name

       or  -1   on error
```

Source Abbreviation

$$SAB^ICDEX(SYS,CDT) ICR 5747

```
Input:

  X     Source Abbreviation or Identifier
  Y     Date used to determine SAB

Output:

  $$SAB 3 Character System Identifier
```

Exclude from Lookup

$$EXC^ICDEX(FILE,IEN) ICR 5747

```
Input:

  FILE   File number 80 or 80.1
  IEN    Internal Entry Number

Output:

  $$EXC  Boolean value 1 = Yes  0 = No
```

4.4.5 DRG Grouper Support

Is Code 1 a Condition of Code 2

$$ISA^ICDEX(IEN1,IEN2,FIELD) ICR 5747

```
Input:

  IEN1   This is the internal entry number (IEN) of a
         code in file 80 that has a relationship with
         the code at IEN2  IEN1 is equivalent to
         Fileman's DA and identifies a code stored in
         a multiple in field 20, 30, 40 or pointed to
         by field 1.11.

  IEN2   This is the internal entry number (IEN) of a
         code in file 80 that may have other codes (IEN1)
         associated with it.  IEN2 is equivalent to
         Fileman's DA(1) and identifies the code in
         the .01 field.

  FIELD  This is a field number in file 80 that contains
```

```
                one or more ICD codes that have a relationship to
                the main entry.   Acceptable field numbers and
                the type of relationships to check include:

                 Field       Relationship
                 20          Code 1 Not Used With Code 2
                 30          Code 1 Required With Code 2
                 40 or 1.11  Code 1 Not Considered CC With Code 2

          Output:

            $$ISA  Boolean value

                   1   Yes/The relationship is True
                   0   No/The relationship is False

             Field        Answers the Question
             -----        -------------------------------------------
             20           Code 1 (identified by IEN1) is not used with
                          Code 2 (identified by IEN2)

             30           Code 1 (identified by IEN1) is required with
                          Code 2 (identified by IEN2)

             40 or 1.11   Code 1 (identified by IEN1) is not considered
                          Complication/Comorbidity (CC) with Code 2
                                (identified by IEN2)
```

Is an ICD Code Valid

$$ISVALID^ICDEX(FILE,IEN,CDT) ICR 5747

```
    Input:

       FILE        File or global root
       IEN         Internal Entry Number
       CDT         Effective date to use (default TODAY)

    Output:

       $$ISVALID   This is a Boolean value

                     1 if the code is valid
                     0 if the code is not valid
```

Does a Condition Exist

$$EXIST^ICDEX(IEN,FIELD) ICR 5747

```
          Input:

            IEN    Internal Entry to file 80
            FIELD  Type of condition to check
```

```
          20   Code Not Used With
          30   Code Required With
          40   Code Not Considered CC With

Output:

  $$EXIST  Boolean value

         1   Yes/True
         0   No/False

   Field   Answers the Question
   -----   ------------------------------------------------
   20      Are there any codes required with this code (IEN)
   30      Are there any codes that should not be used
           with this code (IEN)
   40      Are there any codes that are not considered
           Complication/Comorbidity (CC) with this code
           (IEN)
```

DRGs for a Fiscal Year

$$GETDRG^ICDEX(FILE,IEN,CDT,MDC) ICR 5747

```
Input

   FILE   ICD file number used to retrieve
          the DRGs (Required):

            80   = ICD Diagnosis file
            80.1 = ICD Operation/Procedure file

   IEN    Internal Entry Number (IEN) in the
          file specified (Required)

   CDT    This is the Code Set Versioning date
          (Fileman format) used to identify the
          DRGs that were appropriate on that
          date (optional, default TODAY)

   MDC    Major Diagnostic Category (pointer to
          file 80.1) used as a screen to limit
          the DRGs to a MDC.  This input parameter
          only applies to the OPERATIONS/PROCEDURE
          file 80.1 which has multiple MDCs, each
          with a possibility of multiple DRGs.

Output

   3 piece semi-colon delimited string

       1  DRGs delimited by ^
       2  Fiscal Year
       3  Status flag
```

```
                0 inactive
                1 active

          Example output:

             907^908^909^;3071001;1

          On Error:

             -1;No DRG level;0
```

MDC DRGs

MD^ICDEX(FILE,IEN,CDT,ARY,FLAG) ICR 5747

```
     Input

        FILE   File Number/Identifier
        IEN    Internal entry in file
        CDT    Code Set Versioning Date
       .ARY    Array name passed by reference
        FLAG   Flag   I=Internal (default)
                      E=External

     Output

       ICD Procedures file 80.1 (multiple MDC)

          ARY(<fiscal year>,<MDC>)=DRG^;FY;STA
          ARY(<fiscal year>,<MDC>)="DRG^DRG^;FY;STA

          If Flag contains "E"
          ARY(<fiscal year>,"E",<MDC>)=MDC Name
          ARY(<fiscal year>,"E",<MDC>,<DRG>)=DRG Name
          ARY(<fiscal year>,"E",<MDC>)=MDC Name
          ARY(<fiscal year>,"E",<MDC>,<DRG>)=DRG Name
          ARY(<fiscal year>,"E",<MDC>,<DRG>)=DRG Name
          ARY(<fiscal year>,"E","FY")=External FY

       ICD Diagnosis file 80 (single MDC)

          ARY(<fiscal year>,<MDC>)="DRG^DRG^;FY;STA

          If Flag contains "E"
          ARY(<fiscal year>,"E",<MDC>)=MDC Name
          ARY(<fiscal year>,"E",<MDC>,<DRG>)=DRG Name
          ARY(<fiscal year>,"E",<MDC>,<DRG>)=DRG Name
          ARY(<fiscal year>,"E","FY")=External FY

       NOTE:  If no Fiscal Year found for the input
       date then the first (earliest) Fiscal Year is
       used.
```

Convert External Date to FM

$$EFM^ICDEX(EDT) ICR 5747

```
Input:

  X      External Date

Output:

  $$EFM  Internal Fileman Date

Replaces unsupported $$DGY2K^DGPTOD0(X)
```

FY 4 Digit Year from Fileman Date

$$FY^ICDEX(CDT) ICR 5747

```
Input:

  X      Internal Fileman Date

Output:

  $$FY   FY Year YYYY

Replaces unsupported $$FY^DGPTOD0(X)
```

Versioned MDC for Diagnosis

$$VMDCDX^ICDEX(IEN,CDT) ICR 5747

```
Input

   IEN    Internal Entry Number file 80
   CDT    Code Set Versioning Date

Output

  $$VMDCDX  Versioned MDC
```

Versioned MDC for Operations/Procedure

$$VMDCOP^ICDEX(IEN,MDC,CDT) ICR 5747

```
Input

   IEN    Internal Entry Number file 80
   MDC    Major Diagnostic Category
   CDT    Code Set Versioning Date

Output

  $$VMDCOP  4 piece "^" delimited string
```

```
1   Fiscal Year  Fileman format
2   MDC          Pointer to file 80.3
3   Fiscal Year  pointer to sub-file 80.171
                 (formerly known as DADRGFY)
4   MDC          pointer to sub-file 80.1711
                 (formerly known as DAMDC)
```

Set up an Array of MDCs

MDCG^ICDEX(IEN,CDT,ARY) ICR 5747

```
Input

   IEN    ICD Diagnosis (IEN)
   CDT    Code Set Versioning Date
  .ARY    Array name passed by reference

Output

   ARY    Array listing MDCs for all DRGs

     ARY=MDC
     ARY(MDC)=""
```

Multiple MDC for Operation/Procedure Code

$$MDCT^ICDEX(IEN,CDT,ARY,FMT) ICR 5747

```
Input:

   IEN  Internal Entry Number for file 80.1
   CDT  Code Set Versioning Date
  .ARY  Array of MDCs passed by reference (required)
   FMT  Output Format (optional)

          0   Boolean value only (default)
          1   2 piece "^" delimited string
                 1   Boolean value
                 2   String of matching MDCs delimited by ";"
Output:

 $$MDCT  Boolean value

          0  The ICD Procedure code identified by IEN
             does not include any of the MDCs passed
             in .ARY(MDC) on the date specified (CDT)

          1  The ICD Procedure code identified by IEN
             includes one or more of the MDCs passed
             in .ARY(MDC) on the date specified (CDT)
```

Check for Default MDC

$$MDCD^ICDEX(IEN,MDC) ICR 5747

```
Input:

   IEN  Internal Entry Number for file 80.1
   MDC  Major Diagnostic Category
   CDT  Code Set Versioning Date (optional)
        If not passed, the first FY is used

Output:

  $$MDCD  Boolean value

          0  MDC Does not exist
          1  MDC Exist
```

Major Diagnostic Category Name

$$MDCN^ICDEX(IEN) ICR 5747

```
Input:

   IEN    Internal Entry Number for file 80.3

Output:

  $$MDCN  Major Diagnostic Category Name

Replaces ICR 1586
```

Major O.R. Procedure

$$MOR^ICDEX(IEN) ICR 5747

```
Input:

   IEN   Internal Entry Number for file 80.1

Output:

  $$MOR  Major O.R. Procedure
```

Unacceptable as Principle Diagnosis

$$UPDX^ICDEX(IEN) ICR 5747

```
Input:

   IEN   Internal Entry Number for file 80

Output:

  $$PDX  Boolean value only (default)

           0 No, Code is Acceptable as Principle DX
           1 Yes, Code is Unacceptable as Principle DX
```

Code NOT Used With

$$NOT^ICDEX(IEN,SUB,FMT) ICR 5747

```
Input:

  IEN     Internal Entry Number in file 80
  SUB     TMP global array subscript name.
          If not provided, the subscript
          "ICDNOT" will be used.
  FMT     Format of Output
            0 - Total number only (default)
            1 - Total number with global array

Output:

  $$NOT   The number of ICD codes that can not
          be used with the ICD code identified
          by IEN (FMT=0 or 1)

  TMP global array as follows (FMT=1):

          ^TMP("SUB",$J,IEN)=CODE
          ^TMP("SUB",$J,"B",(CODE_" "),IEN)=""
```

Code Required With

$$REQ^ICDEX(IEN,SUB,FMT) ICR 5747

```
Input:

  IEN     Internal Entry Number in file 80
  SUB     TMP global array subscript name.
          If not provided, the subscript
          "ICDREQ" will be used.
  FMT     Format of Output
            0 - Total number only (default)
            1 - Total number with global array

Output:

  $$REQ   The number of ICD codes requires when
          the ICD code identified by IEN is used.
          (FMT=0 or 1)

  TMP global array as follows (FMT=1):

          ^TMP("SUB",$J,IEN)=CODE
          ^TMP("SUB",$J,"B",(CODE_" "),IEN)=""
```

Code not Considered CC With

$$NCC^ICDEX(IEN,SUB,FMT) ICR 5747

```
Input:

  IEN     Internal Entry Number in file 80
  SUB     TMP global array subscript name.
          If not provided, the subscript
          "ICDNCC" will be used.
  FMT     Format of Output
            0 - Total number only (default)
            1 - Total number with global array
Output:

  $$NCC   The number of ICD codes not considered
          as Complication/Comorbidity with the
          ICD code identified by IEN.
          (FMT=0 or 1)

  TMP global array as follows (FMT=1):

          ^TMP("SUB",$J,IEN)=CODE
          ^TMP("SUB",$J,"B",(CODE_" "),IEN)=""

          Codes are taken from the DRG CC EXCLUSIONS
          file #82.13.  If not found, and the code
          is a legacy code (ICD-9) then the codes
          will be taken from the ICD CODES NOT CC
          WITH field #40.
```

ICD Identifier was found for Code

$$ICDID^ICDEX(FILE,ID,CODE) ICR 5747

```
Input:

  FILE     File Number or root (required)
             80 or ^ICD9 or 80.1 or ^ICD0
  ID       Diagnosis/Procedure code identifier (required)
  CODE     Diagnosis/Procedure code IEN (required)

Output:

  $$ICDID  Boolean value
             1 if identifier was found
             0 if identifier was not found

                  or upon error -1^error message
```

ICD Identifier String (legacy)

$$IDSTR^ICDEX(FILE,IEN) ICR 5747

```
  Input:

    FILE      File Number or root (required)
                80 or ^ICD9 or 80.1 or ^ICD0
    IEN       Diagnosis/Procedure code IEN (required)

  Output:

    $$IDSTR   String of Identifiers delimited by a semi-colon

                ID;ID;ID
```

All ICD Identifiers assigned to a Code

$$ICDIDS^ICDEX(FILE,CODE,.ARY) ICR 5747

```
Input:

  FILE      File Number or root (required)
              80 or ^ICD9 or 80.1 or ^ICD0
  CODE      Diagnosis/Procedure code IEN (required)
  ARY       Array Name passed by reference (required)

Output:

  $$ICDIDS  Number of Identifiers found
            0 (zero) if no identifiers found

            or upon error -1^error message

  ARY       Array of identifiers found
                     ARY(<identifier)=""
```

ICD is own CC – Return CC

$$ISOWNCC^ICDEX(IEN,CDT,FMT) ICR 5747

```
Input:

  IEN         Internal Entry Number for file 80 (required)
  CDT         Date to use to extract CC (default TODAY)
  FMT         Output Format
                 0 = CC only (default)
                 1 = CC ^ Effective Date
Output:

  $$ISOWNCC   Complication/Comorbidity (CC)

              DX is Own CC  Format  Output
              ------------  ------  -------------------------
                   Yes         0     CC Value
                   Yes         1     CC Value ^ Effective Date
                   No        N/A     0 (zero)

                   or upon error -1^error message
```

DRG Complication/Comorbidity/Major CC

$$ICDRGCC^ICDEX(DRG,CDT) ICR 5747

```
Input:

  DRG         Internal Entry Number for file 80.2 (required)
  CDT         Date to use to extract CC/MCC flag
              (default TODAY)

Output:

  $$ICDRGCC   Complication/Comorbidity/Major CC flag

                 0   No CC or MCC
                 1   CC present
                 2   MCC present
                 3   CC or MCC present

                   or upon error -1^error message
```

Inquire to the ICD Files (interactive)

INQ^ICDEX ICR 5747

```
User will be prompted for:

          Effective Date
          File
          Code

Displays  Code
          Short Text
          Description
          Description Warnings (if any)
            Text may be inaccurate, Effective Date
               Predates Code Set Versioning
               Predates Coding System Implementation
               Predates Initial Activation Date
          Activation Warnings (if any)
             Code is Inactive
             Code is pending (activated in the future)
```

Get Effective date in range (interactive)

EFD^ICDEX ICR 5747

```
Prompts for Effective Date for DRG grouper

The lower boundary for the date is the ICD-9
implementation date October 1, 1978.

The upper boundary for date is either

      3 years from the ICD-10 implementation date or
      3 years from TODAY

      Whichever is further into the future

Input:

  None

Output:

  $$EFF  3 piece ^ delimited string

         1   Date Fileman format         nnnnnnn
         2   Date External Short Format  mm/dd/yyyy
         3   Date External Long Format   Mmm dd, yyyy
```

Primary Diagnosis Exclusion Code

$$PDXE^ICDEX(IEN) ICR 5747

```
Input

  IEN     Internal Entry Number (IEN) for file #80

Output

  $$PDXE  Pointer to DRG CC Exclusions file #82.13
```

DRG Information

$$DRG^ICDEX(IEN,CDT) ICR 5747

```
Input:

   CODE    DRG code, internal or external format (Required)
   CDT     Date, FileMan format (default = TODAY)
              If CDT < 10/1/1978, use 10/1/1978
              If CDT > DT, validate with In/Activation Dates
              If CDT is year only, use first of the year
              If CDT is year and month, use first of the month
Output:
          Returns an 22 piece string delimited by the
          up-arrow (^) the pieces are:

             1  DRG name (field #.01)
             2  Weight (field #2)
             3  Low Trim (days) (field #3)
             4  High Trim (days) (field #4)
             5  MDC (field #5)
             6  Surgery Flag (field #.06)
             7  <null>
             8  Avg Length of Stay (days) (field 10)
             9  Local Low Trim Days (field #11)
            10  Local High Trim Days (field #12)
            11  <null>
            12  Local Breakeven (field #13)
            13  Activation Date (.01 of the 66 multiple)
            14  Status (.03 of the 66 multiple)
            15  Inactivation Date (.01 of the 66 multiple)
            16  Effective date (.01 of the 66 multiple)
            17  Internal Entry Number (IEN)
            18  Effective date (.01 of the 66 multiple)
            19  Reference (field #900)
            20  Weight (Non Affil) (field #7)
            21  Weight (Int Affil) (field #7.5)
            22  Message
```

DRG Description (formatted)

$$DRGDES^ICDEX(IEN,CDT,ARY,LEN) ICR 5747

```
Input:

   IEN     Internal Entry Number of DRG file 80.2
   CDT     Date to screen against (default = TODAY)
  .ARY     Output Array passed by reference
   LEN     Length of each array node
             Missing        Defaults to 79
             Less than 25   Defaults to 25
Output:

   $$DRGD  Number of lines in description output array
   ARY     Description in array of length specified
```

DRG Description (unformatted)

$$DRGD^ICDEX(IEN,CDT,ARY,LEN) ICR 5747

```
Input:

   CODE    ICD Code, Internal or External Format (required)
   ARY     Output Array Name for description
             e.g. "ABC" or "ABC("TEST")"
             Default = ^TMP("DRGD",$J)
   CDT     Date to screen against (default = TODAY)
             If CDT < 10/1/1978, use 10/1/1978
             If CDT > DT, use DT
             If CDT is year only, use first of the year
             If CDT is year/month only, use first of the month

Output:

   $$DRGD  Number of lines in description output array

   ARY     Description in array

             @ARY(1:n) - Description (lines 1-n) (field 68)
             @ARY(n+1) - Blank
             @ARY(n+1) - Message: CODE TEXT MAY BE INACCURATE

           or

             -1^Error Description

** NOTE - USER MUST INITIALIZE ^TMP("DRGD",$J), IF USED **
```

Get the DRG Weighted Work Unit (WWU)

$$DRGW^ICDEX(CODE) ICR 5747

```
Input:

   IEN      Internal Entry Number file 80.2

Output:

   $$WT     Weight

Replaces ICR 48
```

Get the DRG Code of an IEN

$$DRGC^ICDEX(IEN) ICR 5747

```
Input:

   IEN      Internal Entry Number file 80.2

Output:

   $$DRGC   Code (field .01)

Replaces ICR 370
```

Get the IEN of a DRG Code

$$DRGN^ICDEX(CODE) ICR 5747

```
  Input:

    CODE     DRG code

 Output:

    $$DRGN   IEN of DRG code
             or
             -1 on error
```

Calculate Effective Date from Patient Data

$$GETDATE^ICDEX(IEN) ICR 5747

```
Input:

   IEN        Internal Entry Number of the PTF file #45

Output:

   $$GETDATE  Returns the correct "EFFECTIVE DATE"
              for a patient to uses retrieving and
              calculating DRG/ICD/CPT data (default
              TODAY)

              Derived from:
                Census Date      ^DGPT         0;13   ICR 5822
                Discharge Date   ^DG(45.86     0;1    ICR 5821
                Surgery Date     ^DGPT(D0,"S"  0;1    ICR 5822
                Movement Date    ^DGPT(D0,"M"  0;10   ICR 5822

 Input:
```

4.4.6 Special Lookup

Special Lookup called by Fileman (DIC)

LK^ICDEX ICR 5747
ICDEXLK ICR 5747

```
This is the Special Lookup program for files 80 and 80.1.
Only the ^DIC call honors the special lookup routines.
Those calls that allow the user to specify the indexes
(IX^DIC and MIX^DIC1), and the Data Base Server calls
(FIND^DIC, $$FIND1^DIC, and UPDATE^DIE) all ignore the
Special Lookup Program.  Also, if DIC(0) contains an
"I" then the Special Lookup program will be ignored.

Local Variables NEWed or KILLed by Calling Application

   ICDVDT   Versioning Date (Fileman format) (OLD, CSV)

              If supplied only active codes on that date
              will be included in the selection list.

              If not supplied, the date will default to
              TODAY and all codes may be selected, active
              and inactive.

              In both cases the display will be altered
              based on the date.

   ICDSYS   Coding System (from file 80.4) (NEW)
```

```
   1    ICD    ICD-9-CM
   2    ICP    ICD-9 Proc
  30    10D    ICD-10-CM
  31    10P    ICD-10-PCS

ICDFMT   Display Format (numeric, 1-4) (NEW)

         1 = Fileman format, code and short text (default)

             250.00    DMII WO CMP NT ST UNCNTR

         2 = Fileman format, code and description

             250.00    DIABETES MELLITUS WITHOUT MENTION OF
                       COMPLICATION, TYPE II OR UNSPECIFIED
                       TYPE, NOT STATED AS UNCONTROLLED

         3 = Lexicon format, short text followed by code

             DMII WO CMP NT ST UNCNTR (250.00)
```

```
            4 = Lexicon format, description followed by code

                DIABETES MELLITUS WITHOUT MENTION OF
                COMPLICATION, TYPE II OR UNSPECIFIED TYPE, NOT
                STATED AS UNCONTROLLED (250.00)

Special Lookup
   ^DD(80,0,"DIC")="ICDEXLK"
   ^DD(80.1,0,"DIC")="ICDEXLK"

FileMan Variables

X          If DIC(0) does not contain an A, then the variable
           X must be defined equal to the value you want to
           find in the requested Index(es).

DIC        Global root or File Number

               ^ICD9( or 80
               ^ICD0( or 80.1

DIC(0)     (Optional) A string of characters which alter how
           DIC responds. Default value for ICD files "AEM"

           Applicable to a versioned file
              A   Ask the entry; if erroneous, ask again
              B   Only the B index is used
              E   Echo information
              F   Forget the lookup value
              I   Ignore the special lookup program
              M   Multiple-index lookup allowed
              O   Only find one entry if it matches exactly
              S   Suppresses display of .01
              T   Search until user selects or enters ^^
              X   EXact match required
              Z   Zero node in Y(0), external form in Y(0,0)

           Not Applicable to a versioned file
              C   Versioned cross-references not turned off
              K   Primary Key not established
              L   Learning a new entry LAYGO not allowed
              N   Uppercase, IEN lookup allowed (not forced)
              n   ICD has no pure numeric entries
              Q   Input is pre-processed, ?? not necessary
              U   All values are external
              V   Verification is not optional

DIC("A")   (Optional) A prompt that is displayed prior to the reading
           of the X input. If DIC("A") is not defined, a prompt will
           be supplied by the special lookup routines.

DIC("B")   (Optional) The default answer which is presented to the
           user when the lookup prompt is issued. If a terminal
           user simply presses the Enter/Return key, the DIC("B")
           default value will be used, and returned in X.
           DIC("B") will only be used if it is non-null.
```

DIC("S") (Optional) DIC("S") is a string of M code that DIC executes to screen an entry from selection. DIC("S") must contain an IF statement to set the value of $T. Those entries that the IF sets as $T=0 will not be displayed or selectable. When the DIC("S") code is executed, the local variable Y is the internal number of the entry being screened and the M naked indicator is at the global level @(DIC_"Y,0)").

DIC("W") (Optional) An M command string which is executed when DIC displays each of the entries that match the user's input. The condition of the variable Y and of the naked indicator is the same as for DIC("S"). If DIC("W") is defined, it overrides the display of any identifiers of the file. Thus, if DIC("W")="", the display of identifiers will be suppressed.

DIC("?N",<file>)=n
(Optional) The number "n" should be an integer set to the number of entries to be displayed on the screen at one time when using "?" help in a lookup.

FileMan Variables not used:

DIC("DR")
DIC("PTRIX",<from>,<to>,<file>)
DIC("T")
DIC("V")
DIC("?PARAM",<file>,"INDEX")
DIC("?PARAM",<file>,"FROM",<subscript>)
DIC("?PARAM",<file>,"PART",<subscript>)

FileMan Variables KILLed:

DLAYGO
DINUM

Output

Y	IEN ^ Code	Fileman

If DIC(0) contains "Z"

Y(0)	0 Node	Fileman
Y(0,0)	Code	Fileman
Y(0,1)	$$ICDDX or $$ICDOP	Non-Fileman
Y(0,2)	Long Description	Non-Fileman

Silent Lookup (GUI)

$$LKTX^ICDEX(X,ROOT,CDT,SYS,VER,OUT) ICR 5747

```
Input

  TXT    Text to Search for (Required)

            Diagnosis or Procedure Code
            Diagnosis or Procedure Descriptive Text

  ROOT   Global Root/File # to Search (Fileman DIC, Required)

            ^ICD9(
            ^ICD0(

  CDT    Date (default = TODAY) (Optional)

  SYS    Coding System (Optional but encouraged)

             1      ICD-9-CM
             2      ICD-9 Proc
            30      ICD-10-CM
            31      ICD-10-PCS

  VER    Versioned Lookup

            0  No, include all codes, active and inactive
            1  Yes, include only Active codes for date CDT

  OUT    Output Format

            1  Fileman, Code and Short Text (default)

               250.00    DMII WO CMP NT ST UNCNTR

            2  Fileman, Code and Description

               250.00    DIABETES MELLITUS WITHOUT MENTION OF
                         COMPLICATION, TYPE II OR UNSPECIFIED
                         TYPE, NOT STATED AS UNCONTROLLED

            3  Lexicon, Short Text and Code

               DMII WO CMP NT ST UNCNTR (250.00)

            4  Lexicon, Description and Code

               DIABETES MELLITUS WITHOUT MENTION OF
               COMPLICATION, TYPE II OR UNSPECIFIED TYPE,
               NOT STATED AS UNCONTROLLED (250.00)

Output (if successful)

  $$LK  Number of entries found
```

Global Array of entries found:

^TMP(ID,$J,"SEL")
^TMP(ID,$J,"SEL",0)=# of entries
^TMP(ID,$J,"SEL",#)=IEN ^ Display Text

Where ID is a package namespaced subscript:

ICD9 - for the Diagnosis file #80
ICD0 - for the Operations/Procedure file #80.1

Local Variables used but NEWed or KILLed Elsewhere

DIC(0)

Extract Fileman Y Variable

Y(ROOT,IEN,CDT,FMT) ICR 5747

```
Input
   ROOT      Global Root (DIC) or file Number
   IEN       Internal Entry Number
   CDT       Versioning date (default TODAY)
   FMT       Format of output
               0   Standard Fileman Y IEN ^ CODE
               1   Expanded Y as if DIC(0) contained a "Z"
Output

   Y         IEN ^ Code                Fileman Compliant

   If FMT greater than 0

   Y(0)      0 Node (Code)             Fileman Compliant
   Y(0,0)    .01 Field (Code)          Fileman Compliant
   Y(0,1)    $$ICDDX or $$ICDOP        Non-Fileman
   Y(0,2)    Long Description          Non-Filema
```
n

TOKEN(ROOT,ROOT,SYS,.ARY) ICR 5747

```
Input
   TEXT      This is a text string to parse.
   ROOT      This is a global root or file number (required)
                ^ICD9(  or  80
                ^ICD0(  or  80.1
   SYS       This is the coding system  (Required)
                 1  or  ICD  or  ICD-9-CM
                 2  or  ICP  or  ICD-9 Proc
                30  or  10D  or  ICD-10-CM
                31  or  10P  or  ICD-10-PCS

Output
  .ARY       This is the output array passed by reference
             containing words parsed from the input string TEXT
             and arranged by frequency of use (Required)
```

```
   ARY(USE,SYS)=WORD

Where USE is the number of times the word was used
in the file identified by ROOT and coding system
SYS and WORD is a single word found in designated
coding system
```

$$WORD(WORD,ROOT,SYS) ICR 5747

```
Input
    WORD      This is a single word.
    ROOT      This is a global root or file number (required)
                 ^ICD9(  or  80
                 ^ICD0(  or  80.1
    SYS       This is the coding system  (Required)
                  1  or  ICD  or  ICD-9-CM
                  2  or  ICP  or  ICD-9 Proc
                 30  or  10D  or  ICD-10-CM
                 31  or  10P  or  ICD-10-PCS
 Output
    $$WORD    This is a Boolean value indicating if a word is
              contained in a set (file or system).

                1 = Word was found

                    If ROOT is not supplied, the word was
                    found in either file 80 or 80.1

                    If SYS is not supplied, the word was
                    found in the file designated by ROOT
                    in any coding system in the file

                    If both ROOT and SYS are supplied,
                    the word was found in the specified
                    coding system

                0 = Word was not found
```

5. Files

The ICD data dictionaries may not be modified. The file descriptions of these files will be so noted.

5.1 Globals to Journal

There are no globals to journal in the ICD package.

5.2 File List

File #	File Name	Global
80	ICD DIAGNOSIS	^ICD9(
80.1	ICD OPERATION/PROCEDURE	^ICD0(
80.2	DRG	^ICD
80.3	MAJOR DIAGNOSTIC CATEGORY	^ICM
80.4	ICD CODING SYSTEMS	^ICDS(
82	DRG DIAGNOSIS IDENTIFIER CODES	^ICDID(
82.1	DRG PROCEDURE IDENTIFIER CODES	^ICDIP(
82.11	DRG PROCEDURE CODE COMBINATIONS	^ICDIDP(
82.12	DRG DIAGNOSIS CODE COMBINATIONS	^ICDIDD(
82.13	DRG CC EXCLUSIONS	^ICDCCEX(

5.3 Condensed Data Dictionary Listing

5.3.1 ICD DIAGNOSIS file #80

```
FIELD     FIELD
NUMBER    NAME

.01       CODE NUMBER (RF), [0;1]
1.1       CODING SYSTEM (*P80.4'), [1;1]
1.11      PDX EXCLUSION CODE (P82.13'), [1;11]
1.2       IDENTIFIER (F), [1;2]
1.3       UNACCEPTABLE AS PRINCIPAL DX (S), [1;3]
1.4       MDC13 (NJ2,0), [1;4]
1.5       MDC24 (S), [1;5]
1.6       MDC25 (S), [1;6]
1.7       ICD EXPANDED (S), [1;7]
1.8       EXCLUDE FROM LOOKUP (CJ1), [ ; ]
1.9       POA EXEMPT (S), [1;9]
10        SEX (Multiple-80.04), [5;0]
          .01  SEX EFFECTIVE DATE (D), [0;1]
          1    SEX (S), [0;2]
11        AGE LOW (Multiple-80.011), [6;0]
          .01  AGE LOW EFFECTIVE DATE (D), [0;1]
          1    AGE LOW (NJ2,0), [0;2]
12        AGE HIGH (Multiple-80.012), [7;0]
          .01  AGE HIGH EFFECTIVE DATE (D), [0;1]
          1    AGE HIGH (NJ3,0), [0;2]
20        ICD CODES NOT TO USE WITH (Multiple-80.01), [N;0]
          .01  ICD CODE NOT TO USE WITH (MP80'X), [0;1]
30        ICD CODES REQUIRED WITH (Multiple-80.02), [R;0]
          .01  ICD CODE REQUIRED WITH (MP80'X), [0;1]
40        ICD CODES NOT CC WITH (Multiple-80.03), [2;0]
          .01  ICD CODE NOT CC WITH (MP80'), [0;1]
66        STATUS (Multiple-80.066), [66;0]
          .01  STATUS EFFECTIVE DATE (RD), [0;1]
          .02  STATUS (RS), [0;2]
67        DIAGNOSIS (Multiple-80.067), [67;0]
          .01  DIAGNOSIS EFFECTIVE DATE (MRD), [0;1]
          1    DIAGNOSIS (RF), [0;2]
68        DESCRIPTION (Multiple-80.068), [68;0]
          .01  DESCRIPTION EFFECTIVE DATE (MRD), [0;1]
          1    DESCRIPTION (RF), [1;1]
          2    NON-SDO DESCRIPTIVE KEYWORDS (F), [2;1]
71        DRG GROUPER (Multiple-80.071), [3;0]
          .01  DRG GROUPER EFFECTIVE DATE (MMD), [0;1]
          1    DRG (Multiple-80.711), [1;0]
               .01  DRG (MP80.2'), [0;1]
72        MAJOR DIAGNOSTIC CATEGORY (Multiple-80.072), [4;0]
          .01  MDC EFFECTIVE DATE (MD), [0;1]
          1    MDC (P80.3'), [0;2]
73        DRG DIAGNOSIS IDENTIFIER CODES (Multiple-80.073), [73;0]
          .01  DRG DIAGNOSIS IDENTIFIER CODE (MP82'), [0;1]
103       COMPLICATION/COMORBIDITY (Multiple-80.0103), [69;0]
          .01  CC EFFECTIVE DATE (D), [0;1]
          1    COMPLICATION/COMORBIDITY (S), [0;2]
```

```
          2    PRIMARY (S), [0;3]
```

5.3.2 ICD OPERATION/PROCEDURE file #80.1

```
FIELD     FIELD
NUMBER    NAME

.01       CODE NUMBER (RF), [0;1]
1.1       CODING SYSTEM (*P80.4'), [1;1]
1.2       IDENTIFIER (F), [1;2]
1.5       MDC24 (S), [1;5]
1.7       ICD EXPANDED (S), [1;7]
1.8       EXCLUDE FROM LOOKUP (CJ1), [ ; ]
10        SEX (Multiple-80.11), [3;0]
          .01  SEX EFFECTIVE DATE (D), [0;1]
          1    SEX (S), [0;2]
20        MAJOR O.R. PROC (F), [M;1]
66        STATUS (Multiple-80.166), [66;0]
          .01  STATUS EFFECTIVE DATE (RD), [0;1]
          .02  STATUS (RS), [0;2]
67        OPERATION/PROCEDURE (Multiple-80.167), [67;0]
          .01  OPER/PROCEDURE EFFECTIVE DATE (MRD), [0;1]
          1    OPERATION/PROCEDURE (RF), [0;2]
68        DESCRIPTION (Multiple-80.168), [68;0]
          .01  DESCRIPTION EFFECTIVE DATE (MRD), [0;1]
          1    DESCRIPTION (RF), [1;1]
          2    NON-SDO DESCRIPTIVE KEYWORDS (F), [2;1]
71        DRG GROUPER (Multiple-80.171), [2;0]
          .01  DRG GROUPER EFFECTIVE DATE (MD), [0;1]
          1    MAJOR DIAGNOSTIC CATEGORIES (Multiple-80.1711), [1;0]
               .01  MAJOR DIAGNOSTIC CATEGORY (MP80.3'), [0;1]
               1    DRG (Multiple-80.17111), [1;0]
                    .01  DRG (MP80.2'), [0;1]
73        DRG PROCEDURE IDENTIFIER CODES (Multiple-80.173), [73;0]
          .01  DRG PROCEDURE IDENTIFIER CODE (MP82.1'), [0;1]
```

5.3.3 DRG file 80.2

```
FIELD     FIELD
NUMBER    NAME

.001      NUMBER (NJ4,0), [ ]
.01       NAME (R), [0;1]
.06       SURGERY (S), [0;6]
1         DESCRIPTION (Multiple-80.21), [1;0]
          .01  DESCRIPTION (MF), [0;1]
2         WEIGHT (NJ8,3), [0;2]
3         LOW TRIM(days) (NJ2,0), [0;3]
4         HIGH TRIM(days) (NJ3,0), [0;4]
5         MDC# (RP80.3'), [0;5]
7         WEIGHT(nonAffil) (NJ8,2), [0;7]
7.5       WEIGHT(IntAffil) (NJ8,2), [0;11]
10        AVG LENGTH OF STAY(days) (NJ8,2), [0;8]
11        LOCAL LOW TRIM(Days) (NJ2,0), [0;9]
12        LOCAL HIGH TRIM(Days) (NJ3,0), [0;10]
13        LOCAL BREAKEVEN (NJ5,1), [0;12]
```

```
14        ACTIVATION DATE (D), [0;13]
15        INACTIVE (S), [0;14]
16        INACTIVATION DATE (D), [0;15]
20        FISCAL YEAR WEIGHTS&TRIMS (Multiple-80.22), [FY;0]
          .01  FISCAL YEAR WEIGHTS&TRIMS (MDX), [0;1]
          2    WEIGHT (NJ9,3), [0;2]
          2.5  WEIGHT(nonAffil) (NJ7,2), [0;8]
          2.6  WEIGHT(IntAfill) (NJ8,2), [0;10]
          3    LOW TRIM(days) (NJ2,0), [0;3]
          4    HIGH TRIM(days) (NJ3,0), [0;4]
          4.5  AVG LENGTH OF STAY(days) (NJ9,2), [0;9]
          5    *** (NJ5,0), [0;5]
          6    LOCAL LOW TRIM(days) (NJ2,0), [0;6]
          7    LOCAL HIGH TRIM(days) (NJ3,0), [0;7]
30        BREAKEVEN FISCAL YEAR/QUARTER (Multiple-80.23), [BE;0]
          .01  BREAKEVEN FISCAL YEAR/QUARTER (NJ3,0XO), [0;1]
          1    SERVICE (Multiple-80.24), [S;0]
               .01  SERVICE (SX), [0;1]
               1    BREAK EVEN DAYS (RNJ5,1), [0;2]
          2    MEDICAL CENTER BREAKEVEN (RNJ5,1), [0;2]
66        EFFECTIVE DATE (Multiple-80.266), [66;0]
          .01  EFFECTIVE DATE (RD), [0;1]
          .03  STATUS (RS), [0;3]
          .05  MDC# (RP80.3'), [0;5]
          .06  SURGERY (RS), [0;6]
68        DESCRIPTION (VERSIONED) (Multiple-80.268), [68;0]
          .01  EFFECTIVE DATE (MD), [0;1]
          1    DESCRIPTION (Multiple-80.2681), [1;0]
               .01  DESCRIPTION (MF), [0;1]
71        DRG GROUPER EFFECIVE DATE (Multiple-80.271), [2;0]
          .01  DRG GROUPER EFFECIVE DATE (D), [0;1]
          1    REFERENCE (F), [0;3]
900       REFERENCE (F), [MC1;1]
```

5.3.4 MAJOR DIAGNOSTIC CATEGORY file 80.3

```
FIELD     FIELD
NUMBER    NAME

.001      NUMBER (NJ2,0), [ ]
.01       NAME (R), [0;1]
1         DISORDER/PROCEDURE (Multiple-80.31), [1;0]
          .01  DISORDER/PROCEDURE (MF), [0;1]
          1    SURGERY (S), [0;2]
          2    DRGa (NJ3,0), [0;3]
          3    DRGb (NJ3,0), [0;4]
          4    DRGc (NJ3,0), [0;5]
          5    DRGd (NJ3,0), [0;6]
          6    DRGe (NJ3,0), [0;7]
          7    DRGf (NJ3,0), [0;8]
          99   MUMPS CODE (Multiple-80.32), [1;0]
               .01  MUMPS CODE (MF), [0;E1,200]
```

5.3.5 ICD CODING SYSTEMS file 80.4

```
FIELD     FIELD
NUMBER    NAME

.001      ICD CODING SYSTEM (NJ4,0), [ ]
.01       ICD CODING SYSTEM NOMENCLATURE (F), [0;1]
.02       CODING SYSTEM ABBREVIATION (F), [0;2]
.03       ICD FILE (*P1'), [0;3]
.04       IMPLEMENTATION DATE (D), [0;4]
```

5.3.6 DRG DIAGNOSIS IDENTIFIER CODES File #82

```
FIELD     FIELD
NUMBER    NAME

.01       IDENTIFIER CODE (RF), [0;1]
1         DESCRIPTION (RF), [0;2]
```

5.3.7 DRG PROCEDURE IDENTIFIER CODES File #82.1

```
FIELD     FIELD
NUMBER    NAME

.01       IDENTIFIER CODE (RF), [0;1]
1         DESCRIPTION (RF), [0;2]
```

5.3.8 DRG PROCEDURE CODE COMBINATIONS file #82.11

```
FIELD     FIELD
NUMBER    NAME

.01       IDENTIFIER CODE (RP82.1'), [0;1]
1         BLOCK (Multiple-82.111), [BL;0]
          .01  BLOCK (MRNJ4,0), [0;1]
          1    ONE OF (Multiple-82.1111), [ONE;0]
               .01  ONE OF (MRMP80.1'), [0;1]
          2    WITH ONE OF 1 (Multiple-82.1112), [WITH1;0]
               .01  WITH ONE OF 1 (MRP80.1'), [0;1]
          3    WITH ONE OF 2 (Multiple-82.1113), [WITH2;0]
               .01  WITH ONE OF 2 (MRP80.1'), [0;1]
          4    WITH ONE OF 3 (Multiple-82.1114), [WITH3;0]
               .01  WITH ONE OF 3 (MRP80.1'), [0;1]
          5    WITH ONE OF 4 (Multiple-82.1115), [WITH4;0]
               .01  WITH ONE OF 4 (MRP80.1'), [0;1]
          6    MDC (Multiple-82.1116), [MDC;0]
               .01  MDC (MRP80.3'), [0;1]
               1    DRG (Multiple-82.11161), [DRG;0]
                    .01  DRG (MRP80.2'), [0;1]
```

5.3.9 DRG DIAGNOSIS CODE COMBINATIONS file #82.12

```
FIELD     FIELD
NUMBER    NAME

.01       IDENTIFIER CODE (RP82'), [0;1]
1         BLOCK (Multiple-82.121), [BL;0]
          .01  BLOCK (MRNJ4,0), [0;1]
          1    ONE OF (Multiple-82.1211), [ONE;0]
               .01  ONE OF (MRP80'), [0;1]
          2    WITH ONE OF 1 (Multiple-82.1212), [WITH1;0]
               .01  WITH ONE OF 1 (MRP80'), [0;1]
          3    WITH ONE OF 2 (Multiple-82.1213), [WITH2;0]
               .01  WITH ONE OF 2 (MRP80'), [0;1]
          4    WITH ONE OF 3 (Multiple-82.1214), [WITH3;0]
               .01  WITH ONE OF 3 (MRP80'), [0;1]
          5    WITH ONE OF 4 (Multiple-82.1215), [WITH4;0]
               .01  WITH ONE OF 4 (MRP80'), [0;1]
          6    MDC (Multiple-82.1216), [MDC;0]
               .01  MDC (MRP80.3'), [0;1]
               1    DRG (Multiple-82.12161), [DRG;0]
                    .01  DRG (MRP80.2'), [0;1]
```

5.3.10 DRG CC EXCLUSIONS file #82.13

```
FIELD     FIELD
NUMBER    NAME

.01       EXCLUSION CODE (RF), [0;1]
1         PDX (Multiple-82.131), [1;0]
          .01  PDX (MRP80'), [0;1]
```

5.4 Detailed Data Dictionary Listing

Using Fileman, select the "DATA DICTIONARY UTILIITIES" menu, then select the "LIST FILE ATTRIBUTES" option. At the "START WITH WHAT FILE" prompt, enter one of the ICD file numbers (80 for diagnosis, 80.1 for procedures, or 80.4 for coding systems). Accept default values for the remaining prompts. This will display a detailed listing of the selected file.

6. Routines

Legacy Routines	**ICDAPIU**	**API Utilities**
	ICDCODE	**Get Code Data**
DRG Routines	**ICDDRG***	**DRG Calculations**
	ICDGTDRG	**Get DRG Data**
	ICDREF	**DRG Reference**
	ICDSUPT	**DRG Support**
	ICDTBL*	**Tables**
Data Extraction Routines	**ICDXCODE**	**Interim ICD-9/10**
	ICDEX*	**Data Extraction**
Lookup/Help	**ICDEXLK***	**Special Lookup**
	ICDSAPI	**Interim ICD-9/10 DIC call**
	ICDDIC	**DIC/Prototype**
	ICDDICA	**DIC/Prototype**
	ICDHLPD	**Diagnostic Identifiers**
	ICDHLPO	**Operation Identifiers**
	ICDID	**File Identifiers**
	ICDCOD	**Inquire to ICD Codes**
Cross-Reference	**ICDIDX***	**Re-Index**
	ICDIDX2	**Re-Index Histories**
	ICDTOKN	**Parse Text to Words**
No longer used	**ICDUPDT**	**Update Protocol**

A complete listing of routines with checksums can be displayed using the XTSUMBLD-CHECK option. At the "New or Old Checksums" prompt, enter "new" and when prompted for "Package" or "Build," select "Package." When prompted for "All routines," respond "No" and enter the namespace ICD* (include the asterisk). You will be presented with a complete list of routines and checksums for the ICD package.

7. Templates

None

8. Options

8.1 ICD DRG GROUPER

Menu Text: DRG Grouper
DESCRIPTION: Used to calculate DRG based on Diagnosis and Operation/Procedure codes entered.
Runs Routine ICDDRGM

9. Protocols

9.1 ICD CODE UPDATE EVENT ICR 4126

TEXT: ICD Code Update
TYPE: Extended Action
DESCRIPTION: Protocol Event for Notifying Applications that an update to File #80 or File #80.1 has occurred. It is commonly invoked by the LEXICAL SERVICES PROTOCOL when the Lexicon installs ICD data.

10. Integration Control Registrations (ICRs) Summary

10.1 ICRs with ICD as the Custodian

10.1.1 Retired/Withdrawn

Files

ICR	File	Scope	Subscriber	Status	Date
368	**^ICD9(**	**Private**	**IB**	**Retired**	**Nov 15, 2008**
369	**^ICD0(**	**Private**	**IB**	**Retired**	**Nov 15, 2008**
647	**^ICD9(**	**Private**	**IB**	**Retired**	**Nov 15, 2008**
1161	**^ICD9(**	**Private**	**VAM**	**Retired**	**Nov 15, 2008**
1275	**^ICD9(**	**Private**	**GMTS**	**Retired**	**Nov 15, 2008**
1276	**^ICD0(**	**Private**	**GMTS**	**Retired**	**Nov 15, 2008**
1294	**^ICD9(**	**Controlled**	**PX/TIU/OR**	**Retired**	**Nov 15, 2008**
1487	**^ICD9(**	**Private**	**ACKQ**	**Retired**	**Nov 15, 2008**
3482	**^ICD9(**	**Controlled**	**DENT**	**Withdrawn**	**Nov 26, 2001**
3840	**^ICD9(**	**Controlled**	**N/A**	**Withdrawn**	**Apr 02, 2003**
5028	**^ICD9(**	**Controlled**	**PL**	**Withdrawn**	**Aug 21, 2007**
5682	**^ICD10DX(**	**Private**	**LEX**	**Withdrawn**	**Jun 07, 2011**

ICR	File	Scope	Subscriber	Status	Date
5683	**^ICD10PR(**	**Private**	**LEX**	**Withdrawn**	**Jun 07, 2011**
10082	**^ICD9(**	**Supported**	**All**	**Withdrawn**	**NOV 15,2008**
10083	**^ICD0(**	**Supported**	**All**	**Withdrawn**	**NOV 15,2008**

Routine

ICR	Routine	Scope	Subscriber	Status	Date
5684	**ICDXCD**	**Supported**	**All**	**Withdrawn**	**Jun 08, 2011**
5685	**ICDXAU**	**Supported**	**All**	**Withdrawn**	**Jun 08, 2011**
5686	**ICDXLK**	**Supported**	**All**	**Withdrawn**	**Jun 08, 2011**

Other

ICR	Component	Scope	Subscriber	Status	Date
5758	**Protocol**	**Controlled**	**PL/GMRC/PXRM**	**Withdrawn**	**Jan 03, 2012**

10.1.2 Active/Pending

Files

ICR	File	Scope	Subscriber	Status	Date
48	**^ICD**	**Private**	**YS**	**Active**	**Jul 25, 1990**
280	**^ICD9(**	**Private**	**HBH**	**Active**	**Sep 13, 1993**
365	**^ICD9(**	**Private**	**QAM**	**Active**	**Mar 03, 1994**
370	**^ICD(**	**Private**	**IB/DSS**	**Active**	**Mar 09, 1994**
582	**^ICD9(**	**Private**	**ICR**	**Active**	**Apr 21, 2003**
1586	**^ICM**	**Controlled**	**IBD/PX**	**Active**	**Aug 08, 1996**
2435	**^ICD9(**	**Private**	**PXRM**	**Active**	**Jun 19, 1998**
2436	**^ICD0(**	**Private**	**PXRM**	**Active**	**Jun 19, 1998**
4485	**^ICD9(**	**Private**	**LEX**	**Active**	**Jul 28, 2004**
4486	**^ICD0(**	**Private**	**LEX**	**Active**	**Jul 28, 2004**
4487	**^ICD(**	**Private**	**LEX**	**Active**	**Jul 28, 2004**
4488	**^ICM(**	**Private**	**LEX**	**Active**	**Jul 28, 2004**
5388	**^ICD9(**	**Supported**	**All**	**Active**	**Mar 16, 2009**
5404	**^ICD0(**	**Supported**	**All**	**Active**	**Mar 17, 2009**
5755	**^ICDS**	**Private**	**LEX**	**Pending**	**Dec 24, 2011**

Routines

ICR	Routine/Entry	Scope	Subscriber	Status	Date
371	**ICDDRG**	**Controlled**	**IB/YS**	**Active**	**Mar 09, 1994**

ICR	Routine/Entry	Scope	Subscriber	Status	Date
	ICDDRG				
3990	**ICDCODE**	**Supported**	**All**	**Active**	**Mar 12, 2003**
	$$ICDDX(CODE,CDT,DFN,SRC)				
	$$ICDOP(CODE,CDT,DFN,SRC)				
	$$ICDD(CODE,'OUTARR',CDT)				
	$$CODEN(CODE,FILE)				
	$$CODEC(CODE)				
3991	**ICDAPIU**	**Supported**	**All**	**Active**	**MAR 12,2003**
	$$STATCHK(CODE,CDT)				
	$$NEXT(CODE)				
	$$PREV(CODE)				
	$$HIST(CODE,ARY)				
	$$DTBR(CDT,CS)				
	$$MSG(CDT,CS)				
	PERIOD(CODE,ARY)				
4052	**ICDGTDRG**	**Supported**	**All/FB/IB/DG**	**Active**	**Jul 14, 2003**
	$$DRG(CODE,EDT)				
	$$GETDRG(CODE,DGNDT,FILE)				
	$$GETDATE(PATNUM)				
	$$ISVALID				
	$$DRGD(CODE,ARRAY,DFN,DATE)				
5699	**ICDXCODE**	**Supported**	**All**	**Pending**	**Aug 02, 2011**
	$$ICDDATA(CSYS,CODE,DATE,FRMT)				
	$$ICDDESC(CSYS,CODE,DATE,OUTARR)				
	$$HIST(SYS,CODE,.ARY)				
	$$NEXT(SYS,CODE)				
	$$PREV(SYS,CODE)				
	$$STATCHK(SYS,CODE,CDT)				
	$$PERIOD(SYS,CODE,.ARY)				
5747	**ICDEX**	**Controlled**	**LEX/PRCA/IB/FB**	**Pending**	**Nov 06, 2011**
	HELP^ICDEX				
	$$ICDDX(CODE,CDT,SYS,FMT)				
	$$ICDOP(CODE,CDT,SYS,FMT)				
	$$ICDD(CODE,.ARY,CDT,SYS,LEN)				
	$$CODEN(CODE,FILE)				
	$$CODEC(FILE,IEN)				
	$$CODEBA(CODE,ROOT)				
	$$CODEABA(CODE,ROOT,SYS)				
	$$CODEFI(CODE)				
	$$CODECS(CODE,FILE,CDT)				

ICR	Routine/Entry	Scope	Subscriber	Status	Date
	$$CSI(FILE,IEN)				
	$$VMDC(IEN,CDT,FMT)				
	$$VAGEL(IEN,CDT,FMT)				
	$$VAGEH(IEN,CDT,FMT)				
	$$VCC(IEN,CDT,FMT)				
	$$VCCP(IEN,CDT,FMT)				
	$$VSEX(FILE,IEN,CDT,FMT)				
	$$SAI(FILE,IEN,CDT)				
	$$VST(FILE,IEN,CDT)				
	$$VLT(FILE,IEN,CDT)				
	$$VSTD(IEN,CDT)				
	$$VSTP(IEN,CDT)				
	$$VLTD(IEN,CDT)				
	$$VLTP(IEN,CDT)				
	$$SD(FILE,IEN,CDT,.ARY,LEN)				
	$$LD(FILE,IEN,CDT,.ARY,LEN)				
	PAR(.ARY,LEN)				
	$$STATCHK(CODE,CDT,SYS)				
	$$DTBR(CDT,STD,SYS)				
	$$IMP(SYS,CDT)				
	$$MSG(CDT,STD,SYS)				
	$$SEL(FILE,IEN)				
	$$NEXT(CODE,SYS,CDT)				
	$$PREV(CODE,SYS,CDT)				
	$$HIST(CODE,.ARY,SYS)				
	$$PERIOD(CODE,.ARY,SYS)				
	$$OBA(FILE,CODE,SYS,REV)				
	$$OD(FILE,WORD,SYS,REV)				
	$$DLM(FILE,IEN,FIELD,CDT)				
	$$CS(FILE,FMT)				
	$$EFF(FILE,IEN,CDT)				
	$$IA(FILE,IEN)				
	$$LA(FILE,IEN,CDT)				
	$$LI(FILE,IEN,CDT)				
	$$LS(FILE,IEN,CDT)				
	$$NUM(CODE)				
	$$COD(NUM)				
	$$IE(CODE)				
	$$FILE(SYS)				
	$$ROOT(SYS)				

ICR	Routine/Entry	Scope	Subscriber	Status	Date
	$$SYS(SYS,CDT,FMT)				
	$$SINFO(SYS,CDT)				
	$$SNAM(SYS)				
	$$SAB(SYS,CDT)				
	$$EXC(FILE,IEN)				
	$$VER(SYS,REL)				
	$$HDR(FILE)				
	$$ISA(IEN1,IEN2,FIELD)				
	$$ISVALID(FILE,IEN,CDT)				
	$$EXIST(IEN,FIELD)				
	$$GETDRG(FILE,IEN,CDT,MDC)				
	MD(FILE,IEN,CDT,.ARY,FLAG)				
	$$EFM(CDT)				
	$$FY(CDT)				
	$$VMDCDX(IEN,CDT)				
	$$VMDCOP(IEN,MDC,CDT)				
	$$REF(IEN,CDT)				
	MDCG(IEN,CDT,.ARY)				
	$$MDCT(IEN,CDT,.ARY,FMT)				
	$$MDCD(IEN,MDC,CDT)				
	$$MDCN(IEN)				
	$$MOR(IEN)				
	$$UPDX(IEN)				
	$$NOT(IEN,SUB,FMT)				
	$$REQ(IEN,SUB,FMT)				
	$$NCC(IEN,SUB,FMT)				
	$$ICDID(FILE,ID,CODE)				
	$$IDSTR(FILE,IEN)				
	$$ICDIDS(FILE,IEN,.ARY)				
	$$ISOWNCC(IEN,CDT,FMT)				
	$$ICDRGCC(DRG,CDT)				
	INQ				
	EFD(X)				
	PDXE(IEN)				
	$$DRG(CODE,CDT)				
	$$DRGW(IEN)				
	$$DRGDES(IEN,CDT,.ARY,LEN)				
	$$DRGD(CODE,OUTARR,CDT)				
	$$DRGN(CODE)				
	$$DRGC(IEN)				

ICR	Routine/Entry	Scope	Subscriber	Status	Date
	$$GETDATE(IEN)				
	LK				
	$$LKTX(X,ROOT,CDT,SYS,VER,OUT)				
	Y(ROOT,IEN,CDT,FMT)				
	TOKEN(X,ROOT,SYS,.ARY)				
	WORD(X,ROOT, SYS)				
5757	**ICDSAPI**	**Supported**	**All**	**Pending**	**DEC 29,2011**
	$$SEARCH(FILE,SCR,FMPAR,CDT)				

Other

ICR	Component	Scope	Subscriber	Status	Date
2184	**GROUP**	**Private**	**PXRM**	**Active**	**Oct 15, 1997**
4126	**Protocol**	**Supported**	**All**	**Active**	**Jul 21, 2003**

10.1.3 Planned for Retirement

ICR	Scope		Replace with
48	**Private**	**YS**	**$$DRGW^ICDEX(IEN)**
280	**Private**	**HBH**	**$$CODEC^ICDEX(FILE,IEN)**
365	**Private**	**QAM**	**$$CODEC^ICDEX(FILE,IEN)**
370	**Private**	**IB/DSS**	**$$DRGC^ICDEX(IEN) and $$DRGDES^ICDEX(IEN,CDT,ARY,LEN)**
582	**Private**	**IMR**	**$$CODEC^ICDEX(FILE,IEN)**
1586	**Subscription**	**AICS/PCE**	**$$MDCN^ICDEXD2(IEN)**
2435	**Private**	**PXRM**	**$$HDR^ICDEX(FILE)**
2436	**Private**	**PXRM**	**$$HDR^ICDEX(FILE)**
3990	**Supported**	**ICDCODE**	**Appropriate ICDEX APIs**
3991	**Supported**	**ICDAPIU**	**Appropriate ICDEX APIs**
4052	**Supported**	**ICDGTDRG**	**Appropriate ICDEX APIs**
5388	**Supported**	**File 80**	**$$CODEC^ICDEX(FILE,IEN) $$IA^ICDEXS(FILE,IEN) $$CODEABA^ICDEXC(CODE,ROOT,SYS) $$OBA^ICDEXA3(FILE,CODE),SYS,REV) and $$OD^ICDEXA3(FILE,WORD,SYS,REV)**
5404	**Supported**	**File 80.1**	**$$CODEC^ICDEX(FILE,IEN) $$IA^ICDEXS(FILE,IEN) $$CODEABA^ICDEXC(CODE,ROOT,SYS) and $$OBA^ICDEXA3(FILE,CODE),SYS,REV)**
5699	**Supported**	**ICDXCODE**	**Appropriate ICDEX APIs**
5757	**Supported**	**ICDSAPI**	**DIC Special Lookup**

11. ICRs Supporting External References

11.1 External Global References

Global Reference	ICR/SACC
^%ZOSF("RSEL"	**ICR 10096**
^%ZOSF("RSUM"	**ICR 10096**
^%ZOSF("TEST"	**ICR 10096**
^%ZOSF("UCI"	**ICR 10096**
^%ZOSF("UCICHECK"	**ICR 10096**
^DG(45.86,	**ICR 5821**
^DGPT(	**ICR 5822**
^DISV(	**ICR 510**
^DPT(	**ICR 10035**
^TMP(NAME,$J)	**SACC 2.3.2.5.1**
^UTILITY($J	**ICR 10011**
^XTMP(	**SACC 2.3.2.5.2**

11.2 External Routine References

Routine Reference	ICR
^%DT	**ICR 10003**
DD^%DT	**ICR 10003**
^%DTC	**ICR 10000**
^%ZIS	**ICR 10086**
HOME^%ZIS	**ICR 10086**
^%ZISC	**ICR 10089**
^%ZTLOAD	**ICR 10063**
CLRMSG^DDS	**ICR 5846**
HLP^DDSMSG	**ICR 5847**
^DIC	**ICD 10006**
IXALL^DIK	**ICR 10013**
^DIM	**ICR 10016**
$$GET1^DIQ	**ICR 2056**
EN^DIQ1	**ICR 10015**
^DIR	**ICR 10026**
^DIWP	**ICR 10011**
$$DT^XLFDT	**ICR 10103**
$$FMADD^XLFDT	**ICR 10103**
$$FMTE^XLFDT	**ICR 10103**
$$UP^XLFSTR	**ICR 10103**

Routine Reference	ICR
H^XUS	**ICR 10044**

12. Archiving and Purging

Archiving and purging capabilities are not applicable as the data is a national table.

13. External/Internal Relations

Minimums of VA FileMan V. 22.0, Kernel V. 8.0, PCE V. 1.0, and PIMS (MAS) V. 5.3 are required to run this package.

14. Package-wide Variables

ICDVDT This variable always refers to a versioning date (FileMan format) used during lookups to determine if a code or text is active or inactive. It also is used by the file 80 and 80.1 identifiers to display a code. It is commonly set to the date that service was provided to the patient. If not provided, TODAY is used.

15. SACC Exemptions/Non-Standard Code

A SACC exemption was granted on May 9, 2013 to the Clinical Lexicon package (distribution package for ICD data) for the purpose of enabling unsubscripted global kills in the pre-install using FileMan DIU2 utility. This is used when a "full file" distribution is made (delete file 80/80.1 and replace). The exemption reads as follows:

> Clinical Lexicon requests an exemption to use $ZU in the pre and post install routines for future LEX patches. This exemption will expire with the release of LEX 3.0. Calling $ZU(68,28,0) to enable an unsubscripted global kill prior to installing the latest ICD files leaves the possibility that a global will be killed by another process during a lengthy installation. Placing the call in the pre (or post) install, instead of making the call manually before and after the install, cuts this window down to a few seconds.

16. How to Generate Online Documentation

This section describes some of the various methods by which users may secure ICD technical documentation. Online technical documentation pertaining to the ICD software, in addition to that, which is located in the help prompts, may be generated through utilization of several kernel options. These include XINDEX and VA FileMan List File Attributes. Further

information about other utilities, which supply online technical documentation, may be found in the Kernel Reference Manual.

16.1 XINDEX

This option analyzes the structure of a routine(s) to determine in part if the routine(s) adheres to VistA Programming Standards. The XINDEX output may include the following components: compiled list of errors and warnings, routine listing, local variables, global variables, naked globals, label references, and external references. By running XINDEX for a specified set of routines, the user is afforded the opportunity to discover any deviations from VistA Programming Standards which exist in the selected routine(s) and to see how routines interact with one another, that is, which routines call or are called by other routines.

To run XINDEX for the ICD package, specify the following namespace at the "routine(s) ?>" prompt: ICD*. ICD initialization routines, which reside in the UCI in which XINDEX is being run, as well as local routines found within the ICD namespace, should be omitted at the "routine(s)? >" prompt. To omit routines from selection, preface the namespace with a minus sign (-).

16.2 List File Attributes

This VA FileMan option allows the user to generate documentation pertaining to files and file structure. Utilization of this option via the "Standard" format will yield the following data dictionary information for a specified file(s): file name and description, identifiers, cross-references, files pointed to by the file specified, files which point to the file specified, input templates, print templates, and sort templates. In addition, the following applicable data is supplied for each field in the file: field name, number, title, global location, description, help prompt, cross-reference(s), input transform, date last edited, and notes.

Using the "Global Map" format of this option generates an output which lists all cross-references for the file selected, global location of each field in the file, input templates, print templates, and sort templates. For a comprehensive listing of CPT files, please refer to the Files section of this manual.

17. Security

17.1 General Security

The ICD data dictionaries may not be modified.

17.2 Security Keys

There are no security keys in the ICD package.

17.3 VA FileMan Access Codes

Below is a list of recommended VA FileMan access codes associated with each file contained in the CPT package. This list may be used to assist in assigning users appropriate VA FileMan access codes.

FILE #	FILE NAME	DD ACCESS	RD ACCESS	WR ACCESS	DEL ACCESS	LAYGO ACCESS
80	**ICD DIAGNOSIS**	**@**	**D**	**@**	**@**	**@**
81.1	**ICD OPERATION/PROCEDURE**	**@**	**D**	**@**	**@**	**@**

18. Appendix A: Integration Control Registrations Detailed

18.1 Integration Control Registration Status

The following Integration Control Registration (ICR) status is as of June, 25, 2012, and is subject to change:

ICR	Coverage	Scope	Status	Comments
48	DRG file #80.2 field 2	Private	Active	Planned Retirement
280	ICD Diagnosis file #80, field .01	Private	Active	Planned Retirement
365	ICD Diagnosis file #80, field .01	Private	Active	Planned Retirement
368	ICD Diagnosis file #80, fields 3, 9.5, and 100	Private	Retired	Nov 15, 2008
369	ICD Procedure file #80.1, fields 4 and 102	Private	Retired	Nov 15, 2008
370	DRG file #80.2 fields .01 and 1	Private	Active	Planned Retirement
371	Routine ICDDRG	Subscription	Active	
582	ICD Diagnosis file #80, field .01	Private	Active	Planned Retirement
647	ICD Diagnosis file #80, field .01 and 3	Private	Retired	Nov 15, 2008
1161	ICD Diagnosis file #80, field .01 and 3	Private	Retired	Nov 15, 2008
1275	ICD Diagnosis file #80, field .01, 3 and 10	Private	Retired	Nov 15, 2008
1276	ICD Procedure file #80.1, fields .01, 4 and 10	Private	Retired	Nov 15, 2008
1294	ICD Diagnosis file #80, fields .01, 3, 5, 10, 100, 102, "AB", "BA" and "D"	Subscription	Retired	Nov 15, 2008
1487	ICD Diagnosis file #80, fields .01, 3 and "BA"	Private	Retired	Nov 15, 2008
1586	ICD MDC file 80.3 field .01	Subscription	Active	Planned Retirement
2184	Application Group PXRS	Private	Active	
2435	ICD Diagnosis file #80 0 (zero) node	Private	Active	Planned Retirement
2436	ICD Procedure file #80.1 0 (zero) node	Private	Active	Planned Retirement
3482	ICD Diagnosis file #80 change notification	Subscription	Pending	To be Withdrawn
3840	Access to DRG File #80.2	Subscription	Withdrawn	
3990	Routine ICDCODE APIs	Supported	Other	Scheduled to be Retired *
3991	Routine ICDAPIU APIs	Supported	Active	Scheduled to be Retired *
4052	Routine ICDGTDRG APIs	Supported	Active	

4485	ICD Diagnosis file #80 privileges	Private	Active	
4486	ICD Procedure file #80.1 privileges	Private	Active	
4487	ICD DRG file #80.2 privileges	Private	Active	
4488	ICD MDC file 80.3 privileges	Private	Active	
5028	ICD Diagnosis file #80 .01 and "AST"	Subscription	Pending	
5388	ICD Diagnosis file #80 fields .01, "AB", "BA", "D", "AST" and "ACT"	Supported	Active	Scheduled to be Retired *
5404	ICD Procedure file #80.1 fields .01, "BA" and "ACT"	Supported	Active	Scheduled to be Retired *
5682	ICD-10 Diagnosis file 8010	Private	Pending	To be Withdrawn
5683	ICD-10 Procedure file 8010.1	Private	Pending	To be Withdrawn
5684	Routine ICDXCD	Supported	Pending	To be Withdrawn
5685	Routine ICDXAU	Supported	Pending	To be Withdrawn
5686	Routine ICDXLK Special Lookup	Supported	Pending	To be Withdrawn
5699	Routine ICDXCODE APIs	Supported	Pending	Scheduled to be Retired *
5747	Routine ICDEX APIs	Subscription	Pending	Replaces 280, 365, 582, 3990, 3991, 5388 AND 5404
5755	ICD Coding System file 80.4 privileges	Private	Pending	
5757	Routine ICDSAPI	Supported	Pending	Scheduled to be Retired *
5758	ICD CODE UPDATE EVENT Protocol	Subscription	Pending	
5773	Routine ICDEXLK	Supported	Pending	
10082	ICD Diagnosis file #80, fields .01, 3, 5, 9.5, 10, 100, 102, "ACT" and "BA"	Supported	Retired	Nov 15, 2008
10083	ICD Procedure file #80.1 fields .01, 4, 9.5, 10, 100, 102, "ACT" and "BA"	Supported	Retired	Nov 15, 2008

*** Scheduled to be retired 18 months after the ICD-10 implementation date**

18.2 ICD as a Subscriber

1118 ICD Codes update in PTF

```
  CUSTODIAL PACKAGE: REGISTRATION
SUBSCRIBING PACKAGE: DRG GROUPER
              USAGE: Private              ENTERED: JAN 11,1995
             STATUS: Active               EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE: 45.89                   ROOT: DIC(45.89,
        DESCRIPTION:                         TYPE: File
   This is to enable the annual DRG Grouper ICD release to include updates to
   the PTF Expanded Code file (#45.89).  New entries are added, updating
   fields .01, CATEGORY; and .02, DIAGNOSIS/PROCEDURE CODE.  Several codes
```

are inactivated, adding entries to their .03, INACTIVE DATE field.

1153 Package File References Cleanup

```
  CUSTODIAL PACKAGE: KERNEL
SUBSCRIBING PACKAGE: DRG GROUPER
              USAGE: Controlled Subscri  ENTERED: FEB 24,1995
             STATUS: Active              EXPIRES:
           DURATION: Next Version        VERSION:
               FILE: 9.4                    ROOT: DIC(9.4)
        DESCRIPTION:                        TYPE: File
```

Loop through the "C" cross-reference on the PACKAGE file and delete any extra entries with the subscribing package namespace. Where necessary, the name of a package may be changed to make it unique.

4306 LEXICAL SERVICES UPDATE Protocol

```
  CUSTODIAL PACKAGE: LEXICON UTILITY
SUBSCRIBING PACKAGE: DRG GROUPER
                        The subscribing protocol is:  ICD CODE UPDATE EVENT
              USAGE: Controlled Subscri  ENTERED: DEC  3,2003
             STATUS: Active              EXPIRES:
           DURATION: Till Otherwise Agr  VERSION:
               FILE:                        ROOT:
        DESCRIPTION:                        TYPE: Other
```

This protocol is used to notify other applications and processes when the Lexicon Utility or the Lexicon Change file is updated.

The Lexicon is updated using a temporary maintenance global, ^LEXM. This global is processed by the routine LEXXGI. Once processed, this protocol is triggered and the global ^LEXM is deleted.

Required Variable LEXSCHG Array contains a listing of those Lexicon Files (#757 - 757.41) that were updated as a result of a recent install. In the case of the CHANGE LOG (file #757.9), new changes to SDO controlled files will be indicated by file number and the internal entry number to the CHANGE LOG.

The variable LEXSCHG is created while processing the Lexicon Maintenance global ^LEXM. It will indicate what files were updated.

Example:

```
LEXSCHG(757,0)=""
LEXSCHG(757.001,0)=""
LEXSCHG(757.01,0)=""
LEXSCHG(757.02,0)=""
LEXSCHG(757.1,0)=""
LEXSCHG(757.11,0)=""
LEXSCHG(757.9,0)=""
LEXSCHG(757.9,2)=80
LEXSCHG(757.9,3)=80.1
LEXSCHG(757.9,4)=81
LEXSCHG(757.9,"B",80,2)=""
LEXSCHG(757.9,"B",80.1,3)=""
LEXSCHG(757.9,"B",81,4)=""
```

If ICD-9-CM and/or CPT-4 changes are included in the ^LEXM global, then the following entries will be found in the local array LEXSCHG:

```
LEXSCHG(80,0)=""
LEXSCHG(80.1,0)=""
LEXSCHG(81,0)=""
```

4404 ID Nodes in ICD Dx file (#80)

```
  CUSTODIAL PACKAGE: VA FILEMAN
SUBSCRIBING PACKAGE: DRG GROUPER
              USAGE: Private             ENTERED: APR 22,2004
             STATUS: Active              EXPIRES:
           DURATION: Till Otherwise Agr  VERSION:
               FILE: 80                     ROOT: DD(80
        DESCRIPTION:                        TYPE: File
```

The Code Text Descriptors project modifies the identifier on the DIAGNOSIS (#3) field in the ICD DIAGNOSIS file (80).

The new identifier makes a function call into $$IDDXS^ICDID to return versioned data for both the DIAGNOSIS and the status in the INACTIVE FLAG. The function has only one input parameter: the Internal Entry Number for file #80. Routine ICDID will also look to see if the package namespaced variable ICDVDT is in the environment. ICDVDT is a versioning date. If ICDVDT is not found in the environment (not supplied) then TODAY will be used and the DIAGNOSIS and INACTIVE FLAG for TODAY will be displayed. If the variable ICDVDT is found in the environment, and is a date other than TODAY, then the appropriate DIAGNOSIS and INACTIVE FLAG will be displayed for the date.

The identifier will be changed to:

```
^DD(80,0,"ID",3)= D EN^DDIOL((" "_$$IDDXS^ICDID(+Y)),"","?0")
```

This will be exported in the combined build CTD UTIL 1.0, containing ICPT*6.0*19, ICD*18.0*12 and LEX*2.0*30.

4405 ID Nodes in ICD OP file (#80.1)

```
  CUSTODIAL PACKAGE: VA FILEMAN
SUBSCRIBING PACKAGE: DRG GROUPER
              USAGE: Private             ENTERED: APR 22,2004
             STATUS: Active              EXPIRES:
           DURATION: Till Otherwise Agr  VERSION:
               FILE: 80.1                   ROOT: DD(80.1
        DESCRIPTION:                        TYPE: File
```

The Code Text Descriptors project modifies the identifier on the OPERATION/PROCEDURE (#4) in the ICD OPERATION/PROCEDURE file (80.1).

The new identifier makes a function call into $$IDOPS^ICDID to return versioned data for both the OPERATION/PROCEDURE and the status in the INACTIVE FLAG. The function has only one input parameter: the Internal Entry Number for file #80.1. Routine ICDID will also look to see if the package namespaced variable ICDVDT is in the environment. ICDVDT is a versioning date. If ICDVDT is not found in the environment (not supplied)

then TODAY will be used and the OPERATION/PROCEDURE and INACTIVE FLAG for TODAY will be displayed. If the variable ICDVDT is found in the environment, and is a date other than TODAY, then the appropriate OPERATION/PROCEDURE and INACTIVE FLAG will be displayed for the date.

The identifiers will be changed to:

^DD(80.1,0,"ID",4)= D EN^DDIOL((" "_$$IDOPS^ICDID(+Y)),"","?0")

This will be exported in the combined build CTD UTIL 1.0, containing ICPT*6.0*19, ICD*18.0*12 and LEX*2.0*30.

4406 ID Nodes in DRG file (#80.2)

```
  CUSTODIAL PACKAGE: VA FILEMAN
SUBSCRIBING PACKAGE: DRG GROUPER
              USAGE: Private             ENTERED: APR 22,2004
             STATUS: Active              EXPIRES:
           DURATION: Till Otherwise Agr  VERSION:
               FILE: 80.2                   ROOT: DD(80.2
        DESCRIPTION:                        TYPE: File
```

The Code Text Descriptors project modifies the identifier on the INACTIVE (#15) field of the DRG file (80.2).

The new identifier makes a function call into $$IDDGS^ICDID to return versioned data for both the DESCRIPTION and the status in the INACTIVE field. The function has only one input parameter: the Internal Entry Number for file #80.2. Routine ICDID will also look to see if the package namespaced variable ICDVDT is in the environment. ICDVDT is a versioning date. If ICDVDT is not found in the environment (not supplied) then TODAY will be used and the DESCRIPTION and INACTIVE fields for TODAY will be displayed. If the variable ICDVDT is found in the environment, and is a date other than TODAY, then the appropriate DESCRIPTION and INACTIVE fields will be displayed for the date.

The identifiers will be changed to:

^DD(80.2,0,"ID",15)= D EN^DDIOL((" "_$$IDDGS^ICDID(+Y)),"","?0")

This will be exported in the combined build CTD UTIL 1.0, containing ICPT*6.0*19, ICD*18.0*12 and LEX*2.0*30.

5415 ICD Diagnosis File 80 Identifier Update

```
  CUSTODIAL PACKAGE: VA FILEMAN
SUBSCRIBING PACKAGE: DRG GROUPER
                      Patch ICD*18.0*40 deletes the DIAGNOSIS field #3 from
                      the ICD Diagnosis file #80.  The DRG Grouper package
                      needs permission to delete the file identifiers
                      associated with this field and then set a replacement
                      identifier in the data dictionary.

                      This agreement is one time only and expires with the
                      installation of ICD*18.0*40.

              USAGE: Private             ENTERED: MAR 19,2009
             STATUS: Active              EXPIRES:
```

```
         DURATION: Till Otherwise Agr  VERSION:
             FILE: 0                      ROOT: DD(80,
      DESCRIPTION:                        TYPE: File
    ^DD(80,0,'ID')
      Direct Kill of the following DD node in the Post-Install:

        ^DD(80,0,"ID",3)=
          D EN^DDIOL((" "_$$IDDXS^ICDID(+Y)),"","?0")

      Direct Set of the following DD node in the Post-Install:

        ^DD(80,0,"ID",8)=
          D EN^DDIOL((" "_$$IDDXF^ICDID(+Y)),"","?0")

      NOTE:  Field #8 is the ICD EXPANDED field and is used to distinguish
      between national codes and VA codes.  The identifiers are being moved
      from the deleted fields to a static field.
```

5416 ICD Procedure File 80.1 Identifier Update

```
  CUSTODIAL PACKAGE: VA FILEMAN
SUBSCRIBING PACKAGE: DRG GROUPER
                     Patch ICD*18.0*40 deletes the OPERATION/PROCEDURE
                     field #4 from the ICD Procedure file #80.1.  The DRG
                     Grouper package needs permission to delete the file
                     identifiers associated with this field and then set a
                     replacement identifier in the Data Dictionary.

                     This agreement is one time only and expires with the
                     installation of ICD*18.0*40.

           USAGE: Private             ENTERED: MAR 19,2009
          STATUS: Active              EXPIRES:
        DURATION: Till Otherwise Agr  VERSION:
            FILE: 0                      ROOT: DD(80.1,
      DESCRIPTION:                        TYPE: File
    ^DD(80.1,0,'ID')
      Direct Kill of the following DD node in the Post-Install:

        ^DD(80.1,0,"ID",4)=
          D EN^DDIOL((" "_$$IDOPS^ICDID(+Y)),"","?0")

      Direct Set of the following DD node in the Post-Install:

        ^DD(80.1,0,"ID",8)=
          D EN^DDIOL((" "_$$IDOPF^ICDID(+Y)),"","?0")

      NOTE:  Field #8 is the ICD EXPANDED field and is used to distinguish
      between national codes and VA codes.  The identifiers are being moved
      from the deleted fields to a static field.
```

5821 Census Date

```
  CUSTODIAL PACKAGE: REGISTRATION
SUBSCRIBING PACKAGE: DRG GROUPER
                     The ICD DRG Grouper needs to access the Census Date
                     to compute the Effective Date.

           USAGE: Private             ENTERED: JUL  1,2012
```

```
            STATUS: Pending              EXPIRES:
          DURATION: Till Otherwise Agr   VERSION:
              FILE: 45.86                   ROOT: DG(45.86,
       DESCRIPTION:                         TYPE: File
  GLOBAL REFERENCE:
    ^DG(45.86,D0,0)
      .01      DATE                  0;1      Direct Global Read & w/Fileman
       This field contains the census date as established by VACO.
```

5822 Census/Discharge/Movement/Surgery Dates

```
  CUSTODIAL PACKAGE: REGISTRATION
SUBSCRIBING PACKAGE: DRG GROUPER
                     The ICD DRG Grouper needs to access the Census Date,
                     the Movement Date, the Surgery Date, and the
                     Discharge date to compute the Effective Date.

             USAGE: Private              ENTERED: JUL  1,2012
            STATUS: Pending              EXPIRES:
          DURATION: Till Otherwise Agr   VERSION:
              FILE: 45                      ROOT: DGPT(
       DESCRIPTION:                         TYPE: File
  GLOBAL REFERENCE:
    ^DGPT(D0,0)
      13       CENSUS DATE           0;13     Direct Global Read & w/Fileman
       This is a pointer to the PTF CENSUS DATE file #45.86

  GLOBAL REFERENCE:
    ^DGPT(D0,70)
```

5846 Clear ScreenMan Help Area

```
  CUSTODIAL PACKAGE: VA FILEMAN
SUBSCRIBING PACKAGE: DRG GROUPER
                     The ICD Grouper package need to call CLRMSG^DDS
                     inside of its special lookup routines (ICDEXLK*) to
                     allow the special lookup routines to operate inside
                     of ScreenMan.  After displaying the selection list
                     and asking for user response, the special lookup
                     needs to clear the Help Area.

             USAGE: Controlled Subscri  ENTERED: SEP 28,2012
            STATUS: Pending             EXPIRES:
          DURATION: Till Otherwise Agr  VERSION:
       DESCRIPTION:                     TYPE: Routine

     ROUTINE: DDS
   COMPONENT:  CLRMSG
              This API clears the contents of the Help area.
```

5847 Write to ScreenMan Help Area

```
  CUSTODIAL PACKAGE: VA FILEMAN
SUBSCRIBING PACKAGE: DRG GROUPER
                     The ICD Grouper package need to call HLP^DDSMSG
                     inside of its special lookup routines (ICDEXLK*) to
                     allow the special lookup routines to operate inside
                     of ScreenMan.  This API allows the display of the
```

```
                    selection list and prompting for user response in the
                    Help Area.

            USAGE: Controlled Subscri  ENTERED: SEP 28,2012
           STATUS: Pending             EXPIRES:
         DURATION: Till Otherwise Agr  VERSION:
      DESCRIPTION:                        TYPE: Routine

    ROUTINE: DDSMSG
  COMPONENT:  HLP
              This API places text in ScreenMan's Help area.
```

18.3 ICD as a Custodian

48 ^ICD Weight

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: MENTAL HEALTH
            USAGE: Private             ENTERED: JUL 25,1990
           STATUS: Active              EXPIRES:
         DURATION: Till Otherwise Agr  VERSION:
             FILE: 80.2                   ROOT: ICD(
      DESCRIPTION:                        TYPE: File
    ^ICD(D0,0)
      2        WEIGHT              0;2      Read w/Fileman
                                            Used for lookups.
```

280 ^ICD9(Code

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: HOSPITAL BASED HOME CARE
            USAGE: Private             ENTERED: SEP 13,1993
           STATUS: Active              EXPIRES:
         DURATION: Till Otherwise Agr  VERSION:
             FILE: 80                     ROOT: ICD9(
      DESCRIPTION:                        TYPE: File
  We are requesting that a sharing agreement be established between the
  Hospital Based Home Care software and the Global ^ICD9( for the following
  fields.

  FIELD             .01      (node: 0, piece: 1) TYPE OF ACCESS    READ
  GLOBAL            ^ICD9(
```

365 ^ICD9(Code

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: CLINICAL MONITORING SYSTEM
            USAGE: Private             ENTERED: MAR  3,1994
           STATUS: Active              EXPIRES:
         DURATION: Till Otherwise Agr  VERSION:
             FILE: 80                     ROOT: ICD9(
      DESCRIPTION:                        TYPE: File
  Read access to the following fields:

   FIELDS:
```

```
   ELEMENT                          FILE
       DD LEVEL                     FIELD #                  DD NUMBER

   CODE NUMBER                      ICD DIAGNOSIS
       1                   0       .01                       80
```

370 ^ICD DRG Number/Description

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: INTEGRATED BILLING
                     DSS EXTRACTS
              USAGE: Private              ENTERED: MAR  9,1994
             STATUS: Active               EXPIRES:
           DURATION:                      VERSION:
               FILE: 80.2                    ROOT: ICD(
        DESCRIPTION:                         TYPE: File
    Request to store pointers to the DRG (#80.2) file from
       Integrated Billing.  The pointers are needed to retrieve
       data from the file at the time that claims are generated.

       Request to directly reference the following fields in the DRG (#80.2)
       file:

       Field Name (#)         Location   Reason
       NAME (.01)               0;1      Print and display the DRG number
       DESCRIPTION (1)    ^ICD(ien,1,1,0)  Print and display the DRG name
```

371 ICDDRG

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: INTEGRATED BILLING
                     MENTAL HEALTH
              USAGE: Controlled Subscri  ENTERED: MAR  9,1994
             STATUS: Active               EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE:                         ROOT:
        DESCRIPTION:                         TYPE: Routine

     ROUTINE: ICDDRG
   COMPONENT:  ICDDRG
   VARIABLES:  ICDEXP  Type: Input
                                Did patient expire during episode?
               ICDTRS  Type: Input
                                Patient transfer to acute facility?
               ICDDMS  Type: Input
                                Patient have irregular discharge?
               ICDDX(1,2,  Type: Input
                                Set of pointers (X) to diagnosis codes in
                                file #80.
               ICDPRC(1,2  Type: Input
                                Set of pointers (X) to procedures in file
                                #80.1.
               SEX  Type: Input
                                Patient gender (M-Male F-Female)|
               ICDDRG  Type: Output
                                Pointer to assigned DRG in file #80.2.
```

The routine call is made to calculate interim DRGs to determine the expected length for a visit.

582 ^ICD9(Code

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: ICR - IMMUNOLOGY CASE REGISTRY
              USAGE: Private              ENTERED: APR 21,2003
             STATUS: Active               EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE: 80                      ROOT: ICD9(
        DESCRIPTION:                         TYPE: File
     ^ICD9(D0,0)
       .01      CODE NUMBER          0;1      Read w/Fileman
```

1586 ^ICM Name

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: AUTOMATED INFO COLLECTION SYS
                     PCE PATIENT CARE ENCOUNTER
              USAGE: Controlled Subscri  ENTERED: AUG  8,1996
             STATUS: Active               EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE: 80.3                    ROOT: ICM(
        DESCRIPTION:                         TYPE: File
   This will enable reads both directly and through FileMan the code name in
   the MAJOR DIAGNOSTIC CATEGORY file (#80.3)

     ^ICM(D0,0)
       .01      NAME                 0;1      Direct Global Read & w
```

2184 PXRM Application Group

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: CLINICAL REMINDERS
              USAGE: Private              ENTERED: OCT 15,1997
             STATUS: Active               EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE:                         ROOT:
        DESCRIPTION:                         TYPE: Other
   Clinical Reminders use the application group PXRS for screening taxonomy
   selections. The following files need to belong to this application group:
   File 80 - ICD DIAGNOSIS, File 80.1 - ICD OPERATION/PROCEDURE File 81 - CPT
```

2435 ^ICD9(^ICD9(0)

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: CLINICAL REMINDERS
              USAGE: Private              ENTERED: JUN 19,1998
             STATUS: Active               EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE: 80                      ROOT: ICD9
        DESCRIPTION:                         TYPE: File
   Clinical Reminders needs to be able to determine when a new version of
   file 80 has been installed in order to keep its expanded taxonomy cache
   current. In order to do this we would like to do a direct read of pieces 3
   and 4 of the file header, ^ICD9(0).
```

^ICD9(0)

2435 ^ICD0(^ICD0(0)

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: CLINICAL REMINDERS
              USAGE: Private              ENTERED: JUN 19,1998
             STATUS: Active               EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE: 80.1                    ROOT: ICD0
        DESCRIPTION:                         TYPE: File
   Clinical Reminders needs to be able to determine when a new version of
   file 80.1 has been installed in order to keep its expanded taxonomy cache
   current. In order to do this we would like to do a direct read of pieces 3
   and 4 of the file header, ^ICD0(0).
     ^ICD0(0)
```

3990 ICDCODE Legacy APIs

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE:
              USAGE: Supported            ENTERED: MAR 12,2003
             STATUS: Other                EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE:                         ROOT:
        DESCRIPTION:                         TYPE: Routine
   This agreement contains the references to routine ICDCODE for the
   supported APIs.  These entry points will retrieve ICD code related data.

   All entry points will return
     -1^error description in an error condition.  in an error condition.


     ROUTINE: ICDCODE
   COMPONENT:  $$ICDDX(CODE,CDT,DFN,SRC)
   VARIABLES:  CODE  Type: Input
                                  ICD Diagnosis Code, IEN or .01 format
                                  (Required)

               CDT  Type: Input
                                  Code Date to check. (Optional)

                                      If CDT < 10/1/1978, use 10/1/1978.
                                      If CDT > DT, validate with newest
                                          effective dates
                                      If CDT is year only, use first of the
                                          year
                                      If CDT is year and month only, use
                                          first of the month
                                      Default = Today (FileMan format)

               DFN  Type: Input
                                  This variable is not used and not
                                  supported at this time.

               SRC  Type: Input
```

```
                              This is the Source Flag. (Optional)

                                0 = exclude local VA codes, use
                                    national codes only (default)
                                1 = include local VA codes and national
                                    codes

          $$ICDDX  Type: Output
                              19 piece "^" delimited string containing
                              the following information:

                              Piece    Description
                              -----    -----------
                                1      IEN in ^ICD9(
                                2      ICD DX Code (#.01)
                                3      Identifiers (#2)
                                4      Versioned Dx Short Name (#67)
                                5      Unacceptable as Principal Dx
                                       (#101)
                                6      Versioned Major Dx Category
                                       (#72)
                                7      MDC13 (#5.5)
                                8      Do not use, see piece 19 for CC
                                9      ICD Expanded (#8)
                               10      Status (#66)
                               11      Sex (#9.5)
                               12      Inactive Date (#66)
                               13      MDC24 (#5.7)
                               14      MDC25 (#5.9)
                               15      Age Low (#14)
                               16      Age High (#15)
                               17      Activation Date (#66)
                               18      Message - Notice of Textual
                                       Inaccuracy
                               19      Versioned Complication
                                       Comorbidity (CC) (#103)

                                         or

                                       -1^Error Description

             Extrinsic function that returns basic information for an
             ICD Diagnosis Code.

COMPONENT:   $$ICDOP(CODE,CDT,DFN,SRC)
VARIABLES:   CODE  Type: Input
                              ICD Procedure Code, IEN or .01 format
                              (Required)

          CDT  Type: Input
                              Code Date to check. (Optional) (Fileman
                              Format)

                                  If CDT < 10/1/1978, use 10/1/1978.
                                  If CDT > DT, validate with newest
                                    effective dates
                                  If CDT is year only, use first of the
```

```
                                  year
                                If CDT is year and month only, use
                                   first of the month
                                Default = Today (FileMan format)

          DFN  Type: Input
                               This variable is not used and not
                               supported at this time.

          SRC  Type: Input
                               This is the Source Flag. (Optional)

                                 0 = exclude local VA codes, use
                                     national codes only (default)
                                 1 = include local VA codes and national
                                     codes

          $$ICDOP  Type: Output
                               14 piece "^" delimited string containing
                               the following information:

                               Piece    Description
                               -----    -----------
                                 1      IEN in ^ICD9(
                                 2      ICD Procedure Code (#.01)
                                 3      Identifiers (#2)
                                 4      MDC24 (#5)
                                 5      Versioned Oper/Proc (#67)
                                 6      <null>
                                 7      <null>
                                 8      <null>
                                 9      ICD Expanded (#8)
                                10      Status (#66)
                                11      Use with Sex (#9.5)
                                12      Inactive Date (#66)
                                13      Activation Date (#66)
                                14      Message - Notice of Textual
                                        Inaccuracy

                                          or

                                        -1^Error Description

             Extrinsic function that returns basic information for an
             ICD Operation/Procedure Code.

COMPONENT:   $$ICDD(CODE,'OUTARR',CDT)
VARIABLES:   CODE  Type: Input
                               ICD Diagnosis or Procedure Code
                               (Required)

          OUTARR  Type: Both
                               Array to store description
                                   name of array - e.g. "ABC" or
                               "ABC("TEST")"
                                   or temp array.      Default =
                               ^TMP("ICDD",$J)
```

```
                                The calling routine is responsible
                              for killing
                                ^TMP("ICDD",$J) after the call, if
                              used.

                              On return, the array contains
                              corresponding lines of text of the code's
                              versioned description (field 68).
                              OUTARR(1) = 1st line of versioned
                              description (field #68) OUTARR(last) =
                              last line of versioned description (field
                              #68) OUTARR(last+1) = blank line
                              OUTARR(last+2) = NOTICE OF TEXTUAL
                              INACCURACY
                               where last+2 is the value returned by
                              $$ICDD.

            CDT  Type: Input
                              Code Date to check - not used currently,
                                     Included in anticipation of
                              future need.
                                  Default = Today (FileMan format)
                                  If CDT < 10/1/1978, use 10/1/1978.
                                  If CDT > DT, use most recent
                                     description
                                  If CDT is year only, use first of the
                                     year
                                  If CDT is year and month only, use
                                     first of the month

            $$ICDD  Type: Output
                              Contains number of lines (number of
                              subscripts) in the description (array)

               Extrinsic function that returns the full description of a
               code, from the "1" node (field 10) of the ICD9 file or the
               ICD0 file.

COMPONENT:  $$CODEN(CODE,FILE)
VARIABLES:  CODE  Type: Input
                              ICD Code  REQUIRED
            FILE  Type: Input
                              File Number in which to check for ICD
                              code
                                    80 for ICD Diagnosis file
                                  80.1 for ICD Operation/Procedure file
            $$CODEN  Type: Output
                              String, containing the following
                              information in the following "~" pieces:

                              Piece    Description
                              =====    ===========
                                1      ien of the ICD Code
                                2      "^ICD9(" if FILE=80; "^ICD0(" if
                              FILE=80.1

               Extrinsic function that returns the internal entry number
```

```
                and the global root of an ICD Code.

   COMPONENT:  $$CODEC(CODE)
   VARIABLES:  CODE  Type: Input
                                Internal entry number of an ICD Code.
               $$CODEC  Type: Output
                                ICD Code
                  Extrinsic function that returns the ICD Code of an ien.
```

3991 ICDAPIU Legacy APIs

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE:
             USAGE: Supported           ENTERED: MAR 12,2003
            STATUS: Active              EXPIRES:
          DURATION: Till Otherwise Agr  VERSION:
              FILE:                        ROOT:
       DESCRIPTION:                        TYPE: Routine
   This contains the references to routine ICDAPIU for the supported APIs to
   be released with v.20.0 of ICD.

   These include extrinsic functions for retrieving Code History, performing
   Status checks, retrieving Next/Previous Codes, retrieving Dates based on
   the Business Rules, and retrieving a notice of a code's textual
   inaccuracy.

     ROUTINE: ICDAPIU
   COMPONENT:  $$STATCHK(CODE,CDT)
   VARIABLES:  CODE  Type: Input
                                ICD Code  REQUIRED
               CDT  Type: Input
                                Code Date to check, Default = Today
                                (FileMan format)
                                    If CDT < 10/1/1978, use 10/1/1978.
                                    If CDT > DT, validate with newest
                                        In/Activation Dates
                                    If CDT is year only, use first of the
                                        year
                                    If CDT is year and month only, use
                                        first of the month
               $$STATCHK  Type: Output
                                String, containing the following
                                information in the following "^" pieces:

                                Piece    Description
                                =====    ===========
                                  1      STATUS where 1:active;
                                         0:inactive
                                  2      IEN of code, -1 if not found

                  Extrinsic function that returns the Status of an ICD Code.

   COMPONENT:  $$NEXT(CODE)
   VARIABLES:  CODE  Type: Input
                                ICD Code  REQUIRED
               $$NEXT  Type: Output
```

```
                              The Next ICD Code, Null if there is none.
            Extrinsic function that returns the Next ICD Code (active
            or inactive)

COMPONENT:  $$PREV(CODE)
VARIABLES:  CODE  Type: Input
                              ICD Code  REQUIRED
            $$PREV  Type: Output
                              The Previous ICD Code, Null if there is
                              none.

            Extrinsic function that returns the Previous ICD Code
            (active or inactive)

COMPONENT:  $$HIST(CODE,ARY)
VARIABLES:  CODE  Type: Input
                              ICD Code  REQUIRED
            .ARY  Type: Both
                              Array, passed by Reference

                              ARY (which was passed by reference) is
                              returned as follows: ARY(0) = number of
                              history entries, -1 if error ARY(date) =
                              STATUS where 1:active; 0:inactive
                                            'date' is in FileMan
                              format ARY("IEN") = Internal Entry Number
                              of ICD Code
            $$HIST  Type: Output
                              The number of activation history entries
                              are returned, -1 if error

            Extrinsic function that returns the activation history of
            an ICD Code.

COMPONENT:  $$DTBR(CDT,CS)
VARIABLES:  CDT  Type: Input
                              Code Date to check, Default = Today
                              (FileMan format)
                                  If CDT is year only, use first of the
                              year
                                  If CDT is year and month only, use
                              first of the month
            CS  Type: Input
                              Code System (0:ICD, 1:CPT/HCPCS, 2:DRG,
                              Default=0)
            $$DTBR  Type: Output
                              If CDT < 10/1/1978 and CS=0, return
                              10/1/1978 If CDT <  1/1/1989 and CS=1,
                              return  1/1/1989 If CDT < 10/1/1982 and
                              CS=2, return 10/1/1982 Otherwise, return
                              CDT
            Extrinsic function that returns a date after applying
            several Business Rules, depending on the Coding System.

COMPONENT:  $$MSG(CDT,CS)
VARIABLES:  CDT  Type: Input
                              Code Date to check, Default = Today
```

```
                                (FileMan format)
                                    If CDT is year only, use first of the
                                year
                                    If CDT is year and month only, use
                                first of the month
            CS  Type: Input
                                Code System (0:ICD, 1:CPT/HCPCS, 2:DRG,
                                3:LEX, Default=0)
            $$MSG  Type: Output
                                A warning stating: "CODE TEXT MAY BE
                                INACCURATE"

               Extrinsic function that returns a message to inform someone
               that the code text may be inaccurate.

   COMPONENT:  PERIOD(CODE,ARY)
   VARIABLES:  COD  Type: Input
                                ICD Code     REQUIRED
               ARY  Type: Output
                                Array, passed by Reference     REQUIRED
               Function that returns Activation/Inactivation Period in ARY

               ARY(0) = IEN  (or, -1 if error) ARY(Act_date) =
               Inactivation Date^Versioned Short Name Text (field #67)
```

4052 ICDGTDRG

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: FEE BASIS
                     INTEGRATED BILLING
                     REGISTRATION
              USAGE: Supported           ENTERED: JUL 14,2003
             STATUS: Active              EXPIRES:
           DURATION: Till Otherwise Agr  VERSION:
               FILE:                        ROOT:
        DESCRIPTION:                        TYPE: Routine

     ROUTINE: ICDGTDRG
   COMPONENT:  $$DRG(CODE,EDT)
   VARIABLES:  CODE  Type: Input
                                REQUIRED - DGN code, ien or .01 value
               EDT  Type: Input
                                OPTIONAL - Effective date, default =
                                today (Fileman format)
               $$DRG  Type: Output
                                If DRG code DOES exist in the database
                                then the function returns a "^" delimited
                                string with the following pieces:

                                  1  DRG name (field #.01)
                                  2  Weight (field #2)
                                  3  Low Trim (days) (field #3)
                                  4  High Trim (days) (field #4)
                                  5  MDC (field #5)
                                  6  Surgery Flag (field #.06)
                                  7  <null>
```

```
                                  8  Avg Length of Stay (days) (field 10)
                                  9  Local Low Trim Days (field #11)
                                 10  Local High Trim Days (field #12)
                                 11  <null>
                                 12  Local Breakeven (field #13)
                                 13  Activation Date (.01 field of the 66
                                       multiple)
                                 14  Status (.03 field of the 66
                                       multiple)
                                 15  Inactivation Date (.01 field of the
                                       66 multiple)
                                 16  Effective date (.01 field of the 66
                                       multiple)
                                 17  Internal Entry Number (IEN)
                                 18  Effective date of CSV (.01 field of
                                       the 66 multiple)

                               If DRG code DOES NOT exist in the
                               database then the function returns a "^"
                               delimited string with the following
                               pieces:

                                 1  -1
                                 2  NO SUCH ENTRY
                                14  Status 0=inactive

          This DBIA contains a supported DRG API call that can be
          used to access data contained in DRG file (# 80.2). Returns
          a string of information from the DRG file (#80.2) for a
          given DRG code and effective date.

COMPONENT:  $$GETDRG(CODE,DGNDT,FILE)
VARIABLES:  CODE  Type: Input
                               REQUIRED - IEN number of the #80 or #80.1
                               file
            DGNDT  Type: Input
                               OPTIONAL - Effective date, default =
                               today (Fileman format)
            FILE  Type: Input
                               REQUIRED - file to access - 9:ICD9 (#80),
                               0:ICD0 (#80.1)
            $$GETDRG  Type: Output
                               If the code exists in the database, then
                               the function returns a string with ";"
                               delimiters:

                               DRG(s) associated with the code
                               (delimited by "^") - can be 1+ (piece
                               1);Effective date (piece 2);status flag
                               (piece 3)

                               If the code DOES NOT exist in the
                               database then the function returns:

                               Piece #1 : -1 Piece #2 : error message
                               Piece #3 : Status = 0 = Inactive
```

```
            This DBIA contains a supported DRG API call that can be
            used to access data contained in the ICD DIAGNOSIS CODE
            file (#80) or the ICD OPERATION/PROCEDURE CODE file
            (#80.1). It returns a string of information from the file
            for a given ICD DIAGNOSIS or OPERATION/PROCEDURE CODE and
            effective date.

COMPONENT:  $$GETDATE(PATNUM)
VARIABLES:  PATNUM  Type: Input
                              REQUIRED - ien or .01 value for PTF file
                              (#45)
            $$GETDATE  Type: Output
                              The function returns a Fileman-formatted
                              date of the proper date to be used as the
                              effective date.  This date can be either
                              the census, discharge, surgery, or
                              movement date.  If all previous dates are
                              undefined, today's date is returned.

            This DBIA contains a supported DRG API call that can be
            used to access data in the PTF file (#45).  It returns the
            proper effective date for a patient to use in accessing
            Code Set Versioned data.

COMPONENT:  $$ISVALID
VARIABLES:  CODE  Type: Input
                              REQUIRED  - IEN number of the #80 or
                              #80.1 file entry
            DGNDT  Type: Input
                              OPTIONAL -Effective date, default = today
                              (Fileman format)
            FILE  Type: Input
                              REQUIRED - file to access - 9:ICD9(#80),
                              0:ICD0(#80.1)
            $$ISVALID  Type: Output
                              Returns 1 if the code is active/valid for
                              the effective date or 0 if it is
                              undefined or inactive.

            This DBA contains a supported DRG API call that can be used
            to determine if an ICD DIAGNOSIS CODE (#80) or ICD
            OPERATION/ PROCEDURE CODE (#80.1)is active for a given
            effective date.  This API is designed for use in DIC("S")
            Fileman calls.

COMPONENT:  $$DRGD(CODE,ARRAY,DFN,DATE)
VARIABLES:  CODE  Type: Input
                              This is either a DRG Code or an Internal
                              Entry Number (IEN) in the DRG file
                              (#80.2)
            ARRAY  Type: Both
                              An array name in which to store the
                              returned versioned description.  If no
                              name is provided, the default name will
                              be ^TMP("DRGD",$J,.  The calling routine
                              is responsible for killing
                              ^TMP("DRGD",$J) after the call, if used.
```

```
                              On return, the array contains
                              corresponding lines of text of the code's
                              versioned description (field 68)

                               ARRAY(1) = 1st line of description
                               ARRAY(last) = last line of description
                               ARRAY(last+1) = blank line
                               ARRAY(last+2) = NOTICE OF TEXTUAL
                              INACCURACY
                                 where last+2 is the value returned by
                              $$DRGD.
            DATE  Type: Input
                              This is a Fileman compliant date.  Time
                              is ignored.  If the date is not supplied,
                              then today's date is used.  The DRG
                              description (long text) will be
                              appropriate for that date.  If no text is
                              found that corresponds with the date
                              provided, the oldest possible text will
                              be returned and an message will be
                              returned that the "text may be
                              inaccurate".
            DFN  Type: Input
                              This is a pointer to the Patient File #2
                              (for future use)
               Extrinsic function that returns the full versioned
               description of a Diagnostic Related Group (DRG) code, from
               the 68 node (field 68) of the DRG file.
```

4126 ICD CODE UPDATE EVENT Protocol

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE:
             USAGE: Supported           ENTERED: JUL 21,2003
            STATUS: Active              EXPIRES:
          DURATION: Till Otherwise Agr  VERSION:
              FILE:                        ROOT:
       DESCRIPTION:                        TYPE: Other
   attached package protocols will be notified of a code set update.
   Packages may attach protocols using KIDS' "USE AS LINK FOR MENU ITEMS"

     ROUTINE:
   COMPONENT:  ICD CODE UPDATE EVENT
   VARIABLES:     Notify applications that ICD codes have been updated.
```

4485 ^ICD9(Lexicon

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: LEXICON UTILITY
             USAGE: Private             ENTERED: JUL 28,2004
            STATUS: Active              EXPIRES:
          DURATION: Till Otherwise Agr  VERSION:
              FILE: 80                     ROOT: ICD9(
       DESCRIPTION:                        TYPE: File
   Lexicon Utility has all privileges as though it were the custodial
```

```
   package.
```

4486 ^ICD0 Lexicon

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: LEXICON UTILITY
              USAGE: Private              ENTERED: JUL 28,2004
             STATUS: Active               EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE: 80.1                    ROOT: ICD0(
        DESCRIPTION:                         TYPE: File
   Lexicon Utility has all privileges as though it were the custodial
   package.
```

4487 ^ICD(Lexicon

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: LEXICON UTILITY
              USAGE: Private              ENTERED: JUL 28,2004
             STATUS: Active               EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE: 80.2                    ROOT: ICD(
        DESCRIPTION:                         TYPE: File
   Lexicon Utility has all privileges as though it were the custodial
   package.
```

4488 ^ICM(Lexicon

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: LEXICON UTILITY
              USAGE: Private              ENTERED: JUL 28,2004
             STATUS: Active               EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE: 80.3                    ROOT: ICM(
        DESCRIPTION:                         TYPE: File
   Lexicon Utility has all privileges as though it were the custodial
   package.
```

5028 ^ICD9(Problem List

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: PROBLEM LIST
              USAGE: Controlled Subscri  ENTERED: AUG 21,2007
             STATUS: Pending              EXPIRES:
           DURATION: Till Otherwise Agr   VERSION:
               FILE: 80                      ROOT: ICD9('AST'
        DESCRIPTION:                         TYPE: File
     This agreement will allow Problem List to determine if a particular ICD9
   code has a new description change.  This agreement is to view the cross
   reference "AST" to determine if a new description exits.
     ^ICD9(D0,67,D1,0)
       .01      VERSION DATE         0;1      Read w/Fileman
                                               This is the date the
                                              diagnosis text was first used.
       This agreement will use the "AST" cross reference from file #80 -
```

```
^ ICD9("AST",(CODE_" "),EFF,IEN1,IEN2)
```

5388 ^ICD9(Interim ICR

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE:
             USAGE: Supported           ENTERED: MAR 16,2009
            STATUS: Active              EXPIRES:
          DURATION: Till Otherwise Agr  VERSION:
              FILE: 80                     ROOT: ICD9(
       DESCRIPTION:                        TYPE: File
   Applications may conduct Fileman lookups of ICD Diagnosis file #80
   provided the 0 (zero) node is not returned as part of the output from the
   lookup.  Applications may also point to the ICD Diagnosis file #80.  This
   agreement provides very limited access to file 80, primarily the .01 field
   and selected cross-references.  Additional access to file 80 is given
   through the use of APIs in routines ICDCODE and ICDAPIU.

     ^ICD9(D0,0)
       .01      CODE NUMBER          0;1      Direct Global Read & w
     ^ICD9('AB',
        Direct global read of the "AB" cross reference.

     ^ICD9('BA',
        Direct global read of the "BA" cross reference.

     ^ICD9('D',
        Direct global read of the "D" cross reference.

     ^ICD9('AST',
        Direct global read of the "AST" cross reference.

     ^ICD9('ACT'
        Direct global read of the "ACT" cross reference.
```

5404 ^ICD0(Interim ICR

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE:
             USAGE: Supported           ENTERED: MAR 17,2009
            STATUS: Active              EXPIRES:
          DURATION: Till Otherwise Agr  VERSION:
              FILE: 80.1                   ROOT: ICD0(
       DESCRIPTION:                        TYPE: File
   Applications may conduct Fileman lookups of ICD Operation Procedure file
   #80.1 provided the 0 (zero) node is not returned as part of the output
   from the lookup.  Applications may also point to the ICD
   Operation/Procedure file #80.1.  This agreement provides very limited
   access to file 80.1, primarily the .01 field and selected
   cross-references.  Additional access to file 80.1 is given through the use
   of APIs in routines ICDCODE and ICDAPIU.

     ^ICD0(D0,0)
       .01      CODE NUMBER          0;1      Direct Global Read & w
     ^ICD0('BA',
```

```
   Direct global read of the "BA" cross reference.

  ^ICD0('ACT'
   Direct global read of the "ACT" cross reference.
```

5699 ICDXCODE Wrapper (2 file solution)

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE:
             USAGE: Supported           ENTERED: AUG  2,2011
            STATUS: Pending             EXPIRES: APR  1,2016
          DURATION:                     VERSION:
       DESCRIPTION:                        TYPE: Routine
  Routine ICDXCODE was developed to replace ICDCODE during the ICD-10
  project to navigate between the ICD-9 Diagnosis file 80 and the ICD-10
  Diagnosis file 8010 under the two file solution.  The two file solution
  had the ICD-9 codes and ICD-10 codes stored in two separate files.  This
  solution was abandoned in favor of the one file solution where both ICD-9
  and ICD-10 are stored in the same file (ICD Diagnosis file 80).  A one
  file solution of these APIs can be found in the routine ICDEX (ICD Data
  Extraction)  Routine ICDXCODE will be exported to support applications
  through the transition between the one and two file solutions.  It will be
  retired 18 months after the ICD-10 compliance date.


    ROUTINE: ICDXCODE
  COMPONENT:  $$ICDDATA(CSYS,CODE,DATE,FRMT)
              Extract general data of ICD Diagnosis and/or Procedures.

  VARIABLES:  Input     CSYS
                          Coding System (Required)

                          Code Type
                            Diagnosis:          DIAG
                            Procedure:          PROC

                          Code File
                            ICD-9 Diagnosis:    ICD9 or 80
                            ICD-10 Diagnosis:   ICD10DX or 8010
                            ICD-9 Procedure     ICD0 or 80.1
                            ICD-10 Procedure    ICD10PR or 8010.1

                          If Code Type is used (DIAG/PROC) then the input
                          parameter date will be used to determine if the
                          output is ICD-9 or ICD-10.  Dates before the
                          ICD-10 implementation will return ICD-9 data, and
                          dates on or after the ICD-10 implementation will
                          return ICD-10 data.

  VARIABLES:  Input     CODE
                          This is an ICD code or an Internal Entry Number
                          (IEN) or a Variable Pointer:
                             IEN;ICD9(
                             IEN;ICD0(
                             IEN;ICD10DX(
                             IEN;ICD10PR(
```

```
VARIABLES:  Input     DATE
                        Date in Fileman format.  If not supplied it
                        defaults to TODAY.  This date is normally the date
                        that service was provided to the patient (i.e.
                        visit date, movement date, or date of onset).

VARIABLES:  Input     FRMT
                        Format of the input CODE (optional)

                          "E" = external format
                          "I" = internal format (IEN)

                        If supplied, it must be consistent with the CODE
                        input parameter.

                        If a variable pointer is passed as CODE, then "I"
                        internal format is assumed.

VARIABLES:  Output    $$ICDXCODE
                        Diagnosis Code Passed:

                           A 19 piece caret (^) delimited string

                           1 IEN of code in file 80/8010
                           2 ICD-9/ICD-10 Diagnosis Code (#.01)
                           3 Id (#2)
                           4 Versioned Diagnosis Short Name (67 multiple)
                           5 Unacceptable as Principal Diagnosis (#101)
                           6 Major Dx Cat (#5)
                           7 MDC13 (5.5)
                           8 Compl/Comorb (#70)
                           9 ICD Expanded (#8) 1:Yes 0:No (ICD-9 only)
                          10 Status (66 multiple)
                          11 Sex (#9.5)
                          12 Inactive Date (66 multiple)
                          13 MDC24 (#5.7)
                          14 MDC25 (#5.9)
                          15 Age Low (#14)
                          16 Age High (#15)
                          17 Activation Date (.01 of 66 multiple)
                          18 Message (ICD-9 only)
                          19 Versioned Complication/Comorbidity (#103)

                        Procedure Code Passed:

                           A 14 piece caret (^) delimited string

                           1 IEN of code in file 80.1/8010.1
                           2 ICD-9/ICD-10 code (#.01)
                           3 Id (#2)
                           4 MDC24 (#5)
                           5 Versioned Oper/Proc (67 multiple)
                           6 <null>
                           7 <null>
                           8 <null>
                           9 ICD Expanded (#8) 1:Yes 0:No (ICD-9 only)
```

```
                        10 Status (66 multiple)
                        11 Use with Sex (#9.5)
                        12 Inactive Date (66 multiple)
                        13 Activation Date (66 multiple)
                        14 Message

                       or

                       -1^Error Description


COMPONENT:  $$ICDDESC(CSYS,CODE,DATE,OUTARR)
            This API returns the long description of either an ICD-9 or
            ICD-10 code.

VARIABLES:  Input     CSYS
                        Coding System (Required)

                        Code Type
                          Diagnosis:          DIAG
                          Procedure:          PROC

                        Code File
                          ICD-9 Diagnosis:    ICD9 or 80
                          ICD-10 Diagnosis:   ICD10DX or 8010
                          ICD-9 Procedure     ICD0 or 80.1
                          ICD-10 Procedure    ICD10PR or 8010.1

                        Code System
                          ICD-9 Diagnosis     1
                          ICD-10 Diagnosis    30
                          ICD-9 Procedure     2
                          ICD-10 Procedure    31

                        If Code Type is used (DIAG/PROC) then the input
                        parameter date will be used to determine if the
                        output is ICD-9 or ICD-10.  Dates before the
                        ICD-10 implementation date will return ICD-9 data,
                        and dates on or after the ICD-10 implementation
                        date will return ICD-10 data.

VARIABLES:  Input     CODE
                        This is an ICD code (Required)

                        It can be an ICD-9 Diagnosis Code, ICD-9 Procedure
                        code, an ICD-10 Diagnosis Code or an ICD-10
                        Procedure code.  The code should be consistent
                        with the Coding System (first input parameter)

VARIABLES:  Input     DATE
                        Date in Fileman format.  If not supplied it
                        defaults to TODAY.  This date is normally the date
                        that service was provided to the patient (i.e.
                        visit date, movement date, or date of onset). The
                        date is used to return the appropriate description
                        for the date service was provided.
```

```
VARIABLES:  Both      OUTARR
                        Input:  Array Name

                          e.g. "ARY", "ABC" or "ABC("TEST")"
                          Default = ^TMP("ICDD",$J)

                        If ^TMP("ICDD",$J) is used, the calling
                        application is responsible for killing the global
                        variable when no longer needed.

                        Output

                           @ARY(1) = Versioned Description (68 multiple)
                           @ARY(2) = Blank (ICD-9 only)
                           @ARY(3) = Message (ICD-9 only):
                                     CODE TEXT MAY BE INACCURATE

VARIABLES:  Output    $$ICDDESC
                        Number of lines in output array

COMPONENT:  $$HIST(SYS,CODE,.ARY)
            This API returns the effective dates and status from the
            code's status history.

VARIABLES:  Input     SYS
                        This is a pointer to the CODING SYSTEM file
                        80.4

                           1 = ICD-9-CM
                           2 = ICD-9-PCS
                          30 = ICD-10-CM
                          31 = ICD-10-PCS

VARIABLES:  Input     CODE
                        This is an ICD Code (IEN not allowed)

VARIABLES:  Both      .ARY
                        This is a local array name passed by reference
                        that will contain the output.

                          ARY(0)       =  Number of Activation History
                                          Entries or -1 on error

                          ARY(<date>) =  Status    where: 1 is Active
                          ARY("IEN")  =  <ien>

VARIABLES:  Output    $$HIST
                        This output variable mirrors ARY(0) if histories
                        are found or, -1 on error.

COMPONENT:  $$NEXT(SYS,CODE)
            This API returns the next code in a sequence.

VARIABLES:  Input     SYS
                        This is a pointer to the CODING SYSTEM file
                        80.4
```

```
                          1 = ICD-9-CM
                          2 = ICD-9-PCS
                         30 = ICD-10-CM
                         31 = ICD-10-PCS

VARIABLES:  Input     CODE
                        This is an ICD Code (IEN not allowed)

VARIABLES:  Output    $$NEXT
                        The Next ICD Code, or the first ICD code if CODE
                        is null or null if CODE is the last ICD code.

COMPONENT:  $$PREV(SYS,CODE)
            This API returns the previous ICD code in a sequence.

VARIABLES:  Input     SYS
                        This is a pointer to the CODING SYSTEM file
                        80.4

                          1 = ICD-9-CM
                          2 = ICD-9-PCS
                         30 = ICD-10-CM
                         31 = ICD-10-PCS

VARIABLES:  Input     CODE
                        This is an ICD Code (IEN not allowed)

VARIABLES:  Output    $$PREV
                        The Previous ICD Code, or the last ICD code if
                        CODE is null or null if CODE is the first ICD
                        code.

COMPONENT:  $$STATCHK(SYS,CODE,CDT)
            This API returns the status of a code and the code's Internal
            Entry Number (IEN).

VARIABLES:  Input     SYS
                        This is a pointer to the CODING SYSTEM file
                        80.4

                          1 = ICD-9-CM
                          2 = ICD-9-PCS
                         30 = ICD-10-CM
                         31 = ICD-10-PCS

VARIABLES:  Input     CODE
                        This is an ICD Code (IEN not allowed)

VARIABLES:  Input     CDT
                        This is a date in Fileman format that will be used
                        to determine the status of CODE.  (Optional,
                        default = TODAY)

VARIABLES:  Output    $$STATCHK
                        This is a 2-piece "^" delimited string containing
                        the code's status and the IEN if the code exists,
                        else -1.  The following are possible outputs:
```

```
                          1^IEN    Active Code
                          0^IEN    Inactive Code
                          0^-1     Code not Found

COMPONENT:  $$PERIOD(SYS,CODE,.ARY)
            This API returns activation periods (from/to dates).

VARIABLES:  Input     SYS
                        This is a pointer to the CODING SYSTEM file
                        80.4

                           1 = ICD-9-CM
                           2 = ICD-9-PCS
                          30 = ICD-10-CM
                          31 = ICD-10-PCS

VARIABLES:  Input     CODE
                        This is an ICD Code (IEN not allowed)

VARIABLES:  Both      .ARY
                        This is a local array name passed by reference
                        that will contain the output.

                        ARY(0) = IEN ^ Selectable ^ Error Message

                                 Where IEN = -1 if error
                                 Selectable = 0 for unselectable
                                 Error Message if applicable

                        ARY(Activation Date) = Inactivation Date^Short Name

                          Where the Short Name is versioned as follows:

                          Period is active    Short Description for the date
                                              the period became active

                          Period is inactive  Short Description for the date
                                              the period became inactive
```

5747 ICDEX ICD Data Extraction

```
    5747     NAME: ICD Data Extraction
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: LEXICON UTILITY
                     The LEXICON UTILITY has access to all APIs listed in
                     this ICR as if it were the Custodial Package.

                    ACCOUNTS RECEIVABLE
                     ACCOUNT RECEIVABLE (PRCA) package will use the
                     following APIs:

                       $$CODEC^ICDEX
                       $$CODECS^ICDEX
```

```
INTEGRATED BILLING
   INTEGRATED BILLING (IB) will use the following APIs:

     $$SYS^ICDEX
     $$CODEABA^ICDEX
     $$STATCHK^ICDEX
     $$ICDDX^ICDEX
     $$ICDOP^ICDEX
     $$LS^ICDEX

FEE BASIS
   FEE BASIS (FB) package will use the following APIs:

     $$GETDRG^ICDEX
     $$STATCHK^ICDEX
     $$CODEC^ICDEX
     $$CODEABA^ICDEX
     $$CODEN^ICDEX
     $$SD^ICDEX

PROSTHETICS
   PROSTHETICS (RMPR) will use the following APIs:

     $$SINFO^ICDEX
     $$CSI^ICDEX
     $$STATCHK^ICDEX
     $$ICDDX^ICDEX
     $$VLT^ICDEX
     $$LS^ICDEX
     $$CODEC^ICDEX

SCHEDULING
   SCHEDULING (SD) will use the following APIs:

     $$IMP^ICDEX
     $$CSI^ICDEX
     $$VER^ICDEX
     $$SYS^ICDEX
     $$LS^ICDEX
     $$ICDDX^ICDEX
     $$VLTD^ICDEX

REGISTRATION
   REGISTRATION (DG) will use the following APIs:

     $$CSI^ICDEX
     $$CODEC^ICDEX
     $$CODEN^ICDEX
     $$CODEABA^ICDEX
     $$LS^ICDEX
     $$NOT^ICDEX
     $$REQ^ICDEX
     $$SYS^ICDEX
     $$VLT^ICDEX
     $$SINFO^ICDEX
     $$CS^ICDEX
     $$ICDDX^ICDEX
```

```
     $$VST^ICDEX

CLINICAL REMINDERS
   CLINICAL REMINDERS (PXRM) will use the following
   APIs:

     $$CODEN^ICDEX
     $$CODEABA^ICDEX
     $$ICDDX^ICDEX
     $$ICDOP^ICDEX
     $$NEXT^ICDEX
     $$PREV^ICDEX
     $$IMP^ICDEX
     $$ROOT^ICDEX
     $$HDR^ICDEX
     $$CODEC^ICDEX
     $$CSI^ICDEX
     $$SINFO^ICDEX

PHARMACY BENEFITS MANAGEMENT
   PHARMACY BENEFITS MANAGEMENT (PSU) will use the
   following APIs:

     $$CSI^ICDEX
     $$ICDDX^ICDEX
     $$ICDOP^ICDEX

CLINICAL CASE REGISTRIES
   CLINICAL CASE REGISTRIES (ROR) will use the following
   APIs:

     $$CSI^ICDEX
     $$VSEX^ICDEX
     $$UPDX^ICDEX
     $$CODEC^ICDEX
     $$CODEABA^ICDEX
     $$VSTD^ICDEX
     $$VLTD^ICDEX
     $$VSTP^ICDEX
     $$VLTP^ICDEX
     $$FILE^ICDEX
     $$VLT^ICDEX
     $$VST^ICDEX
     $$CODEN^ICDEX
     $$ICDDX^ICDEX
     $$ICDOP^ICDEX
     $$SNAM^ICDEX

CLINICAL PROCEDURES
   CLINICAL PROCEDURES (MD) will use the following APIs:

     $$ICDDX^ICDEX
     $$CSI^ICDEX
     $$IMP^ICDEX
     $$SINFO^ICDEX

SPINAL CORD DYSFUNCTION
```

```
   SPINAL CORD DYSFUNCTION (SPN) package will use the
   following APIs:

     $$OBA^ICDEX
     $$CODEBA^ICDEX
     $$CSI^ICDEX
     $$CODEABA^ICDEX
     $$VLT^ICDEX
     $$VST^ICDEX

HOSPITAL BASED HOME CARE
   HOSPITAL-BASED HOME CARE (HBH) will use the following
   APIs:

     $$SYS^ICDEX
     $$CODEC^ICDEX
     $$VSTD^ICDEX
     $$SAI^ICDEX
     $$CSI^ICDEX

EVENT CAPTURE
   EVENT CAPTURE (EC) package will use the following
   APIs:

     $$SINFO^ICDEX
     $$ICDDX^ICDEX
     $$CODEN^ICDEX

AUTOMATED INFO COLLECTION SYS
   AUTOMATED INFO COLLECTION SYS (IBD) package will use
   the following APIs:

     $$SINFO^ICDEX

LAB SERVICE
   LAB SERVICES (LR) will use the following APIs:

     $$CODEC^ICDEX
     $$ICDDX^ICDEX
     $$ICDOP^ICDEX
     $$IMP^ICDEX
     $$SINFO^ICDEX
     $$CSI^ICDEX
     $$SD^ICDEX
     $$SNAM^ICDEX
     $$CODEN^ICDEX

QUASAR
   QUASAR (ACKQ) will use the following APIs:

     $$CODEC^ICDEX
     $$CSI^ICDEX
     $$CODEN^ICDEX

EMERGENCY DEPARTMENT
   EMERGENCY DEPARTMENT (EDP) package will use the
   following APIs:
```

```
          $$ICDDX^ICDEX
          $$ICDOP^ICDEX
          $$CODEC^ICDEX

       PROBLEM LIST
          PROBLEM LIST (GMPL) will use the following APIs:

          $$CODEC^ICDEX
          $$CSI^ICDEX
          $$SAB^ICDEX

       PCE PATIENT CARE ENCOUNTER
          PATIENT CARE ENCOUNTER - PCE (PX) will use the
          following APIs:

          $$CODEC^ICDEX
          $$CODEN^ICDEX
          $$CSI^ICDEX
          $$SINFO^ICDEX
          $$LD^ICDEX
          $$IE^ICDEX

       MENTAL HEALTH
          MENTAL HEALTH (YS) will use the following APIs:

          $$SINFO^ICDEX

       SURGERY
          SURGERY (SR) package will use the following APIs:

          $$CODEN^ICDEX
          $$LS^ICDEX
          $$SYS^ICDEX
          $$VST^ICDEX
          $$CODEABA^ICDEX
          $$OBA^ICDEX
          $$CSI^ICDEX
          $$CODEC^ICDEX

       ORDER ENTRY/RESULTS REPORTING
          ORDER ENTRY/RESULTS REPORTING (OR) will use the
          following APIs:

          $$CODECS^ICDEX
          $$CSI^ICDEX
          $$SAB^ICDEX

       TEXT INTEGRATION UTILITIES
          TEXT INTEGRATION UTILITIES (TIU) will use the
          following APIs:

          $$CODECS^ICDEX

    USAGE: Controlled Subscri  ENTERED: NOV  6,2011
   STATUS: Pending             EXPIRES:
 DURATION: Till Otherwise Agr  VERSION:
```

```
     DESCRIPTION:                          TYPE: Routine
Application Programmer Interfaces (APIs) in this routine were developed to
remove the need for direct global access to either the DIAGNOSIS file 80
or OPERATIONS/PROCEDURE file 80.1.

These entry points are meant to replace the following active/retired ICRs:

     48  Private       YS File 80.2        Weight (2)
    280  Private       HBH File 80         Code (.01)
    365  Private       QAM File 80         Code (.01)
    368  Private       IB File 80          Retired Nov 15, 2008
    369  Private       IB File 80.1        Retired Nov 15, 2008
    370  Private       IB/DSS 80.2         DRG Name (.01)
    582  Private       IMR File 80         Code (.01)
    647  Private       IB File 80          Retired Nov 15, 2008
   1161  Private       VAM File 80         Retired Nov 15, 2008
   1275  Private       GMTS File 80        Retired Nov 15, 2008
   1276  Private       GMTS File 80.1      Retired Nov 15, 2008
   1294  Subscription  PCE/TIU/OR File 80  Retired Nov 15, 2008
   1487  Private       ACKQ File 80        Retired Nov 15, 2008
   1586  Subscription  AICS/PCE File 80.3  MDC Name (.01)
   2435  Private       PXRM File 80 Hdr    ^ICD9(0)
   2436  Private       PXRM File 80.1 Hdr  ^ICD0(0)
   3990  Supported     Routine ICDCODE     To be retired Apr 2016
   3991  Supported     Routine ICDAPIU     To be retired Apr 2016
   4052  Supported     Routine ICDGTDRG
   5028  Subscription  GMPL File 80
   5388  Supported     File 80             Code (.01), AB/BA/D/AST/ACT
                                           To be retired Apr 2016
   5404  Supported     File 80.1           Code (.01), BA/ACT
                                           To be retired Apr 2016
   5699  Supported     Routine ICDXCODE    To be retired Apr 2016
   5757  Supported     Routine ICDSAPI     To be retired Apr 2016
  10082  Supported     File 80             Retired Nov 15, 2008
  10083  Supported     File 80.1           Retired Nov 15, 2008


  ROUTINE: ICDEX
COMPONENT:  HELP
            This is an interactive help entry point for the input and
            output variables for the APIs contained in the routine ICDEX.

COMPONENT:  $$ICDDX(CODE,CDT,SYS,FMT,LOC)
            This entry point extracts data for an ICD-9 or ICD-10 code in
            the DIAGNOSIS file 80.

            This entry point is intended to replace the ICD-9 Legacy API
            $$ICDDX^ICDCODE (ICR 3990) and $$ICDDATA^ICDXCODE (ICR 5699),
            providing a single point of entry for ICD diagnostic data.

VARIABLES:  Input     CODE
                        This is an ICD diagnosis code in either the
                        external or internal format.  If the internal
                        format is used, then the input variable FMT must
                        be set to "I" (Required).
```

```
VARIABLES:  Input     CDT
                      This is the Code Set Versioning date (Fileman
                      format) used to identify the code and text that
                      was appropriate for the date passed in this input
                      parameter. (Optional, if not supplied, TODAY will
                      be used)

VARIABLES:  Input     SYS
                      This is an ICD coding system identifier (taken
                      from file 80.4).  The following coding systems are
                      found in file 80:

                        1 = ICD-9 Diagnosis
                       30 = ICD-10 Diagnosis

                      (Optional, but highly encouraged)

VARIABLES:  Input     FMT
                      This variable tells the API if the CODE is in
                      External or Internal format.

                        "E" = External (default)
                        "I" = Internal Entry Number

                      (Conditional, required if CODE is in internal
                      format)

VARIABLES:  Output    $$ICDDX
                      This is a 20 piece string delimited by "^"

                         1  IEN of code in ^ICD9(
                         2  ICD-9 Dx Code                (#.01)
                         3  Identifier                   (#1.2)
                         4  Versioned Dx                 (67 multiple)
                         5  Unacceptable as Principal Dx (#1.3)
                         6  Major Dx Cat                 (72 multiple)
                         7  MDC13                        (#1.4)
                         8  Compl/Comorb                 (103 multiple)
                         9  ICD Expanded                 (#1.7)
                        10  Status                       (66 multiple)
                        11  Sex                          (10 multiple)
                        12  Inactive Date                (66 multiple)
                        13  MDC24                        (#1.5)
                        14  MDC25                        (#1.6)
                        15  Age Low                      (11 multiple)
                        16  Age High                     (12 multiple)
                        17  Activation Date              (66 multiple)
                        18  Message
                        19  Complication/Comorbidity     (103 multiple)
                        20  Coding System                (#1.1)
                        21  Primary CC Flag              (103 multiple)
                        22  PDX Exclusion Code           (#1.11)
                       or

                        -1^Error Description
```

```
VARIABLES:  Input     LOC
                        This is a boolean flag used to indicate if the API
                        is to use local VA codes.  It only applies to
                        ICD-9 for backwards compatibility.

                          1 = Use local VA codes
                          0 = Do not use local VA codes (default)

COMPONENT:  $$ICDOP(CODE,CDT,SYS,FMT,LOC)
            This entry point extracts data for an ICD-9 or ICD-10 code in
            the OPERATIONS/PROCEDURE file 80.1

            This entry point is intended to replace the ICD-9 Legacy API
            $$ICDOP^ICDCODE (ICR 3990) and $$ICDDATA^ICDXCODE (ICR 5699),
            providing a single point of entry for ICD procedural data.

VARIABLES:  Input     CODE
                        This is an ICD operation/procedure code in either
                        the external or internal format.  If the internal
                        format is used, then the input variable FMT must
                        be set to "I" (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to identify the code and text that
                        was appropriate for the date passed in CDT.
                        (Optional, if not supplied, TODAY will be used)

VARIABLES:  Input     SYS
                        This is an ICD coding system identifier (taken
                        from file 80.4).  The following coding systems are
                        found in file 80.1:

                          2 = ICD-9 Procedures
                         31 = ICD-10 Procedures

                        (Optional, but highly encouraged)

VARIABLES:  Input     FMT
                        This variable tells the API if the CODE is in
                        External or Internal format.

                          "E" = External (default)
                          "I" = Internal Entry Number

                        (Conditional, required if CODE is in internal
                        format)

VARIABLES:  Output    $$ICDOP
                        This is a 15 piece string delimited by "^"

                           1   IEN of code in ^ICD0(
                           2   ICD procedure code           (#.01)
                           3   Identifier                   (#1.2)
                           4   MDC24                        (#1.5)
                           5   Versioned Oper/Proc          (67 multiple)
                           6   <null>
```

```
                          7   <null>
                          8   <null>
                          9   ICD Expanded                  (#1.7)
                         10   Status                        (66 multiple)
                         11   Use with Sex                  (10 multiple)
                         12   Inactive Date                 (66 multiple)
                         13   Activation Date               (66 multiple)
                         14   Message
                         15   Coding System                 (#1.1)

                        or

                         -1^Error Description

VARIABLES:  Input     LOC
                        This is a boolean flag used to indicate if the API
                        is to use local VA codes.  It only applies to
                        ICD-9 for backwards compatibility.

                          1 = Use local VA codes
                          0 = Do not use local VA codes (default)

COMPONENT:  $$ICDD(CODE,.ARY,CDT,SYS,LEN)
            This API returns the long description of either an ICD-9 or
            ICD-10 code.

            This entry point is intended to replace the ICD-9 Legacy API
            $$ICDD^ICDCODE (ICR 3990) and $$ICDDESC^ICDXCODE (ICR 5699),
            providing a single point of entry for ICD diagnosis/procedure
            descriptions.

VARIABLES:  Input     CODE
                        This is an ICD-9 or ICD-10 code in external format
                        only  (Required).

VARIABLES:  Input     .ARY
                        This is the name of a local array, passed by
                        reference that will contain the output of this
                        API.  (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to identify the text that was
                        appropriate for the date passed in this input
                        parameter. (Optional, if not supplied, TODAY will
                        be used)

VARIABLES:  Input     SYS
                        This is an ICD coding system identifier (taken
                        from file 80.4).  The following coding systems are
                        found in files 80 and 80.1:

                          1 = ICD-9 Diagnosis file 80
                          2 = ICD-9 Procedure file 80.1
                         30 = ICD-10 Diagnosis file 80
                         31 = ICD-10 Procedure file 80.1
```

```
                    (Optional, but highly encouraged)

VARIABLES:  Input    LEN
                      This is the text string length of the description
                      placed in array .ARY.  (Optional, if passed it
                      must be greater than 27 based on the longest word
                      found in a diagnosis or procedure description and
                      not greater than 245.  If not passed it defaults
                      to 245 characters based in the input
                      transformation)

VARIABLES:  Output   $$ICDD
                      This is the number of lines in the output array
                      .ARY or if an error occurs, -1^Error Message

VARIABLES:  Output   ARY
                      This is a local array, passed by reference,
                      containing the long description of an ICD code
                      with string lengths defined by LEN when passed or
                      245 characters.  If there is a warning message
                      about text accuracy (ICD-9 only) it will be
                      appended to the end of the message preceded by a
                      blank line.

                        ARY(1) - Description (length of LEN)
                        ARY(n) - Description (continued if necessary)

                        If there is a warning message (ICD-9 only):

                        ARY(n+1) - blank
                        ARY(n+2) - message: CODE TEXT MAY BE INACCURATE

COMPONENT:  $$CODEN(CODE,FILE)
            This API returns the Internal Entry Number (IEN) of a ICD
            code.

            This entry point is intended to replace the ICD-9 Legacy API
            $$CODEN^ICDCODE (ICR 3990). It is also intended to replace the
            need for direct global access of the 'BA' cross-reference in
            ICRs 5388 and 5404.

VARIABLES:  Input    CODE
                      This is an ICD-9 or ICD-10 code in external format
                      only  (Required).

VARIABLES:  Input    FILE
                      This is the file number where the CODE is stored,
                      either 80 or 80.1 (Required)

VARIABLES:  Output   $$CODEN
                      This is the Internal Entry Number (IEN) of CODE in
                      file FILE appended by a tilde "~" and the global
                      root FILE:

                         IEN~^ROOT

                      or -1^Error Message on error
```

COMPONENT: **$$CODEC(FILE,IEN)**
This entry point returns the ICD-9 or ICD-10 code from a specified ICD file and Internal Entry Number (IEN).

This entry point is intended to replace the ICD-9 Legacy API $$CODEC^ICDCODE (ICR 3990). It is also intended to replace the need for direct global access in ICRs 280, 365, 582, 5388, and 5404.

VARIABLES: Input **FILE**
This is the ICD file number used to retrieve the code (Required)

80 = ICD Diagnosis file
80.1 = ICD Operation/Procedure file

VARIABLES: Input **IEN**
This is the internal entry number in FILE were the code to be retrieved is stored (Required)

VARIABLES: Output **$$CODEC**
This is either the ICD code stored at the Internal Entry Number IEN in the file identified by the FILE input parameter, or upon error:

-1 ^ Error Message

COMPONENT: **$$CODEBA(CODE,ROOT)**
This entry point returns the internal entry number (IEN) of a code found in the 'BA' cross-reference in the file specified.

This entry point is provided in lieu of ICD-9 Legacy entry point $$CODEN^ICDCODE (ICR 3990) which will crash with a <MAXNUMBER> error if the code passed has the letter 'E' in the middle of the code (example, ICD-10 procedure code 041E499 would be interpreted as scientific notation). $$CODEBA^ICDEX is much safer.

If you already know the coding system, please use $$CODEABA^ICDEX instead.

This entry point replaces the need for direct global read access of the 'BA' cross-reference allowed by ICRs 5388 and 5404.

VARIABLES: Input **CODE**
This is either an ICD Diagnosis code or ICD Procedure code (Required)

VARIABLES: Input **ROOT**
This is the global root (or file number) where the code is stored (Required)

VARIABLES: Output **$$CODEBA**
This is the internal entry number (IEN) in the specified file where the code is stored or -1 if

```
                    not found.

COMPONENT:  $$CODEABA(CODE,ROOT,SYS)
            This entry point returns the internal entry number (IEN) of a
            code found in the system specific 'ABA' cross-reference in the
            file specified.

            This entry point is provided in lieu of ICD-9 Legacy entry
            point $$CODEN^ICDCODE (ICR 3990) and new entry point
            $$CODEBA^ICDEX.

            Entry point Comparison:

                $$CODEN^ICDCODE will crash if the code
                has the letter 'E' in the middle of the
                code.  Do not use it.

                $$CODEBA^ICDEX is safer but it will fail to
                return the correct IEN if ICD-9 and ICD-10
                ever have a similar code.

                $$CODEABA^ICDEX will neither crash or fail
                to return the correct IEN.

VARIABLES:  Input     CODE
                      This is either an ICD Diagnosis code or ICD
                      Procedure code (Required)

VARIABLES:  Input     ROOT
                      This is the global root (or file number) where the
                      code is stored (Optional if SYS is supplied)

VARIABLES:  Input     SYS
                      This is an ICD coding system identifier (taken
                      from file 80.4).  The following coding systems are
                      found in files 80 and 80.1:

                         1 = ICD-9 Diagnosis file 80
                         2 = ICD-9 Procedure file 80.1
                        30 = ICD-10 Diagnosis file 80
                        31 = ICD-10 Procedure file 80.1

                      This API will look for the code on one of the
                      system specific cross-references:

                        ^ICD9("ABA",1,CODE,IEN)    ICD-9 Diagnosis
                        ^ICD9("ABA",30,CODE,IEN)   ICD-10 Diagnosis
                        ^ICD0("ABA",2,CODE,IEN)    ICD-9 Procedure
                        ^ICD0("ABA",31,CODE,IEN)   ICD-10 Procedure

                      If not supplied, the API will attempt to determine
                      the system based on code and file.

                      (Optional, but highly encouraged)

VARIABLES:  Output    $$CODEABA
                      This is the internal entry number (IEN) in the
```

```
                    specified file where the code is stored or -1 if
                    not found.

COMPONENT:  $$CODEFI(CODE)
            This entry point tries to resolve which file has an ICD code
            on file.

VARIABLES:  Input     CODE
                      This is either an ICD Diagnosis code or ICD
                      Procedure code (Required)

VARIABLES:  Output    $$CODEFI
                      This is the ICD file number where the specified
                      code was found:

                          80   = ICD Diagnosis file
                          80.1 = ICD Operation/Procedure file

                      or NULL if not found or could not resolve to a
                      single file.

COMPONENT:  $$CODECS(CODE,FILE,CDT)
            This entry point tries to resolve the Coding System based on a
            code, a file and a date.

VARIABLES:  Input     CODE
                      This is either an ICD Diagnosis code or ICD
                      Procedure code (Required)

VARIABLES:  Input     FILE
                      This is the ICD file number used to resolve the
                      coding system:

                          80   = ICD Diagnosis file
                          80.1 = ICD Operation/Procedure file

                      (Optional, but encouraged)  If not supplied, an
                      attempt to resolve the input variable FILE will be
                      made using the entry point $$CODEFI(CODE).

VARIABLES:  Input     CDT
                      This is the Code Set Versioning date (Fileman
                      format) used to resolve the coding system.

                      This date is ONLY used if a code is found in both
                      ICD-9 and ICD-10 systems.  If that ever happens,
                      the date passed will determine the coding system.
                      If the date passed is before the ICD-10
                      implementation date it will be considered an ICD-9
                      code and if it is on or after the ICD-10
                      implementation date then it will be considered
                      ICD-10.

VARIABLES:  Output    $$CODECS
                      This is a 2 piece "^" delimited string containing:

                         1   Coding System (pointer to file 80.4)
```

```
                        2   Coding Nomenclature (commonly used name)

                      Example output values:

                         1^ICD-9-CM
                        30^ICD-10-CM
                         2^ICD-9 Proc
                        31^ICD-10-PCS

                      NULL if the API cannot resolve the coding system
                      based on code, file and date.

COMPONENT:  $$CSI(FILE,IEN)
            This entry point returns the Coding System for an Internal
            Entry Number (IEN).

VARIABLES:  Input     FILE
                      This is the ICD file number used to retrieve the
                      coding system (Required):

                          80   = ICD Diagnosis file
                          80.1 = ICD Operation/Procedure file

VARIABLES:  Input     IEN
                      This is an Internal Entry Number (IEN) in the file
                      specified (Required).

VARIABLES:  Output    $$CSI
                      This is a pointer to the ICD CODING SYSTEMS file
                      #80.4

COMPONENT:  $$VMDC(IEN,CDT,FMT)
            This entry point retrieves the versioned Major Diagnostic
            Category (MDC) for a diagnostic code in the DIAGNOSIS file 80.

VARIABLES:  Input     IEN
                      This is an Internal Entry Number (IEN) in the
                      DIAGNOSIS file 80 (Required)

VARIABLES:  Input     CDT
                      This is the Code Set Versioning date (Fileman
                      format) used to identify the Major Diagnostic
                      Category that was appropriate for the date passed
                      (Optional, if not passed TODAY is used).

VARIABLES:  Input     FMT
                      This is a flag used to determine the output
                      format. Acceptable values are 0 and 1 (Optional,
                      default value is 0).

                          FMT = 0    Major Diagnostic Category (MDC)
                          FMT = 1    MDC^Effective Date

VARIABLES:  Output    $$VMDC
                      This is the Major Diagnostic Category (MDC) that
                      was appropriate for the date passed and the
                      diagnosis code identified by input parameter IEN.
```

The output may also have a second "^" delimited piece containing the MDC Effective Date if the input parameter FMT is set to 1.

COMPONENT: $$VAGEL(IEN,CDT,FMT)
This entry point retrieves the versioned Age Low value for a diagnostic code in the DIAGNOSIS file 80. Age Low is the minimum age value for an age range for which the diagnostic code can be applied.

VARIABLES: Input IEN
This is an Internal Entry Number (IEN) in the DIAGNOSIS file 80 (Required)

VARIABLES: Input CDT
This is the Code Set Versioning date (Fileman format) used to identify the Age Low value that was appropriate for the date passed (Optional, if not passed TODAY is used).

VARIABLES: Input FMT
This is a flag used to determine the output format. Acceptable values are 0 and 1 (Optional, default value is 0).

FMT = 0 Age Low
FMT = 1 Age Low^Effective Date

VARIABLES: Output $$VAGEL
This is the Age Low that was appropriate for the date passed and the diagnosis code identified by the input parameter IEN. The output may also have a second "^" delimited piece containing the Age Low Effective Date if the input parameter FMT is set to 1. Null if Age Low not found for date.

COMPONENT: $$VAGEH(IEN,CDT,FMT)
This entry point retrieves the versioned Age High value for a diagnostic code in the DIAGNOSIS file 80. Age High is the maximum age value for an age range for which the diagnostic code can be applied.

VARIABLES: Input IEN
This is an Internal Entry Number (IEN) in the DIAGNOSIS file 80 (Required)

VARIABLES: Input CDT
This is the Code Set Versioning date (Fileman format) used to identify the Age High value that was appropriate for the date passed (Optional, if not passed TODAY is used).

VARIABLES: Input FMT
This is a flag used to determine the output format. Acceptable values are 0 and 1 (Optional, default value is 0).

```
                        FMT = 0    Age High
                        FMT = 1    Age High^Effective Date

VARIABLES:  Output    $$VAGEH
                      This is the Age High that was appropriate for the
                      date passed and the diagnosis code identified by
                      the input parameter IEN.  The output may also have
                      a second "^" delimited piece containing the Age
                      High Effective Date if the input parameter FMT is
                      set to 1.  Null if Age High is not found for date.

COMPONENT:  $$VCC(IEN,CDT,FMT)
            This entry point retrieves the versioned Complication
            Comorbidity (CC) designation for a diagnostic code in the
            DIAGNOSIS file 80.  A diagnostic code can be designated as:

              Non-Complication Comorbidity (Non-CC)
              Complication Comorbidity (CC)
              Major Complication Comorbidity (MCC)

VARIABLES:  Input     IEN
                      This is an Internal Entry Number (IEN) in the
                      DIAGNOSIS file 80 (Required)

VARIABLES:  Input     CDT
                      This is the Code Set Versioning date (Fileman
                      format) used to identify the CC designation value
                      that was appropriate for the date passed
                      (Optional, if not passed TODAY is used).

VARIABLES:  Input     FMT
                      This is a flag used to determine the output
                      format. Acceptable values are 0 and 1 (Optional,
                      default value is 0).

                          FMT = 0    CC designation
                          FMT = 1    CC designation^Effective Date

VARIABLES:  Output    $$VCC
                      This is the CC designation that was appropriate
                      for the date passed and the diagnosis code
                      identified by the input parameter IEN.  The output
                      may also have a second "^" delimited piece
                      containing the CC designation Effective Date if
                      the input parameter FMT is set to 1.

                        0 = Non-Complication Comorbidity (Non-CC)
                        1 = Complication Comorbidity (CC)
                        2 = Major Complication Comorbidity (MCC)

                      Null if not found for date

COMPONENT:  $$VSEX(FILE,IEN,CDT,FMT)
            This entry point retrieves the versioned sex designation for a
            diagnostic or procedure code in either the ICD DIAGNOSIS file
            80 or the ICD OPERATION/PROCEDURE file 80.1. If a sex
            designation exist then the diagnosis or procedure should be
```

applied only to that sex.

VARIABLES: Input FILE
This is the ICD file number used to retrieve the sex designation:

80 = ICD Diagnosis file
80.1 = ICD Operation/Procedure file

VARIABLES: Input IEN
This is an Internal Entry Number (IEN) in either the DIAGNOSIS file 80 or OPERATION/PROCEDURE file 80.1 (Required)

VARIABLES: Input CDT
This is the Code Set Versioning date (Fileman format) used to identify the sex designation value that was appropriate for the date passed (Optional, if not passed TODAY is used).

VARIABLES: Input FMT
This is a flag used to determine the output format. Acceptable values are 0 and 1 (Optional, default value is 0).

FMT = 0 Sex designation
FMT = 1 Sex designation^Effective Date

VARIABLES: Output $$VSEX
This is the sex designation that was appropriate for the date passed and the code identified by the input parameter IEN. The output may also have a second "^" delimited piece containing the sex designation Effective Date if the input parameter FMT is set to 1.

M = Male
F = Female
Null if sex is N/A or not found for date

COMPONENT: $$SAI(FILE,IEN,CDT)
This entry point retrieves the Status, Activation date and Inactivation date for a diagnosis or procedure on a specified date.

VARIABLES: Input FILE
This is the ICD file number used to retrieve the status and effective dates:

80 = ICD Diagnosis file
80.1 = ICD Operation/Procedure file

VARIABLES: Input IEN
This is an Internal Entry Number (IEN) in either the DIAGNOSIS file 80 or OPERATION/PROCEDURE file 80.1 (Required)

```
VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to retrieve the status and effective
                        dates that were appropriate for the date passed
                        (Optional, if not passed TODAY is used).

VARIABLES:  Output    $$SAI
                        This is a 6 piece "^" delimited string

                           1  Status
                           2  Activation Date
                           3  Inactivation Date
                           4  IEN
                           5  Code
                           6  Short Text

                              If the status is active, the short
                              text will be the most recent.

                              If the status is inactive, the short
                              text will be the text in use on the
                              date it was inactivated.

                              Null if no status for date.

COMPONENT:  $$VST(FILE,IEN,CDT)
            This entry point retrieves the Versioned Short Text for an
            diagnosis or procedure on a specified date.

VARIABLES:  Input     FILE
                        This is the ICD file number used to retrieve the
                        Versioned Short Text:

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in either
                        the DIAGNOSIS file 80 or OPERATION/PROCEDURE file
                        80.1 (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to retrieve the Versioned Short Text
                        that was appropriate for the date passed
                        (Optional, if not passed TODAY is used).

VARIABLES:  Output    $$VST
                        This is the Versioned Short Text from either file
                        80 (DIAGNOSIS) or 80.1 (OPERATION/PROCEDURE) that
                        was appropriate for the date passed and the code
                        identified by the input parameter IEN.  Null if
                        not found.

COMPONENT:  $$VLT(FILE,IEN,CDT)
            This entry point retrieves the Versioned Long Text
            (description) for a diagnosis or procedure on a specified
```

```
            date.

VARIABLES:  Input     FILE
                        This is the ICD file number used to retrieve the
                        Versioned Long Text (description):

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in either
                        the DIAGNOSIS file 80 or OPERATION/PROCEDURE file
                        80.1 (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to retrieve the Versioned Long Text
                        (description) that was appropriate for the date
                        passed (Optional, if not passed TODAY is used).

VARIABLES:  Output    $$VLT
                        This is the Versioned Long Text (description) from
                        either file 80 or 80.1 that was appropriate for
                        the date passed and the code identified by the
                        input parameter IEN.  Null if not found.

COMPONENT:  $$VSTD(IEN,CDT)
            This entry point retrieves the Versioned Short Text for a
            diagnosis on a specified date.

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the
                        DIAGNOSIS file 80 (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to retrieve the Versioned Short Text
                        that was appropriate for the date passed
                        (Optional, if not passed TODAY is used).

VARIABLES:  Output    $$VSTD
                        This is the Versioned Short Text from file 80 that
                        was appropriate for the date passed and the code
                        identified by the input parameter IEN.  Null if
                        not found.

COMPONENT:  $$VSTP(IEN,CDT)
            This entry point retrieves the Versioned Short Text for a
            procedure on a specified date.

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the
                        OPERATION/PROCEDURE file 80.1 (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to retrieve the Versioned Short Text
```

```
                    that was appropriate for the date passed
                    (Optional, if not passed TODAY is used).

VARIABLES:  Output    $$VSTP
                    This is the Versioned Short Text from file 80.1
                    that was appropriate for the date passed and the
                    code identified by the input parameter IEN.  Null
                    if not found.

COMPONENT:  $$VLTD(IEN,CDT)
            This entry point retrieves the Versioned Long Text
            (description) for a diagnosis on a specified date.

VARIABLES:  Input     IEN
                    This is an Internal Entry Number (IEN) in the
                    DIAGNOSIS file 80 (Required)

VARIABLES:  Input     CDT
                    This is the Code Set Versioning date (Fileman
                    format) used to retrieve the Versioned Long Text
                    (description) that was appropriate for the date
                    passed (Optional,  If not passed TODAY is used).

VARIABLES:  Output    $$VLTD
                    This is the Versioned Long Text (description) from
                    file 80 that was appropriate for the date passed
                    and the code identified by the input parameter
                    IEN.  Null if not found.

COMPONENT:  $$VLTP(IEN,CDT)
            This entry point retrieves the Versioned Long Text
            (description) for a procedure on a specified date.

VARIABLES:  Input     IEN
                    This is an Internal Entry Number (IEN) in the
                    OPERATION/PROCEDURE file 80.1 (Required)

VARIABLES:  Input     CDT
                    This is the Code Set Versioning date (Fileman
                    format) used to retrieve the Versioned Long Text
                    (description) that was appropriate for the date
                    passed (Optional,  If not passed TODAY is used).

VARIABLES:  Output    $$VLTP
                    This is the Versioned Long Text (description) from
                    file 80.1 that was appropriate for the date passed
                    and the code identified by the input parameter
                    IEN.  Null if not found.

COMPONENT:  $$SD(FILE,IEN,CDT,.ARY,LEN)
            This entry point retrieves the Versioned Short Text for a
            procedure on a specified date.  This entry point is similar to
            $$VST except you can elect to have the Short Text returned in
            a local array and you can specify the string lengths of the
            text in the array.

VARIABLES:  Input     FILE
```

```
                    This is the ICD file number used to retrieve the
                    Versioned Short Text (Required):

                        80   = ICD Diagnosis file
                        80.1 = ICD Operation/Procedure file

VARIABLES:  Input    IEN
                    This is an internal entry number (IEN) in either
                    file 80 or 80.1 (Required)

VARIABLES:  Input    CDT
                    This is the Code Set Versioning date (Fileman
                    format) used to retrieve the Versioned Short Text
                    that was appropriate for the date passed
                    (Optional,  If not passed TODAY is used).

VARIABLES:  Input    .ARY
                    This is a local array name passed by reference
                    that will contain the Short Text output.

VARIABLES:  Input    LEN
                    This is a number greater than 27 and less than 246
                    representing the desired text string lengths for
                    the Short Text output.  If specified, the output
                    will be parsed into strings not to exceed the
                    length specified (Optional, default 245)

VARIABLES:  Output   $$SD
                    This is the Versioned Short Text from either file
                    80 or 80.1 that was appropriate for the date
                    passed and the code identified by the input
                    parameter IEN.  If not found:

                       -1^Error Message

VARIABLES:  Output   ARY
                    If passed, this is a local array containing the
                    number of text lines, the effective date of the
                    Short Text and the text.  If the input parameter
                    LEN (length) is specified and the length is
                    shorter than the Short Text, then the Short Text
                    will be parsed into test strings not to exceed
                    LEN.

                      ARY(0)=# lines ^ effective date
                      ARY(1)=Short Text

                      LEN is defined shorter than text

                      ARY(0)=# lines ^ effective date
                      ARY(1)=String length not to exceed LEN
                      ARY(n)=String length not to exceed LEN

                    Null if not found

COMPONENT:  $$LD(FILE,IEN,CDT,.ARY,LEN)
            This entry point retrieves the Versioned Long Text
```

```
          (description) for a procedure on a specified date.  This entry
          point is similar to $$VLT except you can elect to have the
          Long Text (description) returned in a local array and you can
          specify the string lengths of the text in the array.

VARIABLES:  Input     FILE
                        This is the ICD file number used to retrieve the
                        Versioned Long Text (description) (Required):

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     IEN
                        This is an internal entry number (IEN) in either
                        file 80 or 80.1 (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to retrieve the Versioned Long Text
                        (description) that was appropriate for the date
                        passed (Optional,  If not passed TODAY is used).

VARIABLES:  Input     .ARY
                        This is a local array name passed by reference
                        that will contain the Long Text (description)
                        output.

VARIABLES:  Input     LEN
                        This is a number greater than 27 and less than 246
                        representing the desired text string lengths for
                        the Long Text (description) output.  If specified,
                        the output will be parsed into strings not to
                        exceed the length specified (Optional, default
                        245)

VARIABLES:  Output    $$LD
                        This is the Versioned Long Text (description) from
                        either file 80 or 80.1 that was appropriate for
                        the date passed and the code identified by the
                        input parameter IEN.  If not found:

                            -1^Error Message

VARIABLES:  Output    ARY
                        If passed, this is a local array containing the
                        number of text lines, the effective date of the
                        Long Text (description) and the text.  If the
                        input parameter LEN (length) is specified and the
                        length is shorter than the Long Text
                        (description), then the Long Text (description)
                        will be parsed into test strings not to exceed
                        LEN.

                          ARY(0)=# lines ^ effective date
                          ARY(1)=Long Text (description)

                          LEN defined shorter than text
```

```
                        ARY(0)=# lines ^ effective date
                        ARY(1)=String length not to exceed LEN
                        ARY(n)=String length not to exceed LEN

COMPONENT:  PAR(.ARY,LEN)
            This entry point takes text in a local array (passed by
            reference) and parses it into string lengths not to exceed the
            length specified.

VARIABLES:  Input      .ARY
                         This is a local array name passed by reference and
                         contains the text to be parsed into strings not to
                         exceed the length specified.

                           ARY(1) = Unparsed Text

VARIABLES:  Input      LEN
                         This is a number representing the desired text
                         string lengths for the text found in ARY().
                         (Optional, default length 79)

VARIABLES:  Output     ARY
                         This is a local array containing the input text
                         parsed so that each text string length does not
                         exceed the length specified.

                           ARY(1)=Parsed Text length not to exceed LEN
                           ARY(n)=Parsed Text length not to exceed LEN

COMPONENT:  $$STATCHK(CODE,CDT,SYS)
            This entry point is used to determine the status (active or
            inactive) of a ICD code.

VARIABLES:  Input      CODE
                         This is either an ICD diagnosis or procedure code
                         (external format) (Required)

VARIABLES:  Input      CDT
                         This is the Code Set Versioning date (Fileman
                         format) used to retrieve the code's status,
                         internal entry number (IEN) and effective date
                         that was appropriate for the date passed
                         (Optional,  If not passed TODAY is used)

VARIABLES:  Input      SYS
                         This is an ICD coding system identifier (taken
                         from file 80.4).  The following coding systems are
                         found in files 80 and 80.1:

                              1 = ICD-9 Diagnosis
                             30 = ICD-10 Diagnosis
                              2 = ICD-9 Procedures
                             31 = ICD-10 Procedures

                         (Optional, but encouraged, if doesn't exist it
                         will try to determine coding system by input
```

```
                    parameter CODE)

VARIABLES:  Output    $$STATCHK
                    This is a three piece "^" delimited string

                       1  Status 1 = Active, 0 = Inactive
                       2  IEN or -1 on error
                       3  Effective Date or error message

                       Error           0 ^ -1 ^ Error message
                       Active Code     1 ^ IEN ^ Effective Date
                       Inactive Code   0 ^ IEN ^ Effective Date

COMPONENT:  $$DTBR(CDT,STD,SYS)
            This entry point returns the business rule date for a coding
            system.  This is in earliest date possible for a coding
            standard and/or a coding system.

VARIABLES:  Input     CDT
                    This is the Code Set Versioning date (Fileman
                    format) used to resolved the business rule date.
                    (Optional, if not passed TODAY is used)

VARIABLES:  Input     STD
                    This is a coding standard from a Standards
                    Development Organization (SDO).  A standard may
                    have one or more coding systems.  (Optional,
                    default is 0)

                        0 = ICD (Default)
                        1 = CPT/HCPCS
                        2 = DRG

VARIABLES:  Input     SYS
                    This is an ICD coding system identifier (taken
                    from file 80.4).  (Optional, there is no default
                    value for this parameter, if it does not exist
                    then it is not used)

                    The following coding systems are found in files 80
                    and 80.1:

                         1 = ICD-9 Diagnosis
                        30 = ICD-10 Diagnosis
                         2 = ICD-9 Procedures
                        31 = ICD-10 Procedures

VARIABLES:  Output    $$DTBR
                    Date adjusted by business rules:

                    If Standard (SDT) = 0 (ICD)

                       If CDT < 2781001 use 2781001
                       If CDT < 3131001 and SYS=30, use 3131001
                       If CDT < 3131001 and SYS=31, use 3131001

                    If Standard (SDT) = 1 (CPT/HCPCS)
```

```
                        If CDT < 2890101 use 2890101

                     If Standard (SDT) = 2 (DRG)

                        If CDT < 2821001 use 2821001

                     If CDT is year only, use first of the year If CDT
                     is year and month only, use first of the month

COMPONENT:  $$IMP(SYS,CDT)
            This entry point returns the date a coding system was
            implemented (taken from file 80.4).

VARIABLES:  Input     SYS
                      This is a coding system (taken from file 80.4) or
                      a coding system identifier that can be resolved to
                      a coding system.

                         1 = ICD-9-CM
                         2 = ICD-9-PCS
                        30 = ICD-10-CM
                        31 = ICD-10-PCS

                        DX, DIAG, 80, ^ICD9(

                          1 = ICD-9-CM if CDT is before the ICD-10
                     implementation date
                        30 = ICD-10-CM if CDT is on or after the ICD-10
                     implementation date

                        PR, PROC, OPER, 80.1, ^ICD0(

                          2 = ICD-9-CM if CDT is before  the ICD-10
                     implementation date
                        31 = ICD-10-CM if CDT is on or after  the
                     ICD-10 implementation date

VARIABLES:  Input     CDT
                      This is the Code Set Versioning date (Fileman
                      format) used to resolve the coding system
                      parameter SYS (Optional, if not passed TODAY is
                      used)

VARIABLES:  Output    $$IMP
                      This is the date that a coding system identified
                      by the input parameters SYS and CDT was
                      implemented in Fileman format or on error:

                      -1 ^ Error message

COMPONENT:  $$MSG(CDT,STD,SYS)
            This entry point returns a warning message that the text may
            be inaccurate for the date specified.  It applies only to
            ICD-9 Diagnosis and Procedures.

VARIABLES:  Input     CDT
```

```
                    This is the Code Set Versioning date (Fileman
                    format) used to determine the accuracy of the text
                    being returned (Optional, if not passed TODAY is
                    used)

VARIABLES:  Input    STD
                    This is a coding standard from a Standards
                    Development Organization (SDO).  A standard may
                    have one or more coding systems.  (Optional,
                    default is 0)

                        0 = ICD (Default)
                        1 = CPT/HCPCS
                        2 = DRG

VARIABLES:  Input    SYS
                    This is an ICD coding system identifier (taken
                    from file 80.4).  (Optional, there is no default
                    value for this parameter, if it does not exist
                    then it is not used)

                    The following coding systems are found in files 80
                    and 80.1:

                         1 = ICD-9 Diagnosis
                        30 = ICD-10 Diagnosis
                         2 = ICD-9 Procedures
                        31 = ICD-10 Procedures

VARIABLES:  Output   $$MSG
                    If coding system is not ICD-10 and the date passed
                    is before the Code Set Versioning project Oct 1,
                    2002, then this variable is set to the warning
                    message, "CODE TEXT MAY BE INACCURATE" otherwise
                    it is null.

COMPONENT:  $$SEL(FILE,IEN)
            This entry point determines if an entry in a file is
            selectable by calling applications.

VARIABLES:  Input    FILE
                    This is an ICD file number:

                        80   = ICD Diagnosis file
                        80.1 = ICD Operation/Procedure file

VARIABLES:  Input    IEN
                    This is an Internal Entry Number (IEN) in the file
                    specified.

VARIABLES:  Output   $$SEL
                    This is a Boolean value:

                       1  Entry IEN in file FILE is Selectable
                       0  Entry IEN in file FILE is NOT Selectable

                    or
```

```
                    -1  on error

COMPONENT:  $$NEXT(CODE,SYS,CDT)
            This entry point returns the Next code in a sequence of codes
            in a coding system.

VARIABLES:  Input     CODE
                      This is either an ICD diagnosis, an ICD procedure
                      code or null to retrieve the first code in a
                      sequence.

VARIABLES:  Input     SYS
                      This is an ICD coding system identifier (taken
                      from file 80.4).  (Optional)

                      The following coding systems are found in ICD
                      files 80 and 80.1:

                           1 = ICD-9 Diagnosis
                          30 = ICD-10 Diagnosis
                           2 = ICD-9 Procedures
                          31 = ICD-10 Procedures

VARIABLES:  Input     CDT
                      This is the Code Set Versioning date (Fileman
                      format) used to determine the next code being
                      returned (Optional, there is no default value for
                      this parameter)

                      If CDT date is not passed then this entry point
                      will return the next code, regardless of status
                      (active or inactive)

                      If CDT date is passed then this entry point will
                      return the next active code.

VARIABLES:  Output    $$NEXT
                      This is the next code in a sequence of codes.  If
                      the input code is null, then it will return the
                      first code of the sequence of codes.  If a date is
                      passed in the input parameter CDT, then it will
                      return the next active code in a sequence of
                      codes.

COMPONENT:  $$PREV(CODE,SYS,CDT)
            This entry point returns the Previous code in a sequence of
            codes in a coding system.

VARIABLES:  Input     CODE
                      This is either an ICD diagnosis, an ICD procedure
                      code or null to retrieve the last code in a
                      sequence.

VARIABLES:  Input     SYS
                      This is an ICD coding system identifier (taken
                      from file 80.4).  (Optional)
```

The following coding systems are found in ICD files 80 and 80.1:

1 = ICD-9 Diagnosis
30 = ICD-10 Diagnosis
2 = ICD-9 Procedures
31 = ICD-10 Procedures

VARIABLES: Input CDT
This is the Code Set Versioning date (Fileman format) used to determine the Previous code being returned (Optional, there is no default value for this parameter)

If CDT date is not passed then this entry point will return the previous code, regardless of status (active or inactive)

If CDT date is passed then this entry point will return the previous active code.

VARIABLES: Output $$PREV
This is the previous code in a sequence of codes. If the input code is null, then it will return the last code of the sequence of codes. If a date is passed in the input parameter CDT, then it will return the previous active code in a sequence of codes.

COMPONENT: $$HIST(CODE,.ARY,SYS)
This entry point returns a code's activation history.

VARIABLES: Input CODE
This is an ICD diagnosis or procedure code.

VARIABLES: Input .ARY
This is a local array name passed by reference that will contain the code's activation history.

VARIABLES: Input SYS
This is an ICD coding system identifier (taken from file 80.4). (Optional)

The following coding systems are found in ICD files 80 and 80.1:

1 = ICD-9 Diagnosis
30 = ICD-10 Diagnosis
2 = ICD-9 Procedures
31 = ICD-10 Procedures

VARIABLES: Output $$HIST
This is set equal to the number of history entries in the local array ARY or -1 if there is an error or the code is not found.

```
VARIABLES:  Output    ARY
                        This is a local array containing the history
                        records

                           ARY(0) = Number of History Entries
                           ARY(<effective date>,<status>) = comment

COMPONENT:  $$PERIOD(CODE,.ARY,SYS)
            This entry point returns all the activation periods for a
            code.  An activation period is defined as the period of time
            between the beginning activation effective date and the ending
            inactivation effective date.  If the code is still active the
            period will have an activation date without an inactivation
            date.

VARIABLES:  Input     CODE
                        This is either an ICD diagnosis or procedure code.

VARIABLES:  Input     .ARY
                        This is a local array name passed by reference
                        that will contain the code's activation periods.

VARIABLES:  Input     SYS
                        This is an ICD coding system identifier (taken
                        from file 80.4).  (Optional)

                        The following coding systems are found in ICD
                        files 80 and 80.1:

                             1 = ICD-9 Diagnosis
                            30 = ICD-10 Diagnosis
                             2 = ICD-9 Procedures
                            31 = ICD-10 Procedures

VARIABLES:  Output    $$PERIOD
                        This is a 2 piece "^" delimited string if
                        successful and 3 piece "^" delimited string if
                        unsuccessful or error.

                           1   IEN of code
                           2   Code is selectable (boolean 1/0)

                         or on error

                           -1 ^ 0 ^ Error Message

VARIABLES:  Output    ARY
                        This is a local array containing the Periods of
                        activation for the code

                        ARY(0)

                           This is a 2 piece "^" delimited string if
                           successful and a 3 piece "^" delimited string
                           if unsuccessful or error.

                             1   IEN of code
```

```
                          2   Code is selectable (boolean 1/0)

                       or on error

                        -1^0^Error Message

                     ARY(Activation Date) = Inactivation Date^Short
                     Name

                        Where the Short Name is the Versioned text, and
                     the
                        text is versioned as follows:

                        Period is active   - Text for TODAY's date
                        Period is inactive - Text for inactivation date

COMPONENT:  $$OBA(FILE,CODE,SYS,REV)
            This entry point is used to $ORDER through the BA or ABA
            cross-references and replaces the need to access the BA/ABA
            cross-references in a FOR loop.  This entry point is meant to
            replace BA cross-reference in ICRs 5388 and 5404.

              $$OBA(<file>,<code>,<system>) replaces:

                $O(^ICD9("BA",(<code>_" ")) and
                $O(^ICD0("BA",(<code>_" "))

              Examples:

              F  S CODE=$$OBA(80,CODE,1) Q:'$L(CODE)  D
              F  S CODE=$$OBA(80,CODE,30) Q:'$L(CODE)  D
              F  S CODE=$$OBA(80.1,CODE,2) Q:'$L(CODE)  D
              F  S CODE=$$OBA(80.1,CODE,31) Q:'$L(CODE)  D

VARIABLES:  Input     FILE
                        This is the ICD file number used to determine the
                        global root to $ORDER through (Required):

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     CODE
                        This is either an ICD diagnosis or procedure code
                        to $ORDER from (required):

                           $O(^ROOT("BA",(CODE_" ")))
                           $O(^ROOT("ABA",SYS,(CODE_" ")))

VARIABLES:  Input     SYS
                        This is either an ICD diagnosis or procedure

                        This is an ICD coding system identifier (taken
                        from file 80.4).  (Optional)

                        The following coding systems are found in ICD
                        files 80 and 80.1:
```

```
                          1 = ICD-9 Diagnosis
                         30 = ICD-10 Diagnosis
                          2 = ICD-9 Procedures
                         31 = ICD-10 Procedures

                       If the coding system can be identified then the
                       "BA" cross-reference is ignored and the $ORDER
                       will be performed on the "ABA" cross-reference:

                          $O(^ROOT("ABA",SYS,(CODE_" ")))

                       The "ABA" cross-reference is a coding system
                       specific cross-reference.

VARIABLES:  Used     REV
                       This is a Reverse $ORDER flag, if set to 1, the
                       $ORDER operation will be in the reverse direction
                       of "BA" or "ABA" cross-reference (Optional,
                       default is 0, $ORDER forward)

                       If equal to 1

                          $O(^ROOT("BA",(CODE_" ")),-1)
                          $O(^ROOT("ABA",SYS,(CODE_" ")),-1)

VARIABLES:  Output   $$OBA
                       This is the Next or Previous Code in the "BA" or
                       "ABA" cross-reference depending on the $ORDER
                       direction established by the input parameter REV.

COMPONENT:  $$OD(FILE,WORD,SYS,REV)
            This entry point is used to $ORDER through the "D" or "AD"
            cross-reference and replaces the need to access the D/AD
            cross-references in a FOR loop.  This entry point is meant to
            replace the D cross-reference in ICRs 5388 and 5404.

              $$OD(<file>,<word>,<system>) replaces:

                $O(^ICD9("D",(<word>_" ")) and
                $O(^ICD0("D",(<word>_" "))

              Examples:

              F  S WORD=$$OD(80,WORD,1) Q:'$L(WORD)  D
              F  S WORD=$$OD(80,WORD,30) Q:'$L(WORD)  D
              F  S WORD=$$OD(80.1,WORD,2) Q:'$L(WORD)  D
              F  S WORD=$$OD(80.1,WORD,31) Q:'$L(WORD)  D

VARIABLES:  Input    FILE
                       This is the ICD file number used to determine the
                       global root to $ORDER through (Required):

                           80   = ICD Diagnosis file
                           80.1 = ICD Operation/Procedure file

VARIABLES:  Input    WORD
                       This is a one or two piece "^" delimited string
```

```
                    1  WORD  This is a single word parsed from the
                             codes description.

                    2  IEN   This is the internal entry number where
                             the description can be found that
                             contains the parsed word

                    WORD and IEN can be null.

                  $$OD $ORDER through "WORD^IEN" on either the D or
                  AD cross-references

                  Coding System unknown:  $O(^ROOT("D",WORD,IEN))
                  Coding System known:
                  $O(^ROOT("AD",SYS,WORD,IEN))

VARIABLES:  Input     SYS
                    This is an ICD coding system identifier (taken
                    from file 80.4).  (Optional)

                    The following coding systems are found in ICD
                    files 80 and 80.1:

                         1 = ICD-9 Diagnosis
                        30 = ICD-10 Diagnosis
                         2 = ICD-9 Procedures
                        31 = ICD-10 Procedures

                    If the coding system can be identified then the
                    "D" cross-reference is ignored and the $ORDER will
                    be performed on the "AD" cross-reference:

                       $O(^ROOT("AD",SYS,(CODE_" ")))

                    The "AD" cross-reference is a coding system
                    specific cross-reference.

VARIABLES:  Input     REV
                    This is a Reverse $ORDER flag, if set to 1, the
                    $ORDER operation will be in the reverse direction
                    of "D" or "AD" cross-reference (Optional, default
                    is 0, $ORDER forward)

                    If equal to 1

                       $O(^ROOT("D",WORD)),-1)
                       $O(^ROOT("AD",SYS,WORD)),-1)

VARIABLES:  Output    $$OD
                    This is a 2 piece "^" delimited string containing
                    the Next or Previous Word in the "D" or "AD"
                    cross-reference and accompanying IEN depending on
                    the $ORDER direction established by the input
                    parameter REV.

                    WORD^IEN taken from cross-references
```

```
                    ^ROOT("D",WORD,IEN) or
                    ^ROOT("AD",SYS,WORD,IEN)

COMPONENT:  $$DLM(FILE,IEN,FIELD,CDT)
            This entry point returns the date a record or field was last
            modified.  If the field number is passed, then the date last
            modified (based on date) for the field is returned.  If the
            field is not passed, then the date last modified (based on
            date) for the record at IEN is returned.  The following are
            valid versioned fields:

               File 80

               10   Sex                          5;0
               11   Age Low                      6;0
               12   Age High                     7;0
               66   Status                      66;0
               67   Diagnosis                   67;0
               68   Description                 68;0
               71   DRG Grouper                  3;0
               72   Major Diagnostic Category    4;0
              103   Complication/Comorbidity    69;0

               File 80.1

               10   Sex                          3;0
               66   Status                      66;0
               67   Operation/Procedure         67;0
               68   Description                 68;0
               71   DRG Grouper                  2;0


VARIABLES:  Input     FIELD
                        This is the ICD file number used to determine the
                        global root to $ORDER through (Required):

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the file
                        specified (Required)

VARIABLES:  Input     FIELD
                        This is the field number of a versioned data
                        element in the file specified.  (Optional, with no
                        default value)

                        If the field number is provided then this API will
                        return the date that the field was last modified.

                        If the field number is not provided then this API
                        will return the date that the record was last
                        modified.

VARIABLES:  Input     CDT
```

```
                    This is the Code Set Versioning date (Fileman
                    format) used to determine the date last modified
                    (Optional, if not provided then TODAY is used)

VARIABLES:  Output   $$DLM
                    This is the date last modified for the record
                    identified by the input parameters FILE and IEN.
                    If the input parameter FIELD is set to a valid
                    versioned field then this will be the date that
                    the field was last modified.

                    or -1 ^ message on error

COMPONENT:  $$CS(FILE,FMT,CDT)
            This is an interactive entry point to select a coding system.

VARIABLES:  Input    FILE
                    This is the ICD file number used to select a
                    coding system (Optional, if not provided you will
                    be prompted for an ICD file Number):

                        80   = ICD Diagnosis file
                        80.1 = ICD Operation/Procedure file

VARIABLES:  Input    FMT
                    This is a flag to determine the display format for
                    the prompts:

                       E  Display External only (default)
                       I  Display External with Internal

                       Prompt using External only, default:

                       FMT=E      1   ICD-9-CM
                                  2   ICD-10-CM

                       Prompt using External with Internal:

                       FMT=I      1   ICD-9-CM       (#1)
                                  2   ICD-10-CM      (#30)

VARIABLES:  Input    CDT
                    This is an optional date to use in selecting a
                    coding system.  If passed, only coding systems
                    with an implementation date on or before the date
                    passed are selectable (optional)

VARIABLES:  Output   $$CS
                    This is a 2 piece "^" delimited string

                    1  Coding System (internal) 2  Coding System
                    (external)

                    or  -1  on error or non-selection or  ^^  double
                    up-arrows or  ^   timeout or single up-arrow

COMPONENT:  $$EFF(FILE,IEN,CDT)
```

```
            This entry point returns a codes status, inactivation date and
            activation date (replaces EFF^ICDSUPT)

VARIABLES:  Input     FILE
                        This is an ICD file number (Required):

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the file
                        specified (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to determine the status and effective
                        dates on the date specified (Optional, if not
                        provided then TODAY is used)

VARIABLES:  Output    $$EFF
                        This is a 3 piece "^" delimited string

                            1   Status
                                1 - Active
                                0 - Inactive
                            2   Inactivation Date
                            3   Activation Date

                           or
                             -1^error message

COMPONENT:  $$LA(FILE,IEN,CDT)
            This entry point returns the last activation effective date
            based on a date passed.

VARIABLES:  Input     FILE
                        This is an ICD file number (Required):

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the file
                        specified (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to determine the last activation date
                        based on the date specified (Optional, if not
                        provided then TODAY is used)

VARIABLES:  Output    $$LA
                        This is the last activation date (Fileman format)
                        or

                        -1^Not activated on or before date specified
```

COMPONENT: **$$LI(FILE,IEN,CDT)**
This entry point returns the last inactivation effective date based on a date passed.

VARIABLES: **Input** **FILE**
This is an ICD file number (Required):

80 = ICD Diagnosis file
80.1 = ICD Operation/Procedure file

VARIABLES: **Input** **IEN**
This is an Internal Entry Number (IEN) in the file specified (Required)

VARIABLES: **Input** **CDT**
This is the Code Set Versioning date (Fileman format) used to determine the last inactivation date based on the date specified (Optional, if not provided then TODAY is used)

VARIABLES: **Output** **$$LI**
This is the last inactivation date (Fileman format) or

-1^Not inactivated on or before date specified

COMPONENT: **$$LS(FILE,IEN,CDT)**
This entry point returns the last code status based on a date passed.

VARIABLES: **Input** **FILE**
This is an ICD file number (Required):

80 = ICD Diagnosis file
80.1 = ICD Operation/Procedure file

VARIABLES: **Input** **IEN**
This is an Internal Entry Number (IEN) in the file specified (Required)

VARIABLES: **Input** **CDT**
This is the Code Set Versioning date (Fileman format) used to determine the last code status based on the date specified (Optional, if not provided then TODAY is used)

VARIABLES: **Output** **$$LS**
This is the last code status based on the date passed.

1 - Active
0 - Inactive

or

-1^No status on or before date specified

COMPONENT: **$$NUM(CODE)**
This entry point converts a code to a numeric representation (found on the AN cross-reference)

VARIABLES: **Input** **CODE**
This is either an ICD diagnosis or procedure code (Required) (This is the opposite of $$COD)

VARIABLES: **Output** **$$NUM**
This is a numeric representation of a code.

COMPONENT: **$$COD(NUM)**
This entry point converts a numeric representation of a code to a code (found on the AN cross-reference)

VARIABLES: **Input** **NUM**
This is a numeric representation of an ICD diagnosis or procedure code (This is the opposite of $$NUM)

VARIABLES: **Output** **$$COD**
This is an ICD diagnosis or procedure code.

COMPONENT: **$$IE(CODE)**
This entry point determines if a code is in an external or internal format without plusing (+) the code.

If you have an ICD-10 code with the letter "E in the center and plus it you will receive a MAXNUMBER error.

Example: If you plus (+) the ICD-10 procedure code "041E499" it will be interpreted as a scientific notation (E499 is a really big number). Applications that plus the ICD code can use this entry point to safely determine a code's format.

VARIABLES: **Input** **CODE**
This is either an ICD diagnosis or procedure code (Required)

VARIABLES: **Output** **$$IE**
This is a set of codes as follows:

I CODE is in an internal format (IEN)
E CODE is in an external format (Code)

or

Null on error

COMPONENT: **$$FILE(SYS)**
This entry point will return an ICD file number.

VARIABLES: **Input** **SYS**
This is a coding system, a global root or a file identifier.

Global roots ^ICD9(and ^ICD0(are acceptable

```
                    Coding Systems can be found in file 80.4 File
                    Identifier:  DX or PR
                                        DIAG or PROC or OPER

VARIABLES:  Output    $$FILE
                    This is an ICD file number 80 or 80.1

                    or -1 on error

COMPONENT:  $$ROOT(SYS)
            This entry point will return an ICD global root.

VARIABLES:  Input     SYS
                    This is a coding system, file number, a file
                    identifier or even an ICD code, provided the code
                    is unique to a file.

                    Coding Systems can be found in file 80.4 File
                    Number 80 or 80.1 File Identifier:  DX or PR
                                        DIAG or PROC or OPER

VARIABLES:  Output    $$ROOT
                    This is a global root ^ICD9( or ^ICD0( or Null on
                    error

COMPONENT:  $$SYS(SYS,CDT,FMT)
            This entry point will return a coding system.

VARIABLES:  Input     SYS
                    This can be either a Coding System name,
                    Abbreviation, system identifier (uses date) or a
                    code.

                    Coding System Names: ICD-9-CM, ICD-9 Proc,
                    ICD-10-CM or ICD-10-PCS

                    Coding System Abbreviations: ICD, ICP, 10D or 10P

                    System Identifier (with date CDT)

                      Date is before the ICD-10 implementation date

                        DIAG, ICD9, 80, DX =  1
                        PROC, OPER, ICD0, ICP9, 80.1, PR = 2

                      Date is on or after the ICD-10 implementation
                    date

                        DIAG, ICD9, 80, DX =  30
                        PROC, OPER, ICD0, ICP9, 80.1, PR = 31

                    An ICD code

                       If an ICD code is unique to an ABA
                       cross-reference then the Coding System
                       can be determined from a code
```

```
                    ^ICD9("ABA",1,(CODE_" "))   = 1
                    ^ICD9("ABA",30,(CODE_" "))  = 30
                    ^ICD9("ABA",2,(CODE_" "))   = 2
                    ^ICD9("ABA",31,(CODE_" "))  = 31

VARIABLES:  Input    CDT
              This is the Code Set Versioning date (Fileman
              format) used to determine the coding system based
              on a system identifier (Optional, if not provided
              then TODAY is used)

VARIABLES:  Input    FMT
              This is a single character identifying the desired
              output format (Optional, default is "I"):

                  I  Internal (default)
                  E  External
                  B  Both Internal ^ External

VARIABLES:  Output   $$SYS
              This is the Coding System in the format specified
              by the input parameter FMT:

                FMT=I     FMT=E        FMT=B
                Internal  External     Both

                  1       ICD-9-CM     1^ICD-9-CM
                  2       ICD-9 Proc   2^ICD-9 Proc
                 30       ICD-10-CM    30^ICD-10-CM
                 31       ICD-10-PCS   31^ICD-10-PCS

                 or

                  -1   on error

COMPONENT:  $$SINFO(SYS,CDT)
            This entry point returns coding system information taken from
            file 80.4.

VARIABLES:  Input    SYS
              This can be either a Coding System name,
              Abbreviation, system identifier, file number or a
              code.  (system identifier and code uses date).

              Coding System Names:

                ICD-9-CM
                ICD-9 Proc
                ICD-10-CM or
                ICD-10-PCS

              Coding System Abbreviations:

                ICD, ICP, 10D or 10P

              System Identifier/File Number (with date CDT)
```

```
                        Date is before the ICD-10 implementation date

                          DIAG, ICD9, 80, DX =  1
                          PROC, OPER, ICD0, ICP9, 80.1, PR = 2

                        Date is on or after the ICD-10 implementation
                      date

                          DIAG, ICD9, 80, DX =  30
                          PROC, OPER, ICD0, ICP9, 80.1, PR = 31

VARIABLES:  Input     CDT
                      This is the Code Set Versioning date (Fileman
                      format) used to determine the coding system based
                      on a system identifier (Optional, if not provided
                      then TODAY is used)

VARIABLES:  Output    $$SINFO
                      This is a 6 piece "^" delimited string

                          1      IEN to file 80.4
                          2      Coding System
                          3      Coding System Nomenclature
                          4      Coding system Abbreviation
                          5      File where the Coding System is stored
                          6      Implementation Date

                       or

                         -1   on error

COMPONENT:  $$SNAM(SYS)
            This entry point returns the coding system name.

VARIABLES:  Input     SYS
                      This is a pointer to the coding system file 80.4

VARIABLES:  Output    $$SNAM
                      This the coding system name, file 80.4 (.01)

                          ICD-9-CM
                          ICD-9 Proc
                          ICD-10-CM
                          ICD-10-PCS

                        Or  -1   on error

COMPONENT:  $$SAB(SYS,CDT)
            This entry point returns the coding system abbreviation.

VARIABLES:  Input     SYS
                      This can be either a Coding System name,
                      Abbreviation, system identifier (uses date) or a
                      code.

                      Coding System Names: ICD-9-CM, ICD-9 Proc,
                      ICD-10-CM or ICD-10-PCS
```

```
                    Coding System Abbreviations: ICD, ICP, 10D or 10P

                    System Identifier (with date CDT)

                       Date is before the ICD-10 implementation date

                         DIAG, ICD9, 80, DX =  1
                         PROC, OPER, ICD0, ICP9, 80.1, PR = 2

                       Date is on or after the ICD-10 implementation
                    date

                         DIAG, ICD9, 80, DX =  30
                         PROC, OPER, ICD0, ICP9, 80.1, PR = 31

VARIABLES:  Input    CDT
                      This is the Code Set Versioning date (Fileman
                      format) used to determine the source abbreviation
                      based on a system identifier (Optional, if not
                      provided then TODAY is used)

VARIABLES:  Output   $$SAB
                      3 Character Coding System abbreviation, file 80.4
                      (.02)

                          ICD
                          ICP
                          10D
                          10P

                        Or  -1   on error

COMPONENT:  $$EXC(FILE,IEN)
            This entry point returns a boolean value indicating if an
            entry in the specified file is to be excluded from lookup.  If
            it is to be excluded, then the entry will not be placed on the
            selection list for a user to select from.  Used primarily for
            the special lookup.

VARIABLES:  Input    FILE
                      This is an ICD file number:

                          80   = ICD Diagnosis file
                          80.1 = ICD Operation/Procedure file

VARIABLES:  Input    IEN
                      This is an Internal Entry Number (IEN) in the file
                      specified.

VARIABLES:  Output   $$EXC
                      Boolean value

                         1 = Yes, exclude from lookup
                         0 = No, include in the lookup

COMPONENT:  $$ISA(IEN1,IEN2,FIELD)
```

This entry point returns a boolean value indicating that one code is a "condition" of another. Conditions include:

Code 1 is Not Used With Code 2
Code 1 is Required With Code 2
Code 1 is Not Considered CC With Code 2

VARIABLES: Input IEN1
This is the internal entry number (IEN) of a code in file 80 that has a relationship with the code at IEN2 IEN1 is equivalent to Fileman's DA and identifies a code stored in a multiple in field 20, 30, 40 or pointed to by field 1.11.

VARIABLES: Input IEN2
This is the internal entry number (IEN) of a code in file 80 that may have other codes (IEN1) associated with it. IEN2 is equivalent to Fileman's DA(1) and identifies the code in the .01 field.

VARIABLES: Input FIELD
This is a field number in file 80 that contains one or more ICD codes that have a relationship to the main entry. Acceptable field numbers and the type of relationships to check include:

Field	Relationship
20	Code 1 Not Used With Code 2
30	Code 1 Required With Code 2
40 or 1.11	Code 1 Not Considered CC With Code 2

VARIABLES: Output $$ISA
This is a Boolean value

1 Yes/The relationship is True
0 No/The relationship is False

Field	Answers the Question
20	Code 1 (identified by IEN1) is not used with Code 2 (identified by IEN2)
30	Code 1 (identified by IEN1) is required with Code 2 (identified by IEN2)
40 or 1.11	Code 1 (identified by IEN1) is not considered Complication Comorbidity (CC) with Code 2 (identified by IEN2)

COMPONENT: $$EXIST(IEN,FIELD)
This entry point determines if special condition ICD codes exist.

```
VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the
                        DIAGNOSIS file 80 (Required)

VARIABLES:  Input     FIELD
                        This is a field number in file 80 that contains
                        one or more ICD codes that have a relationship to
                        the main entry (Required)  Acceptable field
                        numbers to check include:

                           20   Code Not Used With
                           30   Code Required With
                           40   Code Not Considered CC With

VARIABLES:  Output    $$EXIST
                        Boolean value

                           1   Yes/True, codes exist
                           0   No/False, codes do not exist

                           Field   Answers the Question
                           -----   ------------------------------
                           20      Are there any codes that should
                                   not be used with this code (IEN)
                           30      Are there any codes required
                                   with this code (IEN)
                           40      Are there any codes that are not
                                   considered CC with this code
                                   (IEN)

COMPONENT:  $$GETDRG(FILE,IEN,CDT,MDC)
            This entry point returns a string of DRGs for an ICD Diagnosis
            or Procedure code.

VARIABLES:  Input     FILE
                        This is the ICD file number used to retrieve the
                        DRGs (Required):

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the file
                        specified (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to identify the DRGs that were
                        appropriate on that date (Optional, if not passed
                        then TODAY is used)

VARIABLES:  Input     MDC
                        This is a Major Diagnostic Category (pointer to
                        file 80.3) used as a screen to limit the DRGs to
                        an MDC.  This input parameter only applies to the
                        ICD OPERATIONS/PROCEDURE file 80.1 which has
                        multiple MDCs, each with a possibility of multiple
```

```
                    DRGs  (Conditional)

VARIABLES:  Output    $$GETDRG
                    3 piece semi-colon ";" delimited string

                       1  DRGs delimited by ^
                       2  Fiscal Year
                       3  Status flag
                          0 inactive
                          1 active

                    Example output:

                       907^908^909^;3071001;1

                    On Error:

                       -1;No DRG level;0

COMPONENT:  MD(FILE,IEN,CDT,.ARY,FLAG)
            This entry point returns an array of Major Diagnostic
            Categories (MDCs) and Diagnosis Related Groups (DRGs)

VARIABLES:  Input     FILE
                    This is the ICD file number used to retrieve the
                    Major Diagnostic Categories (Required):

                        80   = ICD Diagnosis file
                        80.1 = ICD Operation/Procedure file

VARIABLES:  Input     IEN
                    This is an Internal Entry Number (IEN) in the file
                    specified (Required)

VARIABLES:  Input     CDT
                    This is the Code Set Versioning date (Fileman
                    format) used to identify the MDCs that were
                    appropriate on that date (Optional, if not passed
                    then TODAY is used)  NOTE:  If no Fiscal Year is
                    found for the input date then the first (earliest(
                    Fiscal Year is used.

VARIABLES:  Input     .ARY
                    This is a local array name passed by reference
                    that will contain a list of MDCs by effective date

VARIABLES:  Input     FLAG
                    This is a flag that determines the output format:

                       I   = Internal (default) Internal values
                             are always returned

                       E   = Include External values with Internal
                             values

VARIABLES:  Output    ARY
                    ICD Procedures file 80.1 (multiple MDC)
```

```
                        ARY(<fiscal year>,<MDC>)=DRG^;FY;STA
                        ARY(<fiscal year>,<MDC>)="DRG^DRG^;FY;STA

                        If Flag contains "E"

                        ARY(<fiscal year>,"E",<MDC>)=MDC Name
                        ARY(<fiscal year>,"E",<MDC>,<DRG>)=DRG Name
                        ARY(<fiscal year>,"E",<MDC>)=MDC Name
                        ARY(<fiscal year>,"E",<MDC>,<DRG>)=DRG Name
                        ARY(<fiscal year>,"E",<MDC>,<DRG>)=DRG Name
                        ARY(<fiscal year>,"E","FY")=External FY

                     ICD Diagnosis file 80 (single MDC)

                        ARY(<fiscal year>,<MDC>)="DRG^DRG^;FY;STA

                        If Flag contains "E"

                        ARY(<fiscal year>,"E",<MDC>)=MDC Name
                        ARY(<fiscal year>,"E",<MDC>,<DRG>)=DRG Name
                        ARY(<fiscal year>,"E",<MDC>,<DRG>)=DRG Name
                        ARY(<fiscal year>,"E","FY")=External FY

                     NOTE:  If no Fiscal Year found for the input date
                     then the first (earliest) Fiscal Year is used.

COMPONENT:  $$EFM(CDT)
            This entry point converts an external date to a Fileman
            internal date.  This entry point replaces unsupported
            $$DGY2K^DGPTOD0(X)

VARIABLES:  Input     CDT
                        External date (Required), examples of valid dates:

                           JAN 20 1957 or 20 JAN 57
                           1/20/57 or 012057
                           T   (for TODAY)
                           T+1 (for TOMORROW),  T+2,  etc.
                           T-1 (for YESTERDAY)
                           T-3W (for 3 WEEKS AGO), etc.

VARIABLES:  Output    $$EFM
                        Internal Fileman Date

                        or -1 on error

COMPONENT:  $$FY(CDT)
            This entry point returns the 4 digit fiscal year for a
            specified date.  This entry point replaces unsupported
            $$FY^DGPTOD0(X)

VARIABLES:  Input     CDT
                        This is an internal Fileman date.

VARIABLES:  Output    $$FY
                        This is a 4 digit fiscal year (YYYY) for the date
```

```
                    specified or null on error.

COMPONENT:  $$VMDCDX(IEN,CDT)
            This entry point returns the versioned Major Diagnostic Code
            for an ICD Diagnosis.

VARIABLES:  Input     IEN
                      This is an Internal Entry Number (IEN) in the
                      DIAGNOSIS file 80 (Required)

VARIABLES:  Input     CDT
                      This is the Code Set Versioning date (Fileman
                      format) used to identify the MDCs that was
                      appropriate on that date (Optional, if not passed
                      then TODAY is used)

VARIABLES:  Output    $$VMDCDX
                      This is a single MDC (pointer to file 80.3) active
                      on the date specified.

COMPONENT:  $$VMDCOP(IEN,MDC,CDT)
            This entry point returns the versioned Major Diagnostic Codes
            for an ICD Procedure.

VARIABLES:  Input     IEN
                      This is an Internal Entry Number (IEN) in the
                      OPERATION/PROCEDURE file 80.1 (Required)

VARIABLES:  Input     MDC
                      This is a Major Diagnostic Category (pointer to
                      file 80.3) used as a screen to limit the results
                      to a single MDC (Required)

VARIABLES:  Input     CDT
                      This is the Code Set Versioning date (Fileman
                      format) used to identify the MDC that was
                      appropriate on that date (Optional, if not passed
                      then TODAY is used)

VARIABLES:  Output    $$VMDCOP
                      4 piece "^" delimited string

                         1   Fiscal Year, Fileman format
                         2   MDC, pointer to file 80.3
                         3   Fiscal Year, pointer to sub-file
                             80.171 (formerly DADRGFY)
                         4   MDC, pointer to sub-file 80.1711
                             (formerly DAMDC)

COMPONENT:  MDCG(IEN,CDT,.ARY)
            This entry point sets up an array of MDCs (later used in
            $$MDCT)

VARIABLES:  Input     IEN
                      This is an Internal Entry Number (IEN) in the
                      DIAGNOSIS file 80 (Required)
```

```
VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to identify the MDCs that were
                        appropriate on that date (Optional, if not passed
                        then TODAY is used)

VARIABLES:  Input     .ARY
                        This is a local array name passed by reference
                        that will contain a list of MDCs (Required)

VARIABLES:  Output    ARY
                        This is an array listing MDCs for all DRGs
                        associated with a diagnosis on the date specified.

                           ARY(MDC)=""
                           ARY(MDC)=""

COMPONENT:  $$MDCT(IEN,CDT,.ARY,FMT)
            This entry point compares a single entry in the ICD
            OPERATIONS/PROCEDURE file 80.1 to an array of Major Diagnostic
            Categories to see if the ICD procedure is assigned to one or
            more of the MDCs in the array.

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the
                        OPERATIONS/PROCEDURE file 80.1 (Required)

VARIABLES:  Input     CDT
                        This is the Code Set Versioning date (Fileman
                        format) used to identify the MDCs that were
                        appropriate on that date (Optional, if not passed
                        then TODAY is used)

VARIABLES:  Input     .ARY
                        This is a local array passed by reference
                        containing a list of MDCs for comparison
                        (Required)

VARIABLES:  Input     FMT
                        This is a flag defining the output format
                        (optional):

                           0   Boolean value only (default)
                           1   2 piece "^" delimited string

                               1   Boolean value
                               2   String of matching MDCs
                                   delimited by ";"

VARIABLES:  Output    $$MDCT
                        Boolean value

                           0  The ICD Procedure code identified by IEN
                              does not include any of the MDCs passed
                              in .ARY(MDC) on the date specified (CDT)

                           1  The ICD Procedure code identified by IEN
```

```
                              includes one or more of the MDCs passed
                              in .ARY(MDC) on the date specified (CDT)

                         Assuming the following input parameters:

                            IEN=4
                            CDT=3111110
                            ARY(2)=""
                            ARY(21)=""

                            Output format when input parameter FMT=0
                            (default)

                               $$MDCT(IEN,CDT,.ARY) = "1"

                            Output format when input parameter FMT=1

                               $$MDCT(IEN,CDT,.ARY) = "1^2;21"

COMPONENT:   $$MDCD(IEN,MDC,CDT)
             This entry point checks for a Major Diagnostic Category MDC in
             the ICD OPERATION/PROCEDURE file.

VARIABLES:   Input     IEN
                         This is an Internal Entry Number (IEN) in the
                         OPERATIONS/PROCEDURE file 80.1 (Required)

VARIABLES:   Input     MDC
                         This is a Major Diagnostic Category (pointer to
                         file 80.3) (Required)

VARIABLES:   Input     CDT
                         This is the Code Set Versioning date (Fileman
                         format) used to identify the MDCs that were
                         appropriate on that date (Optional, if not passed
                         then the first FY is used)

VARIABLES:   Output    $$MDCD
                         Boolean value

                            0  MDC does not exist on date specified
                            1  MDC exist on date specified

COMPONENT:   $$MOR(IEN)
             This entry point returns the Major O.R. Procedure string

VARIABLES:   Input     IEN
                         This is an Internal Entry Number (IEN) in the
                         OPERATIONS/PROCEDURE file 80.1 (Required)

VARIABLES:   Output    $$MOR
                         Major O.R. Procedure or Null if the procedure is
                         not defined as a Major O.R. Procedure or is not
                         found

                         Major O.R. Procedure definitions include:
```

```
                    1   Bowel                  2  Chest          3
                    Lymphoma/Leukemia     4  Joint 5   Pancreas/Liver
                         6  Pelvic 7   Shoulder/Elbow        8
                    Thumb/Joint 9   Head/Neck             A  Cardio M
                    Musculoskeletal       B  Spine

COMPONENT:  $$UPDX(IEN)
            This entry point determines if a diagnosis is unacceptable as
            a principle diagnosis.

VARIABLES:  Input     IEN
                      This is an Internal Entry Number (IEN) in the
                      DIAGNOSIS file 80 (Required)

VARIABLES:  Output    $$UPDX
                      Boolean value, answers the question:

                         Is the diagnosis UNACCEPTABLE as a Principle
                      DX?

                         1   Yes  Code is Unacceptable as Principle DX
                         0   No   Code is Acceptable as Principle DX

COMPONENT:  $$NOT(IEN,SUB,FMT)
            This entry point returns the number of ICD codes that cannot
            be used with a specified code.  It can also return a global
            array containing a list of the codes that cannot be used with
            the specified code.

VARIABLES:  Input     IEN
                      This is an Internal Entry Number (IEN) in the
                      DIAGNOSIS file 80 (Required)

VARIABLES:  Input     SUB
                      This is a subscript name used in a ^TMP global
                      array (Optional, if not provided, the subscript
                      "ICDNOT" will be used)

                          ^TMP(SUB,$J)

VARIABLES:  Input     FMT
                      This is a flag defining the output format.

                        0 - Total number only (default)
                        1 - Total number with global array

VARIABLES:  Output    $$NOT
                      The number of ICD codes that cannot be used with
                      the ICD code identified by IEN (FMT=0 or 1)

                      TMP global array as follows (FMT=1):

                         ^TMP(SUB,$J,IEN)=CODE
                         ^TMP(SUB,$J,"B",(CODE_" "),IEN)=""

COMPONENT:  $$REQ(IEN,SUB,FMT)
            This entry point returns the number of ICD codes that are
```

```
          required when the specified code is used.  It can also return
          a global array containing a list of the codes that are
          required when the specified code is used.

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the
                        DIAGNOSIS file 80 (Required)

VARIABLES:  Input     SUB
                        This is a subscript name used in a ^TMP global
                        array (Optional, if not provided, the subscript
                        "ICDREQ" will be used)

                            ^TMP(SUB,$J)

VARIABLES:  Input     FMT
                        This is a flag defining the output format.

                          0 - Total number only (default)
                          1 - Total number with global array

VARIABLES:  Output    $$REQ
                        The number of ICD codes required when the ICD code
                        identified by IEN is used. (FMT=0 or 1)

                        TMP global array as follows (FMT=1):

                           ^TMP(SUB,$J,IEN)=CODE
                           ^TMP(SUB,$J,"B",(CODE_" "),IEN)=""

COMPONENT:  $$NCC(IEN,SUB,FMT)
          This entry point returns the number of ICD codes that are not
          considered CC with a specified code.  It can also return a
          global array containing a list of the codes that are not
          considered CC with a specified code.

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the
                        DIAGNOSIS file 80 (Required)

VARIABLES:  Input     SUB
                        This is a subscript name used in a ^TMP global
                        array (Optional, if not provided, the subscript
                        "ICDNCC" will be used)

                            ^TMP(SUB,$J)

VARIABLES:  Input     FMT
                        This is a flag defining the output format.

                          0 - Total number only (default)
                          1 - Total number with global array

VARIABLES:  Output    $$NCC
                        The number of ICD codes not considered CC with the
                        code identified by IEN. (FMT=0 or 1)
```

```
                    TMP global array as follows (FMT=1):

                       ^TMP(SUB,$J,IEN)=CODE
                       ^TMP(SUB,$J,"B",(CODE_" "),IEN)=""

COMPONENT:  LK
            Special Lookup (called by DIC)

            This is the Special Lookup program for files 80 and 80.1.
            Only the ^DIC call honors the special lookup routines.  Those
            calls that allow the user to specify the indexes (IX^DIC and
            MIX^DIC1), and the Data Base Server calls (FIND^DIC,
            $$FIND1^DIC, and UPDATE^DIE) all ignore the Special Lookup
            Program.  Also, if DIC(0) contains an "I" then the Special
            Lookup program will be ignored.

            This routine uses a majority of the variables used in calling
            Fileman ^DIC.  In addition to the Fileman variables, there are
            three special variables that aid in controlling the lookup
            that can be set and killed by the calling application;

              Versioning Date (Fileman format)

                 ICDVDT or
                 ^TMP("ICDEXLK",$J,"ICDVDT")=<versioning date>

              Coding System (from file 80.4)

                 ICDSYS or
                 ^TMP("ICDEXLK",$J,"ICDSYS")=<coding system>

              Display Format (numeric, 1-4) (new)

                 ICDFMT or
                 ^TMP("ICDEXLK",$J,"ICDFMT")=<display format>

VARIABLES:  Input     ICDVDT
                        Versioning Date (Fileman format)

                          ICDVDT or
                          ^TMP("ICDEXLK",$J,"ICDVDT")=<date>

                        This is a Code Set Versioning Date (in Fileman
                        format).  If set, it must also be killed by the
                        calling application.

                        If supplied, it is assumed that the lookup is to
                        be a versioned lookup and only active codes on
                        that date will be included in the selection list.

                        If not supplied, the date will default to TODAY
                        and all codes may be selected, active and
                        inactive.

                        In both cases the display will be altered based on
                        the date.
```

```
VARIABLES:  Input     ICDSYS
                      Coding System (from file 80.4)

                        ICDSYS or
                        ^TMP("ICDEXLK",$J,"ICDSYS")=<coding system>

                      This is the Coding System taken from file 80.4.
                      If set, it must be killed by the calling
                      application.  It may be any of the following:

                          1    ICD    ICD-9-CM
                          2    ICP    ICD-9 Proc
                         30    10D    ICD-10-CM
                         31    10P    ICD-10-PCS

                      If supplied, the lookup will only look in the
                      cross-references specific for that coding system.

VARIABLES:  Input     ICDFMT
                      Display Format (numeric, 1-4)

                        ICDFMT or
                        ^TMP("ICDEXLK",$J,"ICDFMT")=<display format>

                      This is a flag defining a Display Format (numeric,
                      1-4).  If set, it must be killed by the calling
                      application.

                      1 = Fileman format, code and short text
                          (default)

                          250.00    DMII WO CMP NT ST UNCNTR

                      2 = Fileman format, code and description

                          250.00    DIABETES MELLITUS WITHOUT
                                    MENTION OF COMPLICATION,
                                    TYPE II OR UNSPECIFIED TYPE,
                                    NOT STATED AS UNCONTROLLED

                      3 = Lexicon format, short text followed
                          by code

                          DMII WO CMP NT ST UNCNTR (250.00)

                      4 = Lexicon format, description followed
                          by code

                          DIABETES MELLITUS WITHOUT MENTION OF
                          COMPLICATION, TYPE II OR UNSPECIFIED
                          NOT STATED AS UNCONTROLLED (250.00)


VARIABLES:  Input     X
                      This is the user's input, if not available the
                      user will be prompted for input.
```

```
VARIABLES:  Input     FILEMAN
                      FileMan Variables used

                         DIC, DIC(0), DIC("A"), DIC("B"),
                         DIC("S"), DIC("W"), DIC("?N",<file>)

                      FileMan Variables not used:

                         DIC("DR"),DIC("PTRIX",<fm>,<to>,<file>),
                         DIC("T"), DIC("V"), DIC("?PARAM")

                      DIC(0) parameters applicable to a versioned file

                         A   Ask the entry; if erroneous, ask again
                         B   Only the B index is used
                         E   Echo information
                         F   Forget the lookup value
                         I   Ignore the special lookup program
                         M   Multiple-index lookup allowed
                         O   Only find one entry if it matches exactly
                         S   Suppresses display of .01
                         T   Search until user selects or enters ^^
                         X   EXact match required
                         Z   Zero node in Y(0), external form in Y(0,0)

                      DIC(0) parameters NOT applicable to a versioned
                      file and not used

                         C   Versioned cross-references not turned off
                         K   Primary Key not established
                         L   Learning a new entry LAYGO not allowed
                         N   Uppercase, IEN lookup allowed (not forced)
                         n   ICD has no pure numeric entries
                         Q   Input is pre-processed, ?? not necessary
                         U   All values are external
                         V   Verification is not optional

                      FileMan Variables KILLed:

                         DLAYGO
                         DINUM


VARIABLES:  Output    Y
                      Fileman Compliant:

                         Y        IEN ^ Code

                         If DIC(0) containing "Z"

                         Y(0)     0 Node
                         Y(0,0)   Code

                      Non-Fileman Compliant, DIC(0) contains "Z"

                         Y(0,1)   $$ICDDX or $$ICDOP
                         Y(0,2)   Long Description
```

COMPONENT: **$$LKTX(X,ROOT,CDT,SYS,VER,OUT)**
This entry point is a lookup for text in either file 80 or 80.1 It is similar to the special lookup except there is no prompt for input or display for selection (silent) and intended for GUI applications.

VARIABLES: **Input** **X**
This is a string of text to search for.

VARIABLES: **Input** **ROOT**
This is either a global root or file number to indicate either the DIAGNOSIS file 80 or the OPERATIONS/PROCEDURE file 80.1

VARIABLES: **Input** **CDT**
This is the Code Set Versioning date (Fileman format) used to determine the status of a code (active or inactive) It normally represents the date that service was provided to the patient (HIPAA). However, it may also represent the date of onset, visit date or movement date depending on the application calling the lookup.

VARIABLES: **Input** **SYS**
This is a coding system identifier (pointer to file 80.4)

1 = ICD-9-CM
2 = ICD-9-PCS
30 = ICD-10-CM
31 = ICD-10-PCS

VARIABLES: **Input** **VER**
This is the versioned flag (boolean) to indicate if the lookup is to be versioned or not:

0 No Include all codes, active and inactive

1 Yes Include only Active codes for date specified

VARIABLES: **Input** **OUT**
This is a flag that defines the output format:

1 Fileman, Code and Short Text (default)

250.00 DMII WO CMP NT ST UNCNTR

2 Fileman, Code and Description

250.00 DIABETES MELLITUS WITHOUT MENTION OF COMPLICATION TYPE II OR UNSPECIFIED TYPE, NOT STATED AS UNCONTROLLED

```
                        3  Lexicon, Short Text and Code

                           DMII WO CMP NT ST UNCNTR (250.00)

                        4  Lexicon, Description and Code

                           DIABETES MELLITUS WITHOUT MENTION OF
                           COMPLICATION, TYPE II OR UNSPECIFIED,
                           NOT STATED AS UNCONTROLLED (250.00)

VARIABLES:  Output    $$LK
                        This is the number of entries found

                        The entries will be included in a ^TMP Global
                        Array:

                           ^TMP(ID,$J,"SEL")
                           ^TMP(ID,$J,"SEL",0)=# of entries
                           ^TMP(ID,$J,"SEL",#)=IEN ^ Display Text

                          Where ID is a package namespaced subscript:

                           ICD9 - for file #80
                           ICD0 - for file #80.1

COMPONENT:  $$VER(SYS,REL)
            This API returns the current Coding System version, the
            previous Coding System version or the next Coding System
            version based on input parameters.

VARIABLES:  Input     SYS
                        This is a pointer to the coding system file 80.4

VARIABLES:  Input     REL
                        This input parameter indicates the relationship of
                        the output coding system to the input coding
                        system (Optional)

                           0   N/A - Return the current version (default)
                           1   Return the next version
                          -1   Return the previous version

VARIABLES:  Output    $$VER
                        This is a 5 piece string containing:

                             1   Coding System (pointer to file 80.4)
                             2   Coding System Nomenclature
                             3   Coding System Abbreviation
                             4   File Number containing the Coding System
                             5   Date Coding System was Implemented or
                            -1   on error

COMPONENT:  Y(ROOT,IEN,CDT,FMT)
            Given the global root or file number, the Internal Entry
            Number (IEN) and a date, this API will return the equivalent
            of FileMan's output variable Y without having to perform the
            lookup.
```

```
VARIABLES:  Input     ROOT
                        This is either an ICD global root or file number.

VARIABLES:  Input     IEN
                        This is an Internal Entry Number in the file
                        identified by the input parameter ROOT.

VARIABLES:  Input     CDT
                        This is a code set versioning date used to
                        returned versioned (date sensitive) data from the
                        ICD files.

VARIABLES:  Input     FMT
                        This is a output format flag (optional, default
                        0).

                          0  Return standard Fileman Y - IEN ^ CODE
                          1  Return Expanded Y as if DIC(0) contained a
                        "Z"

VARIABLES:  Output    Y
                        Input parameter FMT = 0 or 1

                          Y = IEN ^ Code

                        Input parameter FMT = 1

                          FileMan Compliant

                            Y(0)    =  0 Node    (aka Code)
                            Y(0,0)  =  .01 Field (aka Code)

                          Non-FileMan Compliant

                            Y(0,1)  =  $$ICDDX or $$ICDOP
                            Y(0,2)  =  Versioned Long Description

COMPONENT:  TOKEN(TEXT,ROOT,SYS,ARY)
            This API parses text into words/tokens and saves them in a
            local array for later processing.  Words and tokens not found
            in the file and coding system identified by the input
            parameters are not included in the output array.

VARIABLES:  Input     TEXT
                        This is a text string to parse.

VARIABLES:  Input     ROOT
                        This is a global root or file number (required)

                           ^ICD9(  or  80
                           ^ICD0(  or  80.1

VARIABLES:  Input     SYS
                        This is the coding system  (Required)

                            1  or  ICD  or  ICD-9-CM
```

```
                          2  or  ICP  or  ICD-9 Proc
                         30  or  10D  or  ICD-10-CM
                         31  or  10P  or  ICD-10-PCS

VARIABLES:  Both      .ARY
                       This is the output array passed by reference that
                       contains a list of words parsed from the input
                       string X and arranged by frequency of use

                          ARY(0)=# of words
                          ARY(#)=word

                       The least frequently used word will be ARY(1) and
                       the most frequently used word will be ARY($O(ARY("
                       "),-1)).  Words not found in the file and coding
                       system will not appear in the parsed array.

COMPONENT:  $$WORD(WORD,ROOT,SYS)
            This API determines if a word is found in a file or a coding
            system identified by the input parameters

VARIABLES:  Input     WORD
                       This is a single word.

VARIABLES:  Input     ROOT
                       This is a global root or file number (optional)

                          ^ICD9(  or  80
                          ^ICD0(  or  80.1

VARIABLES:  Input     SYS
                       This is the coding system (Optional)

                           1  or  ICD  or  ICD-9-CM
                           2  or  ICP  or  ICD-9 Proc
                          30  or  10D  or  ICD-10-CM
                          31  or  10P  or  ICD-10-PCS

VARIABLES:  Output    $$WORD
                       This is a Boolean value indicating if a word is
                       contained in a set (file or system).

                         1 = Word was found

                             If ROOT is not supplied, the word was
                             found in either file 80 or 80.1

                             If SYS is not supplied, the word was
                             found in the file designated by ROOT
                             in any coding system in the file

                             If both ROOT and SYS are supplied,
                             the word was found in the specified
                             coding system

                         0 = Word was not found
```

```
COMPONENT:  $$ICDIDS(FILE,CODE,ARY)
            This API returns an array of Diagnosis or Procedure code
            Identifiers used in the calculation of DRG groups.

VARIABLES:  Input     FILE
                        This is the ICD file number used to retrieve the
                        identifier codes (Required):

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     CODE
                        This is an Internal Entry Number (IEN) in the file
                        specified (Required).

VARIABLES:  Both      ARY
                        This is a local array of identifiers found for the
                        code identified input parameters FILE and CODE.

                               ARY(<identifier>)=""

VARIABLES:  Output    $$ICDIDS
                        This is the number of identifiers found for the
                        code identified by the input parameters FILE and
                        CODE, or upon error:

                            -1^error message

COMPONENT:  $$ICDID(FILE,ID,CODE)
            This API checks if a specified ICD identifier exist for a code
            identified by the input parameters FILE and CODE.

VARIABLES:  Input     FILE
                        This is the ICD file number used to retrieve the
                        identifier codes (Required):

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     ID
                        This is a Diagnosis or Procedure code identifier
                        (required)

VARIABLES:  Input     CODE
                        This is an Internal Entry Number (IEN) in the file
                        specified (Required).

VARIABLES:  Output    $$ICDID
                        Boolean value

                           1 if identifier was found for code
                           0 if identifier was not found for code

                           or upon error -1^error message

COMPONENT:  $$ISOWNCC(IEN,CDT,FMT)
            This API returns the Complication/Comorbidity (CC) value for
```

```
          an ICD Diagnosis code when the primary diagnosis is its own
          CC/MCC.

VARIABLES:  Input     IEN
                        This is the Internal Entry Number (IEN) of the ICD
                        Diagnosis file #80.

VARIABLES:  Input     CDT
                        Date to use to extract CC (default TODAY)

VARIABLES:  Input     FMT
                        This is a flag that controls the output format:

                           0 = CC only (default)
                           1 = CC ^ Effective Date

VARIABLES:  Output    $$ISOWNCC
                        Complication/Comorbidity (CC)

                           DX is Own CC  Format  Output
                           ------------  ------  -------------------------
                                Yes        0     CC Value
                                Yes        1     CC Value ^ Effective Date
                                No        N/A    0 (zero)

                           or upon error -1^error message

COMPONENT:  $$ICDRGCC(DRG,CDT)
          This API returns the CC/MCC flag from DRG file #80.2

VARIABLES:  Input     DRG
                        This is an Internal Entry Number for the DRG file
                        80.2 (required)

VARIABLES:  Input     CDT
                        Date to use to extract CC/MCC flag (default TODAY)

VARIABLES:  Output    $$ICDRGCC
                        This is the Complication/Comorbidity/Major CC flag

                            0   No CC or MCC
                            1   CC present
                            2   MCC present
                            3   CC or MCC present

                        or upon error -1^error message

COMPONENT:  $$DRG(CODE,CDT)
          This API returns basic information about a DRG.

VARIABLES:  Input     CODE
                        DRG code, internal or external format (Required)

VARIABLES:  Input     CDT
                        Date to check status for, FileMan format (default
                        = TODAY)
```

```
                        If CDT < 10/1/1978, use 10/1/1978
                        If CDT > DT, validate with In/Activation Dates
                        If CDT is year only, use first of the year
                        If CDT is year and month, use first of the
                      month

VARIABLES:  Output    $$DRG
                      Returns an 22 piece string delimited by the
                      up-arrow (^) the pieces are:

                         1  DRG name (field #.01)
                         2  Weight (field #2)
                         3  Low Trim (days) (field #3)
                         4  High Trim (days) (field #4)
                         5  MDC (field #5)
                         6  Surgery Flag (field #.06)
                         7  <null>
                         8  Avg Length of Stay (days) (field 10)
                         9  Local Low Trim Days (field #11)
                        10  Local High Trim Days (field #12)
                        11  <null>
                        12  Local Breakeven (field #13)
                        13  Activation Date (.01 field, 66 multiple)
                        14  Status (.03 field, 66 multiple)
                        15  Inactivation Date (.01 field, 66 multiple)
                        16  Effective date (.01 field, 66 multiple)
                        17  Internal Entry Number (IEN)
                        18  Effective date (.01 field, 66 multiple)
                        19  Reference (field #900)
                        20  Weight (Non Affil) (field #7)
                        21  Weight (Int Affil) (field #7.5)
                        22  Message

                         or

                        -1^Error Description

COMPONENT:  $$DRGD(CODE,ARY,CDT)
            Returns an unformatted DRG Description.

VARIABLES:  Input     CODE
                      ICD Code, Internal or External Format (required)

VARIABLES:  Both      ARY
                      Input:   Name of Output Array for description

                        e.g. "ABC" or "ABC("TEST")"

                        Default = ^TMP("DRGD",$J)

                      Output:  Description in array

                         @ARY(1:n) - Description (lines 1-n)
                         @ARY(n+1) - Blank
                         @ARY(n+1) - Warning Message

                           or
```

```
                        -1^Error Description

                      NOTE:

                         User must initialize ^TMP("DRGD",$J)
                         if used.  The data is place in the
                         array unformatted, exactly as it is in
                         the DESCRIPTION multiple (sub-files
                         #80.068 or #80.168)

                      SEE ALSO:

                         $$DRGDES^ICDEX(IEN,CDT,.ARY,LENGTH) to
                         retrieve the description formatted into
                         string lengths specified by input
                         parameter for length.

VARIABLES:  Input     CDT
                        Date to screen against (default = TODAY)

                           If CDT < 10/1/1978, use 10/1/1978
                           If CDT > DT, use DT
                           If CDT = year only, use 01/01/yyyy
                           If CDT = year & month, use mm/01/yyyy

VARIABLES:  Output    $$DRGD
                        This is the number of lines in description output
                        array.

COMPONENT:  $$DRGDES(IEN,CDT,ARY,LEN)
            This API returns the DRG Description formatted into string
            lengths specified by the calling application.

VARIABLES:  Input     IEN
                        Internal Entry Number of DRG file 80.2

VARIABLES:  Input     CDT
                        Date to screen against (default = TODAY)

VARIABLES:  Both      .ARY
                        This is a local array passed by reference
                        containing the DRG description.  The text is
                        formatted into string lengths specified by the LEN
                        input parameter.

VARIABLES:  Input     LEN
                        Length of line of the description in the output
                        array

                           Missing        Defaults to 79
                           Less than 25   Defaults to 25

VARIABLES:  Output    $$DRGDES
                        This is the number of lines in description output
                        array.
```

```
COMPONENT:  $$DRGN(CODE)
            This API returns the Internal Entry Number (IEN) of the DRG
            specified by a DRG code.

VARIABLES:  Input     CODE
                        This is a DRG code.

VARIABLES:  Output    $$DRGN
                        This is the IEN of the DRG code specified.

COMPONENT:  $$EFD(X)
            This is an interactive API that will prompt the user for an
            effective date in a range of dates.

VARIABLES:  Output    $$EFD
                        This is a 3 piece "^" delimited string containing
                        an effective date in both internal and external
                        formats:

                           1   Date Fileman format         nnnnnnn
                           2   Date External Short Format  mm/dd/yyyy
                           3   Date External Long Format   Mmm dd, yyyy

                        or
                           "^^" if the user enters double up-arrows
                           "^"  if the user enters a single up-arrow
                           ""   if the user times out

                        The earliest possible date is Oct 1, 1978, the
                        initial ICD implementation date in the VA.

                        If today's date is less than the implementation
                        date of ICD-10, then the latest possible date is 3
                        years from the ICD-10 implementation date.

                        If today's date is greater than the implementation
                        date of ICD-10, then the latest possible date is 3
                        years from today's date.

COMPONENT:  $$GETDATE(IEN)
            This API calculates the Effective Date to use retrieving
            ICD/DRG data based on a patient's treatment.

VARIABLES:  Input     IEN
                        This is an Internal Entry Number of the PTF file
                        #45

VARIABLES:  Output    $$GETDATE
                        This is the correct "EFFECTIVE DATE" for a patient
                        to be used retrieving DRG/ICD/CPT data (default
                        TODAY)

                        "EFFECTIVE DATE" Derived from:

                          Census Date      ^DGPT         0;13
                          Discharge Date   ^DG(45.86     0;1
                          Surgery Date     ^DGPT(D0,"S"  0;1
```

```
                        Movement Date     ^DGPT(D0,"M"  0;10
                        Default           $$NOW^XLFDT

COMPONENT:  $$IA(FILE,IEN)
            This API returns an codes Initial Activation Date based on a
            file number and the codes Internal Entry Number.  The Initial
            Activation date may be different from the Last Activation date
            (see $$LA) if the code was re-used.

VARIABLES:  Input      FILE
                         This is a Global Root or File Number for either
                         the ICD Diagnosis or ICD Procedure files
                         (Required)

VARIABLES:  Input      IEN
                         This is an Internal Entry Number (IEN) in the
                         specified file (Required)

VARIABLES:  Output     $$IA
                         Initial Activation Date

                         OR

                         -1 ^ Error Message

COMPONENT:  $$IDSTR(FILE,IEN)
            This API returns a string of ICD identifier associated with
            either an ICD Diagnosis or ICD Procedure code (supports legacy
            APIs)

VARIABLES:  Input      FILE
                         File Number or root (required)

                             80 or ^ICD9      = File #80
                             80.1 or ^ICD0    = File #80.1

VARIABLES:  Input      IEN
                         This is a Diagnosis/Procedure code IEN (required)

VARIABLES:  Output     $$IDSTR
                         This is a string of Identifiers delimited by a
                         semi-colon

                            ID;ID;ID

COMPONENT:  $$ISVALID(FILE,IEN,CDT)
            This API determine is an ICD code is valid.

VARIABLES:  Input      FILE
                         This is a file number or global root for either
                         the ICD Diagnosis file or the ICD Procedure file

VARIABLES:  Input      IEN
                         This is an Internal Entry Number (IEN) in the file
                         specified.

VARIABLES:  Input      CDT
```

```
                    This is the date to use to determine if the code
                    is valid for date (default TODAY)

VARIABLES:  Output   $$ISVALID
                    This is a Boolean value

                       1 if the code is valid
                       0 if the code is not valid

COMPONENT:  $$PDXE(IEN)
            This API returns the Primary Diagnosis Exclusion Code.

VARIABLES:  Input    IEN
                    This is an Internal Entry Number (IEN) for the ICD
                    Diagnosis file #80

VARIABLES:  Output   $$PDXE
                    This is a pointer to DRG CC Exclusions file #82.13

COMPONENT:  $$REF(IEN,CDT)
            This API returns the name of the DRG Reference Table.

VARIABLES:  Input    IEN
                    This is an Internal Entry Number (IEN) of the DRG
                    file #80.2

VARIABLES:  Input    CDT
                    Effective date to use (default TODAY)

VARIABLES:  Output   $$REF
                    Table reference associated with a DRG entry or
                    null if not found

COMPONENT:  $$VCCP(IEN,CDT,FMT)
            This API returns the CC Primary Flag for a diagnosis.

VARIABLES:  Input    IEN
                    This is an Internal Entry Number (IEN) in the ICD
                    Diagnosis file 80 (required)

VARIABLES:  Input    CDT
                    This is the date to use to Extract CC Primary Flag
                    (default TODAY)

VARIABLES:  Input    FMT
                    Is is a flag to determine the output format
                    (optional):

                     0 = CC Primary Flag only (default)
                     1 = CC Prim Flag^Effective Date^Value

VARIABLES:  Output   $$VCCP
                    This the CC Primary Flag in one of two formats:

                      CC Primary Flag only (FMT=0)
                      CC Primary Flag^Effective Date^Value  (FMT=1)
```

```
COMPONENT:  $$DRGW(IEN)
            This API returns the DRG Weighted Work Unit (WWU)

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) of the DRG
                        file 80.2

VARIABLES:  Output    $$DRGW
                        This is the Weighted Work Unit (WWU) for a DRG

COMPONENT:  $$DRGC(IEN)
            This API returns the DRG code.

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) of the DRG
                        file 80.2

VARIABLES:  Output    $$DRGC
                        This is a DRG Code (field .01)

COMPONENT:  $$MDCN(IEN)
            This API returns the name of a Major Diagnostic Category (MDC)

VARIABLES:  Input     IEN
                        This is the Internal Entry Number (IEN) for file
                        80.3

VARIABLES:  Output    $$MDCN
                        This is a Major Diagnostic Category Name

COMPONENT:  $$HDR(FILE)
            This API returns the header node of either file 80 or 80.1.

VARIABLES:  Input     FILE
                        This is a File Number or Global Root

                           80    or  ^ICD9(
                           80.1  or  ^ICD0(

VARIABLES:  Output    $$HDR
                        This is the header node of either the ICD
                        Diagnosis file 80 or the Operation Procedure file
                        80.1

                          ^ICD9(0)
                          ^ICD0(0)

COMPONENT:  $$IEN(CODE,ROOT,SYS)
            This API returns an internal entry number for a code based on
            file/global root and coding system.

            This API is similar to $$CODEABA^ICDEX except it will also
            return IENs for codes excluded from lookup and VA Local Codes.
            Its primary purpose to support file maintenance.  Use with
            great caution.

            DO NOT USE in any application that requires codes and text to
```

```
           be versioned (date sensitive).

VARIABLES:  Input     CODE
                        This is an ICD Diagnosis or Procedure Code from
                        either the ICD-9 or ICD-10 coding systems
                        (required)

VARIABLES:  Input     ROOT
                        This is a file number or global root (optional)

                           ^ICD9( or 80
                           ^ICD0( or 80.1

VARIABLES:  Input     SYS
                        This is a coding system (optional)

                          1 = ICD-9 Diagnosis
                          2 = ICD-9 Procedure
                         30 = ICD-10 Diagnosis
                         31 = ICD-10 Procedure

VARIABLES:  Output    $$IEN
                        Returns the Internal Entry Number (IEN) for a CODE
                        or -1 if not found

COMPONENT:  $$SDH(FILE,IEN,ARY)
            This API returns a history of Short Description changes by
            date.

VARIABLES:  Input     FILE
                        This is an ICD file number:

                            80   = ICD Diagnosis file
                            80.1 = ICD Operation/Procedure file

VARIABLES:  Input     IEN
                        This is an Internal Entry Number (IEN) in the file
                        specified.

VARIABLES:  Input     .ARY
                        This is a local array name passed by reference
                        that will contain the code's short description
                        history.

VARIABLES:  Output    $$SDH
                        This is a three piece "^" delimited string
                        containing:

                          1   The number of short descriptions found
                          2   The earliest date found
                          3   The latest date found

VARIABLES:  Output    ARY
                        This is a local array containing a history of
                        Short Descriptions by date:

                           ARY(0)= # ^ Earliest Date ^ Latest Date
```

```
                          ARY(DATE)=Long Description

   COMPONENT:  $$LDH(FILE,IEN,ARY)
               This API returns a history of Long Description changes by
               date.

   VARIABLES:  Input     FILE
                         This is an ICD file number:

                              80   = ICD Diagnosis file
                              80.1 = ICD Operation/Procedure file

   VARIABLES:  Input     IEN
                         This is an Internal Entry Number (IEN) in the file
                         specified.

   VARIABLES:  Input     .ARY
                         This is a local array name passed by reference
                         that will contain the code's long description
                         history.

   VARIABLES:  Output    $$LDH
                         This is a three piece "^" delimited string
                         containing:

                           1   The number of long descriptions found
                           2   The earliest date found
                           3   The latest date found

                         OR -1 ^ Error Message

   VARIABLES:  Output    ARY
                         This is a local array containing a history of Long
                         Descriptions by date:

                            ARY(0)= # ^ Earliest Date ^ Latest Date
                            ARY(DATE)=Long Description
```

5755 ^ICDS Lexicon

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: LEXICON UTILITY
              USAGE: Private             ENTERED: DEC 24,2011
             STATUS: Pending             EXPIRES:
           DURATION: Till Otherwise Agr  VERSION:
               FILE: 80.4                   ROOT: ICDS(
        DESCRIPTION:                        TYPE: File
   Lexicon Utility has all privileges as though it were the custodial
   package.
```

5757 ICDSAPI ICD Search Wrapper (2 file solution)

```
      5757     NAME: SEARCH ICD FILES
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE:
              USAGE: Supported           ENTERED: DEC 29,2011
```

```
          STATUS: Pending              EXPIRES: APR  1,2016
        DURATION:                      VERSION: 18
     DESCRIPTION:                         TYPE: Routine
Routine ICDSAPI was developed as a wrapper routine for DIC lookups during
the ICD-10 project to navigate between the ICD-9 Diagnosis file 80 and the
ICD-10 Diagnosis file 8010 under the two file solution.  The two file
solution had the ICD-9 codes and ICD-10 codes stored in two separate
files.  This solution was abandoned in favor of the one file solution
where both ICD-9 and ICD-10 are stored in the same file (ICD Diagnosis
file 80).  A one file solution of these APIs can be found in the routine
ICDEXLK (ICD Data Extraction, special lookup).  Routine ICDSAPI will be
exported to support applications through the transition between the one
and two file solutions.  It will be retired 18 months after the ICD-10
compliance date.


  ROUTINE: ICDSAPI
COMPONENT:  $$SEARCH(FILE,SCR,FMPAR,CDT)
            This API conducts a search of the ICD files (80 or 80.1) for a
            code, a diagnosis or a procedure using Fileman.  This API was
            developed at a time when ICD-10 and ICD-9 codes were in
            different file (aka, the two file solution).  It is being
            maintained at the request of the calling applications.  Now
            the ICD-10 and ICD-9 codes are in the same file (aka, the one
            file solution).  While this API still works, a much better
            option is available using the special lookup routine in file
            80 and 80.1.

VARIABLES:  Input     FILE
                        This can be either a file number, a file root, a
                        file identifier, a coding system or a source
                        abbreviation that can be resolved to a file
                        number.
                                                  Coding    Source Number
                        Root     ID     System    Abbreviation
                         80     ^ICD9(   DIAG   1 or 30   ICD or 10D
                         80.1   ^ICD0(   PROC   2 or 31   ICP or 10P

VARIABLES:  Input     SRC
                        This is a string of MUMPS code that is executed to
                        screen an entry from selection.  It must contain
                        an IF statement to set the value of $T.  Those
                        entries that the IF statement sets $T to 0 (false)
                        will not be displayed or selectable.

VARIABLES:  Input     FMPAR
                        This is the Fileman Lookup parameter consisting of
                        a string of alphabetic characters which that alter
                        how the lookup responds. Default value is "AEMQZ".
                        DIC(0) will be set to the contents of this
                        parameter.

                        Parameters applicable to a versioned file

                           A   Ask the entry; if erroneous, ask again
                           B   Only the B index is used
                           E   Echo information
```

```
                        F   Forget the lookup value
                        I   Ignore the special lookup program
                        M   Multiple-index lookup allowed
                        O   Only find one entry if it matches exactly
                        S   Suppresses display of .01
                        T   Search until user selects or enters ^^
                        X   EXact match required
                        Z   Zero node in Y(0), external form in Y(0,0)

                      Parameters not Applicable to a versioned file and
                      ignored by this lookup

                        C   Versioned cross-references not turned off
                        K   Primary Key not established
                        L   Learning a new entry LAYGO not allowed
                        N   Uppercase, IEN lookup allowed (not forced)
                        n   ICD has no pure numeric entries
                        Q   Input is pre-processed, ?? not necessary
                        U   All values are external
                        V   Verification is not optional

VARIABLES:  Input     CDT
                      This is the Code Set Versioning Date (Fileman
                      format)

                        If supplied only active codes on that date
                        will be included in the selection list.

                        If not supplied, the date will default to
                        TODAY and all codes may be selected, active
                        and inactive.

                        In both cases the display will be altered
                        based on the date.

VARIABLES:  Output    $$SEARCH
                      This is the value of Fileman's Y output variable.

                        Y          IEN ^ Code
                                    or
                                   -1 iF not found
```

5758 ICD CODE UPDATE EVENT Protocol

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: PROBLEM LIST
                        PROBLEM LIST (GMPL) attaches protocol GMPL SELECTION
                        LIST CSV EVENT that generates a mail message
                        containing inactivated ICD codes on the selection
                        list.

                     CONSULT/REQUEST TRACKING
                        CONSULTS (GMRC) attaches protocol ORCM GMRC CSV EVENT
                        that generates a mail message consult or procedure
                        quick orders that have an inactive ICD code.

                     CLINICAL REMINDERS
```

CLINICAL REMINDERS (PXRM) attaches protocol PXRM CODE SET UPDATE ICD that generates a mail message containing inactive code in the dialog file 801.41.

```
           USAGE: Controlled Subscri  ENTERED: JAN  3,2012
          STATUS: Pending             EXPIRES:
        DURATION: Till Otherwise Agr  VERSION:
            FILE:                        ROOT:
     DESCRIPTION:                        TYPE: Other
```

This protocol is used to notify other applications and processes when the ICD-9/10 Code Set is updated.

This is an extended action protocol. Applications may attach actions on this protocol that should be taken in the event of an ICD update.

NOTE: This protocol is commonly invoked by the LEXICAL SERVICES UPDATE protocol when there is a change in ICD data.

5773 DD(80 and DD(80.1 Special Lookup

```
  CUSTODIAL PACKAGE: DRG GROUPER
SUBSCRIBING PACKAGE: VA FILEMAN
```

Fileman calls this Special Lookup routine when the variable DIC(0) does not contain the letter "I" (Ignore Special Lookup).

```
           USAGE: Controlled Subscri  ENTERED: FEB 24,2012
          STATUS: Pending             EXPIRES:
        DURATION: Till Otherwise Agr  VERSION:
            FILE:                        ROOT:
     DESCRIPTION:                        TYPE: File
```

Applications may conduct Fileman lookups of ICD Diagnosis file #80 and the ICD OPERATIONS/PROCEDURE file #80.1 using ^DIC and the Special Lookup routine ICDEXLK. Applications may also point to these files.

A special lookup program was written for the ICD DIAGNOSIS file #80 and ICD OPERATION/PROCEDURE file #80.1 to navigate through the versioned (date sensitive) data stored in these files. The Name of the special lookup is stored in the Data Dictionary for these files:

```
^DD(80,0,"DIC")="ICDEXLK"
^DD(80.1,0,"DIC")="ICDEXLK"
```

Each time an application makes a ^DIC call to either file 80 or 80.1, the special lookup routine is invoked, provided the FileMan variable DIC(0) does not contain an "I" for "ignore the special lookup."

NOTE: Only the ^DIC call honors the special lookup routine. Those calls that allow the user to specify the indexes (IX^DIC and MIX^DIC1), and the Data Base Server calls (FIND^DIC, $$FIND1^DIC, and UPDATE^DIE) all ignore the Special Lookup Program. As a result, the FileMan calls that ignore the Special Lookup Program will not be able to conduct versioned searches or return versioned data so use IX^DIC, MIX^DIC1 FIND^DIC, and $$FIND1^DIC with a great deal of care. Never use any FileMan entry point that alters the data in these files (i.e., ^DIE, EN^DIB, ^DIK FILE^DIE, UPDATE^DIE and

```
FILE^DICN)

Package Special Lookup Variables

   The following local variables in the ICD namespace should be NEWed
   or KILLed by the calling application.  The global variables may
   be used in instances where local environment variables get NEWed
   and the special lookup values need to be retained.  The calling
   application is responsible for KILLing the ^TMP global variables.


     Versioning Date (Fileman format)

         ICDVDT or ^TMP("ICDEXLK",$J,"ICDVDT")=<versioning date>

                 If supplied only active codes on that date will be
                 included in the selection list.

                    1.  V74.6    SCREENING FOR YAWS
                    2.  V77.5    SCREENING FOR GOUT
                    3.  V76.9    SCREEN-NEOPLASM NOS
                    4.  V76.43   SCREEN MAL NEOP-SKIN
                    5.  V78.8    SCREEN-BLOOD DIS NEC

                 If not supplied, the date will default to TODAY and
                 all codes may be selected, active and inactive.

                   1.  V74.6   SCREENING FOR YAWS
                   2.  V77.5   SCREENING FOR GOUT
                   3.  V76.8   SCREEN-NEOPLASM NEC (Inactive)
                   4.  V76.9   SCREEN-NEOPLASM NOS
                   5.  V76.43  SCREEN MAL NEOP-SKIN

     Coding System (from file 80.4)

         ICDSYS or ^TMP("ICDEXLK",$J,"ICDSYS")=<coding system>

                   1     ICD     ICD-9-CM
                   2     ICP     ICD-9 Proc
                  30     10D     ICD-10-CM
                  31     10P     ICD-10-PCS

                 If supplied only codes belonging to the coding system
                 will be included in the selection list.

                    S ICDSYS=1,X="DIABETES MELLITUS KETOACIDOSIS"

                    2 matches found

                      1.  249.11  SEC DM KETOACD UNCNTRLD (Major CC)
                      2.  249.10  SEC DM KETO NT ST UNCNTR (Major CC)

                    S ICDSYS=30,X="DIABETES MELLITUS KETOACIDOSIS"

                    8 matches found

                       1.  E09.11  Drug/chem diabetes mellitus w
```

```
                    ketoacidosis w coma
               2.   E13.11  Oth diabetes mellitus with
                            ketoacidosis with coma
               3.   E09.10  Drug/chem diabetes mellitus w
                            ketoacidosis w/o coma
               4.   E10.11  Type 1 diabetes mellitus with
                            ketoacidosis with coma
               5.   E13.10  Oth diabetes mellitus with
                            Ketoacidosis without coma

          If not supplied codes from any coding system will be
          included in the selection list.

              S X="DIABETES MELLITUS KETOACIDOSIS"

              10 matches found

                1.  249.11   SEC DM KETOACD UNCNTRLD (Major CC)
                2.  249.10   SEC DM KETO NT ST UNCNT (Major CC)
                3.  E09.11   Drug/chem diabetes mellitus w
                             ketoacidosis w coma
                4.  E13.11   Oth diabetes mellitus with
                             Ketoacidosis with coma
                5.  E09.10   Drug/chem diabetes mellitus w
                             ketoacidosis w/o coma

  Display Format (numeric, 1-4)

      ICDFMT or ^TMP("ICDEXLK",$J,"ICDFMT")=<display format>

           Controls the format of the terms and code presented
           for selection on the selection list, 1-4,
           default = 1

           1     Fileman format, code and short text (default)

                 250.00    DMII WO CMP NT ST UNCNTR

           2   Fileman format, code and description

                 250.00   DIABETES MELLITUS WITHOUT MENTION OF
                          COMPLICATION, TYPE II OR UNSPECIFIED
                          TYPE, NOT STATED AS UNCONTROLLED

           3   Lexicon format, short text followed by code

               DMII WO CMP NT ST UNCNTR (250.00)

           4   Lexicon format, description followed by code

               DIABETES MELLITUS WITHOUT MENTION OF
               COMPLICATION, TYPE II OR UNSPECIFIED TYPE, NOT
               STATED AS UNCONTROLLED (250.00)

Fileman Variables used

  The following are FileMan local variables used by the Special
```

Lookup and should be NEWed or KILLed by the calling application

Input

X (Optional) User's input. If it exists, DIC(0) should not contain "A" for "Ask"

DIC (Required) The file number or an explicit global root in the form ^GLOBAL(or ^GLOBAL(X,Y,

DIC(0) (Optional) A string of alphabetic characters which alter how DIC responds. At a minimum this string must be set to null. (Required) Default value for ICD files "AEM"

The following characters are applicable to a versioned file

- A Ask the entry; if erroneous, ask again
- B Only the B index is used
- E Echo information
- F Forget the lookup value
- I Ignore the special lookup program
- M Multiple-index lookup allowed
- O Only find one entry if it matches exactly
- S Suppresses display of .01
- T Search until user selects or enters ^^
- X EXact match required
- Z Zero node in Y(0), external form in Y(0,0)

The following characters are NOT applicable to a versioned file (not used)

- C Versioned cross-references not turned off
- K Primary Key not established
- L Learning a new entry LAYGO not allowed
- N Uppercase, IEN lookup allowed (not forced)
- n ICD has no pure numeric entries
- Q Input is pre-processed, ?? not necessary
- U All values are external
- V Verification is not optional

DIC("A") (Optional) A prompt that is displayed prior to the reading of the X input. If DIC("A") is not defined, a prompt will be supplied by the special lookup routine.

DIC("B") (Optional) The default answer which is presented to the user when the lookup prompt is issued. If a terminal user simply presses the Enter/Return key, the DIC("B") default value will be used, and returned in X. DIC("B") will only be used if it is non-null.

DIC("S") (Optional) DIC("S") is a string of M code that DIC executes to screen an entry from selection. DIC("S") must contain an IF statement to set the value of $T.

Those entries that the IF sets as $T=0 will not be displayed or selectable. When the DIC("S") code is executed, the local variable Y is the internal number of the entry being screened and the M naked indicator is at the global level @(DIC_"Y,0)")

DIC("W") (Optional) An M command string which is executed when DIC displays each of the entries that match the user's input. The condition of the variable Y and of the naked indicator is the same as for DIC("S"). WARNING: If DIC("W") is defined, it overrides the display of the versioned identifiers for the file. Thus, if DIC("W") is set it will suppress the display of versioned data and there is a risk of displaying unversioned data.

DIC("?N",<file>)=n (Optional) The number "n" should be an integer set to the number of entries to be displayed on the screen at one time when using "?" help in a lookup.

FileMan Variables not used

DIC("DR")
DIC("PTRIX",<from>,<to>,<file>)
DIC("T")
DIC("V")
DIC("?PARAM",<file>,"INDEX")
DIC("?PARAM",<file>,"FROM",<subscript>)
DIC("?PARAM",<file>,"PART",<subscript>)

FileMan Variables KILLed

DLAYGO
DINUM

FileMan Variables Modified

If DIC(0) contains an "L" it will be removed

Output Variables

Always Returned

Y	IEN ^ Code	FileMan

If DIC(0) contains "Z"

Y(0)	0 Node	FileMan
Y(0,0)	Code	FileMan
Y(0,1)	$$ICDDX or $$ICDOP	Non-FileMan
Y(0,2)	Long Description	Non-FileMan

5780 ^ICDS(Supported

CUSTODIAL PACKAGE: DRG GROUPER

```
SUBSCRIBING PACKAGE:
              USAGE: Supported           ENTERED: MAR  5,2012
             STATUS: Pending             EXPIRES:
           DURATION: Till Otherwise Agr  VERSION:
               FILE: 80.4                   ROOT: ICDS(
        DESCRIPTION:                        TYPE: File
   This is a static file containing information about ICD coding systems.
   Applications may conduct FileMan lookups and point to this file.

   Use the API $$SINFO^ICDEX(IEN) to retrieve the information about an ICD
   Coding System (ICR 5747)
```

19. Glossary

TERM	MEANING
API	Application Programmer Interface
CMS	Centers for Medicare and Medicaid Services
CSV	Code Set Versioning
DBIA	Database Integration Agreement
ICD-9-CM	International Classification of Diseases, Ninth Revision, Clinical Modification
ICD-9 Proc	International Classification of Diseases, Ninth Revision, Procedural Classification System
ICD-10-CM	International Classification of Diseases, Tenth Revision, Clinical Modification
ICD-10-PCS	International Classification of Diseases, Tenth Revision, Procedural Classification System
KIDS	Kernel Installation Distribution System
SDO	Standard Development Organization
VISTA	Veterans Health Information Systems and Technology
WHO	World Health Organization