How to Normalize Removal Reasons and Wear Condition

Document version 1.1 - November 21, 2012

Starting with EM Track release 3.5.18 there is a new utility available that allows users to normalize or stream line Removal Reasons and Wear Conditions.

A little history

Reason codes and their descriptions are common to the database in use. When you first install a fresh copy of EM Track-III your system is pre-loaded with a set of standard codes. As time progresses users have the option to add their own codes using table maintenance. Although this is a most practical feature, in real life it may have become a nuisance as different users may have added descriptions on their own without first consulting the table to see if a similar description is already present. This resulted in many similar codes, which by itself does not matter, but does cloud some reports as each reason is treated separately.

Things can get even worse when users export and import accounts amongst themselves. When data get imported into a given database then any new reason found in the importing database is merged into the destination database, and when accounts in that database are later exported again, the sum of all codes ever found is sent along with the data.

Assigning of Responsibility

It should be understood that changing these codes, particularly in a multi-location environment, must be well coordinated.

When you are only the recipient of data from other locations and you never send your data back to that originating location then you are free to make any changes you so desire. Whatever changes you make will be retained with that database and any future imports into that database will receive these same changes, no matter where the data come from.

When data flow back and forth amongst many locations it is suggested that only the central location originate changes. Although both sides can make changes, it may be difficult to later find out who made what change if questions arise.

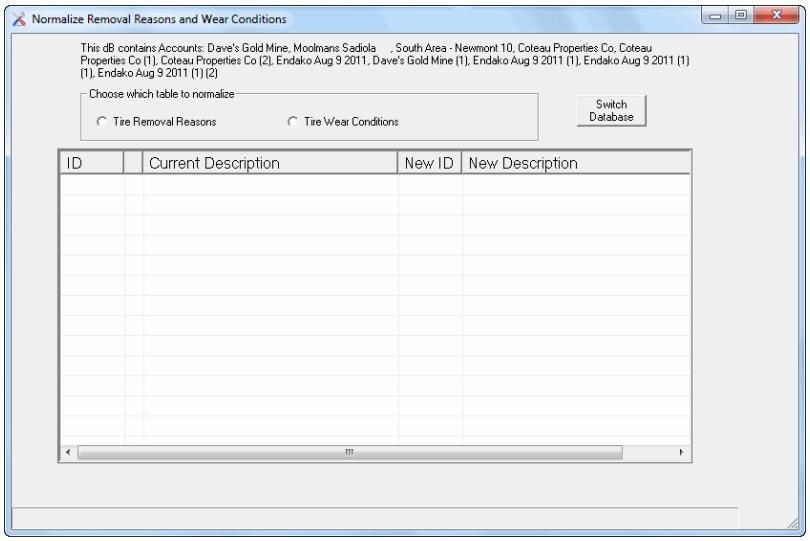
The Solution

Goodyear created a special utility that can clean up your Removal and Wear Condition codes. The program is located in your EM Track-III programs folder, called **NormalizeRemovalAndWearCodes.exe**.

Changes should be grouped into sessions as some changes may only be valid for your site and others may be global. For an explanation of these two versions please read the chapter "Finalizing Changes"

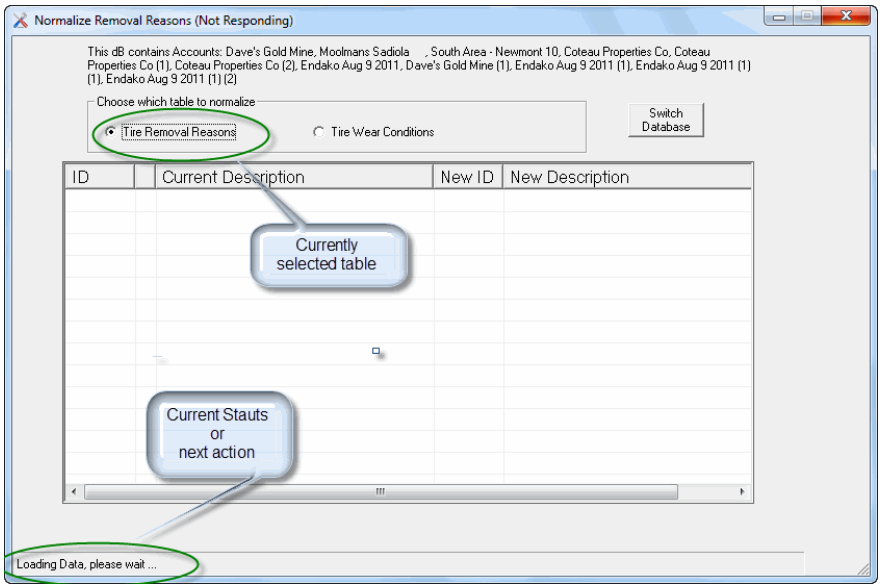
In the following we explain step-by-step the cleanup process.

When you first start the program you see an empty screen similar to this:

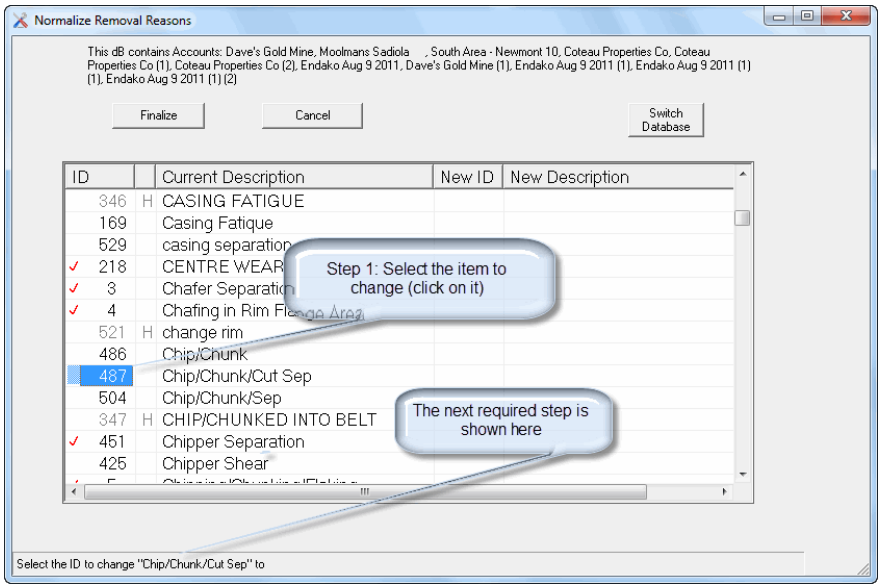


At the top of the screen you will see the account names found in the current database. This is for verification as you may be working with several databases using the "Switch Database" button. When you first start the program your default database is always active.

Next you will be selecting the table to normalize. Once you make the table choice, by clicking on it, the list will get populated. The hourglass tells you that this step is in progress. Depending on the size of your database this may take some time. Your screen will look like this:



Once the data is loaded the screen will look similar to the table maintenance screen:

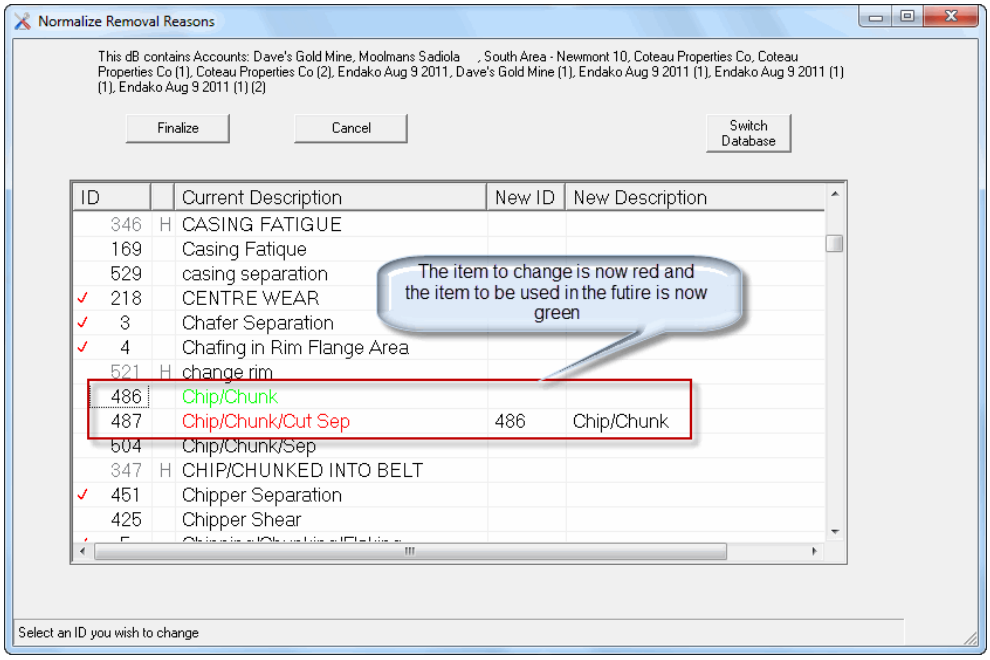


Note the red check-marks in the ID column. A checkmark indicates that this particular code is actually in use somewhere in your database. When there is no checkmark present you have the additional option of deleting this code if you do not want it to ever be used again. See “Special Handling Commands” described later.

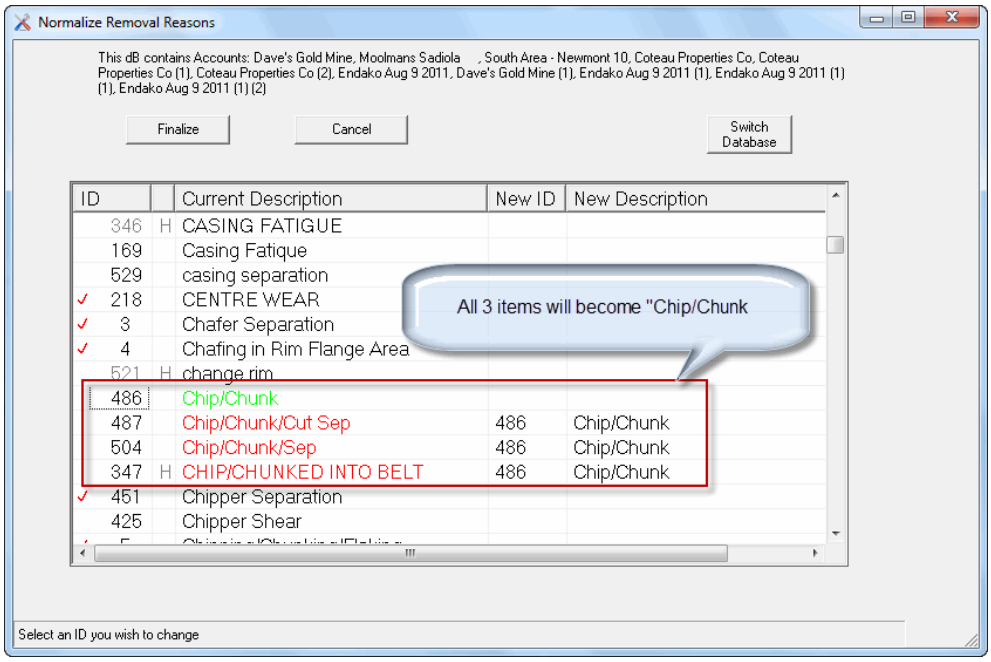
The character “H” in the second column indicates that this item is marked on Hold. Users could not actually use this item. You can manipulate this status as described under “Special Handling Commands” later.

In the example described here we want to merge the three codes 487, 504 and 347 to all become code 486: Chip Chunk, leaving only this one code when we are done.

Each merging, or assigning one code to another, is a two-step process. First you click on the code you want to change, then click on the code you want it to change to. After the second click your screen will look like this:



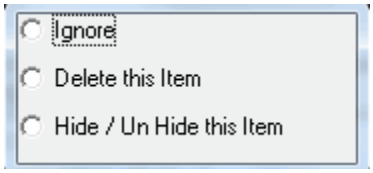
We now repeat this same process with codes 504 and 486, then again with code 347 and 486. The screen will then look like this:



You repeat this as many times and with as many codes as you wish to change.

Special Handling commands

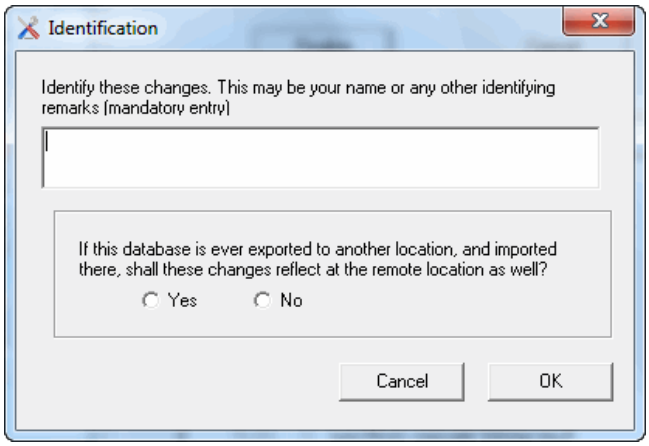
- If you make a mistake you can “Un-Change” an action by clicking on any ID of a red item. This will un-associate that change.
- **Right-clicking** on any ID opens a special option window:



  - Ignore: This closes the option window and will return without any action
  - Delete: If this item is listed and you do not want it to be seen again to the users you may delete it. This action is the same as if you were doing it in table maintenance.
  - Hide / Un Hide: This toggles the Hide status. This action is the same as if you were doing it in table maintenance.

Finalizing Changes

When you are finished making your changes you click the Finalize button. It presents the following screen



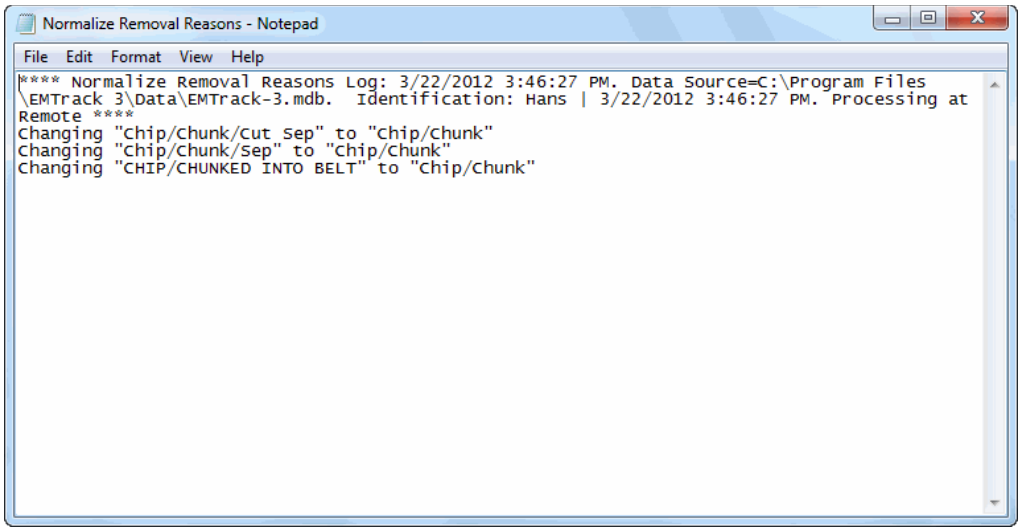
In the field: **Identify these changes** you enter any text that may clarify this set of changes for later reference in the log file. You must enter something in this field.

The next option choice is critical to understand. It affects how these changes will be reflected if accounts in this database are ever exported to another location. If you choose “Yes” then all the changes recorded with this set will also execute at the receiving location after the database is imported there. Normally this is what you should select.

The “No” choice may be used by a home office that consolidates many accounts for consolidated reporting. In such a situation you may normalize many more codes just for yourself so that reports look nice and clean. Data exported to a remote site and imported there will then leave these codes still there so that users may continue to use them. Any such set of changes should be made in a separate session and not done together with other code changes that you want to be everywhere.

After you click the OK button the changes will be written into the database and all references in Tire and Tire History will change to the newly defined codes. A log file is created in your EMTrack3 programs folder and automatically displayed on your screen. This log file is accumulative, meaning that any subsequent changes will be appended to the log so that you have a full history of your changes.

There are actually two (2) log files created, one for Removal Codes (Normalize Removal Codes.txt) and one for Wear Conditions (Normalize Wear Conditins.txt).



What happens at the Remote Site

When accounts from an exported database are imported into another database, then any Removal and Wear changes created at the source location (where the data originate from) will reflect in the receiving database as well, unless the changes were marked not to reflect (see Finalizing Changes).

A report of any changes made is automatically created and displayed. Two (2) log files are located in the EM Track Program folder. One is for Removal Code Changes (Merge RMVCODE Table.txt) and the other is for Wear Condition Changes (Merge WEARCOND Table.txt)

###