ODM Data Loader

**An application for loading data into the CUAHSI
Hydrologic Information System Observations Data Model**

March 2012

Prepared by:

**Jeffery S. Horsburgh
Utah Water Research Laboratory
Utah State University**

Distribution

The ODM Data Loader application and all associated source code and documentation are available at the following URL: http://his.cuahsi.org/.

The ODM Data Loader software, source code, and documentation are provide free of charge under the Berkeley Software Distribution (BSD) license. Please see the following license information:

Disclaimers

Although much effort has been expended in the development and testing of the ODM Data Loader application, errors and inadequacies may still occur. Users must make the final evaluation as to the usefulness of the ODM Data Loader for their application.

Acknowledgements

Technical Support

There is no formal ongoing support for this freely distributed open source software. However, we are interested in feedback. If you find errors, have suggestions, or are interested in any later versions, please contact:

Jeffery S. Horsburgh
Utah State University
8200 Old Main Hill
Logan, UT 84322-8200
jeff.horsburgh@usu.edu

Table of Contents

1.0 INTRODUCTION AND SOFTWARE DESCRIPTION

The CUAHSI Hydrologic Information System (HIS) Project has developed information technology infrastructure to support hydrologic science. One of the components of the HIS is a point Observations Data Model (ODM) (Horsburgh et al., 2008; Tarboton et al., 2008), which is a relational database schema that was designed for storing time series data. The purpose of the ODM is to provide a framework for optimizing data storage and retrieval for integrated analysis of information collected by multiple investigators. The CUAHSI HIS ODM has been implemented by a number of data publishers and researchers throughout the country, and these researchers are using ODM as a mechanism for publication of individual investigator data and for registering these data with the National HIS.

The ODM Data Loader (ODMDL) application was created to allow administrators of local instances of the ODM to load data into an ODM database. The development of the ODMDL application has several advantages. First, ODMDL protects the security and consistency of an ODM database because it provides users with a set of tools for validating and loading their data into ODM. This minimizes the potential for human caused errors in loading these data into an ODM database. The ODMDL input file formats are similar to the table structures in ODM, but they do provide users with some flexibility in specifying the required metadata. Users do not need to perform any specialized programming to parse and load the data, and ODMDL ensures that the data are fully qualified with valid metadata when they are loaded.

1.1 GENERAL FUNCTIONALITY

The main objective of ODMDL application is to provide data managers with a set of tools for loading data into an ODM database. ODMDL is implemented as a single executable file. However, it can be run using its simple graphical user interface or as a command line executable with arguments that specify the database connection and the files to be loaded. ODMDL is a file based data loader. It is capable of loading all of the individual tables within ODM separately from separate input files (i.e., one file per table), or it can do bulk data loading of all of the ODM tables from a single input file. In general, a single execution of ODMDL loads a single file. ODMDL can be executed using a command line batch file that repeatedly runs the executable with command line arguments so that complex data loading tasks using multiple tables can be automated and repeated. Additionally, ODMDL logs all successful data loads and errors in a text log file. To protect the integrity of data within an ODM database, the ODMDL treats each input file as a single data loading transaction. Because of this, the ODMDL will not load any data unless it can successfully load all of the data contained within the file.

1.2 SUPPORTED DATA FILE FORMATS

ODMDL was designed to load data from individual files. It currently supports comma delimited text files (.csv) and Microsoft Excel 2003 files (.xls) that have a one row header that uses ODM field names in the header, followed by the data in subsequent rows. When loading data from Excel, the data should be located in a worksheet that has the same name as the file. Each file to be loaded must conform to the input file templates provided in Appendix A of this document. These input templates generally conform to the table structure of ODM, with some flexibility for specifying alternative information for database generated IDs. For example, when loading data values, the SiteCode and VariableCode can be used rather than the SiteID and VariableID.

1.2 PLATFORM AND MINIMUM SYSTEM REQUIREMENTS

ODMDL was designed to run on Microsoft Windows XP, Vista, 7, 2003 Server, or 2008 Server based computers. It is recommended that machines running the ODMDL software have at least 100 MB of free disk space and at least 1 gigabyte of RAM. In addition, computers running ODMDL application must have the Microsoft .Net Framework Version 3.5 installed prior to installing the ODMDL. Instructions for obtaining the .Net Framework Version 3.5 from Microsoft are included in the Installation Instructions section below.

ODMDL is a client application. It must be connected to an instance of the CUAHSI HIS ODM that has been implemented in Microsoft SQL Server 2005 or 2008. The SQL Server database can be located on the same machine as the ODMDL application, or ODMDL has the capability to connect to a remote ODM database provided that the database server name and ODM database name are known and the user has been given access to, and SQL Server authentication information for, that server and database.

1.3 ODM COMPATIBILITY

Starting with Version 1.1.6, the ODMDL is compatible with both ODM Version 1.1 and ODM Version 1.1.1. Please consult the ODM design specifications documents for the differences in ODM Versions. ODM and the ODM Design specifications documents can be obtained from the following URL: http://his.cuahsi.org.

2.0 INSTALLATION INFORMATION

2.1 INSTALLATION PREREQUISITES

Prior to running the ODMDL installation, you must first install the Microsoft .Net Framework Version 3.5 (if it is not installed already). If you have Microsoft SQL Server 2008 installed, Version 3.5 of the .Net framework will be installed already. The .Net Framework Version 3.5 is free, and is required to run software applications developed in Microsoft's Visual Studio .Net 2008. Instructions for downloading and installing the .Net Framework Version 3.5 can be obtained from the Microsoft website via the following URL:

http://www.microsoft.com/downloads/details.aspx?FamilyId=AB99342F-5D1A-413D-8319-81DA479AB0D7&displaylang=en

Once the .Net Framework Version 3.5 has been installed, you can continue with the ODMDL installation.

> **NOTE:** ODMDL requires that you have an ODM database implemented in Microsoft SQL Server 2005 or 2008. If you do not already have an instance of Microsoft SQL Server running, you can download and install Microsoft SQL Server 2008 Express from Microsoft for free. It is recommended that you download and install either the "Runtime with Management Tools" or the "Runtime with Advanced Services" version of SQL Server 2008 express. You can get these products and instructions for installing them at the following Microsoft URL:
>
> http://www.microsoft.com/express/sql/download/

2.2 INSTALLING THE ODM DATA LOADER APPLICATION

Install the ODMDL using the following steps:

1. First, ensure that you have installed the Microsoft .Net Framework Version 3.5. See the previous section if you have not done so.
2. Double click on the ".msi" installation file. This will begin the installation of the ODMDL application. After a few moments, a window similar to the following will appear:
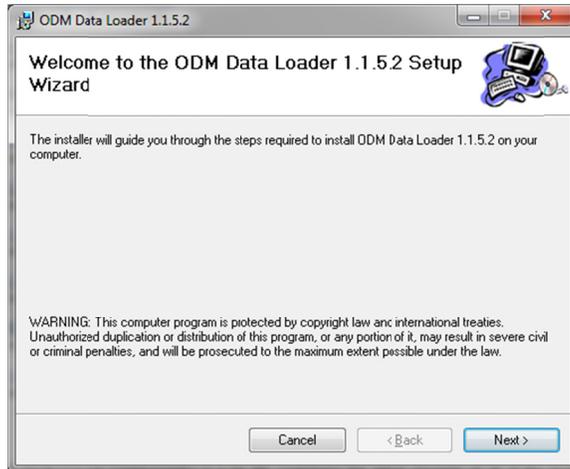
3. Click the "Next" button to continue with the ODMDL installation. You will see the following window.

4. Read the license and then click on the radio button next to "I Agree" to accept the license. Click the "Next" button. The following window will appear.
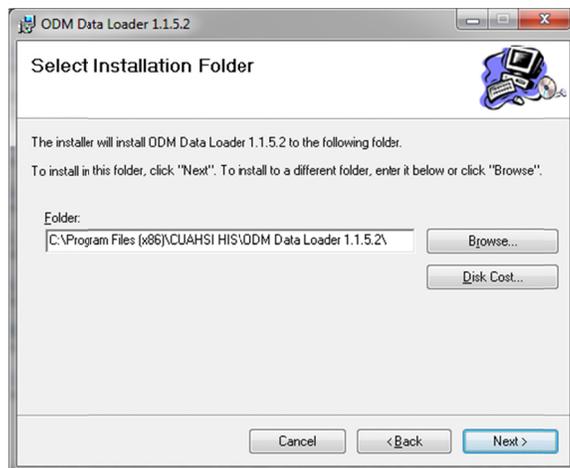
5. It is recommended that you leave the destination folder as is. Click the "Next" button to continue. The following window will appear.
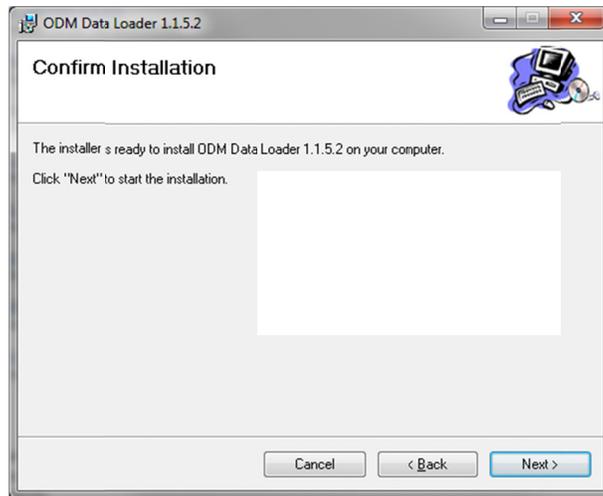
6. Click the "Next" button to continue. A window will pop up stating that the ODMDL is being installed. If you get a security warning, click "Yes" to allow the installation to continue.
7. When the installation is complete, you will see the following window. Click the "Close" button to complete the ODMDL installation. You will find a shortcut to ODMDL on your desktop and under the CUAHSI HIS group in the Windows Start menu.
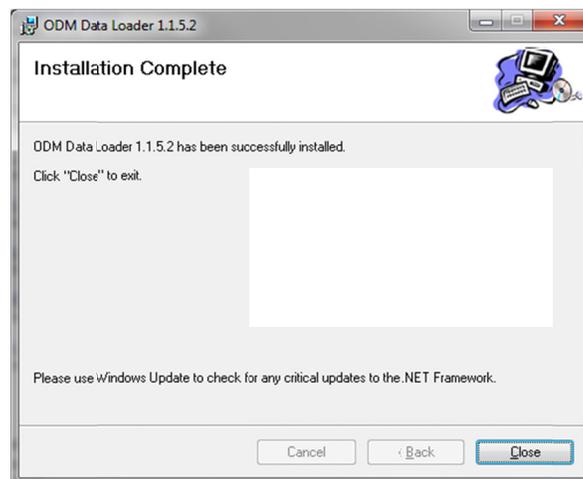
3.0 LOADING DATA USING THE ODM DATA LOADER

Loading data using the ODMDL requires some knowledge of the structure of ODM. The purpose of this section is to provide the necessary background information that is needed to understand how the ODMDL operates. Please refer to the ODM design specifications document (Tarboton et al., 2008), which is available on the CUAHSI HIS website http://his.cuahsi.org, for more a more detailed description of ODM and its features.

3.1 ODM BACKGROUND

ODM is a relational data model. As such, there are relationships between most of the tables in ODM that must be maintained in order to protect the integrity of the data within the database. For example, there is a relationship between the Sites table and the DataValues table, with a primary key called SiteID in the Sites table that points to a foreign key called SiteID in the DataValues table. This linkage defines the relationship between observation values in the DataValues table and the monitoring site at which they were measured in the Sites table. Similar relationships exist for Variables, Methods, Samples, QualityControlLevels, Sources, etc. Because these relationships exist in ODM, DataValues cannot be loaded into the database until the records that define the site at which they were measured, the variable that they represent, the method used to make the measurements, etc., have been added to the associated tables within ODM.

Most of the tables within ODM have a primary key field that contains a unique integer identifier for each record in the table. For example, the Sites table has a primary key called SiteID, the Variables table has a primary key called VariableID, etc. Beginning with ODM Version 1.1, these primary key fields were converted to identity/autonumber fields to take advantage of the capability of commercial relational database management systems to protect the integrity of data in the database through automatic assignment of the primary key values. This means that the database manages the assignment of the primary key ID values rather than the user. ***In other words, you cannot import data into these primary key fields, the database creates the IDs automatically.***

Beginning with ODM Version 1.1, relationship also exist between the controlled vocabulary tables and the fields that they constrain. Because of this, terms cannot be loaded into fields that have a corresponding controlled vocabulary table unless those terms already exist in the controlled vocabulary tables. If appropriate terms do not exist in the controlled vocabulary tables for the data that you would like to load, please use the online forms that can be accessed through the CUAHSI HIS website (http://his.cuahsi.org) to request additions to the central ODM controlled vocabularies. Once new terms are added to the central controlled vocabulary system, they can be added to your local ODM database using the ODM Tools application.

3.2 ORDER OF OPERATIONS

The primary key values (including controlled vocabulary terms) must be established before they can be used as foreign keys, so the order in which the data and its associated metadata are loaded matters. For example, a site must be loaded into the Sites table before data values measured at that site can be loaded into the DataValues table. When a site is loaded into the Sites table, the database automatically assigns it an integer SiteID. That SiteID is then used as the foreign key in the DataValues table to identify all of the observations collected at that site. Generally speaking, when data are loaded into ODM all of the metadata (i.e., sites, variables, methods, etc.) should be loaded *first*, followed by data values. With reference to the ODM schema diagram that can be found in the ODM Design Specifications document, you should start at the outside and work toward the DataValues table,

which is in the center of the diagram. In general, the following order of operations can be followed in loading data into ODM to avoid potential primary/foreign key constraint violations:

1. Load Sites
2. Load Variables
3. Load OffsetTypes
4. Load ISOMetadata
5. Load Sources
6. Load Methods
7. Load LabMethods
8. Load Samples
9. Load Qualifiers
10. Load QualityControlLevels
11. Load DataValues
12. Load GroupDescriptions
13. Load Groups
14. Load DerivedFrom
15. Load Categories

3.3 METHODS FOR LOADING DATA

In general, there are two methods for loading data using the ODMDL: 1) loading individual tables using separate input files for each ODM table, and 2) loading multiple or all of the ODM tables from a single input file. When using method 1, the ODMDL is executed multiple times (once for each file to be loaded) using the simple table-based GUI, the command line interface, or the simple wizard. The above order should be followed in loading the data using method 1. When using method 2, the ODMDL will manage the order of operations to make sure that everything gets loaded correctly. However, it should be noted that the ODMDL is capable of loading the DataValues table with nearly any combination of or subset of the above tables (e.g., Sites and DataValues only, Variables and DataValues only, etc.). If you choose to use this advanced functionality, care must be taken to avoid potential errors. For example, if you wish to load Sites and DataValues from the same file, but that file does not include all of the Variable information, you must ensure that all of the Variable information has already been loaded into the database and that correct Variable references are set in the file you are trying to load (i.e., it contains a VariableID or VariableCode column to identify the variables associated with the data).

3.4 CREATING INPUT FILES

Appendix A of this document contains the templates for the input files that are accepted by the ODMDL. The ODMDL provides some flexibility in how you can specify the contents of the input data files to be loaded. For example, after you have loaded sites, you can use either the SiteID OR the SiteCode to identify the appropriate site for data values to be loaded. The input file templates specify the format of the input files and the alternatives that can be used in creating them. Input files must follow these templates or errors will occur. In general, files should only include the fields as specified in Appendix A and should not contain extra fields (i.e., do not include a DataValue column in your file if you only want to load the Sites table).

> **NOTE:** Bulk data loading (i.e., loading multiple tables from a single file) can only be done when you are loading DataValues. Use the DataValues table template in Appendix A for bulk data loading tasks.

4.0   USING THE SIMPLE GRAPHICAL USER INTERFACE

Use the simple, table-based GUI when you wish to perform one-time data loads from a small number of files and when you wish to preview the contents of each file on the screen before they are loaded.

NOTE: The steps in this and the following sections assume that you already have an ODM database set up and running within an instance of Microsoft SQL Server 2005 or 2008. If you do not have SQL Server 2005 or 2008 or a working ODM database, please consult the information in the Installation Prerequisites section above and in the appendices of this document for information on how to remedy this situation.

The first step in loading data is to create input files that match the templates listed in Appendix A and save them as comma delimited text files, or Microsoft Excel files. Once you have done this, use the following steps to load data into your ODM database using the Simple Graphical User Interface.

1. Open the ODMDL Simple GUI by clicking on the shortcut on your desktop (ODM DL 1.1.X) or by selecting "ODM DL 1.1.X" from the Windows Start Menu (Start – All Programs – CUAHSI HIS – ODM DL 1.1.X). The following window will appear.
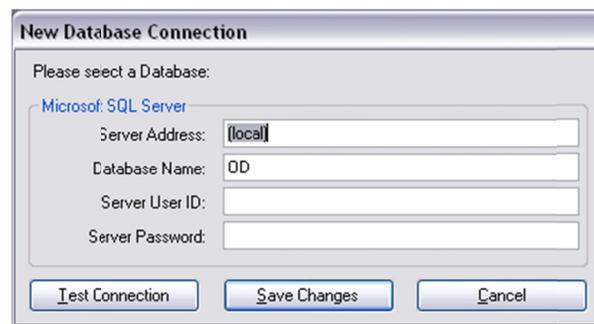
This is the form on which you will create a connection to the ODM database into which you want to load data. Enter an appropriate database name and SQL Server User Name and password and then click on "Save Changes."

NOTE: The ODMDL only supports SQL Server Authentication. It does not support Windows Authentication in SQL Server. You must have a valid SQL Server account with read and write access to the database that you would like to load data into.

If your connection is successful, a small window will pop up notifying you. Please click "OK" to continue. The following window will appear. This is the simple, table-based GUI of the ODMDL.
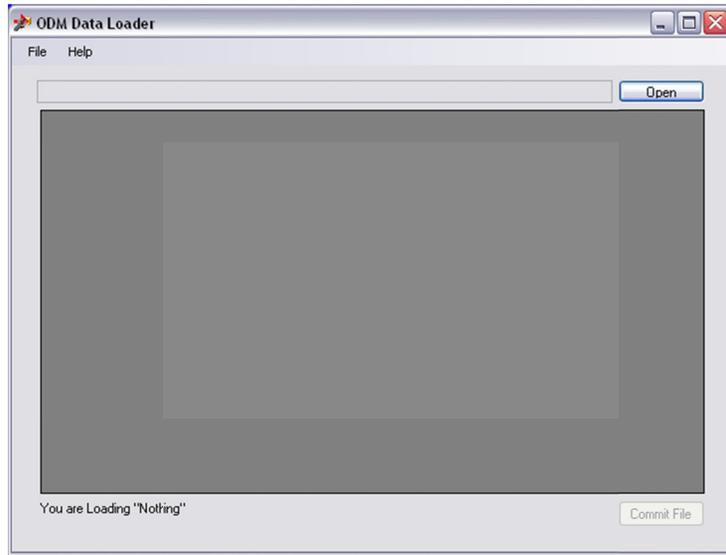
2. To load a data file, click on the "Open" button and navigate to the location of the file that you want to load. In this example, we will load a Sites file that contains records for several monitoring sites in the Little Bear River.
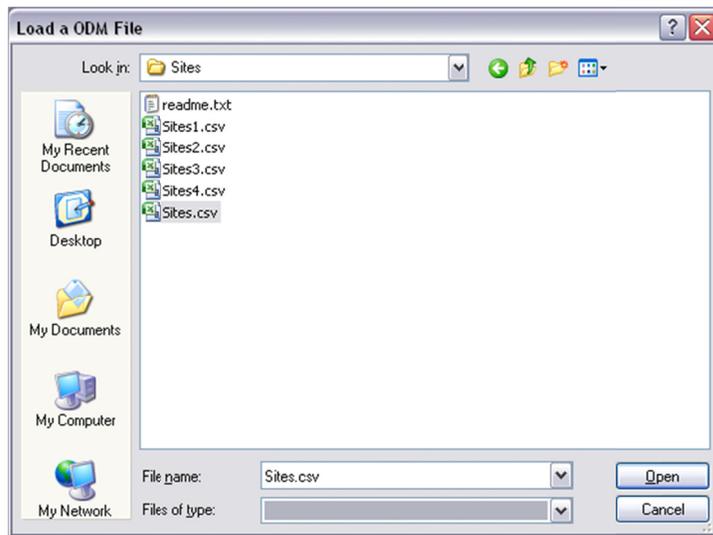
Once you find the file to be loaded, click on the "Open" button. The ODMDL will read the file and display it in the table on the main form. See below.
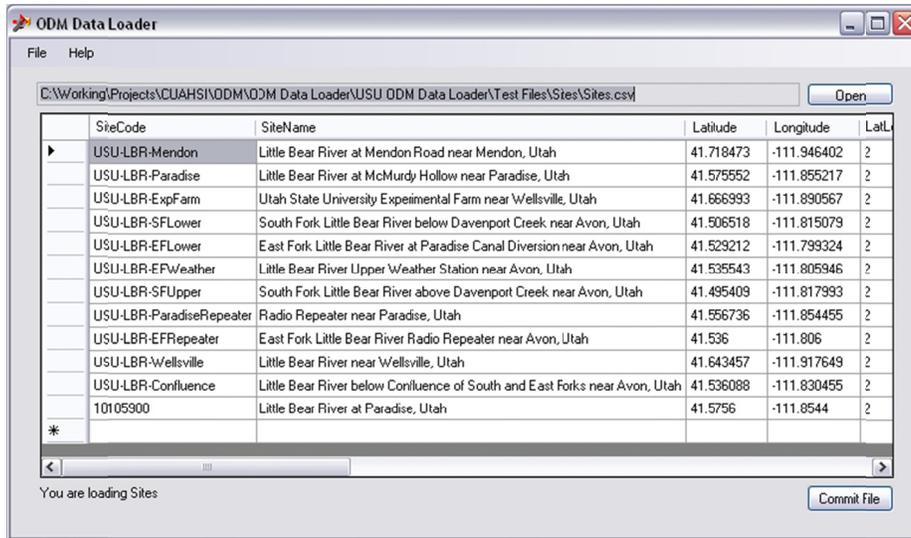
You will notice that at the bottom of the form the ODMDL has recognized that we are loading a Sites file. The ODMDL figures out what type of file you are loading from the header at the top of the file, so you do not have to tell it beforehand what you are trying to do. However, this does mean that you need to follow the input file templates given in Appendix A.

3. The simple, table-based GUI of the data loader provides you with a preview of the data before it gets committed to the database. You can scroll through the data to make sure that it has been read correctly. To commit the data in your file to the database, click the "Commit File" button. If the data loader was successful in loading the data into the database, the a window similar to the following will appear.
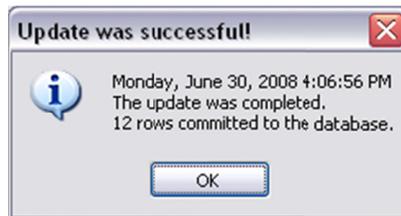
NOTE: If the load was unsuccessful, you will be provided with an error message that details why the data were rejected. Remember that the ODMDL will not load any data into the database unless it can successfully load the entire file. If there are any errors found in the input file, the entire file will be rejected. This protects the integrity of the data in the database and avoids the potential problems caused by partial data loads.

4. Click "OK" to acknowledge the successful load. You will notice that the ODMDL has returned to its beginning state. It is now ready to load another file.
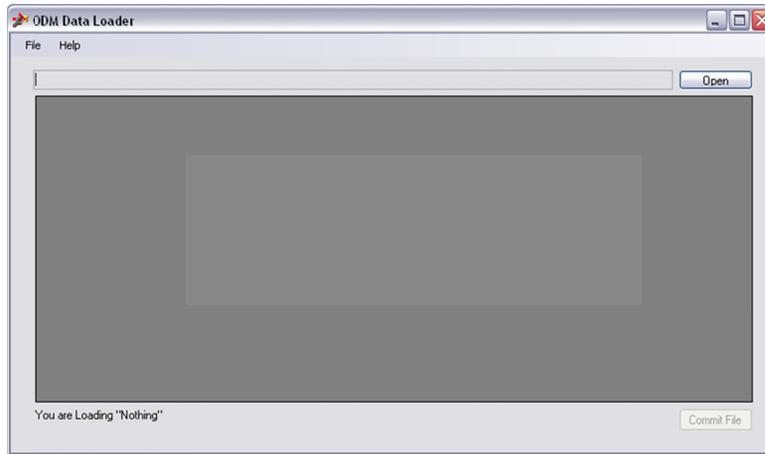
NOTE: As long as you do not close the ODMDL application, your connection to the database will remain open. You can load as many files as you want. If you are done, or you wish to load data into a different database, just close the ODM Data loader. When you open it again you will be prompted to supply database connection information.

---

5. Each time the ODMDL is executed, it writes entries to a log file. You can view this log file by navigating to its location on disk, which is in the application data folder on your machine (e.g., C:\Users\jeff\AppData\Local\CUAHSI, where "jeff" will be your Windows user name – within this folder you will find a folder that looks something like "ODMLoader.exe_Url_j2gfoitf3mydjrzqahzgc41iogwzybid" and your log file will be in that folder). You can open the log file in Notepad or any other text editor. The following is a screen shot of the log file entry that was created by the above example.
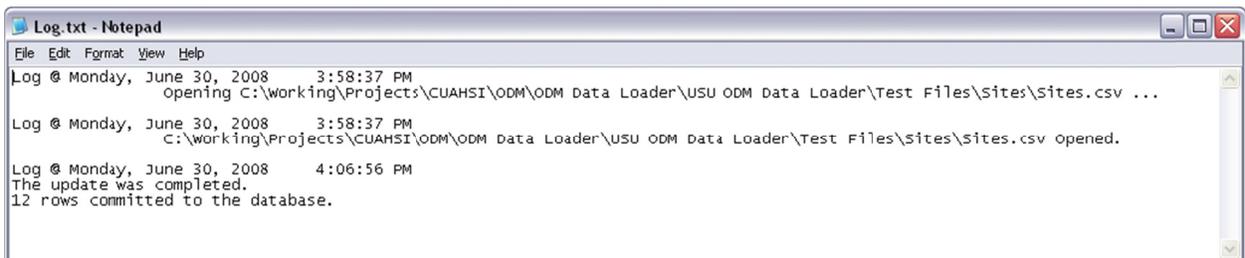
5.0 USING THE COMMAND LINE INTERFACE

Use the command line interface when you wish to set up a data loading task that includes many different input files, or where no user intervention is required. A batch file can be used to automate large tasks using the command line interface, and because of this, the tasks will be repeatable with no user interaction.

In order to enable the automation of data loading tasks, the ODMDL executable also supports a command line interface. This allows the ODMDL to be run from the command line, from within a batch file, or from other applications that are capable of calling an external executable program using command line arguments. The ODMDL command line interface has the following arguments:

-server	This is the name of the server on which the ODM database resides
-db	This is the name of the database into which you would like to load data
-user	This is the SQL Server username that you would like to use to connect to the database
-password	This is the password for the SQL Server account that you would like to use to connect to the database
-file	This is the full or relative path to the file that contains the data you are trying to load
-log	This is the full or relative path to the log file that will be written. If it is omitted, it will be written to the same path as the executable file.

The following is an example of a fully specified command line execution of the ODMDL:

```
odmloader.exe –server (local) –db OD –user sa –password MySAPassword –file
"sites.csv" –log "log.txt"
```

In this example, data is being loaded from a file called "sites.csv" to a database called "OD" on the local instance of SQL Server using the "sa" account. A log file called "log.txt" will be written to the same folder when the executable is located. When the ODMDL is installed, an Environment Variable is set within Windows so that it can be executed from any location (i.e., you do not have to specify the full path the executable or navigate to the folder where it is located before running it). The log file argument is optional, and if it is left out all log messages will be written to the standard log file in the ODMDL executable directory in the Program Files folder.

NOTE: In the example above, the path to the file to be loaded and to the log file is relative to the executable or to the batch file that is being executed. The full paths to these files can be used to ensure that there are no errors and no uncertainty in where the files will be written. The file paths should be enclosed in quotation marks to ensure that they are read correctly.

5.1 EXECUTING THE ODM DATA LOADER FROM THE COMMAND PROMPT

Use the following steps to load a single file into an ODM database from the command prompt using the ODMDL command line interface.

1. Open a command prompt by clicking on Start and then Run (Windows XP) or by typing "run" in the Windows 7 search box. The following Window will appear.
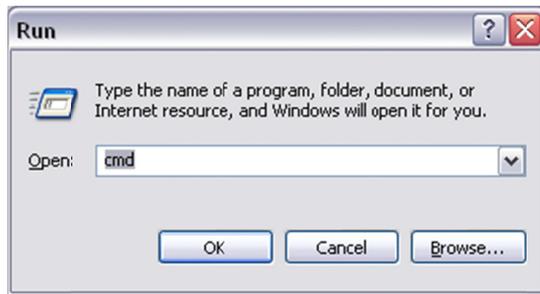
2. In the Run window that pops up type "cmd" in the text box and then click "OK." The following window will appear.

3. At the command prompt enter an execution string that is appropriate for your ODM database and for the file that you are loading. In this example we are loading a file called "variables.csv" to a database called "OD" that is attached to the local instance of SQL Server. The "ODMLoader" user account is used to connect to the database. Your command should look something like the following.
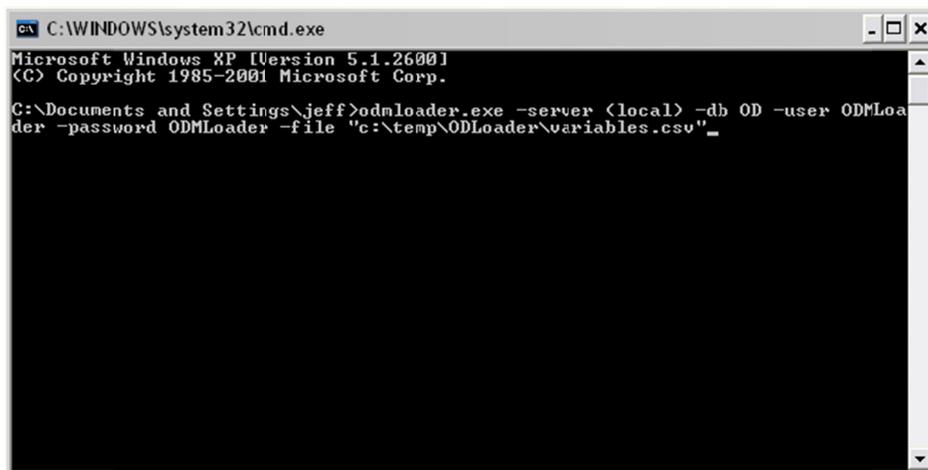
4. Once you have typed in this command hit the "Enter" key to execute the ODMDL. Your command prompt screen will display the log messages that are created when you execute the ODMDL. If the data load was successful, your screen will look something like the following. If the data load was unsuccessful, any error messages will be printed to the screen and in the log file. See the steps above for viewing the log file.


```
C:\WINDOWS\system32\cmd.exe

Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\jeff>odmloader.exe -server (local) -db OD -user ODMLoa
der -password ODMLoader -file "c:\temp\ODLoader\variables.csv"

C:\Documents and Settings\jeff>12:17:44 AM

Log @ Tuesday, July 01, 2008    12:17:44 AM
                Opening c:\temp\ODLoader\variables.csv ...

Log @ Tuesday, July 01, 2008    12:17:44 AM
Opening File: 100%.
                c:\temp\ODLoader\variables.csv Opened.

Log @ Tuesday, July 01, 2008    12:17:44 AM
You are loading "Variables"

Log @ Tuesday, July 01, 2008    12:17:44 AM
Committed 41 rows to the database.
```

NOTE: The process of running the ODMDL from the command prompt is the same for any file that is to be loaded into an ODM database. If you wish to load multiple files individually using this process, simply repeat the above steps for each file.

5.2   EXECUTING THE ODM DATA LOADER USING A BATCH FILE

Each execution of the ODMDL loads a single input file into the database. In order to automate the loading of several files into the database at once, you can create a batch file that calls the ODMDL executable several times, once for each file to be loaded. Use the following steps to execute the ODMDL using a batch file.

1. Create a batch file that contains a fully specified ODMDL execute statement for each file that is to be loaded into the database. You can create a batch file by creating a new .txt file, editing its contents and then renaming it with a ".bat" file extension. You can save this batch file anywhere on disk, but it may be easiest to create a new folder and place the batch file and all of the files to be loaded into the same folder. The following is an example batch file that loads 14 different files (including DataValues) into an ODM database called "OD" that is attached to the local instance of SQL Server. The "ODMLoader" SQL Server account is used. Notice that the order of operations described above is used in the creation of this batch file to ensure that tables are loaded into the ODM database in the correct order.


```
wholedatabase.bat - Notepad

File  Edit  Format  View  Help

odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\sites.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\variables.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\offsettypes.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\isometadata.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\sources.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\methods.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\labmethods.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\samples.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\qualifiers.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\qualitycontrollevels.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\datavalues65.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\groupdescriptions.csv"
odmloader.exe -server (local) -db OD -user ODMLoader -password ODMLoader -file "c:\temp\ODLoader\groups.csv"
```

2. Once your batch file has been created, it can be executed in a couple of different ways. First, you can simply navigate to the location on disk where you have saved your batch file and double click on it. This will open a command window and run the batch file, executing the ODMDL however many times you have listed it in the batch file. If you do this, a window similar to the following will pop up. When the batch file is finished, this window will close automatically. All of the progress and output messages that you see in this window are also written to the log file so that you can review and error messages.
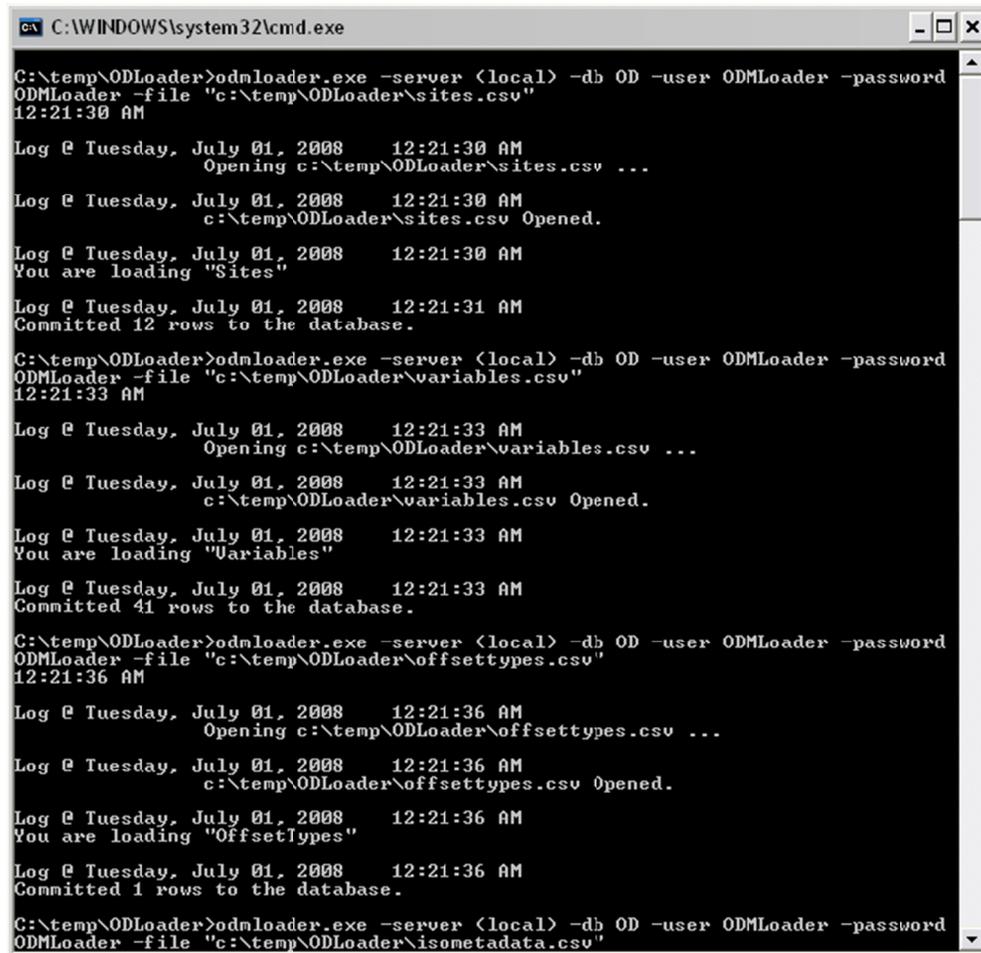
```
C:\WINDOWS\system32\cmd.exe                                    _ □ ✕

C:\temp\ODLoader>odmloader.exe -server (local) -db OD -user ODMLoader -password
ODMLoader -file "c:\temp\ODLoader\sites.csv"
12:21:30 AM

Log @ Tuesday, July 01, 2008    12:21:30 AM
                Opening c:\temp\ODLoader\sites.csv ...

Log @ Tuesday, July 01, 2008    12:21:30 AM
                c:\temp\ODLoader\sites.csv Opened.

Log @ Tuesday, July 01, 2008    12:21:30 AM
You are loading "Sites"

Log @ Tuesday, July 01, 2008    12:21:31 AM
Committed 12 rows to the database.

C:\temp\ODLoader>odmloader.exe -server (local) -db OD -user ODMLoader -password
ODMLoader -file "c:\temp\ODLoader\variables.csv"
12:21:33 AM

Log @ Tuesday, July 01, 2008    12:21:33 AM
                Opening c:\temp\ODLoader\variables.csv ...

Log @ Tuesday, July 01, 2008    12:21:33 AM
                c:\temp\ODLoader\variables.csv Opened.

Log @ Tuesday, July 01, 2008    12:21:33 AM
You are loading "Variables"

Log @ Tuesday, July 01, 2008    12:21:33 AM
Committed 41 rows to the database.

C:\temp\ODLoader>odmloader.exe -server (local) -db OD -user ODMLoader -password
ODMLoader -file "c:\temp\ODLoader\offsettypes.csv"
12:21:36 AM

Log @ Tuesday, July 01, 2008    12:21:36 AM
                Opening c:\temp\ODLoader\offsettypes.csv ...

Log @ Tuesday, July 01, 2008    12:21:36 AM
                c:\temp\ODLoader\offsettypes.csv Opened.

Log @ Tuesday, July 01, 2008    12:21:36 AM
You are loading "OffsetTypes"

Log @ Tuesday, July 01, 2008    12:21:36 AM
Committed 1 rows to the database.

C:\temp\ODLoader>odmloader.exe -server (local) -db OD -user ODMLoader -password
ODMLoader -file "c:\temp\ODLoader\isometadata.csv"
```

3. You can also open a command prompt as described above and use DOS commands to navigate to the folder in which the batch file is located. Once there, simply type the name of the batch file at the command prompt, hit "Enter", and it will be executed. Again, the progress and any error messages are output to the command window and to the log file.

6.0   USING THE SIMPLE WIZARD

Use the simple wizard only when you want to load DataValues and when you want to be guided through the data loading process. Do not use the wizard when you only want to load data to one or two tables, or when you are only loading data to tables other than the DataValues table.

The simple wizard was created to guide you through the data loading process and to ensure that data are loaded into the database in the correct order. It essentially creates a batch file that contains all of the files and options that you select. The batch file is then executed to load the data in the files that you select. You can load one or many files using the wizard.

1. Start the ODMDL Wizard by clicking on the shortcut on your desktop (ODM DL 1.1.X Wizard) or by clicking on Start → All Programs → CUAHSI HIS → ODM DL 1.1.X Wizard. The following window will appear.

On this window you will select a location for the batch file to be created, the log file to be created, and you will specify the information needed to create a connection to the ODM database into which you are loading data. Click the "Browse" buttons to specify the locations where the batch file and the log file will be saved and then fill in the database connection information. Refer to the steps above for information about creating a connection to an ODM database.

NOTE: It may be a good idea to assemble all of the files that you will be loading into a single folder on your hard drive. This will allow you to create the batch file and log file within that folder. This may make it easier to repeat the data loading task if the need arises.

2. Once you have filled out all of the items on the opening form, click the "Next" button to continue. The following form will appear. On this form you will select a file that contains the DataValues that you would like to load. Click the "Browse" button to select a file from disk. The file that you select must conform to

the DataValues template in Appendix A, but you may load any subset of the ODM tables from the DataValues file that you select. See the Note below.
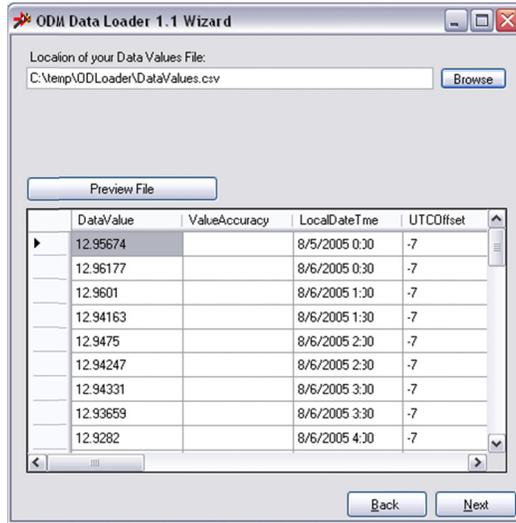
NOTE: Since it is possible to load data to all of the ODM tables from a single DataValues input file, the Wizard begins with the selection of a DataValues file. If you are loading all of your data from a single file, simply select the correct radio button on the following screens indicating that you are not loading Sites, Variables, Methods, etc. from separate files.

3. Once you have selected a DataValues file, you will notice that the file is read and loaded into the preview table on the form. If you type a file path into the text box rather than using the "Browse" button, click on the "Preview File" button to load the file into the preview pane. When you click the "Next" button, the following window will appear.
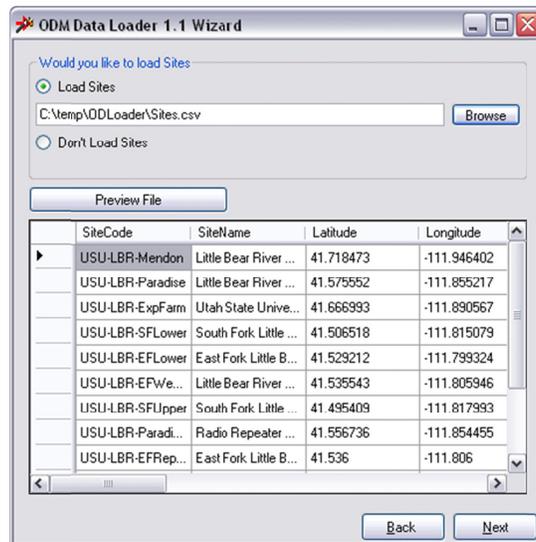
If the DataValues table that you selected in the first screen does not contain enough information to load the Sites table AND your Sites do not already exist in the database, you should click the "Browse" button to select a Sites file to load. If you are loading Sites from the DataValues file that you selected on the previous screen, OR your Sites already exist in your database, you can select the "Don't Load Sites" radio button. Once you have made your selection, click the "Next" button to continue.

4. The next several screens are similar to the Sites screen shown in the previous step. The Wizard takes you through the tables in the order in which they will be loaded and allows you to select an appropriate file from disk or to specify that data for that particular table will not be loaded from a separate file. Navigate through each of these screens and make appropriate selections for your dataset. These screens include: Variables, OffsetTypes, ISOMetadata, Sources, Methods, LabMethods, Samples, Qualifiers, QualityControlLevels, GroupDescriptions, Groups, DerivedFrom, and Categories.

5. When you have navigated through all of the data table selection screens, the following window will appear. On this form, you can select whether to run the data loading task immediately or not. If you select "Yes," the Wizard will immediately run the data loading task by executing the batch file that is created. If you select "No," the Wizard will create the batch file and will then close, but you will have to execute the batch file manually using the steps describe above.
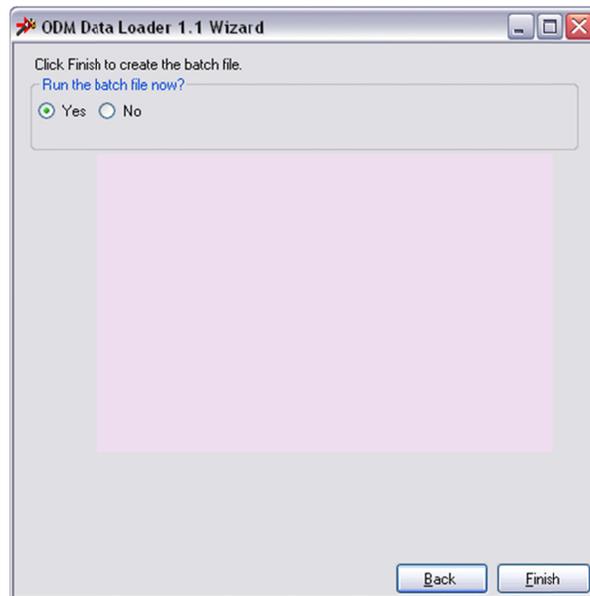
6. If you select "Yes," and then click the "Finish" button, a command prompt window similar to the following will pop up. You can monitor the progress of the batch file in this window while it is running. When the batch file is finished, the command prompt window will close automatically.

7. If you wish to review the batch file that is created by the Wizard, you can navigate to the location on disk where you decided to store it and open it in notepad (right click on the file and select "Edit" – if you double click on the file it will execute). The following is an example.

NOTE: You will notice that in batch files created by the Wizard your SQL Server password is encrypted. If you move a batch file to another computer, it will fail to execute because the other computer will not be able to

decrypt the password. If you wish to move batch files from one computer to another, you can replace the encrypted password strings with your original password.

8. You can view the log file for the batch file execution by navigating to the location on disk that you selected for your log file and opening it in a text editor like Notepad.

7.0 VIEWING THE ODM DATA LOADER LOG FILE

Each time a data loading task is run, information about that task is written to the ODMDL log file. This log file is a text file and is located at the same location as the ODMDL application executable (C:\Program Files\CUAHSI HIS\ODMDL). A summary of the data loading task is written to the log file as well as any errors that are encountered. The following is an excerpt from a log file.

REFERENCES

Horsburgh, J. S., D. G. Tarboton, D. Maidment, and I. Zaslavsky (2008), A Relational Model for Environmental and Water Resources Data, Water Resources Research, Vol. 44, W05406, doi:10.1029/2007WR006392.

Tarboton, D.G., Horsburgh, J.S., and D.R. Maidment. 2008. CUAHSI Community Observations Data Model (ODM) Design Specifications Document: Version 1.1. http://his.cuahsi.org/odmdatabases.html.

The general format for input to the ODMDL is a single file containing a table with a one row header that uses ODM field names in the header, followed by the data in subsequent rows. The templates are such that the input data table format (i.e., the included columns) should either be identical to its destination table within ODM, or in expanded flat file format providing ancillary data associated with each data value sufficient to either load ancillary data tables or identify appropriate existing records in metadata tables. The ODMDL identifies database fields from the input file header names, such that the order of columns in the input file does not matter. The ODMDL identifies the contents of the input file and subsequently what the user is trying to load by parsing its header. The rules for identifying files by header information are given for each table below. If an input file fails to meet one of the rules specified below, an invalid file error will be returned.

Tables that can be loaded into ODM using the ODMDL are listed below in alphabetical order. In the lists of field headers below, (R) indicates required and (O) indicates optional. Where field headers are listed in italics (for example see *SiteColumns* for the DataValues table) users have multiple options for specifying the content of the input file for those fields.

ODM Table: Categories

Identification Rule: ODMDL will identify a Categories file by the appearance of CategoryDescription in the file header list without a DataValue field
Field Headers:
VariableColumns (M) EITHER one and only one of VariableID or VariableCode that corresponds to an existing Variables record in the Variables table.
DataValue (R)
CategoryDescription (R)

ODM Table: DataValues

Identification Rule: ODMDL will identify a DataValues file by the appearance of DataValue in the field header list.
Field Headers:
DataValue (R)
ValueAccuracy (O)
LocalDateTime (R[1])
UTCOffset (R[1])
DateTimeUTC (R[1])
SiteColumns(M) EITHER one and only one of SiteID or SiteCode that corresponds to an existing Sites record in the Sites table, OR the required and optionally the optional columns from the Sites file below.
VariableColumns (M) EITHER one and only one of VariableID or VariableCode that corresponds to an existing Variables record in the Variables table, OR the fields listed for the Variables file below.
OffSetValue (O)
OffsetTypeColumns (O) EITHER OffsetTypeID that corresponds to an existing OffsetTypes record, OR the fields listed for the OffsetTypes file below.
CensorCode (R) – Must contain an existing term from the CensorCodeCV table.
QualifierColumns (O) EITHER QualifierID that corresponds to an existing Qualifiers record in the Qualifiers table, OR the fields listed for the Qualifiers file below.

MethodColumns (R) EITHER MethodID that corresponds to an existing Methods record in the Methods table, OR the fields listed for the Methods file below.

SourceColumns (R) EITHER SourceID that corresponds to an existing Sources record in the Sources table, OR the fields listed for the Sources file below.

SampleColumns (O) EITHER SampleID that corresponds to an existing Samples record in the Samples table, OR the fields listed for the Samples file below.

DerivedFromID (O)

QualityControlLevelColumns (R) EITHER QualityControlLevelID that corresponds to an existing record in the QualityControlLevels table, OR the fields listed for QualityControlLevels file below.

GroupDescription (O) If this matches an existing GroupDescription the corresponding GroupID and ValueID will be added to the Groups table. If this is new, a new GroupDescriptions record will be added and the corresponding IDs added to the Groups table.

---

[1]**Note:** Only two of LocalDateTime, UTCOffset and DateTimeUTC are required. The third will be calculated from the other two.

ODM Table: DerivedFrom

Identification Rule: ODMDL will identify a DerivedFrom file by the appearance of DerivedFromID and ValueID in the field header list without any other fields (requires that DataValues have already been populated)
Field Headers:
DerivedFromID (R)
ValueID (R)

ODM Table: Groups

Identification Rule: ODMDL will identify by a Groups file by the appearance of GroupID and ValueID in the field header list without any other fields (requires that GroupDescriptions and DataValues have already been populated)
Field Headers:
GroupID (R)
ValueID (R)

ODM Table: GroupDescriptions

Identification Rule: ODMDL will identify a GroupDescriptions file by the appearance of GroupDescription in the field header list without a DataValue field.
Field Headers:
GroupDescription (R)

ODM Table: ISOMetadata

Identification Rule: ODMDL will identify an ISOMetadata file by the appearance of TopicCategory in the field header list without DataValue and without Organization.
Field Headers:
TopicCategory (R) – This must contain a valid term from the TopicCategoryCV table.

Title (R)
Abstract (R)
ProfileVersion (R)
MetadataLink (O)

ODM Table: LabMethods

Identification Rule: ODMDL will identify a LabMethods file by the appearance of LabName in the field header list without DataValue and without SampleType.
Field Headers:
LabName (R)
LabOrganization (R)
LabMethodName (R)
LabMethodDescription (R)
LabmethodLink (O)

ODM Table: Methods

Identification Rule: ODMDL will identify a Methods file by the appearance of MethodDescription in the field header list without DataValue.
Field Headers:
MethodDescription (R)
MethodLink (O)

ODM Table: OffsetTypes

Identification Rule: ODMDL will identify an OffsetTypes file by the appearance of OffsetDescription in the field header list without DataValue.
Field Headers:
OffsetUnitsColumn (R) OffsetUnitsID (referring to UnitsID in the Units table) or OffsetUnitsName (referring to UnitsName in the Units Table). One and only one of these columns should be present matching to an existing record in the Units table.
OffsetDescription (R)

ODM Table: Qualifiers

Identification Rule: ODMDL will identify a Qualifiers file by the appearance of QualifierDescription in the field header list without DataValue.
Field Headers:
QualifierCode (O)
QualifierDescription (R)

ODM Table: QualityControlLevels

Identification Rule: ODMDL will identify a QualityControlLevels file by the appearance of QualityControlLevelCode in the field header list without DataValue.

Field Headers:
QualityControlLevelCode (R)
Definition (R)
Explanation (R)

ODM Table: Samples

Identification Rule: ODMDL will identify a Samples file by the appearance of SampleType in the field header list without DataValue.
Field Headers:
SampleType (R) – This must contain a valid Term from the SampleTypeCV table.
LabSampleCode (R)
LabMethodColumns (R) EITHER LabMethodID that corresponds to an existing record in the LabMethods table OR the columns listed for the LabMethods table below.

ODM Table: Sites

Identification Rule: ODMDL will identify a Sites file by the appearance of SiteName in the field header list without the appearance of DataValue.
Field Headers:
SiteCode (R)
SiteName (R)
Latitude (R)
Longitude(R)
LatLongDatumColumn (R) LatLongDatumId (referring to SpatialReferenceID in the SpatialReferences table) or LatLongDatumSRSID (referring to SRSID in the SpatialReferences table) or LatLongDatumSRSName (referring to SRSName in the SpatialReferences table). One and only one of these is required and should be used to identify the corresponding record in the SpatialReferences controlled vocabulary table upon loading.
Elevation_m (O)
VerticalDatum (O) – If used, this must contain a valid term from the VerticalDatumCV table.
LocalX (O)
LocalY(O)
LocalProjectionColumn (O) LocalProjectionID (referring to SpatialReferenceID in the SpatialReferences table) or LocalProjectionSRSID (referring to SRSID in the SpatialReferences table) or LocalProjectionSRSName (referring to SRSName in the SpatialReferences table) (O). One and only one of these is required if LocalX and LocalY are specified and should be used to identify the corresponding record in the SpatialReferences controlled vocabulary table upon loading.
PosAccuracy_m (O)
SiteState (O)
County (O)
Comments (O)
SiteType (O)

> **NOTE:** SiteState is different than the field name in ODM. The use of SiteState in the input file template is necessary to facilitate bulk data loading because there is also a State field in the Sources table. Additionally, you will notice there there are no LatLongDatumSRSID, LatLongDatumSRSName, LocalProjectionSRSID, or

LocalProjectionSRSName fields in ODM. These fields exist only in the template and reference the SRSID and SRSName fields in the SpatialReferences table to provide alternate inputs to the Sites table.

NOTE: SiteType should only be used when you are loading data into an ODM 1.1.1 database.

ODM Table: Sources

Identification Rule: ODMDL will identify a Sources file by the appearance of Organization in the field header list without DataValue.
Field Headers:
Organization (R)
SourceDescription (R)
SourceLink (O)
ContactName (R)
Phone (R)
Email (R)
Address (R)
City (R)
SourceState (R)
ZipCode (R)
MetadataColumns (R) EITHER MetadataID that corresponds to an existing ISOMetadata record OR the columns listed for the ISOMetadata table below.
Citation (R)

NOTE: SourceState is different than the field name in ODM. The use of SourceState in the input file template is necessary to facilitate bulk data loading because there is also a State field in the Sites table.

ODM Table: Variables

Identification Rule: ODMDL will identify a Variables file by the appearance of VariableName in the field header list without the appearance of DataValue.
Field Headers:
VariableCode (R)
VariableName (R) – This must contain a valid term from the VariableNameCV table.
Speciation (R) – This must contain a valid term from the SpeciationCV table.
VariableUnitsColumn (R) VariableUnitsID (referring to UnitsID in the Units table) or VariableUnitsName (referring to UnitsName in the Units Table). One and only one of these columns should be present matching to an existing record in the Units table.
SampleMedium (R) – This must contain a valid term from the SampleMediumCV table.
ValueType (R) – This must contain a valid term from the ValueTypeCV table.
IsRegular (R)
TimeSupport (R)

TimeUnitsColumn (R) TimeUnitsID (referring to UnitsID in the Units table) or TimeUnitsName (referring to UnitsName in the Units Table). One and only one of these columns should be present matching to an existing record in the Units table

DataType (R) – This must contain a valid term from the DataTypeCV table.

GeneralCategory (R) – This must contain a valid term from the GeneralCategoryCV table.

NoDataValue (R)