

**3TL WORKFLOW**

# Table of **Contents**

[Table of Contents 1](#_Toc90473737)

[3TL Flowchart 0](#_Toc90473738)

[Purpose 0](#_Toc90473739)

[3TL Process and Dashboard 1](#_Toc90473740)

[3TL Process Overview 1](#_Toc90473741)

[3TL Dashboard 1](#_Toc90473742)

[Defining the Dashboard Key 2](#_Toc90473743)

[Fact Loaded, Post-Outcome 2](#_Toc90473744)

[Fact Loaded, Pre-Outcome 2](#_Toc90473745)

[Fact Loaded, Partial Load 2](#_Toc90473746)

[Fact Not Loaded 3](#_Toc90473747)

[Fact Ready to Load 3](#_Toc90473748)

[3TL Detail 3](#_Toc90473749)

[File Processed Tracker 4](#_Toc90473750)

[DQR Tracker 4](#_Toc90473751)

[Fact Calendar 4](#_Toc90473752)

[Correspondence 4](#_Toc90473753)

[3T/PR File Received from Member 5](#_Toc90473754)

[Initial Member Communications 5](#_Toc90473755)

[FTP Location 5](#_Toc90473756)

[FTP File Integrity Check 5](#_Toc90473757)

[Files in the FTP Folders Section 5](#_Toc90473758)

[3T/PR Load into Staging Tables 6](#_Toc90473759)

[3T Loop Process 6](#_Toc90473760)

[PR Loop Process 7](#_Toc90473761)

[Post Loop Process\*\* 8](#_Toc90473762)

[Generating Data Quality Reports (DQRs) 8](#_Toc90473763)

[What are Data Quality Reports? 8](#_Toc90473764)

[Accessing DQR Reports 8](#_Toc90473765)

[Exporting DQR Reports 9](#_Toc90473766)

[3T Queries for Data Load Check 9](#_Toc90473767)

[DQR Correspondence w/ Member 10](#_Toc90473768)

[Correspondence Clean Example 10](#_Toc90473769)

[3TL Assist 11](#_Toc90473770)

[Files in the FTP Folders 11](#_Toc90473771)

[Correspondence 11](#_Toc90473772)

[DQR Manager 11](#_Toc90473773)

[Error Records 12](#_Toc90473774)

[Preparing Error Records for the Member 12](#_Toc90473775)

[Signoff Process with Error Records 13](#_Toc90473776)

[Fact Tables Loaded w/ 3T Data 13](#_Toc90473777)

[Load Fact Staging Tables 13](#_Toc90473778)

[Merge All 14](#_Toc90473779)

[Update Outcomes 14](#_Toc90473780)

[Raw Data Download (RDD) 14](#_Toc90473781)

[RDD Request from Members 14](#_Toc90473782)

[Quality Indicators 15](#_Toc90473783)

[Running and Loading the RDD 15](#_Toc90473784)

[Saving the RDD to a CSV 15](#_Toc90473785)

[Delivery of the RDD 16](#_Toc90473786)

# 3TL Flowchart

Graphical user interface

Description automatically generated with medium confidenceText

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generatedA picture containing text

Description automatically generatedText

Description automatically generatedGraphical user interface, text

Description automatically generatedGraphical user interface, text, application

Description automatically generatedText

Description automatically generated

**Manual Process: Jemel**

**Manual Process: Jemel**

**Manual Process: Keith**

**Manual Process: Jemel**

**Manual Process: Jemel**

**Manual Process: Keith**

**Manual Process: Jemel**

**Error Filled Data**

**Clean Data/ Minimal Error Data Sendoff**

**RDD Send out**

**Update Outcomes**

**PSI Load**

**3T/PR Load into Staging Tables**

# Purpose

The purpose of this document is to illustrate how the 3TL Staging process works within the current infrastructure, and how it can be used to assist members with delivering their data to our internal network. This process is precursory in the 3T Staging Process. This document describes how to communicate with members and the step-by-step process in ensuring data submissions are received, cleansed based on member specifications, member sign-off, and loading the finalized data.

# 3TL Process and Dashboard

## 3TL Process Overview

The 3TL (3T Load) process is comprised of the ETL process: extracting, transforming, and loading data. Each piece of ETL and 3TL coincide: 3T/PR Load being the extract piece, the DQR stage being the transformative piece, and the onboarding data piece being the cleanup in our Fact Tables. Our members send their facility data, known as the data submission; the goal is to enforce/enact data quality standards, and provide our members quality reporting. When members send in a data submission, it comes in the form of two files, the 3000 file (3T) and the Physician Roster (PR) files. The way we receive these files is by an FTP server within our system and the members are given credentials to access and send files to their designated “FTP\_VSE” folder.

## 3TL Dashboard

You can access this report on our Intranet site:

Chart, scatter chart

Description automatically generated

Table

Description automatically generatedThe 3TL Dashboard contains our 3T/PR submitting members in accordance with their HCO\_CODE; with a focus on the months/quarters submitted to us.

The goal of this report is to identify all members who have submitted their data and have successfully undergone our preliminary data cleansing steps for quarterly reporting.

## Defining the Dashboard Key

At the bottom of the page, we find our dashboard key. The following sections will further define each color code:



### Fact Loaded, Post-Outcome

All data from the specified HCO and month are completely loaded into the fact tables and are updated with their associated outcomes. This data is RDD and Portal Ready.

### Fact Loaded, Pre-Outcome

All data from the specified HCO and month are completely loaded into the fact tables but have **NOT** been updated with their associated outcomes. This data is **NOT** RDD and Portal Ready.

### Fact Loaded, Partial Load

All data from the specified HCO and month are **partially** loaded into the fact tables and have **NOT** been updated with their associated outcomes. This data is **NOT** RDD and Portal Ready.

### Fact Not Loaded

All data from the specified HCO and month are **NOT** loaded into the fact tables and have **NOT** been updated with their associated outcomes. This data is **NOT** RDD and Portal Ready. This could mean a couple of things, we either never received data from the member for a particular month, or the data could be in the transformation process (DQR). More details in the [**3TL Detail**](#_3TL_Detail) section.

### Fact Ready to Load

All data from the specified HCO and month are **ready to be** completely loaded into the fact tables with their associated outcomes. This data is **NOT** RDD and Portal Ready.

## 3TL Detail

The Dashboard key can only encompass important stages in the handling of member’s data. With 3TL Detail, we can further drill into where a submission may be in the process.

Graphical user interface

Description automatically generated with medium confidenceYou can select into any of the HCO\_CODE squares to open 3TL Detail:

### File Processed Tracker

File Processed Tracker is useful in determining whether a submission has been resubmitted at any point. If the tracker shows a “1”, this signifies the files loaded only once from the FTP to the 3T Staging tables, but if the tracker displays more than “1”; there’s been a resubmission. Keep in mind, this only tracks 3T files, this is with the assumption PR files are included. The Correspondence tables at the bottom will elaborate more on a given situation.

### DQR Tracker

For more information on what DQRs are, [**click here**](#_Generating_Data_Quality). To help identify DQR statuses, [**click here**](#_DQR_Manager).

### Fact Calendar

A picture containing text, street

Description automatically generatedGraphical user interface

Description automatically generatedFact Calendar is used to verify if a submission loaded into the factPatient table, containing every day within the month. There are some instances where the data may not include the appropriate date range or potentially missing days:

This can help identify whether a submission will require a resubmission from the member.

### Correspondence

The Correspondence table is an integral part of 3TL Detail, as it provides comprehensive notation on the “goings-on” of a specified submission. The table also provides an accurate log of when these events take place with the “Entry Date” field.

Text

Description automatically generated

Listed above are **required** notes found in each submission. Any entry that contains parentheses are automated processes, as seen between the [BATCH PROCESSED] and [FACT LOADED]. Please only use these statuses when moving forward in the process. Be sure to write the statuses first, then followed by the additional note:

Ex. DQR SENT: Insert Note Here

* DQR sent is self-explanatory, a manual entry made when the email including the DQR is sent to the member.
* Fact Ready’s entry will update the 3TL dash with the “Fact Ready to Load” color code and signal to load that submission into the fact tables.
* When a file is received in the FTP, add an entry with its submission time and date into Correspondence.
* When loading any Quality Indicator, make sure to notate that particular indicator has been loaded.

# 3T/PR File Received from Member

## Initial Member Communications

In most cases, a member will contact you when they decide to send a file to the FTP.

## FTP Location

To find the data submissions from our members, use this path address to access the FTP folder:

[**\\SQL02\c$\inetpub\ftproot\LocalUser**](file:///\\SQL02\c$\inetpub\ftproot\LocalUser)

Within this folder, you will notice the varying names. These acronyms represent the hospitals and their HCO Codes, for example:

Ex.1) **ftpUserALKA2** = AL 🡪 Athens Limestone

KA2 🡪 HCO\_CODE

Ex.2) **ftpUserHMEA** = HM 🡪 Halifax Medical Center

EA 🡪 HCO\_CODE

# FTP File Integrity Check

If you have questions regarding the file specifications, refer to the [**Data Submission Manual-Vizient Southeast file**](../../Import/b/3000%20File/Data%20Submission%20Manual-Vizient%20Southeast.pdf) for more details.

## Files in the FTP Folders Section

When reviewing files in the FTP, be sure to check [**3TL Assist’s “Files in the FTP Folders” section**](https://www.vizientsoutheast.com:4433/portal/admin/3TL_Assist.aspx) to better review the contents of each 3T/PR file in the FTP:

1. If the filename is following the correct format:
   * HOSPITALNAME\_StartDate\_EndDate\_3T
   * HOSPITALNAME\_StartDate\_EndDate\_PR
2. If the File size is not at 0MB, which may suggest an empty file.

Or in some instances the file may be smaller than what a member usually sends.

1. If the First Row contains data.

No Data in the first row will prevent a proper 3T load, if shown be sure to remove the blank first row.

1. If the end of the file name does **NOT** include any extensions, and if there are any, do be sure to remove said extension.

Ex. HOSPITALNAME\_StartDate\_EndDate\_3T.**txt**

Once complete, the files can be loaded, if completely checked off. Otherwise, correspondence with the member is required to ensure a proper load.

More on 3TL Assist in the [**Correspondence with a Member section**](#_Correspondence_w/_Member).

# 3T/PR Load into Staging Tables

This process is the most integral part of 3TL, due to the multitude of automated processes that occur in each loop. To summarize, our SSIS package is ran when there are data submissions in our FTP. The Staging package runs through each member’s file separately and not simultaneously, running through each loop before moving on to the next FTP folder. There are three primary stages the package will loop through; it starts with a 3T file (3T Loop). Once the file is processed properly, it moves on to the PR file (PR Loop) and completes the run with a post-loop that will run MSDRG related processes.

## 3T Loop Process

1. Moves 3T file from FTP to Staging Folder.
2. Writes entry to **3T\_File\_Submission\_Log** table.
   1. The fields that are given content: *FTP\_FOLDER, FTP\_FILENAME, FTP\_RETRIEVAL\_DATE*
3. Retrieve *BATCH\_ID.*
   1. Using the MAX function, the most recent batch from any given member will be give precedence.
4. Files are checked to determine file type: Fixed Width or Pipe Delimited.
   1. Fixed Width defines the length of characters each field may have. For example, the first column is always 20 characters long, the second column is 7 characters long and so on.
   2. Delimited format defines a character to separate each column on each line. We primarily receive and disseminate CSV (Comma-separated values) to and from our members.
   3. This step also retrieves the last modified date/time.
5. Two separate processes occur depending on the file type:
   1. If the file is Fixed Width:
      * Gets file from Staging folder.
      * Gets the file’s row count.
      * Derive *BATCH\_ID* (from step 3).
      * Loads file into 3T Staging table.
   2. If the file Ragged Right (Pipe):
      * Gets file from Staging folder.
      * Gets the file’s row count.
      * Derive *BATCH\_ID* (from step 3).
      * Loads file into 3T Staging table.
6. Enters row count into the **3T\_File\_Submission\_Log** table.
7. Gets the Error Records for 3T Error table.
   1. Uses the **3T\_Data\_Quality\_Rules** table to determine the errors.
   2. This table looks primarily for required fields.
8. Loads records into the **3T\_Import** table.
   1. Joins error table by *BATCH\_ID* and excludes errors.
9. Copies Staging Files into Archive folder.
   1. Content of the loading folder are NOT deleted.
10. Inserts into the *FILE\_SUBMISSION\_DATE* and *BATCH\_PROV* fields, and into the **DQR\_MANAGER** and **CORRESPONDENCE** tables.

## PR Loop Process

1. Moves PR file from FTP to Staging Folder.
2. Writes entry to **3T\_File\_Submission\_Log** table.
   1. The fields that are given content: *FTP\_FOLDER, FTP\_FILENAME, FTP\_RETRIEVAL\_DATE*
3. The PR *BATCH\_ID* is replaced with the 3T *BATCH\_ID.*
   1. A separate process takes the *BATCH\_ID* from the 3T Loop.
4. Two separate processes occur depending on the file type:
   1. If the file is Fixed Width:
      1. Gets file from Staging folder.
      2. Gets the file’s row count.
      3. Derive *BATCH\_ID* (from step 3).
      4. Loads file into PR Staging table.
   2. If the file Ragged Right (Pipe):
      1. Gets file from Staging folder.
      2. Gets the file’s row count.
      3. Derive *BATCH\_ID* (from step 3).
      4. Loads file into PR Staging table.
5. Captures Errors before moving on.
6. Enters row count into the **3T\_File\_Submission\_Log** table.
7. Loads records into the **PR\_Import** table.
8. Copies Staging Files into Archive folder.
   1. Content of the loading folder are NOT deleted.
9. Completes PR Import Load.

## Post Loop Process\*\*

Once all loops are complete:

1. Runs the **usp\_UpdateMappedValues** stored procedure.
   1. Updates **3T\_Import** with hospital defined values from mapping table.
2. Retrieves MSDRG grouper data.
3. Retrieves MSDRG using CMS tool.
4. Truncates the MSDRG grouper output table.
   1. Clears out residual data.
5. MSDRG Output files gets loaded into Output table
6. Update MSDRG in **3T Import**
   1. Anywhere in the data that contains Nulls or Blanks are updated.

# Generating Data Quality Reports (DQRs)

## What are Data Quality Reports?

After reviewing the files sent by our members, it is time to create a Data Quality Report, or DQR for short. The purpose of these reports is to quickly view any errors that **PORTAL\_ENGINE** caught when loading the 3T file into the tables; the 3T files are loaded into the **3T\_Import\_Staging** and **3T\_Import\_Staging\_Error** tables. When this happens, these reports are automatically generated with an associated batch ID. These reports are necessary in the staging process, as it will allow members to have an overview of the errors in their data before we release it into the portal.

## Accessing DQR Reports

The quickest way to access a DQR report is on the 3TL Dashboard. You can also view the DQR report in the “Batch ID” column within the Correspondence table:

Text

Description automatically generated

To reach the original location of where the DQR’s are generated, use this link: [**https://www.vizientsoutheast.com:4433/portal/Intranet/3T\_Import\_Error\_Report.aspx**](https://www.vizientsoutheast.com:4433/portal/Intranet/3T_Import_Error_Report.aspx)

You will be brought to a site titled “File Submission Report”. Here, you can select various 3T FTP files by each respective member. If you wish to access a particular report, here is how. In this example, we want to find the errors for Athens Limestone for the month of September.

To view any DQR, just select any link in the batchID column:

Graphical user interface, text, application, email

Description automatically generated

## Exporting DQR Reports

To export a DQR, save the DQR as a PDF:

Graphical user interface, table

Description automatically generated

When sending DQRs to our members, we will send them a PDF of their batch. Be sure to save the DQRs server-side: [**\\SQL02\Import\b\3000 File\DQR\DQR PDFs**](file:///\\SQL02\Import\b\3000%20File\DQR\DQR%20PDFs)

## 3T Queries for Data Load Check

However, it’s important to review the database in case a report either did not generate properly or if a batch has not been loaded yet. The queries below show the batches that have been loaded into the desired table by **HCO\_CODE**/member for all date ranges. You can also change the table from **3T\_Import\_Staging** to either **3T\_Import\_Staging\_Error** or **3T\_Import**, and the query will run the same way.

**3T\_Import\_Staging**

select hco\_code, BATCHID, i.provnum, min(disdate) as minDisDate, max(disdate) as maxDisDate from [3T\_Import\_Staging] as i

join global.dbo.refhco as ref

on i.PROVNUM = ref.PROVNUM

where HCO\_CODE = 'ac1'

group by i.provnum, batchid, hco\_code

order by minDisDate

**3T\_Import\_Staging\_Error**

select hco\_code, BATCHID, i.provnum, min(disdate) as minDisDate, max(disdate) as maxDisDate from [3T\_Import\_Staging\_Error] as i

join global.dbo.refhco as ref

on i.PROVNUM = ref.PROVNUM

where HCO\_CODE = 'ac1'

group by i.provnum, batchid, hco\_code

order by minDisDate

**3T\_Import**

select hco\_code, BATCHID, i.provnum, min(disdate) as minDisDate, max(disdate) as maxDisDate from [3T\_Import] as i

join global.dbo.refhco as ref

on i.PROVNUM = ref.PROVNUM

where HCO\_CODE = 'ac1'

group by i.provnum, batchid, hco\_code

order by minDisDate

# DQR Correspondence with a Member

When sending out the DQR, it’s important to remember the error threshold. We want to make sure the member is aware their submission is fragmented with errors.

1. If the submission is completely unusable, this is the time to ask for a resubmission.
2. If the submission has a small number of errors, we can offer them an [**Error Record**](#_Error_Records) to either clean or remove certain records. This is usually a rare occurrence but necessary in the process.
3. If the submission is clean, then we can move on with the [**Fact Tables Loaded w/3T Data**](#_Fact_Tables_Loaded) stage in processing their data.

Here is a prime example of how this can be communicated to our members:

## Correspondence Clean Example

*Hello,*

*Just wanted to follow up with you on your October data quality reports. Your data is clean, as it has reached within the 1% of error threshold for IP records. Our system will automatically clean up any additional error records automatically.*

*If you have any questions, let me know!*

Please acknowledge that is example may not fit your circumstance and is best to communicate with your team if any issues or concerns may arise.

## 3TL Assist

3TL Assist is the one-stop shop for DQR management. Here you can view the contents of the FTP folder and each submission awaiting further processing, update Correspondence records and DQR Manager for a batch’s DQR status.

### Files in the FTP Folders

Refer to the [**Files in the FTP Folders Section**](#_Files_in_the) for details.

### Correspondence

Here, you can insert notes and updates on a submission’s development. All entries entered can be viewed in the 3TL Dashboard.

- Be sure to fill in all boxes, as the entry cannot be inserted otherwise.

- Be sure to refresh the page to keep entries in chronological order when inserting notes consecutively. If two entries have same exact time/date, it will order by the “Notes” Column.

For more on formatting and entry types, [**click here**](#_Correspondence).

### DQR Manager

Once Correspondence has been completely updated with the necessary entries, you can now update the status of a given DQR. **Be sure to only change the status when ready.** Once the status has been changed, it will no longer editable on the page, unless it’s **status code 5 or 6**. Otherwise, you will need to go into the DATA\_TRACKER database within the Correspondence table to update it accordingly.

Below are the DQR statuses, and their usage:

#### [Status 1] No DQR Sent

A DQR has not been sent to a member. This occurs most when a submission has not been loaded for the month.

#### [Status 2] DQR Sent, No Response

A DQR has been sent, but the member has not responded. The status is used on rare occasions where there potentially is a new member or inconsistency.

#### [Status 3] DQR Sent, Received Response (Clean Data/ Minimal Error Data Sendoff)

A DQR has been sent and the member gave a response. The data has no IP errors or is below the error threshold. This status is the most common, as most members will send clean data. There are times when a DQR has been sent and is clean, but a member does not respond. In most instances, the data is good to pass on to the next stages.

#### [Status 4] DQR Sent, Received Response (Error Data Cleanup)

A DQR has been sent and the member gave a response. Their data went over the 1% error threshold, and we offer the member their [**Error Records**](#_Error_Records), to clean up any of the excess error files they may have. Error records are a rare occurrence, as most members find that resubmitting is the easier option. Many times, if we do send out these records, it may only be a handful of records that may need to be either removed or updated. If there is a lack thereof a response and the data is a majority clean, the data is good enough to pass on to the next stages.

#### [Status 5] No DQR Sent

A DQR was not sent due to the contents of their submission being either incorrectly mapped or the data was almost completely unusable.

#### [Status 6] DQR Ready, Not Sent

A DQR has been generated and is ready to send. This status is automatically assigned when a batch is generated from loading the 3T file from the FTP.

# Error Records

## Preparing Error Records for the Member

Before any DQR email is sent out, make sure to prepare the member’s error record. The DQRs provides a general overview of the errors currently presiding within the database, the error records are the specific records that are either missing values within their respective column or an incorrect datatype. From **3T\_Import\_Staging\_Error**, you will need to copy each batch into an Excel worksheet; this will provide the member a way to communicate which records they can either correct or delete.

To find these error records, go to the [**3T\_Import\_Error\_Report**](https://www.vizientsoutheast.com:4433/portal/Intranet/3T_Import_Error_Report.aspx) page. Look for the batch, then select the “Get Code” link on the far right column:

Table

Description automatically generated with medium confidence

You will want to copy the query with the comment “Get records to share with member using FTP”. Once you run the query, you will need to copy all the records, then paste the records into the template excel worksheet in the [**\\SQL02\f$\Import\b\3000 File**](file:///\\SQL02\f$\Import\b\3000%20File%20) path.

Rename the document by this format: **FTPUSER\_BATCHID\_MONTHYEAR**

Ex. For Tallahasse – TMHAC\_2021219\_102020

## Signoff Process with Error Records

When you have finished putting together the worksheet, place the member’s error records into their FTP folder, that way they can view the worksheet as soon as they receive your DQR email.

To better understand this process, below is a diagram representing a simplified version of how the Signoff process works:

Chart

Description automatically generated

The end goal is to clean up the data in a way that will safely arrive back in **3T\_Import**. It starts in Step 1, where both **3T\_Import\_Staging** and **3T\_Import** are inserted into **3T\_Import\_Awaiting\_Signoff**. The errors will be removed using the **3T\_Import\_Staging\_Error** table. Once those records are cleaned up, they can be removed from both previous tables, and inserted back into **3T\_Import** as it is the final stage before being pushed downstream for reporting. Usually, you want to keep the records in **3T\_Import\_Awaiting\_Signoff**, just in case it does not load properly and to prevent accidentally deleting the whole batch from all tables.

# Fact Tables Loaded w/ 3T Data

## Load Fact Staging Tables

Once the 3T submissions have undergone the DQR process, they are ready to be loaded into the Fact Staging tables. When loading into these tables, we will use the stored procedure **loadStagingTables**. You will need to specify a batch ID, as the tables are initially truncated; thus, cannot have multiple batches loaded at once. The process is as follows:

1. Truncates the initial staging tables.
2. Builds **dimPhysician** with the **MergePhysicianRoster** stored procedure.
3. Builds the Patient header staging table with the **load\_factPatientImport** stored procedure with the associated batch ID.
4. Updates the Patient ID seed, otherwise the key associated.
5. Creates indexes on 3T\_Import and \_factPatientImport for processing performance.
6. Builds the remaining staging tables: Physician, Diagnosis and Procedure**.**
7. Drops all created indexes.

### PSI Load

When the tables are completely loaded, you will need to run through the Quality Indicators processing, as our Portal uses this PSI information for reporting. For more information**,** please refer to the [**Quality Indicators document**](../RDD/Quality%20Indicators%20Walkthrough.docx), as you will need to run this data before any RDD can go out.

## Merge All

Once **loadStagingTables** has completed executing and is reviewed, the data will need to be transitioned into the final fact tables, using the **MergeAll** stored procedure. The process is as follows:

1. All duplicate patient records are moved to the **\_factPatientReprocessed** table.
2. Merge all remaining fact tables, where each import table will insert its contents into the final fact tables.
3. Write an entry into Correspondence 🡪 [FACT LOADED]: BatchID

## Update Outcomes

When the data is completely merged, run the **updateOutcomes** stored procedure to update complications, mortalities, readmissions, and their associated denominators/flags.

# Raw Data Download

The data is now completely loaded and report ready. For more information on Raw Data Downloads, refer to the [**Raw Data Downloads document**](../RDD/Raw%20Data%20Download%20Walkthrough.docx).