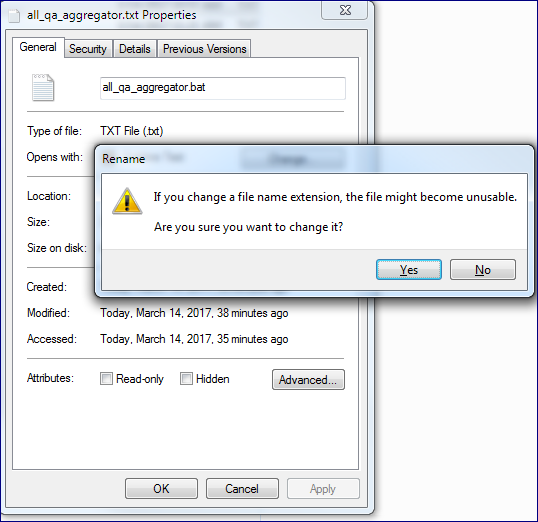
DB Creation Guide

\*You will need “All\_qa\_aggregator.bat” and “Template Pricing DB.mdb” for this walkthrough\*

# 1. Aggregating Data

1. Download a current copy of All\_qa\_aggregator.bat. This file may be required to be stored as a txt file for transfer, so you may need to download All\_qa\_aggregator.txt and then change the file extension from .txt to .bat by right-clicking the file, clicking Properties, and then editing the file name in the Properties window that appears.



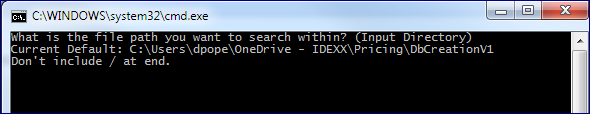
1. Locate the folder containing the folders containing the All\_qa.txt files you wish to aggregate. Copy the file path of the folder.
   1. The folder you will need is most likely be stored in a Windows Share located at:

[\\ECWIFS\grp-vol3\Information\_Services\Data\_Delivery\Reports\Mapping\_Issues\_Spreadsheets](file:///\\ECWIFS\grp-vol3\Information_Services\Data_Delivery\Reports\Mapping_Issues_Spreadsheets)

* 1. In this instance, we plan to aggregate CC17 All\_qa.txt files, so the file path that we need to copy is:

[\\ECWIFS\grp-vol3\Information\_Services\Data\_Delivery\Reports\Mapping\_Issues\_Spreadsheets\CC\CC17](file:///\\ECWIFS\grp-vol3\Information_Services\Data_Delivery\Reports\Mapping_Issues_Spreadsheets\CC\CC17)

* 1. It’s important that the file path does not including a trailing “\” at the end.
  2. Microsoft Access has a file limit of 2gb, the database itself takes some space as well—so it’s important to ensure that your initial all\_qa.txt is not much larger than 1.3GB or so. After cleaning, there will be room to import more, if you would like.

1. Open “All\_qa\_aggregator.bat”. A black window will appear such as the one below.
2. Right-click in the window and paste the file path that you copied from the previous step. This path will represent all the folder locations that will be crawled for All\_qa.txt files. It is also a field that will accept wildcards. This means that you can narrow or widen your search by using the \* symbol (asterisk) to represent ‘any text input’. For instance, with the file path we copied:

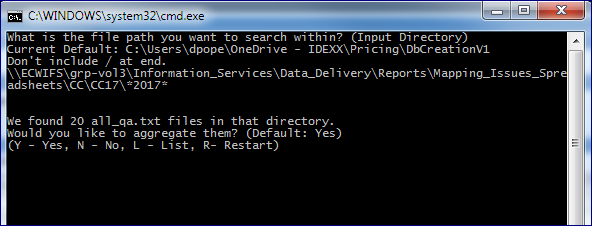
[\\ECWIFS\grp-vol3\Information\_Services\Data\_Delivery\Reports\Mapping\_Issues\_Spreadsheets\CC\CC17](file:///\\ECWIFS\grp-vol3\Information_Services\Data_Delivery\Reports\Mapping_Issues_Spreadsheets\CC\CC17)

This CC17 folder contains folders with project data from December 2016 through, currently, February 2017. If we were to want only data from 2017, then we could modify our input directory, as so, to narrow our search to a specific group of folders:

[\\ECWIFS\grp-vol3\Information\_Services\Data\_Delivery\Reports\Mapping\_Issues\_Spreadsheets\CC\CC17\\*2017\*](file:///\\ECWIFS\grp-vol3\Information_Services\Data_Delivery\Reports\Mapping_Issues_Spreadsheets\CC\CC17\*2017*)

The all\_qa\_aggregated file that we are generating is going to be imported into Microsoft Access, which has a 2gb file limit. After import, an Access database may be as much as a few hundred mb larger than the original text file. Try to keep this in mind while choosing how many All\_qa.txt files to aggregate. You can always add more data to an aggregated file, but it is difficult cleanly remove data. To add more data, you simply choose the same output file name.

1. After entering the file path with any optional modifiers, you’ll be presented with a screen that looks like so:



At this point, you can list the files to be aggregated by typing “L” and hitting enter.

Upon seeing this list, one may realize that there is an All\_qa.txt listed that they do not actually want included. Rather than trying again, there is also a text file in the folder where the All\_qa\_aggregator.bat file is located called “All\_qa\_filepath.txt”. While leaving All\_qa\_aggregator.bat running, one can open this text file and carefully remove the exact lines they don’t wish to include. After saving, one can hit “L” again in the All\_qa\_aggregator.bat window, and should see their changes implemented.

1. Once one is happy with the list of All\_qa.txt files they wish to collect, they may hit “Y” to move on to choosing an output directory. The default directory is the location where the All\_qa\_aggregator.bat file is located, however one can choose any directory they wish. It is important not to include any trailing “\” symbols when entering the output directory. Once you hit enter, the aggregator automatically begins generating your file and will continue until the operation completes.
2. Once the operation completes, one can exit the program, split the file into smaller sections (if needed), or restart the All\_qa\_aggregator.bat script and begin again.
   1. If the same output directory is chosen multiple times in the same day, the All\_qa.txt file located within that directory will be appended to, allowing you to collect many different collections of All\_qa.txt files into the same aggregated file.
   2. Beware that the split function will split your All\_qa.txt file based on file size and has the potential to cut a single record into two corrupt records at each split. With many thousands of transactions, the effect of this may be trivial, however, it is an important factor to consider if/when you need to use the split function.

# 2. Importing Data

1. Download “Template Pricing DB.mdb” to the same directory as your All\_qa\_aggregated file.
2. Rename the database, as appropriate, to either “CC\_P2.mdb”, “Last\_Dose.mdb”, “Nutritional.mdb”, or “Pharma.mdb”. The Pricing Template expects them exactly this way.
3. Open the renamed database and click “Enable Content”, found on the yellow banner.
4. Double-click a form named “Import and Clean” listed under Forms on the left-hand side.
5. Once the form is open, now is a good time to close down any unused, or resource-heavy, applications that you may be running, especially if they are also Office applications.
6. Click the button that is labeled: “Import Data”
7. Once the file dialog box appears, double-click your chosen all\_qa\_aggregated file to begin.
8. This process varies and can take over an hour to complete. Please be patient and work on other important tasks while you wait. In the default example, it took almost 3 hours to import 2.965M records, 1.23GB. You can check the import’s progress by refreshing a File Explorer window that is displaying the Access database file. As the data imports, the database’s size will increase to reflect the added data. Be aware that Access has a 2gb file size limit and that the final size of a freshly imported Access database will be slightly larger than the original All\_qa\_aggregated file.
9. After the import has finished, you’ll receive a dialog informing you of how long it took, and you should also have two new tables: “All\_qa\_aggregated\_” with a date, and an ImportErrors table.

# 3. Removing Bad Data

1. Double-click “Import and Clean”, listed under Forms on the left-hand side.
2. Once the form is open, now is a good time to close down any unused or resource-heavy applications that you may be running, especially if they are also Office applications.
3. Click the button that is labeled: “Clean DB”
4. Wait for the cleaning to finish. It can take some time for this to process. In the default example, 2.965 million records were processed through a series of filters to clean the data. For some filters, this comparison has to be done one by one. The result of the filtering was nearly a million bad records removed in a process which took about 10 minutes to complete.
5. After receiving a confirmation dialog stating that the cleaning is complete and informing you of how long it took to accomplish, confirm that the “ImportErrors” table has gone, then click on the “Database Tools” ribbon at the top of the Microsoft Access window. On the far-left of the “Database Tools” ribbon, click the “Compact and Repair Database” button. After a short amount of time, the procedure will complete and then your database will be complete as well.
6. At this point, you should exit out of the database and confirm that the database name is appropriate for the Project Set you pulled your data from. You can now begin pricing items by using “Template Research Format.xlsm” and “Guide\_Pricing.docx”.
7. You can also distribute your finished database so that many people can use the data set for their purposes. If you would like to make a very large database, you’ll have now freed up a significant portion of the Access 2gb size limit and can now import, clean, and merge more records.

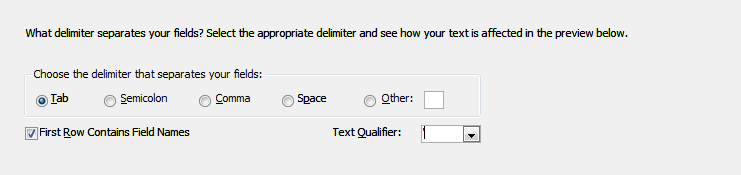
# - On Merging Tables

Merging tables is a simple operation that rarely needs to be completed. The main purpose of merging a table is to add data to a pre-existing Pricing Database. As pricing is only done a few times a year, it’s rare that one may add to a database, most cases require creating a brand new database to hold new pricing data. Nevertheless, if the circumstance arises in which you wish to add data to an existing database, you will need to merge the generated tables with the existing All\_qa table, after you finish the Import and Cleaning steps. To Merge the freshly Cleaned new data, follow these steps:

1. Ensure you have already run the “Clean DB” operation on the freshly collected data.
2. Double-click “Import and Clean”, listed under Forms on the left-hand side.
3. Click the button that is labeled: “Merge Table w/All\_qa Table”
4. A dialog box will appear, prompting you to enter the name of the table you wish to Merge.
5. The utility attempts to guess the appropriate table name, assuming you aggregated your file on this day and did not rename the file— then this table name should be accurate.
6. Otherwise, adjust the “Table Name” to *exactly* reflect the name of the table you wish to merge. The “Table Name” should appear identically to the table name listed on the left-hand side of the main Access window, next to the original All\_qa table you wish to merge with.
7. Once you are satisfied that the names match, click “OK”
8. If you entered a valid table name, then an “Append Records” warning dialog will appear.
9. The dialog warns: “You are about to append \_\_ row(s).” ( \_\_ = # of rows being added to All\_qa)
10. If you are sure that you wish to merge tables, then the correct response to this dialog is: “Yes” (This operation cannot be undone.)
11. After some time, if successfully completed, you should receive a confirmation dialog stating that the merge is complete and informing you of how long it took to accomplish. Upon closing this dialog, the aggregated table that you merged should disappear and only the newly merged “All\_qa” table should remain.
12. Finally, click on the “Database Tools” ribbon at the top of the Microsoft Access window. On the far-left of the “Database Tools” ribbon, click the “Compact and Repair Database” button. After a short amount of time, the procedure will complete and your database may be complete as well.
13. If finished, you should exit out of the database. You can now begin pricing items by using “Template Research Format.xlsm” and “Guide\_Pricing.docx”.
14. You can also distribute your finished database so that many people can use the data set for pricing or other purposes. If you are making a very large database, will need to repeat the Merge step immediately after each set of data that you Import and Clean. It’s important to remember that Access has a 2gb file limit at all times, and so smaller data sets per import will be required as your existing database approaches this limit.

# - On Manually Importing Data

1. Download a current copy of the database template, “Template Pricing DB.mdb”. Move the template to reside in the same directory as your All\_qa file.
2. Rename the database as “CC\_P2.mbd”, “Last\_Dose.mdb”, “Nutritional.mdb”, or “Pharma.mdb”. These must be saved exactly to these names, as the Pricing Template expects them to be this way. Choose the appropriate name for the type of All\_qa files you aggregated.
3. Open the database, click “Enable Content” on the yellow banner.
4. Go to the “External Data” ribbon. Click “Text File”, usually below just right of the ribbon name.
5. When a file browsing window appears, locate and select your All\_qa.txt file. Ensure that “Import the source data into a new table in the current database.” is selected. Click “OK”
6. Ensure “Delimited” is selected, then click “Next”.
7. Ensure “Tab” is selected. Check “First Row Contains Field Names”.
8. Set Text Qualifier to ‘ (single apostrophe), then click “Next”.



1. You must now go through each of the columns potentially being imported. For each of the following columns from the Import Text Wizard window, click them so that their background is black and the text is white. For Mapped Description, “Indexed” should be set to “Yes (Duplicates OK)”. For the following, ‘Do Not Import’ should be checked. (Alt + S = Check “Do Not Import”)

Sap Id, Classification, Notes, Average Quantity (if not Nutra), Minimum Quantity, Minimum Price, Max Quantity, Max Price, Unit of Measure, Mapping Process, User Id, Mapped Date.

1. Click “Next”. Ensure that “Let Access Add Primary Key” is selected, then click “Next” again.
2. Name the new table “All\_qa”. This means that you must delete the date that was added by the All\_qa\_aggregator.bat script. To prepare for import, close any unused applications. When ready, then click “Finish” and wait. This may take some amount of time. There may be a progress bar in the bottom-right of the screen that can help you estimate, however it is not always reliable and the Microsoft Access program may appear to freeze during the import process. It’s advised to be very patient before considering the import failed. The example used in this guide had to process over 2.965 million records to create the database, a very resource-intensive task.

# - On Updating the Import Specification

1. To replace the import specifications in the pricing database because, for instance, a new column was added to the Pit, follow the following steps:

2. The simplest method of updating import specification has you delete the previous import specification, "Import-all\_qa\_aggregated", from Access by clicking 'External Data' -> 'Saved Imports' -> 'Delete'. You may ignore this by giving the new import specification a different name and making additional changes to the ImportData and getImportSpecAndUpdate functions. Specifically, you will need to replace instances of "Import-all\_qa\_aggregated" with your new import specification name, or refactor the code in other ways to allow for different specifications.

3. Follow the steps from the above section for manually importing data as described in the db guide. On the last step, after clicking the finish button and the import completes, check the box which denotes 'Save import steps'

4. Click the textbox for the 'Save As' label that appears asking you for a name for the import you plan to save. It is recommended that you modify the given name to be identical to the original import specification, "Import-all\_qa\_aggregated", however it is not neccessary if appropriate changes are made elsewhere in the file.

5. Open the Saved Imports window again, with 'External Data' -> 'Saved Imports’. Single-click the file path of your new import specification and Ctrl+C copy it.

6. Press Alt+F11 to open the Visual Basic editor. On the upper left panel, locate the tree structure representing the Template Pricing DB. It may appear like "CC\_P2 (Template Pricing DB)". Within the modules folder located underneath the TemplatePricingDB, find the module named "Template Pricing DB - Utilities". Double-click this module.

7. Either scroll down or Ctrl+F and search for the line for the ImportData function. The first line of the function is [ Sub ImportData() ].

8. Within the ImportData function, replace the file path contained in 2 locations of the function with the file path copied from your import specification. It's easiest, typically, to highlight the entire file path without highlight the double quotes and then pressing Ctrl+V to paste. Ensure there are no leading or trailing spaces or other characters between the double quotes, just the file path. If it makes it easier to locate, the original file path at this point is: "C:\Users\dpope\Desktop\Pricing\DB\all\_qa\_aggregated.txt"

9. If you chose the same name for your new import specification, then you are now complete. It's recommended to delete the manually imported table, then proceed to test the automated import function to ensure successful changes were committed.

10. If you chose a different name for your new import specification, then there are more changes to be made. Within the ImportData function, replace "Import-all\_qa\_aggregated" with the name of your new import specification. Be sure to wrap the name in double quotes with no leading or trailing spaces. The line you will be changing currently looks as so: [ DoCmd.RunSavedImportExport "Import-all\_qa\_aggregated" ]

11. Lastly, go the function 'getImportSpecAndUpdate'. The first line of this function is "Function getImportSpecAndUpdate(fileIn As String, fileOut As String)". Update the line containing the original import name to contain your new import name: [ Set allQaImportSpec = CurrentProject.ImportExportSpecifications.Item("Import-all\_qa\_aggregated") ] Specifically, change "Import-all\_qa\_aggregated" to contain your new import name. You are now complete, it is recommended that you test your new import spec by deleting the manual import table and attempting an automated import.

- Typing this into the Immediate Window of the VBA Editor will display your import spec as an XML file.

[ ? CurrentProject.ImportExportSpecifications.Item("Import-all\_qa\_aggregated").XML ]

# - On DB Filters for Removing Bad Data

The cleaning step removes any rows that meet any of these conditions:

* + - * Marked as ImportError in ImportError table
      * Mapped Description Contains “(Bucket)”
      * Marked as “Excluded” in I/E column
      * Mapped Quantity is not equal to Original Quantity
      * Mapped Quantity is less than 1
      * Mapped Quantity is not a whole number
      * APPI is less than or equal to 0
      * APPI is Null

These filters are run one at a time, top-down. So first ImportErrors, then Bucket items, and so on…