Technical Documentation for DLL Version 2.40 for the CMG Classification System Version 2.40

Centers for Medicare & Medicaid Services

August 3, 2009

This documentation is for a revised version of the CMG dynamic-link library (DLL) for use in classifying patients in Inpatient Rehabilitation Facilities (IRFs). This revision represents version 2.40 of the DLL. This revised DLL implements a new version of the CMG classification system, version 2.40.

This documentation and the CMG DLL are published by the Centers for Medicare & Medicaid services (CMS). The documentation and DLL may be copied freely, as our goal is broad dissemination to facilitate accurate classification of Inpatient Rehabilitation Facility (IRF) patients with the Case Mix Group (CMG) classification system.

Disclaimer

To the maximum extent permitted by applicable law, CMS and its contractors/distributors make no representations about the suitability, for any purpose or use, of the software and documents. THE SOFTWARE AND DOCUMENTS ARE PROVIDED "AS IS" AND WITHOUT WARRANTY OF ANY KIND. ANY WARRANTIES, INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR USE, TITLE, AND NON-INFRINGEMENT ARE HEREBY DISCLAIMED. Recipient shall not, in any event, be entitled to, and CMS or contractors/distributors shall not be liable for indirect, special, incidental, or consequential damages of any nature including, without being limited to, loss of use, data, or profit irrespective of the nature of the claim, whether in contract, tort, warranty, or otherwise.

Introduction

This documentation describes a Windows DLL that can be used to calculate a patient's CMG group from Inpatient Rehabilitation Facility Patient Assessment Instrument (IRF-PAI) data. CMG_240.DLL is a 32-bit ActiveX DLL that can be called from Visual Basic or from 32-bit C++ applications. The DLL accepts an input string containing an IRF-PAI data record in the standard CMS submission format and returns the CMG value and associated information. Documentation for the IRF-PAI standard submission format, as well as the DLL and associated documentation, is available at:

http://www.cms.hhs.gov/InpatientRehabFacPPS/06_Software.asp#TopOfPage

The ZIP file (CMG_240.ZIP) containing the DLL software and this documentation may be freely distributed and you may freely use and distribute the DLL in your applications.

Later sections of this document explain how to call the DLL from Visual Basic and from C++. Following these sections, there are several sections that should be read by all users. The first describes the input and output parameters used when calling either DLL. The second section describes the parameter file that must be distributed with the DLL. The third explains the error codes that are returned by the DLL.

Current Changes with DLL Version 2.40

Version 2.40 of the DLL is being released in August, 2009. The prior DLL Version was 2.30. All changes made with DLL Version 2.40 involve addition and deletion of ICD9 comorbidity codes. These comorbidity changes were required by the FY 2010 annual revision to the ICD9 codes. The four comorbidity additions are documented in detail in Appendix C of the Program Documentation (ProgDoc.pdf). The one comorbidity deletion is documented in Appendix D of the Program Documentation (ProgDoc.pdf).

Versioning

There are two separate version designators related to the DLL. The first is the version of the CMG classification model. The current version of the CMG model has been designated by CMS as version 2.40 and this version is based on the FY 2010 annual revision to the ICD9 codes. It is implemented in the DLL that is named CMG_240.DLL to indicate that it computes the version 2.40 model. The class implemented by the DLL is named CMG_240, and the property CmgVersion which is returned by the DLL has a value of "2.40". The name of the DLL, the class, and the CmgVersion will not change unless CMS publishes a new regulation containing a new CMG model.

The DLL also has its own internal version number which can be different than the CMG model version number. The DLL version number is initially set the same as the CMG model version number. Thus, this initial version of CMG_240.DLL has a version number of "2.40" (available from the DLLVersion property). If the DLL for CMG model version 2.40 is later corrected or revised, then a new DLL version number will be used (e.g., "2.41"). However, regardless of the DLL version number, the name of the DLL and of the class that calculates version 2.40 of the CMG model will be CMG_240.

Contents of the CMG_240.ZIP File

The CMG_240.ZIP file contains the following files:

- **Techdoc.pdf --** The current document, which explains the use of the DLL.
- **Progdoc.pdf --** Explains how to calculate CMG group codes. This document is primarily for programmers who wish to write their own CMG calculation software rather than using the DLL. Programmers who wish to use the DLL do not need to read this document to use the DLL correctly.
- **CMG_240.DLL** -- The 32-bit ActiveX DLL that computes CMG group codes.
- **CMG_240.DAT** -- A parameter file that must be distributed with the DLL.
- **VbDemo.frm and VbDemo.vbp** -- Program source code for a Visual Basic 6.0 application which demonstrates how to call and use the DLL. These files should assist in the development of CMG applications in Visual Basic and serve as model for applications in other programming languages.
- **VbDemo_frm.pdf and VbDemo_vbp.pdf --** Adobe Acrobat versions of the Visual Basic source code (508 compliant).
- **CmgTest.txt** -- IRF-PAI assessment test data file used by the Visual Basic application. This test data file can be used by developers in testing their applications using Visual Basic and other programming languages.
- **VbDemo.log --** This file provides the correct CMG classification results for each test record in "CmgTest.txt" and will assist developers in testing their applications. This file is actually the output of the Visual Basic application program when run with the CmgTest.txt assessment test data file.
- **VbDemo_log.pdf** -- Adobe Acrobat version of the "VbDemo.log" file (508 compliant).

Developing CMG Applications

The VbDemo source code illustrates how to call and use the DLL from Visual Basic and should serve as a model for developing Visual Basic applications and applications in other languages. In order to see the operation of the Visual Basic demo program, you must do the following:

1. Compile the VbDemo source code to produce an executable (exe) file. Place that file in an application directory.
2. Register the DLL under Windows using RegSvr32.exe.

3. Place the file CMG_240.DAT file in the application directory (this is a parameter file used by the DLL and its use is explained later in this document).
4. Place the file CMGTEST.TXT in the application directory. This is a file containing IRF-PAI test data that is read by the demo programs.
5. Execute the VbDemo application.

If VbDemo is working correctly, a window titled "CMG Demo Program" will appear and "CMG/DLL Version 2.40/2.40" will be displayed. Click the "Process" button and a results window should indicate 170 records processed with no CMG mismatches. A log file (VbDemo.log) will also be produced that gives the correct and calculated CMG codes (CMG IN/OUT), error code, motor score, cognitive score, age, impairment group, and comorbidities.

The source code for the demo program (VbDemo.frm and VbDemo.vbp) is well commented and explains the DLL usage in detail. Please refer to the source code files for further details.

Calling CMG_240.DLL from Visual Basic

CMG_240.DLL contains a class called CMG_240 that can be called from applications developed using Microsoft Visual Basic 6.0. A sample application called VbDemo (files VbDemo.vbp and VbDeno.frm) is included with CMG_240.ZIP that illustrates the use of the DLL from a Visual Basic application.

When you distribute an application that uses CMG_240.DLL, you must register the DLL with Windows using RegSvr32.Exe. In addition, you must distribute the CMG_240.DAT file that contains calculation parameters used by the DLL. Details about CMG_240.DAT are contained in a later section of the document.

In order to use the CMG class, you must create a local instance of it using a statement like the following:

```
Dim CMG As New CMG_240
```

To perform the CMG calculation, you must follow these three steps:

1. Set certain required properties.
2. Invoke the CmgCalc method.
3. Get the values of output properties.

A similar approach would be taken if calling the DLL from other languages.

A later section of this documentation, *Parameters Used by the DLL*, lists the parameters used to calculate CMGs. The parameters listed as input parameters are the ones that you must set before invoking the CmgCalc method. The parameters listed as output parameters are the ones that you may get after invoking the CmgCalc method.

The statements below illustrate the usage of the CMG class:

```
Dim Cmg As New CMG_240
Dim sRecord As String
Dim sCmg As String
Dim iError As Integer
Dim nMotor As Single
Dim iAge As Integer
Dim iCognitive As Integer
Dim sDllVersion as String

'This code snippet assumes that data from an IRF-PAI record
'has been placed in the string variable sRecord in the CMS
'standard submission format.
```

```
'Set properties to prepare for calculation
Cmg.IrfPaiRecord = sRecord       'Sets IRF-PAI record string

'Invoke CmgCalc method to perform calculation
Cmg.CmgCalc

'Get return value properties
sCmg = Cmg.CmgValue              'Gets calculated CMG group
iError = Cmg.ErrorCode           'Gets calculation error code

'Optional properties which might be useful
nMotor = Cmg.Motor               'Gets motor scale score
iCognitive = Cmg.Cognitive       'Gets cognitive scale score
iAge = Cmg.Age                   'Gets calculated age
sDllVersion = Cmg.DllVersion     'Gets DLL version
```

Parameters Used by the DLL

Note that the parameters listed below correspond exactly with properties. You must "set" the properties listed as input parameters, and "get" the properties listed as output parameters.

Parameter Name	**Parameter Type**	**Data Type**	**Description**
IrfPaiRecord	Input	String	A string containing the IRF-PAI record in standard CMS format (length=1258 bytes).
CmgValue	Output	String	CmgValue is a 5-byte string containing the CMG group. You must set the IrfPaiRecord property and invoke the CmgCalc method for the CMG group to be placed in CmgValue. If no value has been calculated or the ErrorCode property is non-zero, CmgValue will contain 5 spaces.
ErrorCode	Output	Integer	If ErrorCode is equal to zero, no errors occurred and CmgValue will contain a valid CMG group designation. If an error occurred, ErrorCode will be non-zero and CmgValue will contain 5 spaces. The possible values of ErrorCode are listed in a later section of this document.
Motor	Output	Single	Calculated CMG motor score. If ErrorCode is equal to zero, the value of Motor will be between 12.0 and 84.0, otherwise Motor will be equal to zero.
Cognitive	Output	Integer	Calculated CMG cognitive score. If ErrorCode is equal to zero, the value of Cognitive will be between 5 and 35, otherwise Cognitive will be equal to zero.
Age	Output	Integer	Calculated age (computed by comparing date of birth with date of admission). If ErrorCode is equal to zero, the value of age will be between 0 and 140, otherwise Age will be equal to zero.
CmgVersion	Output	String	CmgVersion is a string which returns a value of "2.40", the version number of the present CMG calculation model.
DllVersion	Output	String	DllVersion is a string which returns a value of "2.40", the version number for the present DLL.

Parameter Name	Parameter Type	Data Type	Description
DatPath	Input/Output	String	DatPath is the path containing the CMG_240.DAT parameter file. DatPath can be either an input property (i.e., you can "set" its value) or an output property (i.e., you can get its value). When the DLL is initialized, it attempts to read CMG_240.DAT from the directory that contains the DLL. If you wish to store the CMG_240.DAT file in a different directory, you must set the DatPath property appropriately before invoking the CmgCalc method. When you set the DatPath property, the DLL will automatically attempt to read CMG_240.DAT from the directory you specify (unless it has already successfully read the file, in which case the value you set is ignored). When you get the DatPath property, it will return the path from which it successfully read CMG_240.DAT. If the file has not been successfully read, DatPath will return the last path from which a read was attempted.
Checksum	Output	Double	Checksum returns the check sum value that was computed while reading CMG_240.DAT. If the file was found and read correctly, the checksum value will be 4,058,525,844. If an error occurred reading the DAT file, ErrorCode will have a value of 1, 2, or 3, depending upon the exact error encountered. If the valid checksum value is not obtained, then the DAT file has been altered. When the file has been altered, ErrorCode will be set to 3 and the DLL will not calculate a CMG value (this value will be blank). You do not normally need to check the value of Checksum since ErrorCode will automatically be set if an error is encountered. The Checksum property is included primarily to assist in debugging errors.

CMG_240.DAT File

In order to perform its calculations, the DLL requires certain parameters that are contained in an ASCII file called CMG_240.DAT. This file must be distributed with the DLL. By default, the DLL will attempt to read CMG_240.DAT from its own directory (i.e., the directory which contains the DLL itself). If you wish to place CMG_240.DAT in a different directory, you must set the DatPath property to tell the DLL where to find the file. For example, if you placed CMG_240.DAT in the directory *C:\My Application\Datfile*, you would set the DatPath property as follows:

Cmg.DatPath = "c:\my application\datfile"

You must set this property before invoking the CmgCalc method. If the file cannot be found and successfully read by the DLL prior to calling the CmgCalc method, an ErrorCode of 1, 2, or 3 will be set (according to the exact error encountered).

To insure the integrity of the CMG_240.DAT file, a checksum is calculated when it is read. If the value obtained is not equal to the expected value, then the DAT file has been altered. In this case, ErrorCode will be set to 3 and the DLL will not calculate a CmgValue (this value will be blank).

Error Codes

When the CmgCalc method is invoked, the DLL sets the value for ErrorCode. If no errors occur, the value is set to zero and a non-blank CmgValue is set. Otherwise, ErrorCode is set to a non-zero value and CmgValue is set to 5 spaces. Your code should always check the value of ErrorCode after every call to CmgValue.

Many of the error codes are related to invalid data in the fields that are used to calculate the CMG. The DLL does not check every field in the data record. The data checks applied by the DLL are therefore a subset of CMS's standard IRF-PAI data edits. It is therefore possible for the DLL to return a valid CMG value from an IRF-PAI record that would be rejected by the standard CMS IRF-PAI system. This could happen if the record had an error in a field that is not used to calculate the CMG.

The values returned in ErrorCode are as follows:

0 = no errors
1 = DAT file not found or open error
2 = DAT file read error
3 = DAT file checksum error
4 = Rec_ID not = "B2"
5 = computed age is greater than 140 years
6 = computed age is negative
7 = year of partial birthdate equals year of admission
8 = invalid birth date
9 = invalid admission date
10 = Eating_a (admission value) is out of range
11 = Grooming_a (admission value) is out of range
12 = Bathing_a (admission value) is out of range
13 = Dress_upper_a (admission value) is out of range
14 = Dress_lower_a (admission value) is out of range
15 = Toileting_a (admission value) is out of range
16 = Bladder_a (admission value) is out of range
17 = Bowel_a (admission value) is out of range
18 = Trans_bed_a (admission value) is out of range
19 = Trans_toilet_a (admission value) is out of range
20 = Walk_wheel_a (admission value) is out of range
21 = Stairs_a (admission value) is out of range
22 = Comprehend_a (admission value) is out of range
23 = Expression_a (admission value) is out of range
24 = Soc_interact_a (admission value) is out of range
25 = Prob_solving_a (admission value) is out of range
26 = Memory_a (admission value) is out of range
27 = Comorb_a code has invalid format
28 = Comorb_b code has invalid format
29 = Comorb_c code has invalid format
30 = Comorb_d code has invalid format
31 = Comorb_e code has invalid format
32 = Comorb_f code has invalid format
33 = Comorb_g code has invalid format
34 = Comorb_h code has invalid format
35 = Comorb_i code has invalid format
36 = Comorb_j code has invalid format
37 = Impairment group code is out of range

DLL Version History

The DLL version number is returned by the DllVersion parameter described earlier in this documentation.

Version	Comments
Beta 01	Initial beta release.
Beta 02	Second beta release.
Beta 03	Third beta release.
Beta 04	Fourth beta release.
Beta 05	Fifth beta release.
Beta 06	Not released.
Beta 07	Seventh beta release. Posted on CMS website on 10/2001.
1.10	Second public version of the DLL, posted on CMS website on 6/2002.
1.20	Third public version of the DLL, released in July 2004.
1.21	Internal release – not publicly distributed.
1.22	Fourth public version of the DLL, released in December 2004.
2.00	Fifth public version of the DLL, released August 2005. Implements the second version (Version 2.00) of the CMG classification system.
2.01	Sixth public version of the DLL, released August, 2005. The only change with DLL Version 2.01 was to fix a problem in DLL Version 2.00 that prevented an application from loading DLL Version 2.00 and a prior version (e.g., Version 1.22) simultaneously in a Visual Basic 6 application.
2.02	Seventh public version of the DLL, released March, 2006. Removed one comorbidity that had been included in DLL Version 2.00.
2.10	Eighth public version of the DLL, released August, 2006. Many comorbidities were added, deleted, or changed tiers per the FY 2007 IRF PPS Final Rule due to be published on August 1, 2007 and the FY 2007 annual revision to the ICD9 codes.
2.20	Ninth public version of the DLL, released August, 2007. Several comorbidities were added and 2 comorbidities were deleted per the FY 2008 annual revision to the ICD9 codes.
2.30	Tenth public version of the DLL, released August, 2008. Several comorbidities were added and 1 comorbidity was deleted per the FY 2009 annual revision to the ICD9 codes.
2.40	Eleventh public version of the DLL, released August, 2009. Four comorbidities were added and 1 comorbidity was deleted per the FY 2010 annual revision to the ICD9 codes.

Technical Assistance

If you have questions about the DLL which are not answered by this documentation, please refer to the CMS web site for further information or for contact information to receive assistance:

http://www.cms.hhs.gov/InpatientRehabFacPPS/06_Software.asp#TopOfPage