



Import Manager File Specification

Introduction

MPOG is a collaborative focused on collecting and standardizing anesthesia data for healthcare research and quality improvement. In order to participate with MPOG, contributing organizations must get their anesthesia documentation into MPOG's central repository. To this end, the Import Manager framework is provided to handle many of the complexities arising from merging data from disparate sources and inserting it into a database. This document contains all relevant information for generating files compatible with Import Manager.

While some overall concepts are covered, installation and running Import Manager is outside the scope of this document. For this information refer to the MPOG Playbook.

Key Concepts

The following sections briefly describe some high level concepts utilized by Import Manager and how they impact the files generated.

File Structure

All files that are consumed by Import Manager are plain text, comma-delimited files (CSV). This was chosen over other formats such as XML as CSV is more easily generated by most platforms. Each line in a file represents a row of data to be inserted into a table.

Import Method

How the files are generated or where they come from are details considered specific to each implementation. Some institutions have a single enterprise system from which all files are extracted, while others generate files from several different sources located on different servers.

However they are created, all files get into Import Manager the same way. First, they are placed into a directory. Then, an application polls this directory. If any new files are detected, the application copies the file name and contents into the Import Manager database. Once inserted successfully, the file is deleted from the directory. The Import Manager holds the files internally for further processing. Eventually, the data is transformed into the final format and copied into another database accessible by users for validation.

Modules

There are many different forms of anesthesia data that get extracted. Some examples include vital signs, labs, medications, fluids, intraoperative notes, billing codes, and in-hospital deaths. Given all of these data types are structured differently, Import Manager provides different file formats to handle these differences. The various file formats are called modules. The specification for each of these modules is listed at the end of this document, with descriptions for each column and example data provided.

Instances

Typically, organizations only have one instance of their anesthesia documentation system. However, larger organizations may have many hospitals and therefore may have several instances. Alternatively, an organization may have not only their current anesthesia system but also an archived legacy system containing historical data. To prevent collision between database identifiers coming from different instances, Import Manager partitions different anesthesia systems using an Instance ID.

The most obvious effect this has on the extract is that files coming from separate instances should be placed in separate directories (see Import Method). Import Manager stores a one-to-one relationship between each directory it polls and the instance assigned to it.

It is possible to create a “virtual” instance by filtering data and generating multiple sets of files into different directories. This can be useful if not all organizations within an instance wish to have their data extracted into Import Manager right away. However, be aware that merging instances is not supported, so it is difficult to undo these virtual instances.

Specification Version

It is possible that future iterations of Import Manager will support additional versions of a given module. In order to distinguish this, each file must specify which specification it uses. Currently there is only one specification for each module, therefore the specification version is always “V1”.

Source System

It is common for data in a given module to come from multiple systems at once. For example, point-of-care labs may be stored directly in the anesthesia system while formal labs may come from a separate lab interface. In order to prevent institutions from having to manually merge files prior to import, files can be distinguished by where they came from. This label is referred to as “source system.”

Target Date

Rather than patients or cases, Import Manager is designed to work at a granularity of a single day. This means that the framework processes and logs everything based upon date. The day for which data has been pulled is called the target date and at a minimum, each file should contain one day’s worth of data.

The definition of a target date varies by module and is included in its specification. For example, the definition for the Cases module is that for a given date, a Cases file for that date must include all cases that were started on that date. Most but not all modules refer to this definition.

When generating files, there are two approaches to use and they are differentiated by how they handle extracting large volumes of data.

Single Date Format

The most basic approach to generating files is to create a separate file for each days' worth of data. Extracting historical data using this approach requires looping through every date.

The module specifications assume the single date format.

Multi Date Format

The logistics of creating and handling many different files can be difficult and time-consuming. This is particularly true when receiving billing codes from a third party vendor, who are typically accustomed to creating monthly reports rather than daily ones. As such, Import Manager also supports combining many days' worth of data into a single file.

Multi date files are structured the same as single date files with two key exceptions:

1. The target date portion of the file name is replaced with a dummy date or a label.
2. A target date column is added to the beginning of each row. This target date must be formatted as MM/dd/yyyy, keeping any leading zeroes.

Warning!

While there is no hard upper limit on how much data can be placed into a multi date file, Import Manager's performance can degrade considerably when processing huge files. Furthermore, extracting large date ranges take longer to complete and are more prone to failure.

For these reasons, it is strongly recommended to extract no more than one month into a multi date file.

Pull Date

Pull date is the date on which the extract was executed and the file was created. Compare this to the target date, which is the date for which a given file or row contains data. For example, a file may contain data for a particular date a year ago but was created yesterday (target date and pull date respectively).

Pull date is important as it indicates which version of a file is the most recent and therefore which one should be used for the most up-to-date information.

Extract Schedule

Timely reporting is important to MPOG's quality improvement mission. As such, anesthesia data should be extracted shortly after the surgery occurs. However, the reality of healthcare documentation means that everything might not be finalized until a few days after the surgery. Some data elements such as outcomes or billing codes might not be available until weeks or even months after the case. Having the complete record is important for MPOG's research mission, so then the data should be extracted long after the surgery occurs.

The solution to these conflicting needs is simply to extract the data multiple times. Once testing is complete, an automated daily extract should be created to pull the following four dates:

- T-4: the day four days ago from today
- T-30: the day thirty days ago from today
- T-90: the day ninety days ago from today
- T-365: the day 365 days ago from today

Global Requirements

This section details technical requirements that apply to all modules.

Files Columns

Not all columns are required and sometimes an institution may not be able to populate a given column for any row. Even if a column is never filled in, it must still be represented within a file. Refer to “Representing an Empty Field” for more details.

Also, columns in the file must be in the same order as they are listed in the specification.

Target Date Completeness

Files cannot contain updates for only a few cases or patients at a time. Whenever data for a particular target date and module is extracted, all data for that date must be present in the file. Any data that was extracted previously but is not present in a new file will be interpreted as deleted and will be purged from Import Manager.

Filename Template

All files must be named in one of two ways, depending on whether the single date or multi date format is used. Both target date and pull date must be formatted as YYYYMMDD, with the date always being exactly 8 characters.

Single Date Format

Single date file uses the following template:

MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv

```
Patients_V1_Epic_20170106_20180623.csv
Procedures_V1_AcmeBillingCo_20170106_20180623.csv
```

Multi Date Format

Given multi date files contain the target date in each row, this file format uses a placeholder label instead. This label is not interpreted by the Import Manager and is not used for anything other than distinguishing it from other files. This label can be a readable name or a dummy date.

MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv

```
Patients_V1_Epic_T-4_20180623.csv
Patients_V1_Epic_20000101_20180623.csv
```

Warning!

When extracting a range of dates using the multi-date format, do not put the date range in the file name. Doing so will cause files to be processed incorrectly by the Import Manager.

Header Row

These files must not contain a header row with the column names. The first line in each file must be the first row of data.

Characters to Escape

Import manager uses two types of delimiters in the files, a comma for separating columns and a newline for separating rows. For this reason, these characters must not appear in any field. Instead, anytime one of these characters are used they should be replaced as listed below.

Character to replace	Replace with
Comma (,)	,
Carriage return (\r)	
Newline (\n)	


```
--Original note
Using a miller blade, the patient was intubated successfully.
Airway was not difficult.

--Escaped note
Using a miller blade&#44; the patient was intubated successfully.&#10;Airway was not
difficult.
```

Avoid Additional Delimiters

Aside from using a comma, no other delimiter should be used. Another common scheme for delimiting data is to use quotes around each field, but that should not be done inside these files.

Representing an Empty Field

Empty or null fields can be represented in files one of two ways. They are 1) using no character at all or 2) using the word NULL. It is not correct to include any whitespace in the column if attempting to store NULL. To illustrate, here is how empty fields would look in a hypothetical file with three columns:

```
--No character
A,B,C
,B,C
A,,C
A,B,

--Alternatively, using NULL
A,B,C
NULL,B,C
A,NULL,C
A,B,NULL

--Including whitespace in a blank column is NOT correct
A,B,C
 ,B,C
A, ,C
A,B,
```

For consistency, this document will use the first approach when presenting example data.

Variables

Throughout the module specification there will be pairs of columns used to store various database variables. These pairs consist of an ID column and a name column. The name is the human readable label of the variable and the ID is the database identifier used to represent it.

Some of these variable columns may not be represented with both an ID and name in the original system. A common example is patient gender, which is often stored as one column with the values “M” or “F”. When this happens, store the single value in both the ID and name columns.

Some variable column pairs are optional. In these cases, they both must be NULL or not NULL at the same time. In other words, one cannot be NULL while the other is not NULL.

Data Types

Unless specified otherwise, all columns are assumed to be stored as text with a 100-character limit.

Text(MAX)

There will be some instances where a column is allowed to store large amounts of text. There is an upper limit of around 2GB, but this should not be exceeded in practice.

Boolean

Boolean values can be represented several different ways, listed below. For consistency, the rest of this document will use 1 and 0 when presenting Boolean values.

```
TRUE / FALSE  
1 / 0  
YES / NO  
Y / N
```

If the Boolean column is also nullable, an “unknown” value can be encoded as an empty field.

DateTime

Dates and times are stored together in a single field. A DateTime can be stored in any format that can be interpreted by SQL Server. A common date format is “yyyy-MM-dd HH:mm:ss.fff” which represents time with a 24 hour clock.

```
2017-01-02 15:45:00.000
```

The full list of supported formats can be found here:

<https://docs.microsoft.com/en-us/sql/t-sql/data-types/datetime-transact-sql>

Float

A numerical value that allows decimals.

Integer

A numerical value that allows whole numbers only.

Patients

The patients file contains information regarding each patient. Most importantly, it contains Patient_ID and Medical_Record_Number which are used to link various records.

Warning!

Be aware this file contains significant amounts of protected health information (PHI). While these identifiers are not uploaded to the central database, they are necessary for two reasons. First, they are used for linking different modules together. Second, these fields are used to know which identifiers need to be removed elsewhere in the patient's record. This process is called PHI scrubbing.

Target Date

A file for a given target date should contain all patients that had a case on that started on that date. In other words, for a given target date a Patients file should have all patients which have a case in the Cases file for the same target date.

Be aware that some of the fields in this module can change over time, most commonly due to moving residence or marriage. If the source system keeps a history of patient information, the identifiers used for a patient should be those that were correct on that target date. Otherwise, use the most recent patient information.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Patient_ID	No		The internal patient identifier, must uniquely identify a patient, must not be a form of PHI
Medical_Record_Number	No		Patient's medical record number (MRN)
First_Name	No		Patient's first name
Middle_Name	Yes		Patient's middle name or middle initial
Last_Name	No		Patient's last name
Date_of_Birth	No	DateTime	Patient's date of birth
Social_Security_Number	No		Patient's social security number (SSN) or other national identifier
Gender_ID	No		The identifier of the variable indicating patient gender
Gender_Name	No		The name of the variable for patient gender (e.g. F, M, Female, Male, Unknown, etc.)
Ethnicity_ID	Yes		The identifier of the variable for patient ethnicity
Ethnicity_Name	Yes		The name of the variable for patient ethnicity (e.g. Hispanic, non-Hispanic)
Race_ID	Yes		The identifier of the variable for patient race
Race_Name	Yes		The name of the variable for patient race (e.g. White, Black, Native American, Asian)
Address_Line_1	Yes		First line of patient's street address
Address_Line_2	Yes		Second line of patient's street address
Address_City	Yes		Patient's city
Address_State_Province	Yes		Patient's state or province
Address_Postal_Code	Yes		Patient's ZIP or other postal code
Phone_Number	Yes		Patient's phone number
Medicaid_ID	Yes		Patient's Medicaid ID (commonly left blank)

Additional Column Details

Patient_ID

Due to common documentation errors it is technically acceptable, though not preferred, that a single person can be represented with multiple Patient_IDs. However, each Patient_ID must refer to only one person.

Given patients are extract each time they have a surgical case, a Patient_ID can be duplicated across different Patients files. Therefore, Patient_ID does not need to be unique across files within an instance.

Medical_Record_Number

It is possible for a given Patient_ID to have multiple MRNs. This typically occurs when an instance shared by multiple hospitals. This field should be populated with the MRN most relevant to the hospital within the surgical case occurred for that target date. MRNs can then be linked together using the optional PatientCrosswalk module.

Gender_ID / Gender_Name

These columns store the variable used to represent gender. Many systems don't have for formal database identifier, so it is common to re-use one value for both fields.

Ethnicity_ID / Ethnicity_Name

These columns store the variable used to represent ethnicity, most commonly represented as Hispanic or non-Hispanic. If race and ethnicity are stored together in a single variable, use the same variable for both ethnicity and race.

Race_ID / Race_Name

These columns store the variable used to represent race. If race and ethnicity are stored together in a single variable, use the same variable for both ethnicity and race.

Example Data

Note examples presented here may word wrap. Be aware in these instances are actually a single line in the file.

Single Date Format

```
A-00000001,0123456789098,Jane,,Doe,01/01/1950,987-65-4321,F,Female,123,Non-Hispanic,B,Black,1234 Fake Ave,Apt 01,Our Town,NY,12345,321-123-4567,  
A-00000002,0987654321012,John,Q,Public,02/14/1972,123-45-6789,M,Male,321,Hispanic,W,White,,,,,
```

Multi Date Format

```
01/01/2017,A-00000001,0123456789098,Jane,,Doe,01/01/1950,987-65-4321,F,Female,123,Non-Hispanic,B,Black,1234 Fake Ave,Apt 01,Our Town,NY,12345,321-123-4567,  
01/02/2017,A-00000002,0987654321012,John,Q,Public,02/14/1972,123-45-6789,M,Male,321,Hispanic,W,White,,,,,
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: `MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv`
 - Multi date: `MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv`
- Module Specific Requirements
 - Medical_Record_Number is correctly formatted (no surrounding brackets)
 - Gender_ID / Gender_Name are always NULL or not NULL at the same time
 - Race_ID / Race_Name are always NULL or not NULL at the same time
 - Ethnicity_ID / Ethnicity_Name are always NULL or not NULL at the same time

Cases

The Cases module contains high level information about a procedure. Most importantly, it contains the link between a patient and their surgery-level documentation. It also contains which sub-organization a case belongs to, which is useful for large institutions who wish to partition their cases.

Target Date

For a target date, a Cases file should contain all cases that started on that date. Use anesthesia start to determine when a case starts.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Case_ID	No		The internal case identifier used to link other files together, must be unique across within an instance
Patient_ID	No		The internal patient identifier, must be the same ID as used in the "Patients" file
Encounter_ID	Yes		The encounter / admission / visit identifier
Organization_ID	No		The identifier of which parent institution / organization this case belongs to.
Organization_Name	No		The name of the parent institution / organization
Room_ID	No		The identifier that uniquely identifies a an operating room across an organization
Room_Name	No		The name of the operating room (e.g. "OR-1"), does not necessarily need to be unique
Room_Type_ID	No		The identifier of the room type variable
Room_Type_Name	No		The name of the room type variable that classifies the room (e.g. mixed use room, cardiac OR, etc.)
Admission_Type_ID	No		The identifier of the admission type variable
Admission_Type_Name	No		The name of the admission type variable (e.g. inpatient, outpatient, etc.)
Procedural_Service_ID	Yes		The identifier of the primary procedure service variable
Procedural_Service_Name	Yes		The name of the primary procedure service variable (e.g. cardiac, ortho, medical, radiology, etc.)
Case_Time	No	DateTime	The time the case started.
Diagnosis	Yes		The primary diagnosis text description for the procedure
Scheduled_Procedure	Yes	Text(8000)	The primary text description for the SCHEDULED procedure
Actual_Procedure	Yes	Text(8000)	The primary text description for the ACTUAL procedure

Additional Column Details

Organization_ID / Organization_Name

These columns store the variable used to represent the organization the case was performed in, generally the name of the hospital. These fields determine how cases are partitioned into different MPOG databases or are filtered out entirely. Make sure the values placed in this column support the desired end result.

Room_Type_ID / Room_Type_Name

These columns represent the variable used to represent a room category. For example, “general use OR” or “labor and delivery”. If no explicit room type variable exists, re-use Room_ID and Room_Name respectively.

Case_Time

The date stored in this column must match the target date.

If multiple anesthesia starts are documented on a case, use the earliest one.

Example Data

Single Date Format

```
598437,28579,1387592,5,Main Hospital,23,OR-5,5874,Delivery Room,0,Outpatient,13435,Obstetrics,2017-01-01 17:31:00,Labor,Delivery,C-Section  
824508,23857,,7,Cardiovascular Center,38,CVC-10,38,CVC-10,I,Inpatient,19485,Cardiac,2017-02-14 08:30:00,Coronary Artery Disease,CABG,CABG
```

Multi Date Format

```
01/01/2017,598437,28579,1387592,5,Main Hospital,23,OR-5,5874,Delivery Room,0,Outpatient,13435,Obstetrics,2017-01-01 17:31:00,Labor,Delivery,C-Section  
02/14/2017,824508,23857,,7,Cardiovascular Center,38,CVC-10,38,CVC-10,I,Inpatient,19485,Cardiac,2017-02-14 08:30:00,Coronary Artery Disease,CABG,CABG
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: `MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv`
 - Multi date: `MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv`
- Module Specific Requirements
 - Case_ID is unique within instance, represented only once within one target date and not again for any other target date
 - Organization_ID / Organization_Name supports desired partitioning
 - Organization_ID / Organization_Name are always NULL or not NULL at the same time
 - Room_ID / Room_Name are always NULL or not NULL at the same time
 - Room_Type_ID / Room_Type_Name are always NULL or not NULL at the same time
 - Procedure_Service_ID / Procedure_Service_Name are always NULL or not NULL at the same time
 - Case_Time date matches Target_Date

Labs

The Labs module contains all structured laboratory results. Common examples of labs are glucose or hematocrit.

Target Date

This module has two options for its target date definition. Note that only one option may be chosen per instance.

1. (Recommended) For a given target date, extract all labs taken on that date.
2. For a given target date, determine the list of all cases for that date. Then, extract all labs for those patients that were taken within the date range of 365 days before and after anesthesia start.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Lab_ID	No		The unique row identifier of the lab result
Patient_ID	No		The patient identifier, must be the same identifier given in the "Patients" file
Lab_Type_ID	No		The identifier of the lab type variable
Lab_Type_Name	No		The name of the lab type variable (glucose, hemoglobin, blood gases, etc.)
Was_Point_of_Care_Lab	Yes	Boolean	Whether the lab indicated was a point of care lab or formal lab
Sample_Time	Yes	DateTime	The time the lab sample was drawn
Observation_Time	No	DateTime	The time the lab result was returned
Unit_ID	Yes		The identifier of the unit of measurement variable
Unit_Name	Yes		The name of the unit of measurement variable (e.g. mL, g, cm)
Lab_Value	No		The value of the lab result (either string or number)
Lab_Value_Code	Yes		The value of the lab result has a coded representation, put that code here
Normal_Range_Low	Yes		The low end of the range for a normal lab result of this type
Normal_Range_High	Yes		The high end of the range for a normal lab result of this type
Status_Low_Normal_High	Yes		A code representing if the lab value was low, normal, or high (e.g. LL, L, N, H, HH)
Comment	Yes	Text(MAX)	Any additional comments for the lab value
Lab_Interface_Message	Yes	Text(MAX)	The original full message sent from a lab interface (e.g. the text of a HL7 message)

Additional Column Details

Lab_ID

It may be necessary to manufacture this row identifier if one does not exist in the source system. One recommended approach is to concatenate other identifiers. For example, PATIENTID-LABTYPEID-OBSTIME-LABVALUE. When doing so, be aware of the character limit for this column.

Lab_Value_Code

Rarely, some lab values will have an additional codified representation from a standard lexicon. For example, a value indicating a diagnosis of "diabetes" may also have a value code of "E11.9", the ICD-10 representation of diabetes. The vast majority of lab results do not have such a code and therefore this field is commonly blank.

Lab_Interface_Message

If the full text of the original lab message is available, it can optionally be placed here. However, the other columns must still be populated.

Example Data

Single Date Format

```
P00123-3456-1/1/2017 13:45:32-87,P00123,3456,Glucose,1,,1/1/2017 13:45:32,53,mg/dl,87,,70,150,N,,  
P03245-4784-1/1/2017 14:56:41-45,P03245,4784,Hematocrit,1,1/1/2017 14:45:00,1/1/2017 14:56:41,487,Percent,58,,42,54,H,,
```

Multi Date Format

```
01/01/2017,P00123-3456-1/1/2017 13:45:32-87,P00123,3456,Glucose,1,,1/1/2017 13:45:32,53,mg/dl,87,,70,150,N,,  
01/01/2017,P03245-4784-1/1/2017 14:56:41-45,P03245,4784,Hematocrit,1,1/1/2017 14:45:00,1/1/2017 14:56:41,487,Percent,58,,42,54,H,,
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv
 - Multi date: MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv
- Module Specific Requirements
 - Lab_ID uniquely identifies the row

PeriopAdministrations

The PeriopAdministrations module contains individual measurements of fluids and medications. While called administrations, this module also contains fluid outputs such as blood loss and urine.

Target Date

For a given target date, the PeriopAdministrations module should contain all medication and fluid documentation for cases started on that date.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Admin_ID	No		The row identifier for the administration, must be unique across all files
Case_ID	No		The case identifier, must be the same ID used in the "Cases" file
Phase_of_Care_ID	Yes		The identifier of the phase of care variable
Phase_of_Care_Name	Yes		The name of the phase of care variable (e.g. "Preop", "Intraop", "PACU")
Admin_Type_ID	No		The identifier of the administration type variable
Admin_Type_Name	No		The name of the administration type variable (e.g. PRBC, Lactated Ringers, Propofol)
Unit_ID	No		The identifier of the unit of measurement variable
Unit_Name	No		The name of the unit of measurement variable (e.g. mL, g, cm)
Route_ID	Yes		The identifier of the route of administration variable
Route_Name	Yes		The name of the route of administration variable (e.g. IV)
Dose_Start_Time	No	DateTime	The time of the administration, or in the case of an infusion, the time that the infusion was started
Dose_End_Time	Yes	DateTime	The time that the infusion was ended
Was_Infusion	Yes	Boolean	Whether the administration was given as a bolus or an infusion, if known
Admin_Value	Yes		The dose value or infusion rate
Comment	Yes		Any comments regarding the administration
Entered_Start_Time	Yes	DateTime	The time when the administration/start of the infusion was documented
Entered_End_Time	Yes	DateTime	The time when the end of the infusion was documented
Patient_Dosing_Weight_KG	Yes	Float	The patient's weight as used for the purposes of dosing, measured in kilograms

Additional Column Details

Admin_ID

It may be necessary to manufacture this row identifier if one does not exist in the source system. One recommended approach is to concatenate other identifiers. For example, CASEID-ADMINTYPEID-DOSESTARTTIME-ADMINVALUE. When doing so, be aware of the character limit for this column.

Dose_End_Time

The dose end time often only applies to medication infusions. Infusions pump settings can be adjusted numerous times during a surgery, so infusion rows in this module should represent one setting / rate.

For bolus medications and most fluids, dose end time will be blank.

Admin_Value

The value of the administration, most commonly the size of the dose. Note that this column is nullable, it is not optional in most instances. Currently any rows with a NULL value will be discarded. In the future MPOG may add support for medications that were given but have an undocumented dose.

Example Data

Single Date Format

```
C02345-7432-1/1/2017 15:23:00-50,C02345,Intraop,Intraop,7432,Propofol,4972,MCG/KG/MIN,124,IV,1/1/2017 15:23:00,1/1/2017 15:54:00,1,50,,1/1/2017 15:25:23,1/1/2017 15:55:02,65  
C02345-7432-1/1/2017 15:54:00-75,C02345,Intraop,Intraop,7432,Propofol,4972,MCG/KG/MIN,124,IV,1/1/2017 15:54:00,1/1/2017 16:23:00,1,75,,1/1/2017 15:55:14,1/1/2017 16:26:08,65
```

Multi Date Format

```
01/01/2017,C02345-7432-1/1/2017 15:23:00-50,C02345,Intraop,Intraop,7432,Propofol,4972,MCG/KG/MIN,124,IV,1/1/2017 15:23:00,1/1/2017 15:54:00,1,50,,1/1/2017 15:25:23,1/1/2017 15:55:02,65  
01/01/2017,C02345-7432-1/1/2017 15:54:00-75,C02345,Intraop,Intraop,7432,Propofol,4972,MCG/KG/MIN,124,IV,1/1/2017 15:54:00,1/1/2017 16:23:00,1,75,,1/1/2017 15:55:14,1/1/2017 16:26:08,65
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: `MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv`
 - Multi date: `MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv`
- Module Specific Requirements
 - Admin_ID uniquely identifies the row
 - Phase_of_Care_ID / Phase_of_Care_Name are always NULL or not NULL at the same time
 - Route_ID / Route_Name are always NULL or not NULL at the same time
 - Infusion rate changes should have one row per rate change.
 - Administrations should have multiple rows if multiple measurements were made. Do not attempt to average or total individual measurements.

PeriopObservations

The PeriopObservations module contains most of the text-based documentation from the anesthesia record as well as the numeric vital signs. This module contains the largest volume of data compared to other modules in this specification.

Data contained in this module can result in one of four tables: AIMS_IntraopNotes, AIMS_Preop, AIMS_IntraopPhysiologic, and AIMS_Outcomes.

Target Date

For a given target date, the PeriopObservations module should contain all documentation for cases started on that date, in the following areas:

- Intraoperative timings (e.g. anesthesia start, procedure end, patient in room)
- Preoperative documentation (e.g. height, weight, existing medications, medical history)
- Vital sign monitors
- Outcomes (e.g. PONV, anaphylaxis)
- Perioperative free-text notes

For a complete listing of what can be considered a PeriopObservation, you can review available MPOG concepts at <https://mpog.org/concept-browser/> with the following concept types:

- Intraoperative Events, Interventions, and Observations
- Outcome Observations
- Physiologic Observations
- Preoperative Observations

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Obs_ID	No		The row identifier for the observation, must be unique across all files
Case_ID	No		The case identifier, must be the same ID used in the "Cases" file
Phase_of_Care_ID	Yes		The identifier of the phase of care variable
Phase_of_Care_Name	Yes		The name of the phase of care variable (e.g. "Preop", "Intraop", "PACU", "Outcome")
Obs_Type_ID	No		The identifier of the observation type variable
Obs_Type_Name	No		The name of the observation type variable (e.g. blood pressure, surgery start, height, weight, pulse)
Observation_Time	Yes	DateTime	The time the observation was made
Entered_Time	Yes	DateTime	The time the observation was documented
Was_Deleted	Yes	Boolean	Whether the observation was deleted by the user or was marked as artifact
Was_User_Entered	Yes	Boolean	Whether the observation was user entered
Was_User_Entry_Expected	Yes	Boolean	Whether the observation was supposed to be user entered
Obs_Value	No		The value of the observation
Obs_Value_Code	Yes		If the observation value has a coded representation, enter that code here
Obs_Value_Code_Lexicon	Yes		If Obs_Value_Code is populated, this is the code type used (e.g. ICD9, ICD10, SNOMED, RXCUI, OTHER)
Comment	Yes		Any comments about the observation

Additional Column Details

Was_User_Entered

(1 = user entered, 0 = automatically captured)

Was_User_Entry_Expected

(1 = user entry expected, 0 = monitor capture expected)

Obs_Value_Code

While Obs_Value contains the main value of the observation, Obs_Value_Code can contain an alternative representation of the same value. Two common use cases for Obs_Value_Code are problem lists and home medications, which often can be linked to ICD9/10 codes and RXCUI codes respectively.

Example Data

Single Date Format

```
336411683,1914360244,Intraop,Intraop,536116,Anesthesia Machine Checked,2015-03-01 13:46:12.000,2015-03-01 13:46:13.070,0,,Anesthesia Machine Checked,,,
336411684,1914360244,Intraop,Intraop,536120,Equipment verified,2015-03-01 13:47:12.000,2015-03-01 13:47:13.700,0,,Equipment verified,,,
336412115,1914360244,Intraop,Intraop,36155,Anesthesia Start,2015-03-01 15:38:41.000,2015-03-01 15:38:41.377,0,,Anesthesia Start,,,
1611654115,1914360244,Preop,Preop,524503,PE Height (in),,2015-03-01 13:47:00.000,0,,19.000,,,
1611522835,1914360244,Preop,Preop,524504,PE Weight (kg),,2015-03-01 13:45:00.000,0,,3.000,,,
1611522851,1914360244,Preop,Preop,524507,MC PhysHR,,2015-03-01 13:41:00.000,0,,134.000,,,
1611522899,1914360244,Preop,Preop,524508,MC PhysTemperature (c),,2015-03-01 13:49:00.000,0,,36.500,,,
159827408,1914360244,Intraop,Intraop,149741405,Monitor Isoflurane Forane Insp,2015-03-01 16:18:00.000,2015-03-01 16:18:00.000,0,0,0,1.500,,,
159823671,1914360244,Intraop,Intraop,149737668,Monitor Isoflurane Forane Exp,2015-03-01 16:21:00.000,2015-03-01 16:21:00.000,0,0,0,0.600,,,
159269791,1914360244,Intraop,Intraop,149737672,Monitor Nitrous Exp,2015-03-01 15:47:59.000,2015-03-01 15:47:59.000,0,0,0,9.9999997648258E-02,,,
9564914,1914360244,Intraop,Intraop,32795,NFF-Pul Vent PIP,2015-03-01 15:50:59.000,2015-03-01 15:50:59.000,0,0,0,0.000,,,
9803275,1914360244,Intraop,Intraop,33224,NFF-RR,2015-03-01 16:04:00.000,2015-03-01 16:04:00.000,0,0,0,14.000,,,
12163415,1914360244,Intraop,Intraop,529876,CBD TOF,2015-03-01 17:46:00.000,2015-03-01 17:57:00.000,0,1,1,4 / 4,,,
10017304,1914360244,Intraop,Intraop,33301,NFF-Pul Vent PEEP,2015-03-01 16:13:00.000,2015-03-01 16:13:00.000,0,0,0,6.000,,,
9508896,1914360244,Intraop,Intraop,33309,NF-CV Rhythm,2015-03-01 15:49:00.000,2015-03-01 17:34:00.000,0,1,1,NSR,,,

```

Multi Date Format

```
03/01/2015,336411683,1914360244,Intraop,Intraop,536116,Anesthesia Machine Checked,2015-03-01 13:46:12.000,2015-03-01 13:46:13.070,0,,Anesthesia Machine Checked,,,
03/01/2015,336411684,1914360244,Intraop,Intraop,536120,Equipment verified,2015-03-01 13:47:12.000,2015-03-01 13:47:13.700,0,,Equipment verified,,,
03/01/2015,336412115,1914360244,Intraop,Intraop,36155,Anesthesia Start,2015-03-01 15:38:41.000,2015-03-01 15:38:41.377,0,,Anesthesia Start,,,
03/01/2015,1611654115,1914360244,Preop,Preop,524503,PE Height (in),,2015-03-01 13:47:00.000,0,,19.000,,,
03/01/2015,1611522835,1914360244,Preop,Preop,524504,PE Weight (kg),,2015-03-01 13:45:00.000,0,,3.000,,,
03/01/2015,1611522851,1914360244,Preop,Preop,524507,MC PhysHR,,2015-03-01 13:41:00.000,0,,134.000,,,
03/01/2015,1611522899,1914360244,Preop,Preop,524508,MC PhysTemperature (c),,2015-03-01 13:49:00.000,0,,36.500,,,
03/01/2015,159827408,1914360244,Intraop,Intraop,149741405,Monitor Isoflurane Forane Insp,2015-03-01 16:18:00.000,2015-03-01 16:18:00.000,0,0,0,1.500,,,
03/01/2015,159823671,1914360244,Intraop,Intraop,149737668,Monitor Isoflurane Forane Exp,2015-03-01 16:21:00.000,2015-03-01 16:21:00.000,0,0,0,0.600,,,
03/01/2015,159269791,1914360244,Intraop,Intraop,149737672,Monitor Nitrous Exp,2015-03-01 15:47:59.000,2015-03-01 15:47:59.000,0,0,0,9.9999997648258E-02,,,
03/01/2015,9564914,1914360244,Intraop,Intraop,32795,NFF-Pul Vent PIP,2015-03-01 15:50:59.000,2015-03-01 15:50:59.000,0,0,0,0.000,,,
03/01/2015,9803275,1914360244,Intraop,Intraop,33224,NFF-RR,2015-03-01 16:04:00.000,2015-03-01 16:04:00.000,0,0,0,14.000,,,
03/01/2015,12163415,1914360244,Intraop,Intraop,529876,CBD TOF,2015-03-01 17:46:00.000,2015-03-01 17:57:00.000,0,1,1,4 / 4,,,
03/01/2015,10017304,1914360244,Intraop,Intraop,33301,NFF-Pul Vent PEEP,2015-03-01 16:13:00.000,2015-03-01 16:13:00.000,0,0,0,6.000,,,
03/01/2015,9508896,1914360244,Intraop,Intraop,33309,NF-CV Rhythm,2015-03-01 15:49:00.000,2015-03-01 17:34:00.000,0,1,1,NSR,,,

```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: `MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv`
 - Multi date: `MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv`
- Module Specific Requirements
 - Obs_ID uniquely identifies the row
 - Phase_of_Care_ID / Phase_of_Care_Name are always NULL or not NULL at the same time

PeriopObservationDetails

The PeriopObservationDetails module contains data that has a child relationship to the PeriopObservations module.

Two common uses for details of this are templates and container notes. Templates are text notes that are constructed from multiple options. For example, a note template for endotracheal tube placement might look like “__ mm __ ET tube taped @ __ cm”. A completed template could be “7.5 mm single-lumen cuffed ET tube taped @ 23 cm”, meaning the details would be “7.5”, “single-lumen cuffed”, and “23”.

Container notes are more broad but generally are made of a short parent PeriopObservation with many child PeriopObservationDetails. For example, a note describing spinal anesthesia may have a parent note saying just “Spinal” and children notes detailing the needle type, patient position, approach, number of attempts, etc.

Data contained in this module can appear in the AIMS_IntraopNoteDetails or AIMS_PreopDetails tables. The table is determined by which table the parent observation was sent to. Note that AIMS_IntraopPhysiologic and AIMS_Outcomes does not support child observations, so any details whose parent notes are sent to these tables will not be exported.

Target Date

Any detail observation must have the same target date as its parent PeriopObservation.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Obs_Detail_ID	No		The row identifier for the observation detail, must be unique across all files
Obs_ID	No		The row identifier of the parent observation, must be the same ID as used in the PeriopObservations file
Obs_Detail_Type_ID	No		The identifier of the observation detail type
Obs_Detail_Type_Name	No		The name of the observation detail type (e.g. type of blade used in a intubation note)
Observation_Time	Yes	DateTime	The time the observation detail was made (if different from the parent observation)
Entered_Time	Yes	DateTime	The time the observation detail was documented (if different from the parent observation)
Was_Deleted	Yes		Whether the observation detail was deleted
Obs_Detail_Value	No		The observation detail value
Obs_Detail_Value_Code	Yes		A codified representation of the observation detail value (e.g. pick list ID, ICD 9/10 code)
Obs_Value_Code_Lexicon	Yes		If Obs_Value_Code is populated, this is the code type used (e.g. ICD9, ICD10, SNOMED, RXCUI, OTHER)
Comment	Yes		Any comments regarding the observation detail

Additional Column Details

Was_Deleted

(1 = deleted, 0 = not deleted)

Obs_Value_Code

While Obs_Value contains the main value of the observation, Obs_Value_Code can contain an alternative representation of the same value. Two common use cases for Obs_Value_Code are problem lists and home medications, which often can be linked to ICD9/10 codes and RXCUI codes respectively.

Example Data

Single Date Format

```
877345746,336412369,9249097,Blade Type,,,0,Miller #1,,,
877345746,336412369,9249097,Blade Type,,,1,Miller #2,,,
877345746,336412369,9249098,Attempt,,,1,1st Attempt,,,
877345746,336412369,9249098,Attempt,,,0,3rd attempt,,,
877149138,336412382,9254358,Laryngoscopic View,,,0,Grade 1 - Full view of Vocal Cords,,,
877149138,336412382,9264632,Cricothyroid Pressure,,,0,Yes,,,
877607890,336411989,9330166,ET Tube Size,,,0,3.000,,,
877607890,336411989,9330167,ET Tube Type,,,0,Uncuffed,,,
877607890,336411989,9330168,Taped at @,,,0,9.500,,,
877476818,336411904,586980,L/Min,,,0,5.000,,,
875772882,336412159,531992,Placed on,,,0,Under patient,,,
875772882,336412159,531992,Placed on,,,1,upper body,,,
875772882,336412159,9322517,Temp Setting,,,0,43.000,,,
875772882,336412159,9322517,Temp Setting,,,1,43.000,,,
875969490,336412141,587098,Type,,,0,14449.000,,,
875969490,336412141,587099,Location,,,0,32123.000,,,
875969490,336412141,587100,Side,,,0,14470.000,,,
875969490,336412141,587101,Size,,,0,32138.000,,,
875969490,336412141,527556,Name,,,0,0.000,,,
```

Multi Date Format

```
03/01/2015,877345746,336412369,9249097,Blade Type,,,0,Miller #1,,,
03/01/2015,877345746,336412369,9249097,Blade Type,,,1,Miller #2,,,
03/01/2015,877345746,336412369,9249098,Attempt,,,1,1st Attempt,,,
03/01/2015,877345746,336412369,9249098,Attempt,,,0,3rd attempt,,,
03/01/2015,877149138,336412382,9254358,Laryngoscopic View,,,0,Grade 1 - Full view of Vocal Cords,,,
03/01/2015,877149138,336412382,9264632,Cricothyroid Pressure,,,0,Yes,,,
03/01/2015,877607890,336411989,9330166,ET Tube Size,,,0,3.000,,,
03/01/2015,877607890,336411989,9330167,ET Tube Type,,,0,Uncuffed,,,
03/01/2015,877607890,336411989,9330168,Taped at @,,,0,9.500,,,
03/01/2015,877476818,336411904,586980,L/Min,,,0,5.000,,,
03/01/2015,875772882,336412159,531992,Placed on,,,0,Under patient,,,
03/01/2015,875772882,336412159,531992,Placed on,,,1,upper body,,,
03/01/2015,875772882,336412159,9322517,Temp Setting,,,0,43.000,,,
03/01/2015,875772882,336412159,9322517,Temp Setting,,,1,43.000,,,
03/01/2015,875969490,336412141,587098,Type,,,0,14449.000,,,
03/01/2015,875969490,336412141,587099,Location,,,0,32123.000,,,
03/01/2015,875969490,336412141,587100,Side,,,0,14470.000,,,
03/01/2015,875969490,336412141,587101,Size,,,0,32138.000,,,
03/01/2015,875969490,336412141,527556,Name,,,0,0.000,,,
```

Development Checklist

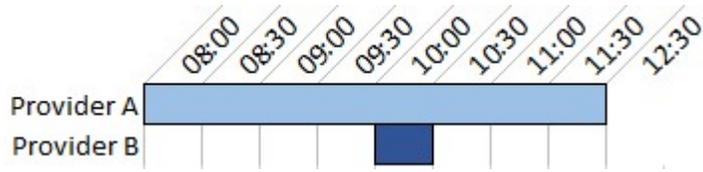
- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: `MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv`
 - Multi date: `MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv`
- Module Specific Requirements
 - Obs_Detail_ID uniquely identifies the row

StaffTracking

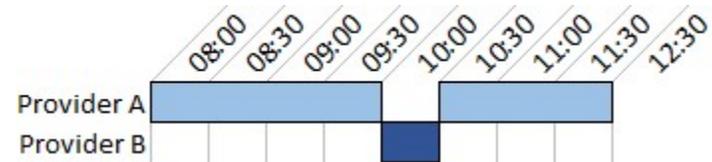
The StaffTracking module contains information related to staff who signed in and out of a surgery.

Warning!

MPOG quality performance metrics rely accurate sign-in and sign-out data. If a provider takes a break but there is no documented sign-out, any activity during that break period may be attributed to that provider.



Staff Example 1: Provider A has a single row for sign-in and sign-out. Provider A is considered signed into the surgery at 10:00AM, even if another provider is signed in at the same time.



Staff Example 2: Provider A has a two rows for sign-in and sign-out, indicating a break. Only Provider B is considered signed into the case at 10:00AM.

Target Date

For a given target date, determine the list of all cases that started on that date (see Cases module target date definition). Extract all staff sign-ins for those cases.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Staff_Tracking_ID	No		The row identifier for the staff sign-in/out, must be unique across all files
Case_ID	No		The case identifier, must be the same ID used in the "Cases" file
Staff_ID	No		The internal identifier for a staff member, must uniquely identify a person
Staff_First_Name	No		The staff's first name
Staff_Last_Name	No		The staff's last name
Staff_Pager	Yes		The staff's pager number
Email	Yes		The staff email address, a work address should be used over a personal one
NPI	Yes		The National Provider Identifier for the staff member
Masked_Public_ID	Yes		An identifier that can be used to display provider performance without listing the provider's name
Staff_Type_ID	No		The identifier of the staff type variable
Staff_Type_Name	No		The name of the staff type variable (e.g. anesthesia attending, anesthesia resident, surgeon, CRNA)
Was_Present	Yes	Boolean	If the staff member was physically present in the operating room
Sign_In_Time	No	DateTime	The time the staff member signed in
Sign_Out_Time	Yes	DateTime	The time the staff member signed out

Additional Column Details

Masked_Public_ID

This optional identifier can be useful if your institution already has an obfuscated identifier for a given staff member. This can be later used in performance reporting.

Example Data

Single Date Format

```
6328934,C8920437,2056681521,Janelle,Barton,86112,Janelle.Barton@myHealthSystem.org,4050029738,,04997,CRNA,1,2017-01-02 14:35:00.000,2017-01-02 17:52:00.000
0855984,C3901596,1217493913,Alexis,Coffey,99112,Alexis.Coffey@myHealthSystem.org,8330747457,,40539,CRNA,1,2017-01-02 09:42:00.000,2017-01-02 10:58:00.000
5576508,C8996770,907523355,Clinton,Ware,99220,Clinton.Ware@myHealthSystem.org,1946656363,,21963,Anesthesia Attending,0,2017-01-02 18:11:00.000,2017-01-02 21:27:00.000
7019040,C0608551,566494144,Kerry,Gill,76838,Kerry.Gill@myHealthSystem.org,3146071721,,29705,CRNA,1,2017-01-02 11:39:00.000,2017-01-02 15:01:00.000
```

Multi Date Format

```
01/02/2017,6328934,C8920437,2056681521,Janelle,Barton,86112,Janelle.Barton@myHealthSystem.org,4050029738,,04997,CRNA,1,2017-01-02 14:35:00.000,2017-01-02 17:52:00.000
01/05/2017,5530661,C5729337,1593734025,Kristine,Sweeney,95645,Kristine.Sweeney@myHealthSystem.org,,,90136,Anesthesia Resident,0,2017-01-05 13:53:00.000,2017-01-05 14:50:00.000
01/04/2017,2318318,C0980138,183256919,Melinda,Stanton,81057,Melinda.Stanton@myHealthSystem.org,,,89522,Anesthesia Attending,0,2017-01-04 13:46:00.000,2017-01-04 17:08:00.000
01/04/2017,3541329,C4552579,1295430981,Carol,Sheppard,73093,Carol.Sheppard@myHealthSystem.org,,,29067,Anesthesia Attending,1,2017-01-04 07:05:00.000,2017-01-04 10:11:00.000
01/05/2017,6332741,C9657012,1210497916,Janet,Blackwell,06478,Janet.Blackwell@myHealthSystem.org,1852397465,,60880,Anesthesia Resident,0,2017-01-05 21:59:00.000,2017-01-05 23:44:00.000
01/01/2017,6654649,C5939056,81304624,Philip,Mc Gee,,Philip.Mc Gee@myHealthSystem.org,0033624747,,05254,CRNA,0,2017-01-01 08:40:00.000,2017-01-01 10:32:00.000
01/02/2017,0855984,C3901596,1217493913,Alexis,Coffey,99112,Alexis.Coffey@myHealthSystem.org,8330747457,,40539,CRNA,1,2017-01-02 09:42:00.000,2017-01-02 10:58:00.000
01/03/2017,1018510,C0158851,2124195857,Lillian,Ayala,73228,Lillian.Ayala@myHealthSystem.org,,,93223,Anesthesia Resident,0,2017-01-03 21:15:00.000,2017-01-03 23:49:00.000
01/04/2017,8502376,C8606800,1586262110,Elizabeth,Garner,42782,Elizabeth.Garner@myHealthSystem.org,,,56207,Anesthesia Attending,1,2017-01-04 14:57:00.000,2017-01-04 18:53:00.000
01/01/2017,5493979,C6841090,2088353260,Kristy,Bowman,01419,Kristy.Bowman@myHealthSystem.org,4537333638,,04662,Anesthesia Resident,0,2017-01-01 21:36:00.000,2017-01-02 01:34:00.000
01/02/2017,5576508,C8996770,907523355,Clinton,Ware,99220,Clinton.Ware@myHealthSystem.org,1946656363,,21963,Anesthesia Attending,0,2017-01-02 18:11:00.000,2017-01-02 21:27:00.000
01/01/2017,1358737,C0079531,736692570,Trina,Mcintosh,,Trina.Mcintosh@myHealthSystem.org,0019323729,,33458,Anesthesia Resident,1,2017-01-01 08:27:00.000,2017-01-01 11:30:00.000
01/01/2017,1382836,C8503407,1673189945,Derek,Mc Donald,92664,Derek.Mc Donald@myHealthSystem.org,7580890483,,29936,Anesthesia Resident,0,2017-01-01 22:50:00.000,2017-01-02 02:29:00.000
01/01/2017,5195507,C9047554,222277392,Joey,Douglas,,Joey.Douglas@myHealthSystem.org,,,04792,Anesthesia Resident,0,2017-01-01 10:03:00.000,2017-01-01 11:28:00.000
01/02/2017,7019040,C0608551,566494144,Kerry,Gill,76838,Kerry.Gill@myHealthSystem.org,3146071721,,29705,CRNA,1,2017-01-02 11:39:00.000,2017-01-02 15:01:00.000
01/04/2017,6445941,C5168313,374303402,Cornelius,Manning,44564,Cornelius.Manning@myHealthSystem.org,,,37780,Anesthesia Attending,0,2017-01-04 17:59:00.000,2017-01-04 21:07:00.000
01/04/2017,6991842,C7325864,626507584,Angelica,Woods,12386,Angelica.Woods@myHealthSystem.org,,,73008,CRNA,1,2017-01-04 21:48:00.000,2017-01-04 23:54:00.000
01/03/2017,6364318,C1666333,884253661,Roxanne,Harrison,58972,Roxanne.Harrison@myHealthSystem.org,,,40807,Surgical Attending,1,2017-01-03 13:18:00.000,2017-01-03 17:01:00.000
01/03/2017,0672072,C5589324,545587177,Jimmy,Woodard,45363,Jimmy.Woodard@myHealthSystem.org,0525300265,,26716,Surgical Attending,0,2017-01-03 12:46:00.000,2017-01-03 15:58:00.000
01/01/2017,0557990,C3983499,1827763575,Heath,Mc Daniel,78801,Heath.Mc Daniel@myHealthSystem.org,,,93943,CRNA,1,2017-01-01 13:00:00.000,2017-01-01 16:19:00.000
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: `MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv`
 - Multi date: `MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv`
- Module Specific Requirements
 - Staff_Tracking_ID uniquely identifies the row
 - A provider should have multiple rows within a single case if they signed in and out multiple times

HospitalMortality

The HospitalMortality module contains dates of in-hospital patient deaths. It can also contain diagnosis codes detailing the cause of death.

This module requires only deaths that occurred within the hospital. However, including additional mortality data from other sources (e.g. a state registry) is acceptable.

Target Date

Each target date should include all deaths on that date.

Alternatively, when using the Reference_Date column each target date must include all deaths that use that reference date.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Medical_Record_Number	No		The patient's medical record number
Date_of_Death	Yes	DateTime	The specific date the patient died. Mutually exclusive with Reference_Date
Reference_Date	Yes	DateTime	The date after which the patient could have died. Mutually exclusive with Date_of_Death
Days_within_Reference_Date	Yes	Integer	The maximum number of days after "Reference Date" in which the patient could have died
Cause_of_Death_Lexicon	Yes		The type of codes used in the cause of death columns (e.g. ICD-9/ICD-10)
Primary_Cause_of_Death	Yes		The diagnosis code primary cause of death
Related_Cause_of_Death_1	Yes		Another diagnosis code related to cause of death
Related_Cause_of_Death_2	Yes		Another diagnosis code related to cause of death
Related_Cause_of_Death_3	Yes		Another diagnosis code related to cause of death
Related_Cause_of_Death_4	Yes		Another diagnosis code related to cause of death
Related_Cause_of_Death_5	Yes		Another diagnosis code related to cause of death

Additional Column Details

Date_of_Death

Date of Death may include a time but doing so is optional.

Reference_Date

The Reference_Date and Days_within_Reference_Date columns should only be used when a specific date of death is not available, but a known range such as 30-day mortality is.

Example Data

Single Date Format

```
123456789012,2017-01-01 09:07:00.000,,,,,,,,,  
987654321987,2017-01-01 15:12:00.000,,,,,,,,,  
543210987654,2017-01-01 18:13:00.000,,ICD-9,410.90,250.1,,,,,  
567890123456,2017-01-01 10:35:00.000,,ICD-10,Z82.41,,,,,  
789012345678,,2017-01-01 00:00:00.000,30,,,,,,,,,
```

Multi Date Format

```
01/01/2017,123456789012,2017-01-01 09:07:00.000,,,,,,,,,  
01/01/2017,987654321987,2017-01-01 15:12:00.000,,,,,,,,,  
01/01/2017,543210987654,2017-01-01 18:13:00.000,,ICD-9,410.90,250.1,,,,,  
01/01/2017,567890123456,2017-01-01 10:35:00.000,,ICD-10,Z82.41,,,,,  
01/01/2017,789012345678,,2017-01-01 00:00:00.000,30,,,,,,,,,
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv
 - Multi date: MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv
- Module Specific Requirements
 - Either Date_of_Death or Reference_Date is filled in, but not both

- Reference_Date / Days_within_Reference_Date are always NULL or not NULL at the same time

Procedures

The Procedures module contains procedure codes billed either during a hospital visit or a surgical procedure.

Institutions are required to contribute procedure codes from a professional fee source. This can be either the anesthesiologist, the surgeon, or both. Procedure codes from hospital discharge billing are also accepted but are not required.

Target Date

For professional fee sources, a given target date should contain all procedures whose date of service start matches that date.

For hospital discharge sources, a given target date should contain all procedures whose date of admission matches that date.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Data_Source	No		(Present for legacy reasons) Please fill in the Source System used in the filename.
Medical_Record_Number	Yes		The patient's medical record number (MRN)
Patient_ID	Yes		The patient's internal database ID. This should match the Patient_ID column in the Patients file.
Visit_ID	Yes		The internal identifier for the encounter/visit/admittance
Case_ID	Yes		The internal case identifier (DO NOT attempt to link to a case unless an explicit link already exists)
Invoice_ID	Yes		The identifier/number for the billing invoice
Procedure_Code_ID	Yes		The unique row identifier for a procedure billing code
Procedure_Source_Type	No		Describes what type of billing the procedure code originated from (professional fee vs hospital discharge)
Date_of_Service_Start	Yes	DateTime	The time the procedure occurred or started
Date_of_Service_End	Yes	DateTime	The time the procedure ended (typically only used for anesthesia time-based charges)
Date_of_Admission	Yes	DateTime	The time when the patient was admitted
Date_of_Discharge	Yes	DateTime	The time when the patient was discharged
Procedure_Code	No		The actual procedure code (e.g. 00222)
Procedure_Code_Lexicon	No		The dictionary the code belongs to (e.g. CPT, ICD-9, ICD-10)
Procedure_Code_Priority	Yes	Integer	The relative importance/rank of the procedure
Was_Primary_Procedure	Yes	Boolean	Whether the procedure was considered the primary procedure of an operation.
Total_Anesthesia_Units	Yes	Float	The total number of time units charged by the anesthesiology provider

Additional Column Details

Procedure_Code

Make sure codes are formatted properly such as including padding zeros, decimals, and trailing zeros (e.g. "00832" is correct, "832" is not)

Warning!

A common problem encountered with billing codes is improper formatting. CPT and ICD codes are often reformatted as numbers by other programs such as Excel. Always handle these types of identifiers as text and not as numeric.

Data_Source

Due to legacy reasons, the data source / source system field is present in the file data as well as being in the file name. Re-use whatever value is used for source system in the filename.

Professional fee data and hospital discharge data are stored in different systems. Different sources can have separate files for the same dates as long as the "SourceSystem" is different in the filename.

Patient_ID

For the same patient this should match the Patient_ID column in the Patients file.

Case_ID

DO NOT attempt to link to a case unless an explicit link already exists. For example, do not attempt to determine the case ID by matching on patient ID and date.

Case_ID is not expected for procedures from hospital discharge billing.

Procedure_Source_Type

The typical values are "Anesthesia Pro Fee", "Surgery Pro Fee", "Unspecified Pro Fee", and "Hospital Discharge".

Was_Primary_Procedure

Only fill this in if Procedure_Code_Priority is unavailable.

Total_Anesthesia_Units

This only applies to time-based, anesthesia procedure codes.

Example Data

The following examples contain different source systems, which are pulled into separate files. For clarity, this separation of files is represented as whitespace.

Single Date Format

```
MyAnesBilling,123456789012,P12345,54781,,784-231546-4,87462168575,Anesthesia Pro-Fee Billing,2017-01-01 09:10:00.000,2017-01-01 11:32:00.000,2017-01-01,2017-01-02,01500,CPT,,1,14.0
```

```
MySurgBilling,123456789012,P12345,54781,,4578-12-7,87462168575,Surgery Pro-Fee Billing,2017-01-01 09:10:00.000,,2017-01-01,2017-01-02,35570,CPT,,1,
```

```
MyHospitalBilling,123456789012,P12345,54781,,489-124-56,87462168575,Hospital Discharge Billing,,,2017-01-01,2017-01-02,02H633Z,ICD-10,,,
```

Multi Date Format

```
01/01/2017,MyAnesBilling,123456789012,P12345,54781,,784-231546-4,87462168575,Anesthesia Pro-Fee Billing,2017-01-01 09:10:00.000,2017-01-01 11:32:00.000,2017-01-01,2017-01-02,01500,CPT,,1,14.0
```

```
01/01/2017,MySurgBilling,123456789012,P12345,54781,,4578-12-7,87462168575,Surgery Pro-Fee Billing,2017-01-01 09:10:00.000,,2017-01-01,2017-01-02,35570,CPT,,1,
```

```
01/01/2017,MyHospitalBilling,123456789012,P12345,54781,,489-124-56,87462168575,Hospital Discharge Billing,,,2017-01-01,2017-01-02,02H633Z,ICD-10,,,
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: `MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv`
 - Multi date: `MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv`
- Module Specific Requirements
 - Procedure_Code contains a standardized code with the correct formatting
 - Medical_Record_Number, Patient_ID, or both must be filled in
 - Date_of_Service_Start, Date_of_Admission, or both must be filled in

ProcedureModifiers

The ProcedureModifiers module contains modifier codes for procedures listed in the Procedures module. These modifier codes provide additional information about the procedure, most commonly regarding how care was somehow unusual (e.g. anesthesia was delivered by the surgeon).

This module is optional.

Target Date

The target date of a procedure modifier should match the target date of its parent procedure.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Data_Source	No		(Present for legacy reasons) Please fill in the data_source used in the filename.
Medical_Record_Number	Yes		The patient's medical record number (MRN)
Patient_ID	Yes		The patient's internal database ID. This should match the Patient_ID column in the Patients file.
Visit_ID	Yes		The internal identifier for the encounter/visit/admittance
Case_ID	Yes		The internal case identifier (DO NOT attempt to link to a case unless an explicit link already exists)
Invoice_ID	Yes		The identifier/number for the billing invoice
Procedure_Code_ID	Yes		The unique row identifier for a procedure billing code
Procedure_Source_Type	No		Describes what type of billing the procedure code originated from (professional fee vs hospital discharge)
Date_of_Service_Start	Yes	DateTime	The time the procedure occurred or started
Date_of_Service_End	Yes	DateTime	The time the procedure ended (typically only used for anesthesia time-based charges)
Date_of_Admission	Yes	DateTime	The time when the patient was admitted
Date_of_Discharge	Yes	DateTime	The time when the patient was discharged
Procedure_Code	No		The procedure code to which the modifier code is applied (e.g. 00222)
Procedure_Code_Lexicon	No		The dictionary the procedure code belongs to (e.g. CPT, ICD-9, ICD-10)
Modifier_Code_Lexicon	No		The dictionary the modifier code belongs to
Modifier_Code	No		The actual modifier code (e.g. QS)

Additional Column Details

Procedure_Code

Make sure codes are formatted properly such as including padding zeros, decimals, and trailing zeros (e.g. "00832" is correct, "832" is not)

Warning!

A common problem encountered with billing codes is improper formatting. CPT and ICD codes are often reformatted as numbers by other programs such as Excel. Always handle these types of identifiers as text and not as numeric.

Data_Source

Due to legacy reasons, the data source / source system field is present in the file data as well as being in the file name. Re-use whatever value is used for source system in the filename.

Professional fee data and hospital discharge data are stored in different systems. Different sources can have separate files for the same dates as long as the "SourceSystem" is different in the filename.

Patient_ID

For the same patient this should match the Patient_ID column in the Patients file.

Case_ID

DO NOT attempt to link to a case unless an explicit link already exists. For example, do not attempt to determine the case ID by matching on patient ID and date.

Case_ID is not expected for procedures from hospital discharge billing.

Procedure_Source_Type

The typical values are "Anesthesia Pro Fee", "Surgery Pro Fee", "Unspecified Pro Fee", and "Hospital Discharge".

Example Data

Single Date Format

```
MyAnesBilling,123456789012,P12345,54781,,784-231546-4,87462168575,Anesthesia Pro-Fee Billing,2017-01-01 09:10:00,2017-01-01 11:32:00,2017-01-01,2017-01-02,01500,CPT,CPT,51
MyAnesBilling,123456789012,P12345,54781,,784-231546-4,87462168575,Anesthesia Pro-Fee Billing,2017-01-01 09:10:00,2017-01-01 11:32:00,2017-01-01,2017-01-02,01500,CPT,HCPCS,GC
MyAnesBilling,123456789012,P12345,54781,,784-231546-4,87462168575,Anesthesia Pro-Fee Billing,2017-01-01 09:10:00,2017-01-01 11:32:00,2017-01-01,2017-01-02,01500,CPT,HCPCS,QS
```

Multi Date Format

```
01/01/2017,MyAnesBilling,123456789012,P12345,54781,,784-231546-4,87462168575,Anesthesia Pro-Fee Billing,2017-01-01 09:10:00,2017-01-01 11:32:00,2017-01-01,2017-01-02,01500,CPT,CPT,51
01/01/2017,MyAnesBilling,123456789012,P12345,54781,,784-231546-4,87462168575,Anesthesia Pro-Fee Billing,2017-01-01 09:10:00,2017-01-01 11:32:00,2017-01-01,2017-01-02,01500,CPT,HCPCS,GC
01/01/2017,MyAnesBilling,123456789012,P12345,54781,,784-231546-4,87462168575,Anesthesia Pro-Fee Billing,2017-01-01 09:10:00,2017-01-01 11:32:00,2017-01-01,2017-01-02,01500,CPT,HCPCS,QS
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv
 - Multi date: MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv
- Module Specific Requirements
 - Procedure_Code contains a standardized code with the correct formatting
 - Medical_Record_Number, Patient_ID, or both must be filled in
 - Date_of_Service_Start, Date_of_Admission, or both must be filled in
 - Each modifier code is listed on a separate row

Diagnoses

The Diagnoses module contains diagnosis codes billed for either a hospital visit or surgical case. These diagnoses are often represented as ICD-9 or ICD-10 codes.

Institutions need to contribute diagnosis codes from hospital discharge billing. Contributing diagnosis codes from professional fee billing is recommended but not required.

Target Date

For professional fee sources, a given target date should contain all diagnoses whose date of service start matches that date.

For hospital discharge sources, a given target date should contain all diagnoses whose date of admission matches that date.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Data_Source	No		(Present for legacy reasons) Please fill in the data_source used in the filename.
Medical_Record_Number	No		The patient's medical record number (MRN).
Patient_ID	Yes		The patient's internal database ID. This should match the Patient_ID column in the Patients file.
Visit_ID	Yes		The internal identifier for the encounter/visit/admittance
Case_ID	Yes		The internal case identifier (DO NOT attempt to link to a case unless an explicit link already exists)
Invoice_ID	Yes		The identifier/number for the billing invoice
Procedure_Code_ID	Yes		The unique row identifier for a procedure billing code
Procedure_Code	Yes		The procedure code to which the diagnosis code is linked
Diagnosis_Code_ID	Yes		The unique row identifier for a diagnosis billing code
Diagnosis_Source_Type	No		Describes what type of billing the diagnosis code originated from (professional fee vs hospital discharge)
Date_of_Service_Start	Yes	DateTime	The time the related procedure occurred or started.
Date_of_Service_End	Yes	DateTime	The time the related procedure ended
Date_of_Admission	Yes	DateTime	The time when the patient was admitted
Date_of_Discharge	Yes	DateTime	The time when the patient was discharged
Date_of_Diagnosis_Start	Yes	DateTime	The date the diagnosis was first observed
Date_of_Diagnosis_End	Yes	DateTime	The date the diagnosis was no longer present in the patient
Diagnosis_Code	No		The actual diagnosis code (e.g. 249.01)
Diagnosis_Code_Lexicon	No		The dictionary the code belongs to (e.g. ICD-9)
Diagnosis_Code_Priority	Yes	Integer	The relative importance/rank of the diagnosis
Was_Primary_Diagnosis	Yes	Boolean	Whether the diagnosis was considered the primary reason for an operation
Present_On_Admission	Yes	Boolean	Whether the diagnosis/condition was present at the time the patient was admitted

Additional Column Details

Diagnosis_Code

Make sure codes are formatted properly such as including padding zeros, decimals, and trailing zeros (e.g. "005.0" is correct, "5" or "50" are not).

Warning!

A common problem encountered with billing codes is improper formatting. CPT and ICD codes are often reformatted as numbers by other programs such as Excel. Always handle these types of identifiers as text and not as numeric.

Data_Source

Due to legacy reasons, the data source / source system field is present in the file data as well as being in the file name. Re-use whatever value is used for source system in the filename.

Professional fee data and hospital discharge data are stored in different systems. Different sources can have separate files for the same dates as long as the "SourceSystem" is different in the filename.

Patient_ID

For the same patient this should match the Patient_ID column in the Patients file.

Case_ID

DO NOT attempt to link to a case unless an explicit link already exists. For example, do not attempt to determine the case ID by matching on patient ID and date.

Case_ID is not expected for procedures from hospital discharge billing.

Diagnosis_Source_Type

The typical values are "Anesthesia Pro Fee", "Surgery Pro Fee", "Unspecified Pro Fee", and "Hospital Discharge".

Example Data

The following examples contain different source systems, which are pulled into separate files. For clarity, this separation of files is represented as whitespace.

Single Date Format

```
MyAnesBilling,123456789012,P12345,54781,,784-231546-4,87462168575,01500,5484736215,Anesthesia Pro-Fee Billing,2017-01-01 09:10:00,2017-01-01 11:32:00,,,,,249.01,ICD-9,,1,
```

```
MySurgBilling,123456789012,P12345,54781,,4578-12-7,87462168575,35570,234657,Surgery Pro-Fee Billing,2017-01-01 09:10:00,,2017-01-01,2017-01-02,,,249.01,ICD-9,,1,
```

```
MyHospitalBilling,123456789012,P12345,54781,,489-124-56,,,412975162,Hospital Discharge Billing,,,2017-01-01,2017-01-02,2017-01-01,,998.9,ICD-9,,,0
```

Multi Date Format

```
01/01/2017,MyAnesBilling,123456789012,P12345,54781,,784-231546-4,87462168575,01500,5484736215,Anesthesia Pro-Fee Billing,2017-01-01 09:10:00,2017-01-01 11:32:00,,,,,249.01,ICD-9,,1,
```

```
01/01/2017,MySurgBilling,123456789012,P12345,54781,,4578-12-7,87462168575,35570,234657,Surgery Pro-Fee Billing,2017-01-01 09:10:00,,2017-01-01,2017-01-02,,,249.01,ICD-9,,1,
```

```
01/01/2017,MyHospitalBilling,123456789012,P12345,54781,,489-124-56,,,412975162,Hospital Discharge Billing,,,2017-01-01,2017-01-02,2017-01-01,,998.9,ICD-9,,,0
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: `MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv`
 - Multi date: `MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv`
- Module Specific Requirements
 - Procedure_Code contains a standardized code with the correct formatting
 - Diagnosis_Code contains a standardized code with the correct formatting
 - Medical_Record_Number, Patient_ID, or both MUST be filled in
 - Date_of_Service_Start, Date_of_Admission, or both MUST be filled in

Payers

The Payers module contains the names of all insurance payers billed for a hospital visit or surgical case.

This module is only required if your institution plans on participating in MPOG's Qualified Clinical Data Registry (QCDR) program.

Target Date

For professional fee sources, a given target date should contain all payer names whose date of service start matches that date.

For hospital discharge sources, a given target date should contain all payer names whose date of admission matches that date.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Data_Source	No		(Present for legacy reasons) Please fill in the Source System used in the filename.
Medical_Record_Number	Yes		The patient's medical record number (MRN).
Patient_ID	Yes		The patient's internal database ID.
Visit_ID	Yes		The internal identifier for the encounter/visit/admittance
Case_ID	Yes		The internal case identifier
Invoice_ID	Yes		The identifier/number for the billing invoice
Payer_Source_Type	No		Describes what type of billing the payer name originated from (professional fee vs hospital discharge)
Date_of_Service_Start	Yes	DateTime	The time the related procedure occurred or started
Date_of_Service_End	Yes	DateTime	The time the related procedure ended
Date_of_Admission	Yes	DateTime	The time when the patient was admitted
Date_of_Discharge	Yes	DateTime	The time when the patient was discharged
Payer_Name	No		The name of the insurance payer

Additional Column Details

Professional Fee data will often come from another source than the hospital discharge data. These two sources can have separate files for the same dates as long as the "SourceSystem" is different in the filename.

Data_Source

Due to legacy reasons, the data source / source system field is present in the file data as well as being in the file name. Re-use whatever value is used for source system in the filename.

Patient_ID

For the same patient this should match the Patient_ID column in the Patients file.

Case_ID

DO NOT attempt to link to a case unless an explicit link already exists. For example, do not attempt to determine the case ID by matching on patient ID and date.

Case_ID is not expected for procedures from hospital discharge billing.

Payer_Source_Type

The typical values are "Anesthesia Pro Fee", "Surgery Pro Fee", "Unspecified Pro Fee", and "Hospital Discharge".

Example Data

Single Date Format

```
MyBillingSystem,123456789012,P12345,564710,,547-7849-4,Hospital Discharge,,,2017-01-01 08:10:00, 2017-01-03 10:13:00,MEDICARE PART B
MyBillingSystem,123456789012,P12345,564710,,547-7849-4,Hospital Discharge,,,2017-01-01 08:10:00, 2017-01-03 10:13:00,BCBS
MyBillingSystem,987654321098,P54321,764284,,748-5163-3,Hospital Discharge,,,2017-01-01 11:34:00, 2017-01-01 15:30:00,SELF PAYER
```

Multi Date Format

```
01/01/2017,MyBillingSystem,123456789012,P12345,564710,,547-7849-4,Hospital Discharge,,,2017-01-01 08:10:00, 2017-01-03 10:13:00,MEDICARE PART B
01/01/2017,MyBillingSystem,123456789012,P12345,564710,,547-7849-4,Hospital Discharge,,,2017-01-01 08:10:00, 2017-01-03 10:13:00,BCBS
01/01/2017,MyBillingSystem,987654321098,P54321,764284,,748-5163-3,Hospital Discharge,,,2017-01-01 11:34:00, 2017-01-01 15:30:00,SELF PAYER
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition

- Extract always fully pulls each date, target date is never incomplete
- If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
- No whitespace in NULL fields
- File name matches the correct template
 - Single date: MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv
 - Multi date: MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv
- Module Specific Requirements
 - Medical_Record_Number, Patient_ID, or both must be filled in
 - Date_of_Service_Start, Date_of_Admission, or both MUST be filled in

PatientCrosswalk

The PatientCrosswalk module contains links between different MRNs that represent the same patient. This is often used with larger hospital systems where a patient can have multiple MRNs. This module is optional.

Target Date

The PatientCrosswalk module shares the same target date definition with the Patients module.

File Columns

Flat File Column	Nullable?	Data Type	Purpose
Medical_Record_Number	No		The patient's MRN used within the anesthesia system (i.e. what is present in the patient module)
Other_Medical_Record_Number	Yes		Another MRN used to represent the same patient elsewhere

Additional Column Details

Medical_Record_Number

For the same patient this should match the Medical_Record_Number column in the Patients file.

Other_Medical_Record_Number

This identifier is commonly a MRN, but in some instances may be another type of patient identifier. This can be useful for linking external data which does not use MRN.

Example Data

Single Date Format

```
123456789012,987654321098  
456789012345,890123456789
```

Multi Date Format

```
01/01/2017,123456789012,987654321098  
01/01/2017,456789012345,890123456789
```

Development Checklist

- Global Requirements
 - All columns are represented in file
 - Columns are in correct order
 - Required columns are never blank
 - Column data types are obeyed
 - File does not contain a header row
 - Delimiter characters have been escaped
 - No additional delimiters are used (e.g. quotes around each field)
 - Target date follows module definition
 - Extract always fully pulls each date, target date is never incomplete
 - If using the multi date format, target date column has been added to the beginning of each row, formatted as MM/dd/yyyy (include leading zeroes)
 - No whitespace in NULL fields
 - File name matches the correct template
 - Single date: `MODULE_SPECVERSION_SOURCESYSTEM_TARGETDATE_PULLDATE.csv`
 - Multi date: `MODULE_SPECVERSION_SOURCESYSTEM_FILELABEL_PULLDATE.csv`