z/OS Batch

Medicare Code Editor Software

Object/Source Installation Manual

Software version 25.0
PBL–010 October 2008

3M Health Information Systems, Inc.
100 Barnes Road
Wallingford, CT 06492

This manual was written, designed, and produced by the Technical Writing Department of 3M Health Information Systems, Inc.

Document number PBL–010
October 2008

About this document

THIS MANUAL CONTAINS the information needed to install version 25.0 of the Medicare Code Editor (MCE) software that runs under the z/OS batch operating system. The manual assumes that the person installing the software has experience working with Basic Assembly Language (BAL) and z/OS Job Control Language (JCL).

The Centers for Medicare and Medicaid Services has approved this version of the Medicare Code Editor software and requires its use by Medicare fiscal intermediaries.

Any page in this manual that is updated after you receive the software and before the next release, is printed on yellow paper. The updated content is identified by a solid black change bar in the margin.

If you have any comments on this manual, please mail them to the Technical Writing Department at the address shown on page ii, or email them to hisdocumentation@mmm.com. We appreciate your interest in our documentation.

Contents

Figures

Tables

Chapter 1

Introduction

Contents

Chapter 1

Introduction

ON APRIL 20, 1983, CONGRESS ENACTED "Prospective Payment for Medicare Inpatient Hospital Services" as Title VI of the Social Security Amendment. Under Title VI, hospitals are paid a fixed price by Diagnosis Related Group (DRG) for treating Medicare patients.

In order to determine the appropriate DRG for a Medicare patient, the age, sex, discharge status, principal diagnosis, secondary diagnoses, and procedures performed must be reported by hospitals to their Medicare fiscal intermediaries. The logic of the DRG Definitions assumes that the patient information provided is accurate, and no attempt is made by the DRG Definitions to edit the data for accuracy. Only for extreme inconsistencies in the medical information will a DRG not be assigned to a patient record.

Types of edits

Three types of edits can be performed before assigning a DRG:

- Code edits examine a record for the correct use of the ICD-9-CM codes that describe a patient's diagnoses and procedures. Code edits include basic consistency checks on the interrelationships of a patient's age, sex, and diagnoses and procedures.
- Coverage edits examine patient type and performed procedures to determine if the services rendered are covered by Medicare and to what extent they are covered.

- Clinical edits examine the clinical consistency of the diagnostic and procedural information on the medical claim to determine if they are clinically reasonable and therefore if they should be paid.

In a first phase of edits, the Centers for Medicare and Medicaid Services (CMS) provides all fiscal intermediaries with a code editing package, referred to as the Medicare Code Editor (MCE). MCE software contains edits that deal primarily with coding and coverage related issues.

Purpose of the software

MCE detects and reports errors in the coding of claims data. While the program identifies and indicates the nature of the error, it does not correct the error. A particular error condition is associated with each type of coding error that is identified.

Versions and date ranges

Table 1–1 lists the versions contained in this release of MCE software. The patient's discharge date determines the version used for processing.

Table 1–1. Program versions with discharge date ranges

MCE version	DRG version	Discharge date range
MCE 25.0	DRG 26	10/01/2008 - 09/30/2009
MCE 24.1	DRG 25.1	04/01/2008 - 09/30/2008
MCE 24.0	DRG 25.0	10/01/2007 - 03/31/2008
MCE 23.0	DRG 24.0	10/01/2006 - 09/30/2007
MCE 22.0	DRG 23.0	10/01/2005 - 09/30/2006
MCE 21.0	DRG 22.0	10/01/2004 - 09/30/2005
MCE 20.0	DRG 21.0	10/01/2003 - 09/30/2004
MCE 19.0	DRG 20.0	10/01/2002 - 09/30/2003
MCE 18.0	DRG 19.0	10/01/2001 - 09/30/2002
MCE 17.0	DRG 18.0	10/01/2000 - 09/30/2001
MCE 16.0	DRG 17.0	10/01/1999 - 09/30/2000
MCE 15.1	DRG 16.0	07/01/1999 - 09/30/1999
MCE 15.0	DRG 16.0	10/01/1998 - 06/30/1999
MCE 14.0	DRG 15.0	10/01/1997 - 09/30/1998
MCE 13.0	DRG 14.0	10/01/1996 - 09/30/1997
MCE 12.0	DRG 13.0	10/01/1995 - 09/30/1996
MCE 11.0	DRG 12.0	10/01/1994 - 09/30/1995
MCE 10.0	DRG 11.0	10/01/1993 - 09/30/1994
MCE 9.0	DRG 10.0	10/01/1992 - 09/30/1993

Table 1–1. Program versions with discharge date ranges *(continued)*

MCE version	DRG version	Discharge date range
MCE 8.0	DRG 9.0	10/01/1991 - 09/30/1992
MCE 7.0	DRG 8.0	10/01/1990 - 09/30/1991
MCE 6.0	DRG 7.0	10/01/1989 - 09/30/1990
MCE 5.0	DRG 6.0	10/01/1988 - 09/30/1989
MCE 4.0	DRG 5.0	10/01/1987 - 09/30/1988
MCE 3.0	DRG 4.0	10/01/1986 - 09/30/1987
MCE 2.0	DRG 3.0	03/01/1984 - 09/30/1986

Chapter 2

Program output

Contents

Chapter 2

Program output

THIS CHAPTER DESCRIBES the output from the Medicare Code Editor (MCE) software program. When conflicting or incorrect information on a medical claim has been identified, the Medicare Code Editor prints a summary of the medical claim information, including the edit message that identifies the potential problem.

Figure 2–1, shown on the next page, illustrates the MCE summary format and content of the printed claim. The illustration is intended to be an example of a claims summary that is generated. No error messages appear in the example.

When error messages occur, they appear to the right of the code in question or at the bottom of the report. The *Definitions of Medicare Code Edits* guide contains more information on the edits that appear in MCE software.

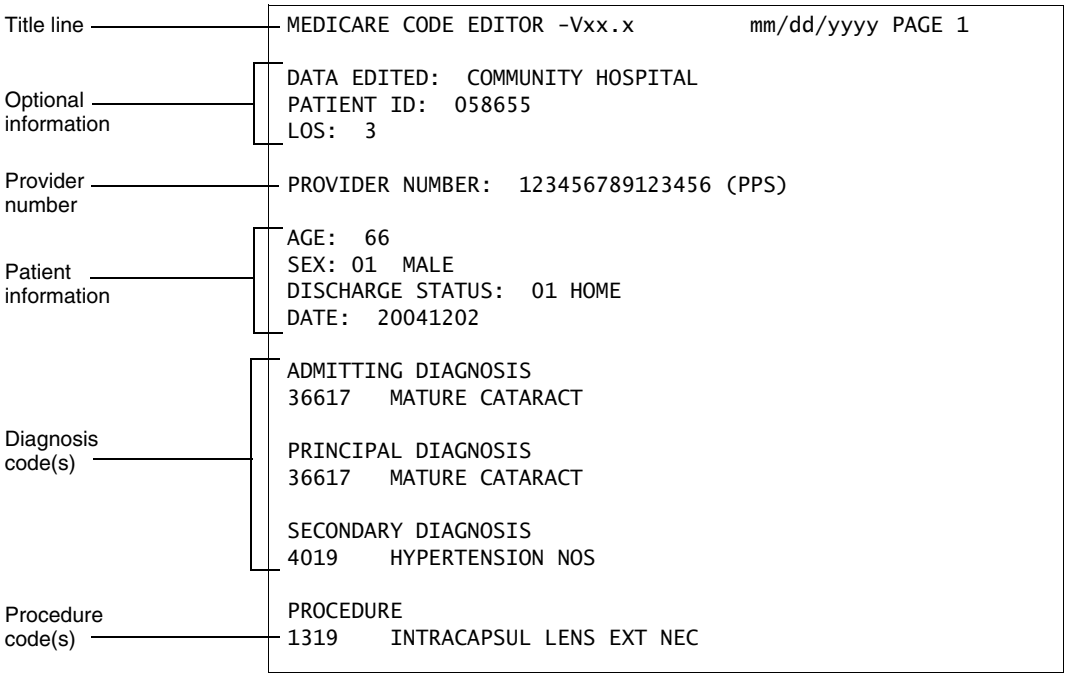


Figure 2–1. Sample output report

Elements in the output report

Data elements in the MCE output report are described below.

Title line

The title line includes the name of the software, the date the report was produced (mm/dd/yyyy format), the program version that processed the claim, and the report's page number. Each record is printed on a separate page.

Optional information

This section contains optional patient information from the claims record. The user may enter up to 11 lines, each 40 characters long, of optional information. For example, a hospital name and claim identifier can be reported as optional fields. The claim identifier can be a medical record number, social security number, patient's name, or any other identifier chosen by the user.

Figure 2–1 shows the hospital name, patient ID, and length of stay reported in the optional fields section. Additional information (e.g., physician number) can also be reported in this section at the user's option. Information on the description of the print program pointer OPTPTR is given in chapter 5.

Provider number

The 15 character Medicare provider number is reported. The type of hospital (i.e., PPS or non-PPS) is also reported in parentheses next to the provider number.

Patient information

This section contains the required patient information from the claims record (i.e., age, sex, discharge status, and discharge date).

Discharge status must be coded according to the UB-04 conventions. *See table 5–2 on page 5.5 for a list of valid discharge status codes.*

Discharge date is displayed in the same format as the date was entered (i.e., yyyymmdd). There are no separators in the 8-character field.

The program uses the discharge date to determine which version of the software will be used to process the claim. When the discharge date is absent or invalid, an error message is displayed and the claim stops processing.

For more information on software version date ranges, see page 1.4.

Diagnosis code(s)

The following diagnosis information is reported:

- ICD-9-CM admitting diagnosis code and English description
- ICD-9-CM principal diagnosis code and English description
- ICD-9-CM secondary diagnosis code(s) and English descriptions

Procedure code(s)

The ICD-9-CM codes and English descriptions of the procedure(s) performed are reported.

Chapter 3

Installing the software

Contents

Chapter 3

Installing the software

THIS CHAPTER DESCRIBES installation of the Medicare Code Editor (MCE) software that evaluates patient data to help identify possible errors in coding. Appendix A lists the edits contained in the program. The *Definitions of Medicare Code Edits* guide (PBL–011) contains more information on coding edits.

Following description of the installation media, you will be instructed how to download the files shown in table 3–1, and test that the installation was successful.

Installation media

The MCE installation media contains the compiled object code for the MCE and print programs, written in the IBM OS Assembler language. The media also contains MCE tables and English description files that are an integral part of the MCE system, and the source for all the executor programs. Table 3–1 lists the files contained on the media.

Table 3–1. Files on the installation media

File	Name	Description
1	MCE.JCL	JCL library of sample JCL to download files 2-5
2	MCE.OBJ	Library of MCE object programs
3	TESTDB	Installation test database
4	I9DSCRP	Combined ICD-9-CM English description file
5	MCE.ASM	Library of source programs and tables
6	DX.EBC	Diagnosis EBCDIC table
7	SG.EBC	Procedure EBCDIC table

Downloading the installation media files

This section gives specific information on the installation files and downloading them.

- Files 1 (MCE.JCL), 2 (MCE.OBJ), and 5 (MCE.ASM) must be copied to disk as partitioned data sets, using the IBM utility IEBUPDTE. The input data contains the "./ ADD" cards required by the IEBUPDTE utility.
- Files 3 (TESTDB), 6 (DX.EBC), and 7 (SG.EBC) must be copied to sequential files, using IEBGENER or any similar utility.
- File 4 (I9DSCRP) must be copied to a VSAM file.

Table 3–2 lists the physical characteristics of the installation files, their comparable disk representations, and their space allocations on 3390 disk packs.

Table 3–2. File characteristics

File	LRECL	BLKSIZE	DSORG tape	DSORG disk	Space
1	80	3120	PS	PO	2 trks
2	80	3120	PS	PO	3 cyls
3	1220	18300	PS	PS	12 trks
4	50	27950	PS	VSAM	1 cyl
5	80	3120	PS	PO	5 cyls
6	69	27945	PS	PS	2 cyls
7	69	27945	PS	PS	6 trks

In the JCL examples that follow, the VVVVVV notation for volume serial numbers should be replaced with the volume serial number of the cartridge you are installing. The number is marked on the cartridge's external label. The JCL examples shown can be run in any order.

File 1 - JCL library

File 1 contains sample JCL to download the various files on the remainder of the cartridge. There also is JCL to run sample COBOL interface programs. Table 3–3 lists the files contained in the JCL library.

Table 3–3. Files contained in JCL library (file 1)

Member	Function
ALTSTJCL	Run sample COBOL program (ALTTEST)
CBTSTJCL	Run sample COBOL program (COBTEST)
DBLOAD	Load test database
EBCLOAD	Load EBCDIC tables
OBJLOAD	Load object library
SRCLOAD	Load source library
VSAMLOAD	Load ICD-9-CM description file

Figure 3–1 shows sample JCL to copy the partitioned data set of sample JCL to disk.

```
//JOB CARD FOR YOUR INSTALLATION
//* ******************************************
//* JCL TO INSTALL THE JCL LIBRARY
//* ******************************************
//  EXEC   PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=*
//SYSIN    DD UNIT=CART,VOL=(,RETAIN,SER=VVVVVV),
//         LABEL=(1,NL),DISP=OLD,
//         DCB=(LRECL=80,BLKSIZE=3120,RECFM=FB)
//SYSUT2   DD UNIT=SYSDA,DISP=(NEW,CATLG,DELETE),
//         DSN=YOURID.MCE250.JCL,
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//         SPACE=(TRK,(2,1,3),RLSE)
/*
```

Figure 3–1. Sample JCL to install the JCL library

File 2 - library of object programs

File 2 on the installation media contains an object library of all MCE programs. Table 3–4 lists the members of the object library.

Table 3–4. Object library members

Number	Name	Description
1	MCE250CN	The main control program (standard interface)
2	MCE250DT	Date calculation program
3	MCE250ED	Editor program
4	MCE250LB	The editor support library
5	MCE250RT	The editor tables
6	MCE250PA	Print program
7	MCE250PB	Print program
8	MCE250PC	Print program
9	MCE250VS	VSAM code description program
10	MCE250CA	The main control program (alternate interface)

Members 1 through 9 comprise the main MCE executor using the standard interface. Substitute MCE250CA for MCE250CN to compile the main grouper executor using the alternate (re-entrant, macro-free) interface.

All the programs contained in the object library were written in IBM Basic Assembly Language (BAL). The programs were written and tested on an IBM 7060 computer.

Figure 3–2 shows the JCL necessary to copy the object library to disk as a partitioned data set.

```
//JOB CARD FOR YOUR INSTALLATION
//* ******************************************
//* JCL TO INSTALL THE OBJECT LIBRARY
//* ******************************************
//  EXEC   PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=*
//SYSIN    DD UNIT=CART,VOL=(,RETAIN,SER=VVVVVV),
//         LABEL=(2,NL),DISP=OLD,
//         DCB=(LRECL=80,BLKSIZE=3120,RECFM=FB)
//SYSUT2   DD UNIT=SYSDA,DISP=(NEW,CATLG,DELETE),
//         DSN=YOURID.MCE250.OBJLIB,
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//         SPACE=(CYL,(3,1,2),RLSE)
/*
```

Figure 3–2. Sample JCL to install the program object library (file 2)

File 3 - installation test database

File 3 contains a test database that is used to verify the integrity of the installed MCE program. The format of the test database is shown in table 3–5.

Table 3–5. Test database format

Field#	Description	Datatype	Location
1	Age	CL3	1-3
2	Sex	CL1	4
3	Discharge Status	CL2	5-6
4	Discharge date	CL8	7-14
5	Diagnoses (26)	26CL8	15-222
6	Procedures (25)	25CL7	223-397
7	Provider	CL15	398-412
8	PPS	CL1	413-413
9	Version	CL3	414-416
10	ADXFLAG	CL1	417-417
11	DXFLAGS	26XL13	418-742
12	SGFLAGS	25XL16	743-1142
13	MCEBUFF	CL78	1143-1220

Figure 3–3 shows the JCL for copying the test database to disk.

```
//JOB CARD FOR YOUR INSTALLATION
//* ******************************************
//* JCL TO INSTALL THE TEST DATABASE
//* ******************************************
//  EXEC   PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DUMMY
//SYSUT1   DD UNIT=CART,VOL=(,RETAIN,SER=VVVVVV),
//         LABEL=(3,NL),DISP=OLD,
//         DCB=(LRECL=1220,BLKSIZE=18300,RECFM=FB)
//SYSUT2   DD UNIT=SYSDA,DISP=(NEW,CATLG,DELETE),
//         DSN=YOURID.MCE250.TESTDB,
//         DCB=(LRECL=1220,BLKSIZE=18300,RECFM=FB),
//         SPACE=(TRK,(12),RLSE)
/*
```

Figure 3–3. Sample JCL to install the test database (file 3)

File 4 - English description VSAM file

File 4, the I9DSCRP file, is written as a key-sequenced data set, and the input file is sorted. This file replaces any English description files that may have been installed for other versions of MCE software. It combines all codes into one file, and has an additional identifier as part of the key.

Downloading the description file is optional. The report programs that use the I9DSCRP file give you the option to bypass descriptions (*see DSCPTR narrative in chapter 5*).

Figure 3–4 is an example of how to load the English description file. The layout of the description file follows the figure.

```
//JOB CARD FOR YOUR INSTALLATION
//* ****************************************
//* JCL TO INSTALL THE ENGLISH DESCRIPTION FILE
//* ****************************************
// EXEC PGM=IDCAMS,REGION=1024K
//SYSPRINT DD SYSOUT=*
//INPUT DD UNIT=CART,VOL=SER=VVVVVV,
//         DCB=(RECFM=FB,LRECL=50,BLKSIZE=27950),
//         LABEL=(4,NL),DISP=OLD
//SYSIN DD *
     DEFINE CLUSTER (NAME(YOURID.MCE250.VSFILE) -
                      VOLUMES(VVVVVV) -
                      CISZ(2048) -
                      RECORDS(19228)) -
             DATA    (KEYS(10 0) -
                      RECORDSIZE(50 50) -
                      NAME(YOURID.MCE250.VSFILE.DATA)) -
             INDEX    (NAME(YOURID.MCE250.VSFILE.INDEX))
     REPRO INFILE(INPUT) -
           OUTDATASET(YOURID.MCE250.VSFILE)
/*
```

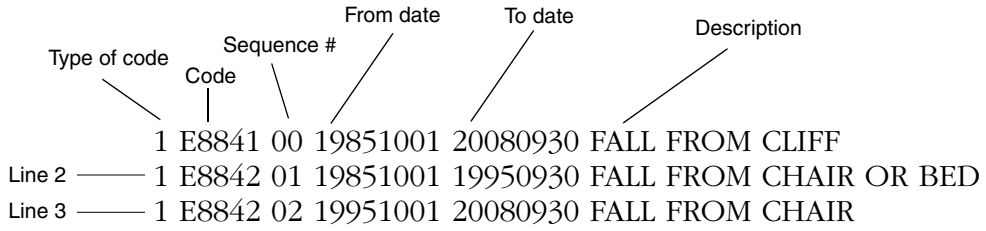
Figure 3–4. Sample JCL to install the English description file (file 4)

Layout of the description file

The layout of the description file follows:

- The first byte indicates whether the code is a diagnosis (1) or procedure (2).
- The next seven bytes (bytes 2-8) contain the code.
- The next two bytes contain the sequence number. When sequence number equals 00, the code description is valid for all MCE versions (first to current).
- The next eight bytes contain the “from” date.
- The next eight bytes contain the “to” date.
- The remaining bytes contain the code description.

An example of the description file layout is shown below.



➤ ***Note:*** *As illustrated in the example above, the "from" and "to" version numbers are replaced with "from" and "to" dates.*

In the first line, the description for diagnosis code E8841 is valid for all MCE versions; therefore, the sequence number zero.

Diagnosis code E8842 has two entries on lines 2 and 3 because the description changed on 19951001. Since there are two entries, each entry is given a sequence number. The description for sequence number 01 is valid for MCE from 19851001 through 19950930. The description for sequence number 02 is valid from 19951001 to 20080930.

File 5 - library of source programs and tables

The source library in file 5 consists of the source for the COBOL test programs and the sources for the MCE editor and print routine programs.

Table 3–6 lists the members of the source library.

Table 3–6. Source library members (file 5)

Number	Name	Description
1	ALTTEST	Sample COBOL program (alternate interface)
2	COBTEST	Sample COBOL (standard interface) program
3	MCE250CA	Control program (alternate interface)
4	MCE250CN	Control program (standard interface)
5	MCE250DT	Date calculation program
6	MCE250ED	Editor program
7	MCE250LB	Editor support library
8	MCE250PA	Print program
9	MCE250PB	Print program
10	MCE250PC	Print program
11	MCE250PR	Print macro
12	MCE250RT	Editor tables
13	MCE250VS	VSAM description file program

Figure 3–5 shows the JCL necessary to copy the source library to disk as a partitioned data set. The sources for all of the MCE executor programs are included.

```
//JOB CARD FOR YOUR INSTALLATION
//* ****************************************
//* JCL TO INSTALL THE SOURCE LIBRARY
//* ****************************************
// EXEC    PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=*
//SYSIN    DD UNIT=CART,VOL=(,RETAIN,SER=VVVVVV),
//         LABEL=(5,NL),DISP=OLD,
//         DCB=(LRECL=80,BLKSIZE=3120,RECFM=FB)
//SYSUT2   DD UNIT=SYSDA,DISP=(NEW,CATLG,DELETE),
//         DSN=YOURID.MCE250.SRCLIB,
//         DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//         SPACE=(CYL,(5,1,4),RLSE)
/*
```

Figure 3–5. Sample JCL to install the program source library (file 5)

Files 6 and 7 - MCE v25.0 EBCDIC tables

The tables that drive the MCE are expressed in Extended Binary Coded Decimal Interchange Code (EBCDIC) as two files:

Diagnosis table. Contains one row per ICD-9-CM diagnosis, with diagnosis attributes.

Procedure table. Contains one row per ICD-9-CM procedure, with procedure attributes.

In the layouts in this section, each field is identified by its position (first column is position 1) and length in a table row. Criteria fields (length 1) are one when the criteria are met and zero otherwise.

Files 6 and 7 contain the diagnosis and procedure tables. Sample JCL for downloading the EBCDIC tables is shown in figure 3–6 on page 3.17.

EBCDIC Diagnosis table

File 6, shown in table 3–7, contains the EBCDIC Diagnosis table.

Table 3–7. Diagnosis table

Name	Pos	Len	Description
dx	1	5	ICD-9-CM diagnosis
effdate	6	8	edit effective date
termdate	14	8	edit termination date
pediatric	22	1	diagnosis for pediatric only
msp	23	1	medicare as secondary payer
maternity	24	1	diagnosis for maternity only
nonspecific	25	1	nonspecific diagnosis
newborn	26	1	diagnosis for newborn only
manifestation	27	1	manifestation
female	28	1	diagnosis for female only
male	29	1	diagnosis for male only
mdc08	30	1	MDC 8
reqsdx	31	1	requires secondary diagnosis
ncov2	32	1	ncov2
qadm	33	1	questionable admission
unacceptable	34	1	unacceptable diagnosis
adult	35	1	diagnosis for adult only
cc	36	1	cc
ncov3	37	1	ncov3

Table 3–7. Diagnosis table *(continued)*

Name	Pos	Len	Description
ncov4	38	1	ncov4
ncov5	39	1	ncov5
ncov2agelt78	40	1	ncov2agelt78
ncov2agelt64	41	1	ncov2agelt64
ncov6	42	1	ncov6
ncov7	43	1	ncov7
ncov89	44	1	ncov89
diabtypeI	45	1	diabetes
UNUSED	46	1	UNUSED
UNUSED	47	1	UNUSED
UNUSED	48	1	UNUSED
UNUSED	49	1	UNUSED
UNUSED	50	1	UNUSED
UNUSED	51	1	UNUSED
UNUSED	52	1	UNUSED
UNUSED	53	1	UNUSED
UNUSED	54	1	UNUSED
UNUSED	55	1	UNUSED
UNUSED	56	1	UNUSED
UNUSED	57	1	UNUSED
UNUSED	58	1	UNUSED
UNUSED	59	1	UNUSED
UNUSED	60	1	UNUSED
UNUSED	61	1	UNUSED
UNUSED	62	1	UNUSED
UNUSED	63	1	UNUSED
UNUSED	64	1	UNUSED
UNUSED	65	1	UNUSED
UNUSED	66	1	UNUSED
UNUSED	67	1	UNUSED
UNUSED	68	1	UNUSED
UNUSED	69	1	UNUSED

EBCDIC Procedure table

File 7, shown in table 3–8, contains the EBCDIC Procedure table.

Table 3–8. Procedure table

Name	Pos	Len	Description
sg	1	5	ICD-9-CM procedure
effdate	6	8	edit effective date
termdate	14	8	edit termination date
noncovered	22	1	noncovered procedure
biopsy	23	1	biopsy
UNUSED	24	1	UNUSED
bilateral	25	1	bilateral procedure
nonspecific	26	1	nonspecific procedure
or	27	1	or indicator
female	28	1	procedure for female only
male	29	1	procedure for male only
kidneyxp	30	1	kidney transplant
ncov8	31	1	ncov8
ncov9	32	1	ncov9
ncov6	33	1	ncov6
ncov7	34	1	ncov7
ncov45	35	1	ncov45
ncov2	36	1	ncov2
lcov_lvrs	37	1	limited coverage - LVRS
lcov_lungxp	38	1	limited coverage - lung transplant
lcov_heartlungxl	39	1	limited coverage - heart/lung transplant
lcov_heartxp	40	1	limited coverage - heart transplant
lcov_heartsys	41	1	limited coverage - heart system transplant
lcov_intxp	42	1	limited coverage - intestine transplant
lcov_liver	43	1	limited coverage - liver transplant
UNUSED	44	1	UNUSED
ncov10a	45	1	ncov10a
ncov10b	46	1	ncov10b
ncov10c	47	1	ncov10c
ncov11	48	1	ncov11

Table 3–8. Procedure table *(continued)*

Name	Pos	Len	Description
ncov12agele60	49	1	ncov12agele60
lcov_kidneyxp	50	1	limited coverage - kidney transplant
lcov_pancreasxp	51	1	limited coverage - pancreas transplant
ncov13a	52	1	ncov13a
ncov13b	53	1	ncov13b
lcov_artheartxp	54	1	limited coverage - artificial heart transplant
UNUSED	55	1	UNUSED
UNUSED	56	1	UNUSED
UNUSED	57	1	UNUSED
UNUSED	58	1	UNUSED
UNUSED	59	1	UNUSED
UNUSED	60	1	UNUSED
UNUSED	61	1	UNUSED
UNUSED	62	1	UNUSED
UNUSED	63	1	UNUSED
UNUSED	64	1	UNUSED
UNUSED	65	1	UNUSED
UNUSED	66	1	UNUSED
UNUSED	67	1	UNUSED
UNUSED	68	1	UNUSED
UNUSED	69	1	UNUSED

Figure 3–6 shows sample JCL for downloading the EBCDIC tables.

```
//JOB CARD FOR YOUR INSTALLATION
//* ******************************************
//* JCL TO INSTALL THE DIAGNOSIS EBCDIC TABLE
//* ******************************************
//  EXEC   PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DUMMY
//SYSUT1   DD UNIT=CART,VOL=(,RETAIN,SER=VVVVVV),
//         LABEL=(6,NL),DISP=(OLD,PASS),
//         DCB=(LRECL=69,BLKSIZE=27945,RECFM=FB)
//SYSUT2   DD UNIT=SYSDA,DISP=(NEW,CATLG,DELETE),
//         DSN=YOURID.MCE250.DXEBC,
//         DCB=(LRECL=69,BLKSIZE=27945,RECFM=FB),
//         SPACE=(CYL,(3),RLSE)
/*
//* ******************************************
//* JCL TO INSTALL THE PROCEDURE EBCDIC TABLE
//* ******************************************
//  EXEC   PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DUMMY
//SYSUT1   DD UNIT=CART,VOL=(,RETAIN,SER=VVVVVV),
//         LABEL=(7,NL),DISP=OLD,
//         DCB=(LRECL=69,BLKSIZE=27945,RECFM=FB)
//SYSUT2   DD UNIT=C21USR,DISP=(NEW,CATLG,DELETE),
//         DSN=YOURID.MCE250.SGEBC,
//         DCB=(LRECL=69,BLKSIZE=27945,RECFM=FB),
//         SPACE=(TRK,(6),RLSE)
/*
```

Figure 3–6. Sample JCL for downloading the EBCDIC tables

Running the test program

➤ ***Note:*** *We strongly recommend running the test program to ensure that the software is correctly installed.*

A copy of the COBOL test program and the test database are included on the cartridge to allow you to test the results of the installation procedure. Figure 3–7 is an example of a compile-link-go to execute the COBOL test program.

If you have not installed the ICD-9-CM description file, change line 55 in the COBOL test program to read:

```
77 DSCFLAG PIC S9(8) COMP VALUE IS +0.
```

Also, exclude the marked (†) line from the JCL in figure 3–7.

```
//JOB CARD FOR YOUR INSTALLATION
//* ******************************************************
//* THIS JOB IS USED TO COMPILE, LINK AND RUN THE MCE
//* COBOL TEST PROGRAM, COBTEST.
//*
//* BOTH OBJECT AND LOAD MODULES ARE TEMPORARY.
//* ******************************************************
//COBUCLG PROC SYSOUT='*'
//*   COBOL FOR MVS COMPILE AND LINK
//COB    EXEC  PGM=IGYCRCTL,PARM='RENT,NODYNAM'
//STEPLIB  DD  DSN=IGY.V2R2M0.SIGYCOMP,DISP=SHR
//SYSLIB   DD  DSN=YOURID.&PROD..SRCLIB,DISP=SHR
//SYSPRINT DD  SYSOUT=&SYSOUT
//SYSIN    DD  DSN=YOURID.&PROD..SRCLIB(COBTEST),DISP=SHR
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT2   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT3   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT4   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT5   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT6   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT7   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSLIN   DD  DSN=&&LOADSET,UNIT=SYSDA,DISP=(MOD,PASS),
//             SPACE=(TRK,(3,3)),DCB=BLKSIZE=800
//*
//LKED   EXEC  PGM=IEWL,PARM='LIST,MAP,AMODE=31,RMODE=ANY',
//             COND=(5,LT,COB)
//SYSLIB   DD  DSN=CEE.SCEELKED,DISP=SHR
//SYSLMOD  DD  DSN=&&GOSET(COBTEST),UNIT=SYSDA,DISP=(,PASS),
//             SPACE=(CYL,(5,1,5))
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD  SYSOUT=&SYSOUT
//SYSLIN   DD  DSN=&&LOADSET,DISP=(OLD,DELETE)
//         DD  DDNAME=SYSIN
//OBJECT   DD DSN=YOURID.&PROD..OBJLIB,DISP=OLD
//*
//GO     EXEC  PGM=COBTEST,COND=((5,LT,COB),(5,LT,LKED))
//STEPLIB  DD DISP=SHR,DSN=&&GOSET
//         DD DISP=SHR,DSN=CEE.SCEERUN
//INFILE   DD DSN=YOURID.&PROD..TESTDB,DISP=SHR
//SYSPRINT DD SYSOUT=&SYSOUT
//MCE250I9 DD DSN=YOURID.&PROD..VSFILE,DISP=SHR†
//RPTFILE  DD SYSOUT=&SYSOUT,DCB=(RECFM=FA,BLKSIZE=81,BUFNO=1)
//         PEND
//*
//PROG1    EXEC COBUCLG,PROD=MCE250
//*
//LKED.SYSIN DD *
 INCLUDE OBJECT(MCE250CN,MCE250ED,MCE250LB,MCE250RT)
 INCLUDE OBJECT(MCE250PA,MCE250VS,MCE250DT)
 ENTRY COBTEST
 NAME COBTEST
/*
```

Figure 3–7. Sample JCL to execute the COBOL test program

If the test is successful, all return results should match the expected results on the test database input, and the report

output should match the printout shown in figure 3–8. There should be 505 records processed. The test should take less than 5 CPU seconds.

Note that the DISCHARGE DATE output field displays in the same format as the date is entered (yyyymmdd).

```
MEDICARE CODE EDITOR - V##.#                 mm/dd/yyyy PAGE     1

LAST RECORD

PROVIDER: 000000000000505  (PPS)
AGE:   33
SEX: 2 FEMALE
DISCHARGE STATUS: 01 HOME
DISCHARGE DATE: 20061001

ADMITTING DIAGNOSIS
 V4586    BARIATRIC SURGERY STATUS     0

PRINCIPAL DIAGNOSIS
 V4586    BARIATRIC SURGERY STATUS     0000000100000

SECONDARY DIAGNOSIS
 79399    ABN FIND-BODY STRUCT NEC     0000000000000

PROCEDURE
 4468     LAPAROSCOP GASTROPLASTY      0000000000000000
```

Figure 3–8. COBOL test program output

Chapter 4

Running the program

Contents

Chapter 4

Running the program

TO EXECUTE THE Medicare Code Editor (MCE) program, you must write an interface program that will perform the following functions:

- Read the input file records.
- Construct the MCE control block (*see chapter 5*).
- Move diagnoses and procedures into contiguous locations if they were not recorded that way on input.
- Recode the discharge status if the coding scheme is not UB-04 standard.
- Call the MCE program, and optionally, one of the report programs.
- Write output records, if applicable.

Note that the MCE system assumes that provider number, PPS indicator, age, sex, discharge status, date, diagnoses and procedures are all EBCDIC (character) data.

Calling the editor

Once the interface program is done and specifies the pointers in the control block where the input data is located, the MCE program is invoked by calling the controller program MCE250CN that determines the MCE version to be called based on the date of discharge.

MCE250CN then calls the appropriate MCE version and returns control to your interface program. If a date is not valid, or is not within the range of the MCE versions (15.0 through 25.0), an error message is displayed and the claim stops processing.

The process is then repeated for each record to be edited. At the call to the control program, general purpose Register 1 must be set to point to the control block. The control block is discussed in chapter 5.

JCL for executing the program

By implementing the CALL...USING statement, COBOL programmers will have Register 1 set by the CALL statement.

Figure 4–1 is an example of compile-link-go JCL to edit only.

For an example of JCL to edit and call the report program, refer to figure 4–2.

If you have not installed the ICD-9-CM description file, exclude the marked (†) line from the JCL in figure 4–2.

```
//JOB CARD FOR YOUR INSTALLATION
//* ****************************************************
//* THIS JOB IS USED TO COMPILE, LINK AND RUN THE MCE
//* COBOL TEST PROGRAM, COBTEST.
//*
//* BOTH OBJECT AND LOAD MODULES ARE TEMPORARY.
//* ****************************************************
//COBUCLG PROC SYSOUT='*'
//*   COBOL FOR MVS COMPILE AND LINK
//COB    EXEC  PGM=IGYCRCTL,PARM='RENT,NODYNAM'
//STEPLIB  DD  DSN=IGY.V2R2M0.SIGYCOMP,DISP=SHR
//SYSLIB   DD  DSN=YOURID.&PROD..SRCLIB,DISP=SHR
//SYSPRINT DD  SYSOUT=&SYSOUT
//SYSIN    DD  DSN=YOURID.&PROD..SRCLIB(COBTEST),DISP=SHR
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT2   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT3   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT4   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT5   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT6   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT7   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSLIN   DD  DSN=&&LOADSET,UNIT=SYSDA,DISP=(MOD,PASS),
//             SPACE=(TRK,(3,3)),DCB=BLKSIZE=800
//*
//LKED   EXEC  PGM=IEWL,PARM='LIST,MAP,AMODE=31,RMODE=ANY',
//             COND=(5,LT,COB)
//SYSLIB   DD  DSN=CEE.SCEELKED,DISP=SHR
//SYSLMOD  DD  DSN=&&GOSET(COBTEST),UNIT=SYSDA,DISP=(,PASS),
//             SPACE=(CYL,(5,1,5))
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD  SYSOUT=&SYSOUT
//SYSLIN   DD  DSN=&&LOADSET,DISP=(OLD,DELETE)
//         DD  DDNAME=SYSIN
//OBJECT   DD DSN=YOURID.&PROD..OBJLIB,DISP=OLD
//*
//GO     EXEC  PGM=COBTEST,COND=((5,LT,COB),(5,LT,LKED))
//STEPLIB  DD DISP=SHR,DSN=&&GOSET
//         DD DISP=SHR,DSN=CEE.SCEERUN
//INFILE   DD DSN=YOURID.&PROD..TESTDB,DISP=SHR
//SYSPRINT DD SYSOUT=&SYSOUT
//RPTFILE  DD SYSOUT=&SYSOUT,DCB=(RECFM=FA,BLKSIZE=81,BUFNO=1)
//         PEND
//*
//PROG1    EXEC COBUCLG,PROD=MCE250
//*
//LKED.SYSIN DD *
 INCLUDE OBJECT(MCE250CN,MCE250ED,MCE250LB,MCE250RT)
 ENTRY COBTEST
 NAME COBTEST
/*
```

Figure 4–1. Sample JCL for edit-only procedure

```
//JOB CARD FOR YOUR INSTALLATION
//* ***************************************************
//* THIS JOB IS USED TO COMPILE, LINK AND RUN THE MCE
//* COBOL TEST PROGRAM, COBTEST.
//*
//* BOTH OBJECT AND LOAD MODULES ARE TEMPORARY.
//* ***************************************************
//COBUCLG PROC SYSOUT='*'
//*   COBOL FOR MVS COMPILE AND LINK
//COB    EXEC  PGM=IGYCRCTL,PARM='RENT,NODYNAM'
//STEPLIB  DD  DSN=IGY.V2R2M0.SIGYCOMP,DISP=SHR
//SYSLIB   DD  DSN=YOURID.&PROD..SRCLIB,DISP=SHR
//SYSPRINT DD  SYSOUT=&SYSOUT
//SYSIN    DD  DSN=YOURID.&PROD..SRCLIB(COBTEST),DISP=SHR
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT2   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT3   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT4   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT5   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT6   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT7   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSLIN   DD  DSN=&&LOADSET,UNIT=SYSDA,DISP=(MOD,PASS),
//             SPACE=(TRK,(3,3)),DCB=BLKSIZE=800
//*
//LKED   EXEC  PGM=IEWL,PARM='LIST,MAP,AMODE=31,RMODE=ANY',
//             COND=(5,LT,COB)
//SYSLIB   DD  DSN=CEE.SCEELKED,DISP=SHR
//SYSLMOD  DD  DSN=&&GOSET(COBTEST),UNIT=SYSDA,DISP=(,PASS),
//             SPACE=(CYL,(5,1,5))
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD  SYSOUT=&SYSOUT
//SYSLIN   DD  DSN=&&LOADSET,DISP=(OLD,DELETE)
//         DD  DDNAME=SYSIN
//OBJECT   DD DSN=YOURID.&PROD..OBJLIB,DISP=OLD
//*
//GO     EXEC  PGM=COBTEST,COND=((5,LT,COB),(5,LT,LKED))
//STEPLIB  DD DISP=SHR,DSN=&&GOSET
//         DD DISP=SHR,DSN=CEE.SCEERUN
//INFILE   DD DSN=YOURID.&PROD..TESTDB,DISP=SHR
//SYSPRINT DD SYSOUT=&SYSOUT
//MCE250I9 DD DSN=YOURID.&PROD..VSFILE,DISP=SHR †
//RPTFILE  DD SYSOUT=&SYSOUT,DCB=(RECFM=FA,BLKSIZE=81,BUFNO=1)
//         PEND
//*
//PROG1    EXEC COBUCLG,PROD=MCE250
//*
//LKED.SYSIN DD *
 INCLUDE OBJECT(MCE250CN,MCE250ED,MCE250LB,MCE250RT)
 INCLUDE OBJECT(MCE250PA,MCE250VS,MCE250DT)
 ENTRY COBTEST
 NAME COBTEST
/*
```

Figure 4–2. Sample JCL for edit-print procedure

Using the alternate interface

The alternate editor control program, (MCE250CA) operates the same as the standard editor control program (MCE250CN) except that it does not contain any macros and is written to be re-entrant, so it should run in a wider variety of mainframe environments. Whereas the standard interface uses GETMAINS to obtain a 20,000 byte work area, the alternate interface requires that the calling program provide the work area. It must do so by providing two additional addresses in the list pointed to by general register 1 (see table 5–1.)

Table 4–1 gives the additional work area parameters required by the alternate interface.

Table 4–1. Work area parameters

Element number	Editor pointers	Full word pointer to...
16	WORKAREA	A buffer of at least 20,000 bytes.
17	WORKSIZE	4-byte binary (PIC 9(8) comp) field containing the actual length in bytes of the work area. The value of this field should not be less than 20,000 bytes, though larger values are acceptable.

To use the alternate interface, substitute MCE250CA for MCE250CN and provide these two extra parameters. See the COBOL program ALTTEST, provided in the source library, for an example of how to set up a work area and pass it to MCE250CA.

Assembler programmers should note that the length of the work area is *not* given in the full word at element number 17 but rather a *pointer* to the full word containing the length is given at element number 17.

Sample JCL for running ALTTEST may be created by modifying the JCL shown in figure 4–1 or 4–2. To modify the JCL, change all occurrences of COBTEST to ALTTEST and change MCE250CN to MCE250CA.

Chapter 5

The control block

Contents

Chapter 5

The control block

THE CONTROL BLOCK IS a block of fullwords which serves as the main reference point for each of the programs in the Medicare Code Editor (MCE) system. Each program uses the control block to locate required input data and to establish the locations of return information.

Table 5–1 lists the control block and elements required for each system component. The first 15 elements are the same for both the editor and the print programs. The pointers from element number 16 on have different meanings depending on which component is being called.

Table 5–1. Control block and elements of MCE system

Element Number	Editor pointers	Print program pointers
1	DXPTR	DXPTR
2	NDXPTR	NDXPTR
3	PRPTR	PRPTR
4	NPRPTR	NPRPTR
5	AGEPTR	AGEPTR
6	SEXPTR	SEXPTR
7	DSTATPTR	DSTATPTR
8	PROVPTR	PROVPTR
9	PPSPTR	PPSPTR

Table 5–1. Control block and elements of MCE system *(continued)*

Element Number	Editor pointers	Print program pointers
10	DATEPTR	DATEPTR
11	VPTR	VPTR
12	ADXFLGPTR	ADXFLGPTR
13	DXFLGPTR	DXFLGPTR
14	PRFLGPTR	PRFLGPTR
15	BUFFPTR	BUFFPTR
16		DSCPTR
17		OPTPTR1
18		OPTPTR2
19		OPTPTR3
20		OPTPTR4
21		OPTPTR5
22		OPTPTR6
23		OPTPTR7
24		OPTPTR8
25		OPTPTR9
26		OPTPTR10
27		OPTPTR11

The following pages explain the pointers listed in table 5–1. Bit values, where documented, are numbered in a left-to-right order, with bit 0 being the left-most bit.

DXPTR Address of the area containing contiguous ICD-9-CM diagnosis codes. Each diagnosis must be left justified and blank filled in an 8-byte field. The eighth byte represents the POA indicator. The first of these codes is presumed to be the admitting diagnosis and the second is presumed to be the principal diagnosis. These codes must be present.

NDXPTR Address of a fullword containing the number of diagnosis codes. This is the maximum number that the area pointed to by DXPTR can hold. The system calculates the number of actual diagnoses. The number must be a binary (PIC 9(8) COMP) fullword. This must be a value of at least 2, as at least two diagnoses must be present. The maximum number of codes allowed is 26. If greater than 26, the software uses only the first 26 fields in the buffer and ignores the rest.

SGPTR Address of the area containing contiguous ICD-9-CM procedure codes. Each code must be seven bytes. Procedures are handled in the same manner as diagnoses by the system.

NSGPTR Address of a fullword containing the number of procedure codes. This is the maximum number that the area pointed to by SGPTR can hold. The number must be a binary (PIC 9(8) COMP) fullword. The maximum number of codes allowed is 25.

AGEPTR Address of a 3-byte variable containing the numeric age in years. The variable must be right-adjusted, with either zero or blank filling allowed. Values in the range 0-124 are valid.

SEXPTR Address of a 1-byte variable containing the numeric sex. The variable must contain the value 1 for males, 2 for females or 0 for unknown.

DSTATPTR Address of a 2-byte variable containing the numeric discharge status code, which must be coded according to the UB-04 code scheme. If discharge status is not available, DSTATPTR should point to a constant with a value of 00. Table 5–2 lists the valid UB-04 discharge status codes in the software.

Table 5–2. UB-04 discharge status codes

Code	Description
00	Unknown
01	Home, self care (routine)
02	Short term hospital
03	SNF
04	ICF
05	Other facility (valid until 03/31/08) Canc/child hosp (effective 04/01/08)
06	Home health service
07	Left against medical advice
08	Home IV service (deleted 10/01/05)
20	Died
30	Still a patient
43	Fed hospital (added 10/01/03)
50	Hospice - home

Table 5–2. UB-04 discharge status codes *(continued)*

Code	Description
51	Hospice - medical facility
61	Swing bed (added 10/01/01)
62	Rehab fac/unit (added 10/01/01)
63	LTC hospital (added 10/01/01)
64	Nursing facility-Medicaid certified (added 10/01/02)
65	Psych hosp/unit (added 10/01/03)
66	Critical access hospital (added 10/01/05)
70	Oth institution (effective 04/01/08)
71	OP services-other facility (10/01/01–09/30/03 only)
72	OP services-this facility (10/01/01–09/30/03 only)

PROVPTR Address of an area containing the 15-byte Medicare provider number. This information is required for the summary record. Refer to BUFFPTR above for a detailed explanation.

PPSPTR Address of a 1-byte numeric variable which must be set to one of the values shown in table 5–3. This information is required for the summary record. Refer to BUFFPTR above for details.

Table 5–3. PPS values

Value	Description
0	PPS status unknown
1	PPS provider
2	Non-PPS provider

DATEPTR Address of the calendar discharge date (yyyymmdd) which is used for determining which MCE version to call. Each of the three components of the date must be numeric and left zero filled. There are no separators. If this date is not valid, the claim stops processing, and the edit flag (see BUFFPTR) will be set to 4. Since the date edit is not a part of the "official" MCE edits, there is no accumulator provided. The flag is included for your convenience only.

VPTR Address of a 3-byte area (Pic 9(3)) where the version identification number is placed by the program. This area contains the number of the MCE version that was run. Selection of an MCE

version is determined by the date passed in DATEPTR. Table 5–4 lists the versions and date ranges.

Table 5–4. Versions

MCE version	Date range
16.0	10/01/1999 – 09/30/2000
17.0	10/01/2000 – 09/30/2001
18.0	10/01/2001 – 09/30/2002
19.0	10/01/2002 – 09/30/2003
20.0	10/01/2003 – 09/30/2004
21.0	10/01/2004 – 09/30/2005
22.0	10/01/2005 – 09/30/2006
23.0	10/01/2006 – 09/30/2007
24.0	10/01/2007 – 03/31/2008
24.1	04/01/2008 – 09/30/2008
25.0	10/01/2008 – 09/30/2009

ADXFLGPTR Address of a 1-byte variable containing the admitting diagnosis edit. The variable will contain the value 0 if the admitting diagnosis is valid or 1 if the admitting diagnosis is invalid.

DXFLGPTR Address of a 325-byte field containing the diagnosis code edits starting with the principal diagnosis. 13 bytes for each of 25 diagnosis codes. The variable will contain the value 0 if the edit was not applicable or 1 if the edit was applicable. Table 5–5 provides a description for each of the 13 edit bytes.

Table 5–5. MCE diagnosis code edits

Byte	MCE diagnosis edit
1	Invalid diagnosis code
2	Sex conflict
3	Age conflict
4	Questionable admission
5	Manifestation code as principal diagnosis
6	Nonspecific principal diagnosis
7	E-code as principal diagnosis

Table 5–5. MCE diagnosis code edits *(continued)*

Byte	MCE diagnosis edit
8	Unacceptable principal diagnosis
9	Duplicate of principal diagnosis
10	Medicare is secondary payer
11	Requires secondary diagnosis
12	Type of age conflict: 0 = No age conflict 1 = Newborn 2 = Pediatric 3 = Maternity 4 = Adult
13	POA indicator invalid or missing *(for future use)*

PRFLGPTR Address of a 400-byte field containing the procedure code edits. 16 bytes for each of 25 procedure codes. The variable will contain the value 0 if the edit was not applicable or 1 if the edit was applicable. Table 5–6 provides a description for each of the 16 edit bytes.

Table 5–6. MCE procedure code edits

Byte	MCE procedure edit
1	Invalid procedure code
2	Sex conflict
3	Nonspecific O.R. procedure
4	Open biopsy check
5	Non-covered procedure
6	Bilateral procedure
7	Limited coverage – Lung volume reduction surgery (LVRS)
8	Limited coverage – Lung transplant
9	Limited coverage – Combination heart/lung transplant
10	Limited coverage – Heart transplant
11	Limited coverage – Implant of heart assist system
12	Limited coverage – Intestine/multi-visceral transplant
13	Limited coverage – Liver transplant
14	Limited coverage – Kidney transplant
15	Limited coverage – Pancreas transplant
16	Limited coverage – Artificial heart transplant

BUFFPTR Address of a 78-byte buffer (MCEBUFF) that must be allocated by your interface program. The software will produce a summary of errors for each record and will put the summarized information in this buffer, along with the provider number, PPS indicator, and edit flag. Table 5–7 is a description of the buffer.

The accumulators at positions 17 through 62, and 69 through 76, contain the counts of the number of occurrences of each of the error conditions related to diagnoses and/or procedures. Those for which the count cannot exceed 1 are designated with an asterisk (*).

Table 5–7. Buffer description

Byte	Datatype	Description
1	pic 9(15).	Medicare provider number
16	pic 9.	PPS indicator
17	pic 99.	Invalid ICD-9-CM code
19	pic 99.	Sex conflict
21	pic 99.	Age conflict
23	pic 99.	* Questionable admission
25	pic 99.	* Manifestation as principal dx
27	pic 99.	* Non-specific principal dx (versions 2.0-23.0 only)
29	pic 99.	* E-code as principal dx
31	pic 99.	* Unacceptable principal dx
33	pic 99.	Duplicate of principal dx
35	pic 99.	MSP alert (versions 15.0–17.0 only)
37	pic 99.	Principal dx requires secondary dx
39	pic 99.	Non-specific procedure (versions 15.0-23.0 only)
41	pic 99.	Open biopsy check
43	pic 99.	Non-covered procedure
45	pic 99.	*Bilateral procedure
47	pic 99.	LVRS - Limited coverage
49	pic 99.	Lung transplant - Limited coverage
51	pic 99.	Combo heart/lung transpl - Limited coverage
53	pic 99.	Heart transplant - Limited coverage
55	pic 99.	Implantable hrt assist - Limited coverage
57	pic 99.	Intest/M. visceral transpl - Limited coverage
59	pic 99.	Liver transplant - Limited coverage
61	pic 99.	* Invalid admit dx

Table 5–7. Buffer description *(continued) (continued)*

Byte	Datatype	Description
63	pic 99.	* Invalid age (not between 0 and 124 years)
65	pic 99.	* Invalid sex (not 1 or 2)
67	pic 99.	* Invalid or missing discharge status code (not 01-08, 20, 30, 43, 50, 51, 61–66) **Note**: Some discharge status codes are not valid for all date ranges. *See table 5–2 on page 5.5*.
69	pic 99.	Kidney transplant - Limited coverage
71	pic 99.	Pancreas transplant - Limited coverage
73	pic 99.	POA indicators invalid or missing *(for future use)*
75	pic 99.	Artificial heart transplant - Limited coverage
77	pic 99.	MCE edit flag

The accumulators at positions 45 and 63 through 68 will have a count of 1 if the error is present, and 0 otherwise. They are effectively the flag bytes for these errors.

Flag values The MCE edit flag is set by the software to values shown in table 5–8.

Table 5–8. Edit flag values

Value	Description
0	No errors
1	Pre-payment error Non-covered procedure Questionable admission Age conflict Sex conflict Invalid ICD-9-CM code E-code as principal diagnosis Manifestation as principal diagnosis Unacceptable principal diagnosis Invalid age, sex or discharge status Duplicate of PDX, Requires secondary dx Limited coverage
2	Post-payment error Non-specific diagnosis Non-specific procedure Bilateral procedure Biopsy check MSP alert (versions 2.0–17.0 only)
3	Both pre-payment and post-payment errors
4	Discharge date invalid or missing
50	MCE table (MCE250RT) could not be opened or is corrupted

EDflag is not set for admitting diagnosis.

DSCPTR Used with the report programs. This is the address of a binary (PIC 9(8) COMP) fullword indicating whether the ICD-9-CM English description file will be accessed. A value of 0 (zero) indicates that no English descriptions are wanted, while a value of 1 indicates that descriptions will be printed. If descriptions are bypassed, their area in the print line is blanked out. No change in print format occurs.

OPTPTR Used with the report programs. Up to 11 OPTPTRs may be present, with each one pointing to a 40-byte user-allocated area containing additional patient information (patient I.D., length of stay, etc.) that is to be included as a line of output on the

report. The report program will print the entire 40 bytes "as is" for each option line present. These lines will be printed immediately below the title line and before the standard information (provider number, PPS status, age, sex, discharge status, date, diagnosis and procedures) is reported.

For MCE250PB and MCE250PC, at least two OPTPTRs must be present, and they must be the 17th and 18th pointers in the control block. The first OPTPTR (#17) must be the address of an area allocated to hold report lines. The second OPTPTR (#18) must be the address of a fullword into which the report programs (MCE250PB or MCE250PC) will place a binary (PIC 9(8) COMP) count of the actual number of report lines used. The remaining OPTPTRs can be used as described above. *See chapter 6 for a full explanation of the report programs.*

It is the user's responsibility to set a flag in the last pointer to indicate the end of the pointer list. To set the end-of-list flag the high-order bit of the last pointer must be turned on. For COBOL programmers, the CALL... USING statement automatically sets the end-of-list indicator.

Chapter 6

The report programs

Contents

Chapter 6

The report programs

THREE REPORT PROGRAMS ARE INCLUDED in the Medicare Code Editor (MCE) program. All versions of the software are compatible with the report programs discussed below.

MCE250PA This standard report program (format A) prints each patient record on a separate page. MCE250PA output is written to a file with the DD name RPTFILE which can be allocated to the printer, a disk, or tape file. RPTFILE is opened on the initial call to MCE250PA. In order to close RPTFILE, MCE250PA must be called with Register 1 set to a value of zero (for COBOL, a CALL MCE250PA with no parameters will have this effect). If you are blocking RPTFILE records, closing the file is essential, or the last block of output may be lost.

MCE250PB This report program (format B) is the same as above, except instead of printing, a mirror of the report page is returned with a count of lines. The first position of each line is the carriage control character, having one of the values shown in table 6–1.

Table 6–1. Carriage control character values

Value	Explanation
1	Skip to new page before printing line
0	Space two lines before printing current line
Blank	Normal print spacing

MCE250PC This report program (format C) returns a series of taglines, one tagline for each diagnosis and procedure on the patient record. The tagline will include any error messages. Each line follows a coded tag which explains the tagline. The tag numbers are 4-byte numerics. There will be two or more lines with the same tag number if more than one error is found for the same diagnosis or procedure. Taglines are explained in table 6–2.

Table 6–2. MCE250PC tagline format

Tag number	Tagline
0101-0116	Diagnosis line, with 101 as the admitting diagnosis, 102 as the principal diagnosis, 103 as the first secondary diagnosis, etc.
0201-0215	Procedure line, with 201 as the first listed procedure, 202 the second, etc.
0301	Line indicating invalid discharge disposition
0401	Line indicating invalid age
0501	Line indicating invalid sex

Table 6–3 is an example of the taglines that might be returned by MCE250PC.

Table 6–3. MCE250PC tagline example

Positions 1-4	Positions 5-84
0101	V1087 Hx of thyroid malignancy
0102	V1087 Hx of thyroid malignancy Unacceptable principal diagnosis
0103	462 Acute pharyngitis
0201	064 Complete thyroidectomy
0202	403 Regional lymph node exc
0301	Invalid discharge disposition

Uses for the report programs

The MCE user may utilize the report programs in a variety of ways. Some uses are suggested below. In each example, your interface program would do the flag testing and decide whether or not to call one of the print programs.

Remember that for MCE250PB and MCE250PC, the first two OPTPTRs are used for passing and receiving line information. Refer to chapter 5 for an explanation of the OPTPTRs.

To run the report programs, Register 1 must again be pointing to the control block.

MCE250PA MCE250PA could be called each time the edit flag (position 77-78 in the MCEBUFF) was returned with a value greater than zero (remember that invalid admitting diagnosis does not set this flag). This would generate a one-page report for the patient record on which errors were detected.

Alternatively, MCE250PA could be called for each record, whether or not any flags were set, which could produce a large volume of print output.

MCE250PA output could be written to a tape or disk file for later printing.

MCE250PB MCE250PB could be called if you wanted to have all of the basic information in the standard report, but wanted to insert lines or edit the report before printing. Your interface program must allocate sufficient space for holding a full report. It must also control all I/O to the printer, as MCE250PB does not actually generate any printing.

The allocated area can be calculated by multiplying the expected maximum number of lines by 81 (the maximum number of characters per line).

MCE250PC MCE250PC could be called if you wanted the editor information for each diagnosis and procedure, but wanted to integrate it with other information tailored to your own reporting requirements.

Your interface program must allocate space. In this instance, the space requirement would be:

```
(maximum diagnosis (NDXPTR value) + maximum procedures (NSGPTR value) +3) x 84.
```

If English descriptions are bypassed, the above statement is excluded. *For more information, see DSCPTR in chapter 5.*

Figure 6–1 illustrates a compile-link-go including the standard report program (MCE250PA). If English descriptions are bypassed, the line marked with the dagger (†) is excluded.

```
 //JOB CARD FOR YOUR INSTALLATION
//* ***************************************************
//* THIS JOB IS USED TO COMPILE, LINK AND RUN THE MCE
//* COBOL TEST PROGRAM, COBTEST.
//*
//* BOTH OBJECT AND LOAD MODULES ARE TEMPORARY.
//* ***************************************************
//COBUCLG PROC SYSOUT='*'
//*   COBOL FOR MVS COMPILE AND LINK
//COB    EXEC  PGM=IGYCRCTL,PARM='RENT,NODYNAM'
//STEPLIB  DD  DSN=IGY.V2R2M0.SIGYCOMP,DISP=SHR
//SYSLIB   DD  DSN=YOURID.&PROD..SRCLIB,DISP=SHR
//SYSPRINT DD  SYSOUT=&SYSOUT
//SYSIN    DD  DSN=YOURID.&PROD..SRCLIB(COBTEST),DISP=SHR
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT2   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT3   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT4   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT5   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT6   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT7   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSLIN   DD  DSN=&&LOADSET,UNIT=SYSDA,DISP=(MOD,PASS),
//             SPACE=(TRK,(3,3)),DCB=BLKSIZE=800
//*
//LKED   EXEC  PGM=IEWL,PARM='LIST,MAP,AMODE=31,RMODE=ANY',
//             COND=(5,LT,COB)
//SYSLIB   DD  DSN=CEE.SCEELKED,DISP=SHR
//SYSLMOD  DD  DSN=&&GOSET(COBTEST),UNIT=SYSDA,DISP=(,PASS),
//             SPACE=(CYL,(5,1,5))
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD  SYSOUT=&SYSOUT
//SYSLIN   DD  DSN=&&LOADSET,DISP=(OLD,DELETE)
//         DD  DDNAME=SYSIN
//OBJECT   DD DSN=YOURID.&PROD..OBJLIB,DISP=OLD
//*
//GO     EXEC  PGM=COBTEST,COND=((5,LT,COB),(5,LT,LKED))
//STEPLIB  DD DISP=SHR,DSN=&&GOSET
//         DD DISP=SHR,DSN=CEE.SCEERUN
//INFILE   DD DSN=YOURID.&PROD..TESTDB,DISP=SHR
//SYSPRINT DD SYSOUT=&SYSOUT
//MCE250I9 DD DSN=YOURID.&PROD..VSFILE,DISP=SHR †
//RPTFILE  DD SYSOUT=&SYSOUT,DCB=(RECFM=FA,BLKSIZE=81,BUFNO=1)
//         PEND
//*
//PROG1    EXEC COBUCLG,PROD=MCE250
//*
//LKED.SYSIN DD *
 INCLUDE OBJECT(MCE250CN,MCE250ED,MCE250LB,MCE250RT)
 INCLUDE OBJECT(MCE250PA,MCE250VS,MCE250DT)
 ENTRY COBTEST
 NAME COBTEST
/*
```

Figure 6–1. Example of print JCL using a COBOL interface program

Appendix A

MCE software edits

Appendix A

Appendix A

MCE software edits

THE EDITS CONTAINED in the current Medicare Code Editor (MCE) software are listed here with their descriptions. These edits are the official MCE edits specified by the Centers for Medicare and Medicaid Services (CMS).

Below is a list of edit messages described in this appendix:

1. Invalid diagnosis or procedure code
2. E-code as principal diagnosis
3. Duplicate of PDX
4. Age conflict
5. Sex conflict
6. Manifestation code as principal diagnosis
7. Non-specific principal diagnosis (*Discontinued as of 10/01/07)*
8. Questionable admission
9. Unacceptable principal diagnosis
10. Non-specific O.R. procedure (*Discontinued as of 10/01/07)*
11. Non-covered procedure
12. Open biopsy check
13. Bilateral procedure
14. Invalid age
15. Invalid sex
16. Invalid discharge status
17. Limited coverage

➤ ***Note:*** *Effective 10/01/01, the MSP (Medicare as secondary payer) alert edit was discontinued and will appear for claims processed using MCE versions 2.0–17.0 only.*

➤ ***Note:*** *Effective 10/01/07, the non-specific principal diagnosis edits and non-specific O.R. procedure edits were discontinued and will appear for claims processed using MCE version 2.0-23.0 only.*

1. Invalid diagnosis or procedure code

MCE software checks each diagnosis code, including the admitting diagnosis, and each procedure code against a table of valid ICD-9-CM codes. If an entered code does not agree with any code on the internal list, the entered code is considered invalid or as having an invalid or missing 4th, 5th, or 6th digit.

➤ ***Note:*** *Effective 10/01/1994, the Invalid 4th or 5th digit edit was discontinued and the Invalid diagnosis or procedure code edit displayed in its place.*

2. E-code as principal diagnosis

E-codes describe the circumstance(s) that caused an injury, not the nature of the injury (e.g., fall from bed), and therefore should not be used as a principal diagnosis. E-codes are all ICD-9-CM diagnosis codes that begin with the letter E.

3. Duplicate of PDX

Whenever a secondary diagnosis is coded the same as the principal diagnosis, the secondary diagnosis is identified by MCE software as a duplicate of the principal diagnosis. This is because the diagnosis code may be considered a complication or comorbidity (CC) and will create an error in DRG assignment if the DRG is affected by the presence of a CC.

4. Age conflict

MCE software detects inconsistencies between a patient's age and any diagnosis on the patient's record. Examples of age conflicts are a five-year-old patient with benign prostatic hypertrophy, and a 78 year-old patient with a delivery. In such cases, either the diagnosis or age is presumed to be incorrect. There are four age code categories: newborn (less than one year), pediatric (0–17 years inclusive), maternity (12–55 years inclusive), and adult (15–124 years inclusive).

5. Sex conflict

MCE software detects inconsistencies between a patient's sex and any diagnosis or procedure on the patient's record. Examples of sex conflicts are a male patient with cervical cancer (diagnosis) and a male patient with a hysterectomy (procedure). In such cases, either the diagnosis, procedure, or sex is presumed to be incorrect.

6. Manifestation code as principal diagnosis

Manifestation codes describe the manifestation of an underlying disease, not the disease itself, and therefore should not be used as a principal diagnosis.

7. Non-specific principal diagnosis

Discontinued as of 10/01/07.

A set of diagnosis codes, particularly those described as "not otherwise specified" (NOS), are identified by the software as non-specific. While these codes are valid ICD-9-CM codes, more precise codes should be used for the principal diagnosis. It should be noted that a diagnosis is considered non-specific only if the patient was discharged alive; patients who have died often do not receive a complete diagnostic workup, and specification of a precise principal diagnosis may not be possible.Non-specific principal diagnosis.

8. Questionable admission

There are some diagnoses which are not usually sufficient justification for admission to an acute care hospital (e.g., benign hypertension). In these cases, the diagnosis code is flagged.

9. Unacceptable principal diagnosis

There are selected codes that describe a circumstance which influences an individual's health status but is not a current illness or injury (e.g., family history of ischemic heart disease) or codes that are not specific manifestations but may be due to an underlying cause. Such codes are considered unacceptable as a principal diagnosis. In a few cases, some unacceptable codes will be acceptable as principal diagnosis if any secondary diagnosis is coded; for these codes, the software displays a "Requires secondary dx" message next to the code in place of the "Unacceptable principal diagnosis" edit.

10. Non-specific O.R. procedure

Discontinued as of 10/01/07.

A set of O.R. procedure codes, particularly those described as "not otherwise specified" (NOS), are identified by the software as non-specific. While these codes are valid ICD-9-CM codes, more precise codes should be used. It should be noted that the non-specific O.R. procedure condition is reported only if all the O.R. procedures performed have been coded as non-specific; if a patient had several O.R. procedures and only one was non-specific, the edit would not be generated.

11. Non-covered procedure

Medicare does not provide reimbursement for some procedures and their codes are flagged by the software. Some non-covered procedures are covered under certain circumstances with particular principal or secondary diagnoses, as specified by CMS.

12. Open biopsy check

Biopsies can be performed as open (i.e., a body cavity entered surgically), percutaneous, or endoscopic procedures. Patients are assigned to different DRGs depending on whether or not the biopsy was open. ICD-9-CM codes are explicit for open and non-open biopsies; however, the distinction made by the codes is not applied uniformly. MCE software identifies all biopsies that are coded as open biopsies, and suggests the corresponding non-open biopsy code to use, if applicable.

13. Bilateral procedure

Certain codes do not accurately reflect procedures that are performed in one admission on two or more different bilateral joints of the lower extremities. A combination of these codes shows a bilateral procedure when, in fact, they could be procedures performed on a single joint (i.e., duplicate procedures). When two or more different joint replacement procedures are coded, this edit instructs the fiscal intermediary to make sure that these procedures were performed on two separate joints.

14. Invalid age

A patient's age is usually needed for correct DRG grouping. If the age reported is outside the valid range (0–124 years), the software assumes the age is in error.

15. Invalid sex

A patient's sex is sometimes needed for correct DRG grouping. The sex code reported must be either 1 (male) or 2 (female). If the entry is not either of these values, the software flags the record.

16. Invalid discharge status

A patient's discharge status is sometimes needed for correct DRG grouping. Discharge status must be coded according to the UB–04 conventions. Note that when an invalid discharge status is reported, the patient is presumed to have been discharged alive for the purpose of performing the non-specific principal diagnosis check.

17. Limited coverage

For certain procedures whose medical complexity and serious nature incur extraordinary associated costs, Medicare limits coverage to a portion of the cost. The edit applies to such procedures as lung volume reduction surgery (LVRS), an implantable heart assist system, and major organ transplants.

Appendix B

Summary of changes

Contents

Appendix B

Summary of changes

MODIFICATIONS MADE to the Medicare Code Editor (MCE) software and effective in the current release are summarized below.

Software

- Basic changes to accommodate table and date range modifications.
- New version 25.0 with an effective date range of 10/01/2008–09/30/2009.

Tables

The tables have been updated with information for MCE software versions 2.0 through 25.0.

Documentation

The following code lists for edits were updated. For more information, refer to chapter 2 of the *Definitions of Medicare Code Edits* guide.

- Diagnoses for Newborns
- Diagnoses for Adults - (age greater than 14)
- Diagnoses for Maternity -age 12-55 years old
- Diagnoses for Females Only
- Unacceptable Principal Diagnoses
- E-Code as Principal Diagnosis
- Bilateral Procedure
- Non-Covered Procedures
- Limited Coverage

Index

E

F

H

I

J

L

M

N

O

P

Q

R

S

T

U

V