

MICRO-OFFICE SYSTEMS, INC.

---

MicroMD Utilities: AnyCode

MicroMD Utilities:

AnyCode

MICRO-OFFICE SYSTEMS, INC.

# **MicroMD Utilities: AnyCode**

---

© 2011 Micro-Office Systems, Inc.  
3825 Severn Road • Cleveland Heights, Ohio 44118  
Phone 216-297-1240 • Fax 216-297-1241

Last Revision: Friday, March 25, 2011

---

# Table of Contents

<b>BASIC ANYCODE CONFIGURATION</b> .....	<b>1</b>
CONNECTION TAB.....	1
MENU TAB.....	1
UNDERSTANDING RVU VALUES.....	1
<b>LOADING PROCEDURE CODE DATA</b> .....	<b>2</b>
SELECTING A CODE LOAD DATA SOURCE.....	2
<i>Provided Codes: 06Medicare.mdb</i> .....	2
<i>Comma-Separated Text Files</i> .....	3
<i>Custom Files</i> .....	3
SELECTING A LOCALITY.....	4
DETERMINING WHICH PROCEDURES TO MODIFY.....	4
<i>Selecting a Code Range</i> .....	4
<i>Understanding Procedure Modifiers</i> .....	4
ACTIONS ON PROCEDURE CODES.....	5
<i>Adding New Procedure Codes</i> .....	5
<i>Update RVU</i> .....	6
<i>Update Fee</i> .....	6
<i>Update Allowed Amount</i> .....	7
<i>Update Description</i> .....	7
<i>Update Dates</i> .....	8
<i>Mark Invalid</i> .....	8
<b>LOADING DIAGNOSIS CODE INFORMATION</b> .....	<b>9</b>
SELECTING A CODE LOAD DATA SOURCE.....	9
DETERMINING WHICH DIAGNOSIS CODES TO MODIFY.....	10
<i>Selecting a Code Range</i> .....	10
<i>Understanding Diagnosis Type</i> .....	10
ACTIONS ON DIAGNOSIS CODES.....	10
<i>Adding New Diagnosis Codes</i> .....	10
<i>Update Description</i> .....	11
<i>Update Dates</i> .....	11
<i>Mark Invalid</i> .....	11

---

## Basic AnyCode Configuration

### Connection Tab

You should connect to the database with the 'dba' logon.

### Menu Tab

There are two options at the bottom. One adds a menu option and one adds a hot key. Since AnyCode is only called rarely, usually you do NOT want to check these options. Instead, AnyCode should be called directly when it is needed.

### Understanding RVU values

There are three different RVU values associated with each procedure code: the "work RVU", the "practice RVU", and the "malpractice RVU". For a procedure code, the "practice RVU" can further be refined to either the "facility based RVU" or the "non-facility based RVU".

Additionally, each region of the country also has associated with it the same three categories of RVU values.

To determine the RVU for a procedure at a specific location, the following formula is used:

$$\begin{aligned} &(\text{Location Work RVU} * \text{Procedure Work RVU}) + \\ &(\text{Location Practice RVU} * \text{Procedure Practice RVU}) + \\ &(\text{Location Malpractice RVU} * \text{Procedure Malpractice RVU}) \end{aligned}$$

## Loading Procedure Code Data

AnyCode 2011

Connection | Log | MicroMD Menu | Procedure | Procedure Default | Diagnosis

Code Load Data: C:\Users\Dan\Workstation\Product\MicroMD\Utilities\AnyCode\2011-01-03\11\_CMS\_PROC.mdb Update...

Effective Dates: 1/ 1/2011 thru 12/31/2011 2011 Medicare (12/19/10 Release)

Locality: FLORIDA - Fort Lauderdale, FL ?

Code Range: thru

Provider Filter: ..

Modifiers

1 mod per proc code

modifier	proctype	loc
26	2	
TC	T	
53	5	

Allowed Amount

Plans: ..

Ins Class: ..

Ins Cat: ..

POS: PROCEDURE DEFAULT

Fee Calculation

Increase Allowed By: 0.0 %

Round to: 0.01

Actions

Use Facility-based Practice Expense

Add New Codes  Update Fee  Update RVU

Include Fee  Update Allowed  Update Desc

Mark Invalid  Update Contract Allowed  Update Dates

Fee

Plans: ..

Ins Class: ..

Ins Cat: ..

Chg Class: ..

Update Standard Charge

Update Medicare Charge

Use Allowed Amount

About... View Log... Support...

## Selecting a Code Load Data Source

### Provided Codes: 06Medicare.mdb

AnyCode comes with the procedure and diagnosis codes in an Access database available for download from the Micro-Office Systems website. These files are located in the directory within the AnyCode folder. You can use the ellipses “...” button next to the Code Load Data field to select the file. In this documentation, this database will be referred to as the “code load database”.

Once the data source is selected, a number of fields will be filled in, including the name associated with the data source, which will appear immediately below. It should say “YYYY Medicare (release date)” where YYYY is the current year.

### **Comma-Separated Text Files**

In addition to the provided Access database, AnyCode also supports loading procedure code information from a comma-separated value text file. Any field containing a comma may have double quotes surrounding the field value. For example:

```
12345,Procedure ABC,5.3  
23456,"Procedure DEF, Left",8.6
```

In order to load text file information into AnyCode, you must give the text file an extension of “csv”. For example, “c:\mydata.csv”. Then, in the “Code Load Data” section, browse to the ‘csv’ file and select it. When you do this for the first time, a dialog box will appear showing the first few rows of data along with some places at the bottom of the dialog for specifying which field contains the procedure code, the procedure code description, the fee, and the RVU value. The description, fee, and RVU fields may be set to 0 if that information is not present in the text file. In most cases, the code and the description should both be present.

Once that information is entered and accepted, a new file will be created in the directory containing the ‘csv’ file. It will have the same base name, but instead of the ‘csv’ extension, the new file will have a ‘layout’ extension. This file contains the column information specified in the dialog box. Once this has been done, the text file will load exactly like the 05Medicare.mdb file, and you can Add/Update/Delete procedure code information using this data. Note that the ‘csv’ file only allows for a single RVU value specification, and it does not allow for multiple locations. A single ‘All’ location will be created, and it must be selected.

If you make a mistake when setting up the layout information for the text file, simply manually delete the ‘layout’ file. Then, the next time AnyCode is run and the text file is selected, you will be re-prompted to enter the layout information.

*Note: CSV file support is currently NOT available for diagnosis codes.*

### **Custom Files**

If you wish to have an alternate set of codes configured for use with AnyCode, please contact Micro-Office Systems, Inc. to discuss pricing.

## Selecting a Locality

Along with the procedure code information, the Access database also contains a list of all localities along with their associated RVU information. Please select the appropriate locality from the dropdown list. The RVU values will be used to construct the fee and allowed amounts. Note that there are sometimes multiple regions within a state. If you are unsure exactly which counties are included in a particular region, select the region and then click on the “?” button to the right of the region name. That will display a dialog box with extra information about the region.

There is a checkbox for “Facility-based Practice Expense”. This checkbox determines which of the two RVU values will be used when composing the practice component in the RVU formula.

## Determining Which Procedures to Modify

### Selecting a Code Range

In the “Code Range” section, select a range of codes to load. You need to only specify prefixes, so “V” thru “V” will load all codes beginning with a “V”. Leaving the values blank will load all procedure codes in the database.

### Understanding Procedure Modifiers

When the code load database is selected, the “Modifiers” section of the Procedure Tab will be loaded with all of the modifiers present in the database. Each of these modifiers can be optionally marked for loading or not loading. For example, to only load procedures with a ‘26’ modifier, check only the box next to the ‘26’.

Additionally, you can specify a procedure type code to be associated with each modifier type. In MicroMD, each 5 character procedure code can have an optional sixth character associated with it. Two procedure codes with different modifiers must have different procedure type codes, so if you are loading more than one modifier type, you must give them different procedure type code extensions.

Finally, there is one last checkbox in the modifier section. If you check “1 modifier per procedure code”, then only a single modifier type will be loaded for each procedure code, even if multiple options exist in the code load database. It will always be the one which appears first alphabetically, with “no modifier” always taking priority. For example, if a procedure code exists in the code load database with both a 26 modifier and with no modifier, and the “1 modifier per procedure code” option is checked, then only the version with no modifier will be loaded into the MicroMD database, even if both the “blank” modifier and the “26” modifier are checked for loading into MicroMD. Note also that if “1 modifier per procedure code” is checked, then it is NOT necessary to specify a separate procedure type code for each modifier, though you certainly have the option to do so.

## Actions on Procedure Codes

The “Actions” section of the Procedure tab contains six different checkboxes. Each corresponds to an action that will be performed when running the AnyCode load operation.

### Adding New Procedure Codes

Procedure codes from the code load database are matched to existing procedure codes in MicroMD by the 5 character procedure code and the modifier. They are NOT matched based on the procedure type code, which is the optional 6<sup>th</sup> character in MicroMD. Therefore, even if in the modifier section of AnyCode you mapped modifier ‘26’ to a procedure type of ‘2’, AnyCode will still find a match in MicroMD for a given procedure code if there is a modifier of ‘26’ even if the procedure type in MicroMD is NOT ‘2’.

Within the range specified in the code range area, any procedure code / modifier combination in the code load database that does not exist in the MicroMD database will be marked for adding to MicroMD. The only exception to this is if the “1 modifier per procedure code” box is checked. If it is checked, then only a single version of a procedure code will be marked for addition, even if the code load database contains multiple versions of the procedure code (with different modifier values).

When a new procedure code is added, many of the default values come from the “Procedure Default” tab. The other procedure code fields are determined as follows.

Field	Value
Procedure Code	The 5 char procedure code with the optional 6 <sup>th</sup> character procedure type extension as defined in the “Modifier” section of the Procedure tab.
Modifier	The modifier from the Code Load database
Description	The description from the Code Load database
RVU	The composite RVU built from the procedure and location RVU values. The formula is described earlier in this document.
Standard Charge / Medicare Charge	The composite RVU is multiplied by the procedure’s “Conversion Factor”, which is defined in the code load database. The fee is then increased and rounded as defined in the “Fee Calculation” section of the “Procedure” tab. If “Use Allowed” is checked for the Medicare Charge, the allowed amount described below will be used for the Medicare value instead.
Allowed	The composite RVU multiplied by the procedure’s “Conversion Factor”, rounded to two decimal places.
Active From / Active To	If the code load database has a date range defined, and if the “Update Dates” action is checked, then that range will be entered into the active dates for the new procedure. For example, the 05Medicare.mdb database comes with a defined date range of 1/1/2005 through 12/31/2005. Otherwise, the active dates will be left blank in MicroMD.

## **Update RVU**

If the “Update RVU” action is checked, then when procedure / modifier combinations are found in MicroMD that match codes in the code load database, the new composite RVU value is compared to the value in the MicroMD database. If, when rounded to two decimal places, the values are different, then the new RVU value will replace the old one in MicroMD.

## **Update Fee**

If the “Update Fee” action is checked, then when procedure / modifier combinations are found in MicroMD that match codes in the code load database, fee information can be updated in MicroMD. Which fee information gets updated depends on other options in the Procedure tab. **If the “Update Fee” checkbox is not checked, then the other fee options are ignored.**

### **Updating Standard Fee**

If the “Update Standard Charge” checkbox is checked, then the standard charge associated with the procedure (at the procedure level) will be updated if needed. The new standard charge is calculated by multiplying the composite RVU with the procedure’s “Conversion Factor”, which is defined in the code load database. That value is then increased and rounded as defined in the “Fee Calculation” section of the Procedure tab.

### **Updating Medicare Fee**

If the “Update Medicare Charge” checkbox is checked, then the Medicare charge associated with the procedure (at the procedure level) will be updated if needed. The new Medicare charge is calculated by multiplying the composite RVU with the procedure’s “Conversion Factor”, which is defined in the code load database. If the “Use Allowed” checkbox is not checked, then that value is then increased and rounded as defined in the “Fee Calculation” section of the Procedure tab.

### **Updating Plan-Level Fees**

The “Fee” box on the Procedure tab contains three fields for entering plan, insurance class, and insurance category information, respectively. For each box, pressing the button to the right will bring up a list of all of the information from MicroMD and allow for easy selection of values. If any of these three fields are filled in, AnyCode will build a list of corresponding plan values. Thus, if you choose the “Medicare” insurance class, all plans associated with that insurance class will be included in the update.

For each of these plans, AnyCode will check to see if there are already plan-level fees associated with the procedure. If there are, the fee information will be updated, and the writeoff will be adjusted based on the existing allowed amount. If there are not plan-level fees associated with the procedure, new records will be created. The calculated allowed amount will be associated as well when the record is created.

### **Updating Charge Class Fees**

The “Fee” box on the Procedure tab contains a field for entering charge class information. Pressing the button to the right will bring up a list of all of the charge classes in MicroMD and allow for easy selection of classes.

If any charges classes are selected, then for each of these classes, AnyCode will check to see if there are already charge class fees associated with the procedure. If there are, the fee information will be updated. If there are not charge class fees associated with the procedure, new records will be created.

### **Update Allowed Amount**

If the “Update Allowed” action is checked, then when procedure / modifier combinations are found in MicroMD that match codes in the code load database, allowed amount information can be updated in MicroMD. Which information gets updated depends on other options in the Procedure tab. **If the “Update Allowed” checkbox is not checked, then the other allowed amount options are ignored.**

### **Updating Plan-Level Allowed Amounts**

The “Allowed Amount” box on the Procedure tab contains three fields for entering plan, insurance class, and insurance category information, respectively. For each box, pressing the button to the right will bring up a list of all of the information from MicroMD and allow for easy selection of values. If any of these three fields are filled in, AnyCode will build a list of corresponding plan values. Thus, if you choose the “Medicare” insurance class, all plans associated with that insurance class will be included in the update.

For each of these plans, AnyCode will check to see if there are already plan-level allowed amounts associated with the procedure. If there are, the information will be updated, and the corresponding calculated fields will be recalculated. If there are not plan-level allowed amounts associated with the procedure, new records will be created. The fee will be associated with the plan as well when the record is created.

### **Update Description**

If the “Update Description” action is checked, then when procedure / modifier combinations are found in MicroMD that match codes in the code load database, the new description is compared to the value in the MicroMD database. If the values are different, then the new description will replace the old one in MicroMD. It is very common for practices to modify procedure descriptions to suit their particular purpose, and therefore, this option should usually be turned off, since otherwise, custom description information could be overwritten.

## **Update Dates**

If the “Update Dates” action is checked, then when procedure / modifier combinations are found in MicroMD that match codes in the code load database, if the existing code has an end date earlier than the code load database’s end date, the end date will be updated. If there is no end date associated with the code in MicroMD, no end date will be updated.

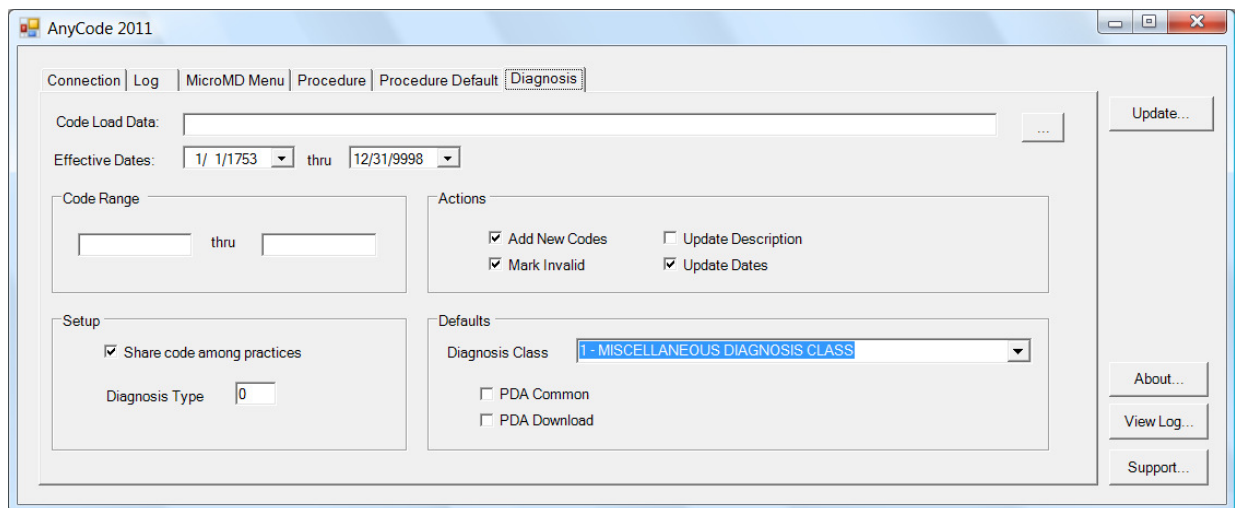
## **Mark Invalid**

If there are procedure codes in the specified range in MicroMD that do NOT appear in the code load database, and if this value is checked, the codes will be marked invalid. Marking a code invalid means that the end date associated with the code will be updated to be the day before the code load database’s start date. For example, using the Medicare.mdb code load database, codes marked invalid would be given an end date of 12/31 of the current year.

MicroMD procedure codes will only be marked invalid if all of the following are true:

- They fall within the range of codes specified
- The procedure code does not appear in the code load database AT ALL. For example, if the code load database has 12345 with a modifier of 26, then no procedure code in MicroMD starting with 12345 will be marked invalid, regardless of modifier or procedure type.
- The procedure code is at least 5 characters long. This eliminates most “custom” procedure codes, which are usually shorter.
- The procedure code in MicroMD has no end date or has an end date past the start date of the code load database. Basically, if the code is already invalid, it does not need to be re-invalidated.
- The first two characters of the MicroMD procedure code are used as the first two characters of some procedure code in the code load database. This prevents custom codes, like “INTRST”, from being marked invalid, since there are no “real” procedure codes beginning with “IN”.

## Loading Diagnosis Code Information



### Selecting a Code Load Data Source

AnyCode comes with the Medicare procedure and diagnosis codes in an Access database. These files are located in the directory within the AnyCode folder. You can use the “...” button next to the Code Load Data field to select the file. In the documentation, this database will be referred to as the “code load database”. If you wish to have an alternate set of codes configured for use with AnyCode, please contact Micro-Office Systems, Inc. to discuss pricing.

Once the data source is selected, a number of fields will be filled in, including the name associated with the data source, which will appear immediately below. It should say “YYYY Medicare (full)” where YYYY is the current year.

**Note:** Since it is possible in the future for procedure and diagnosis codes to be distributed separately, they are loaded separately in the diagnosis and procedure tabs.

## Determining Which Diagnosis Codes to Modify

### Selecting a Code Range

In the “Code Range” section, select a range of codes to load. You need to only specify prefixes, so “V” thru “V” will load all codes beginning with a “V”. Leaving the values blank will load all diagnosis codes in the database.

### Understanding Diagnosis Type

Besides the normal diagnosis code, MicroMD allows for a 1 character “Diagnosis Type” to be associated with the diagnosis. This allows for further refining a diagnosis code within the MicroMD system. External code load databases do NOT use the concept of a diagnosis type. Since the default diagnosis type in MicroMD is ‘0’, it is recommended that you leave the diagnosis type for AnyCode set to ‘0’. In order for a diagnosis code in the code load database to match a diagnosis code in MicroMD (for update), both the diagnosis code and the AnyCode-specified diagnosis type must match. Therefore, if diagnosis code 123.45 is defined in MicroMD with a diagnosis type of ‘1’ and a diagnosis type of ‘2’, and AnyCode has the diagnosis type set to ‘0’, then AnyCode will return with an operation of “Add 123.45/0”.

## Actions on Diagnosis Codes

The “Actions” section of the Diagnosis tab contains four different checkboxes. Each corresponds to an action that will be performed when running the AnyCode load operation.

### Adding New Diagnosis Codes

Diagnosis codes from the code load database are matched to existing diagnosis codes in MicroMD by the diagnosis code and by the diagnosis type. Therefore, if MicroMD contains diagnosis code 123.45 with a diagnosis type of ‘1’, and you have selected a diagnosis type of ‘0’ in AnyCode, then 123.45 with a diagnosis type of ‘0’ will be created. Since for almost every practice each diagnosis code exists with at least a type of ‘0’, it is generally advisable to leave the diagnosis type at ‘0’.

Within the range specified in the code range area, any diagnosis code / diagnosis type combination in the code load database that does not exist in the MicroMD database will be marked for adding to MicroMD.

When a new diagnosis code is added, the fields are determined as follows.

Field	Value
Diagnosis Code	The new code from the code load database
Diagnosis Type	As defined on the Diagnosis tab
Description	The description from the Code Load database
Diagnosis Class	Defined in the “Defaults” section of the Diagnosis tab
PDA Download	Defined in the “Defaults” section of the Diagnosis tab
PDA Common	Defined in the “Defaults” section of the Diagnosis tab
Active From / Active To	If the code load database has a date range defined, and if the “Update Dates” action is checked, then that range will be the active date range for the new diagnosis code. Otherwise, the active dates will be left blank in MicroMD.

## **Update Description**

If the “Update Description” action is checked, then when diagnosis code/type combinations are found in MicroMD that match codes in the code load database, the new description is compared to the value in the MicroMD database. If the values are different, then the new description will replace the old one in MicroMD. It is very common for practices to modify diagnosis descriptions to suit their particular purpose, and therefore, this option should usually be turned off, since otherwise, custom description information could be overwritten.

## **Update Dates**

If the “Update Dates” action is checked, then when diagnosis code/type combinations are found in MicroMD that match codes in the code load database, if the existing code has an end date earlier than the code load database’s end date, the end date will be updated. If there is no end date associated with the code in MicroMD, no end date will be updated.

## **Mark Invalid**

If there are diagnosis codes in the specified range in MicroMD that do NOT appear in the code load database, and if this value is checked, the codes will be marked invalid. Marking a code invalid means that the end date associated with the code will be updated to be the day before the code load database’s start date. For example, using the Medicare.mdb code load database, codes marked invalid would be given an end date of 12/31 of the current year.

MicroMD diagnosis codes will only be marked invalid if all of the following are true:

- They fall within the range of codes specified
- The diagnosis code does not appear in the code load database.
- The diagnosis code in MicroMD has no end date or has an end date past the start date of the code load database. Basically, if the code is already invalid, it does not need to be re-invalidated.