Class List of Characteristics:

Changing the landscape of Estimating by eliminating Bias

Introduction

After nearly 30 years in Aerospace Estimating and having been through the pendulum swing from subject matter expert focus to Data Driven I have come to realize that neither by themselves are inherently "right". I think focusing on one without the other is one of the biggest mistakes an estimator can make and CLOC helps bridge the gap by bringing them together and creating a universal language that pays dividends both up and down stream.

What is CLOC?

Simply put, CLOC (Short for Class List of Characteristics) is a coding system wherein each digit of the code indicates one of the cost drivers for the product or activity. The power of CLOC comes from the ability to *"Call a thing by what it really is"* and creating an unhindered universal language that can be correctly utilized for any purpose. The code defines the product essentially and does NOT include Bias like quantity, cost, model, Platform etc.. These things are used in conjunction with the code by the user to do any kind of analysis needed.

With today's estimating environment realizing the importance of data driven estimating, there have been great advancements in the way we analyze data. A large issue however has surfaced from that: "Is it the *right* data?". CLOC gives the ability to bridge the gap between technical knowledge and the analyst. Once the Bias has been removed, the data can be compared apples to apples and the Bias can be re-introduced to place a laser focus on the analysis.

How do you use CLOC?

Imagine everything in your product line has a code. An analyst could sort for like codes and either use historical costs for that code to estimate similar product (that is ACTUALLY similar) or look at the possibility of making several like products into 1 increasing commonality and reducing cost of manufacture. One could also compare techniques needed to manufacture and pair together with a supply base that excels in that field. These are just a few of the uses once the ability to look at a product for what it is exists.

For Example: Coded parts are compared to their Historical Recurring Costs

Class List	Total Setup	Total Run	Hours Per Part	Processing Per Part	Matl. cost per part	NRC	Total Recurring Cost Per Part
M-4-A-3-B-7075-t7351	2.58	22.13	0.82	$ 23.00	$ 126.61	$ 5.50	$ 240.21
M-4-A-3-B-7075-t7351	2.58	22.13	0.82	$ 23.00	$ 126.61	$ 5.50	$ 240.21
M-4-T-4-A-6al4v	2.58	28.54	0.62	$ 7.00	$ 44.80	$ 3.67	$ 120.28
M-4-T-4-A-6al4v	2.58	26.04	0.57	$ 7.00	$ 91.39	$ 3.67	$ 161.37
M-5-T-3-C-6al4v	2.58	128.08	6.53	$ 40.00	$ 3,043.04	$ 5.50	$ 3,801.71
M-5-T-3-C-6al4v	2.58	128.08	6.53	$ 40.00	$ 3,043.04	$ 5.50	$ 3,801.71
M-4-A-4-B-7075-t7351	2.83	18.17	0.53	$ 4.00	$ 21.01	$ 3.67	$ 82.76
M-4-T-4-C-6al4v	4.08	30.63	1.16	$ 20.00	$ 474.37	$ 737.00	$ 621.63
M-4-T-4-C-6al4v	3.58	30.63	1.14	$ 20.00	$ 474.36	$ 737.00	$ 619.79
M-4-T-4-C-6al4v	3.58	30.63	1.14	$ 20.00	$ 403.52	$ 737.00	$ 548.95
M-4-T-4-C-6al4v	3.58	30.63	1.14	$ 20.00	$ 403.53	$ 737.00	$ 548.96
M-4-T-4-C-6al4v	3.58	30.63	1.14	$ 20.00	$ 355.44	$ 737.00	$ 500.87
M-4-T-4-C-6al4v	3.58	30.63	1.14	$ 20.00	$ 355.44	$ 737.00	$ 500.87
M-4-S-3-A-15-5PH	2.58	61.88	1.29	$ 23.00	$ 65.45	$ 372.17	$ 230.26
M-4-S-3-A-15-5PH	2.58	61.88	1.29	$ 23.00	$ 64.05	$ 372.17	$ 228.86
M-4-S-4-A-15-5PH	2.58	33.63	1.21	$ 8.00	$ 16.69	$ 3.67	$ 157.45

Then sorted by code…

Class List	Total Recurring Cost Per Part
M-4-A-3-A-7050-T7451	$ 706.54
M-4-A-3-A-7050-T7451	$ 706.54
M-4-A-3-A-7075-t7351	$ 144.39
M-4-A-3-A-7075-t7351	$ 144.39
M-4-A-3-A-7075-t7351	$ [illegible]
M-4-A-3-A-7075-t7351	$ 57.30
M-4-A-3-A-7075-t7351	$ 40.88
M-4-A-3-A-7075-t7351	$ 40.88
M-4-A-3-A-7075-t7351	$ 33.50
M-4-A-3-A-7075-t7351	$ 33.50
M-4-A-3-A-7075-t7351	$ 33.50
M-4-A-3-A-7075-t7351	$ 33.50
M-4-A-3-A-7075-t7351	$ 30.48
M-4-A-3-A-7075-t7351	$ 30.48
M-4-A-3-A-7075-t7351	$ 30.69
M-4-A-3-A-7075-t7351	$ 30.69
M-4-A-3-A-7075-t7451	$ 94.93
M-4-A-3-A-7050-T7351	$ 56.66
M-4-A-3-A-7050-T7351	$ 56.66
M-4-A-3-A-7050-T7451	$ 36.03
M-4-A-3-A-7050-T7451	$ 36.03
M-4-A-3-A-7050-T7451	$ 39.16
M-4-A-3-A-7050-T7451	$ 39.16
M-4-A-3-A-7050-T7451	$ 58.41
M-4-A-3-A-7050-T7451	$ 58.41
M-4-A-3-A-7050-T7451	$ 46.96
M-4-A-3-A-7050-T7451	$ 46.96
M-4-A-3-A-7050-T7451	$ 46.68
M-4-A-3-A-7050-T7451	$ 46.68
M-4-A-3-A-7050-T7451	$ 42.13
M-4-A-3-A-7050-T7451	$ 48.71
M-4-A-3-A-7050-T7451	$ 48.72

As you can see, similar parts have an avg. cost of ~$60 yet 2 of them are huge outliers. Why is that?

In this particular example the outliers were purchased at a much lower quantity than all of the others. This allowed the estimator to show in a data driven way the impact of order quantities on cost for that type of part.

How does CLOC create a common language?

One of the biggest elephants in the room for an estimator is the language gap when looking for data to support an activity. I have seen large errors so many times due to the Technical folks not understanding what the estimator is looking for or vice versa. Communicating with CLOC coding eliminates that problem and creates a bridge that enables clean communication. Imagine trying to find data on a similar past activity and being able to tell the technical focal exactly what you are looking for and KNOWING that the activity is similar based on the code. This creates a data-driven basis for a subject matter experts opinion.

Adding Value Upstream

Another use of CLOC is at the birth stage of a product. When an Engineer is designing a product, he/she could look at the code for the product. is the code complex? If so then the engineer could look at the complex portions of the code and try to simplify. This would inherently help the engineer "design for cost". Along with that the before after code would tell every stakeholder if the change was increasing or decreasing cost.

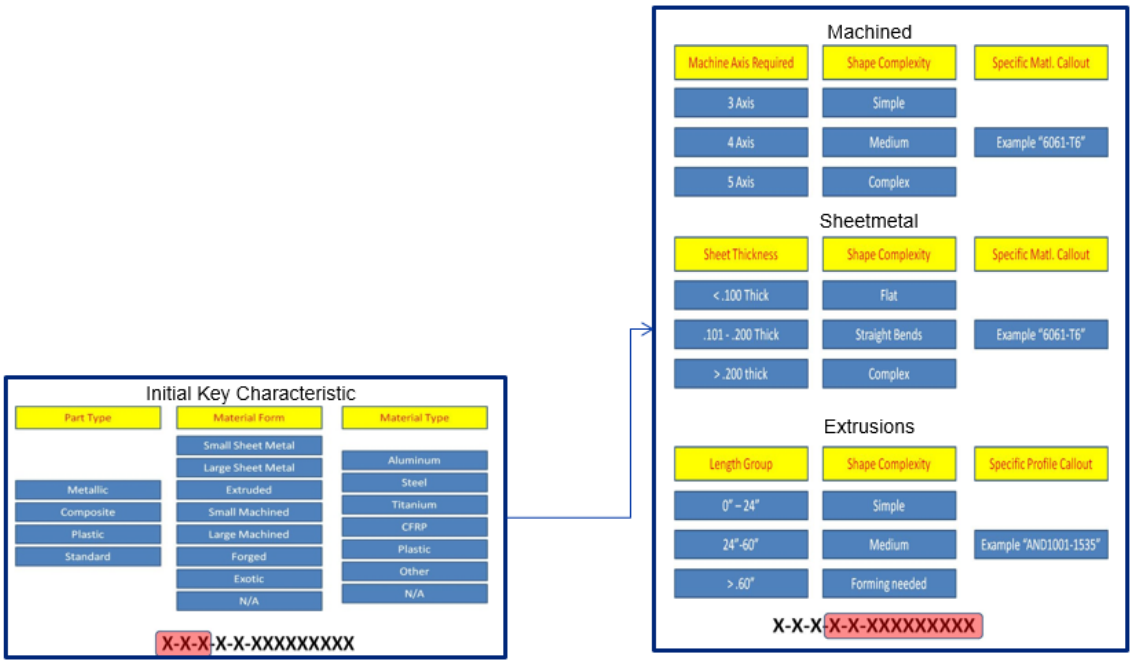
CLOC creates the ultimate CER

With everything coded, an Analyst is not only be able to have infinite ways to create cost estimating relationships, it allows apples to apples looks across competitors and industries opening the door for creative analysis. The data crunching & sorting that can be done with CLOC virtually eliminates the struggle of normalization both educating and empowering the analyst.

Creating a CLOC base code

1. Code Structure

   The structure or taxonomy of the code can be shaped around and largely depends on it's intended usage. The code should have enough "space" to be able to capture the level of detail for ALL product/activity types intended to be captured. The structure should be designed in a left to right "funnel" shape going from highest commodity level to lowest. In the example below of a simple code structure the first 3 digits denote the initial "type" while the second three are specific to the type denoted in the first 3 digits. This is done so that an analyst can sort left to right from high level to low level in an organized manner.



2. Initial code design

When designing the code for each digit successively one must take care to ensure the the “levers” or rules for a product that make it “X” or “Y” are Binary decisions to keep subjectivity out of the mix. For example, a good rule would be “if >3 inches long it shall be code “X”, if 3 inches or less it shall be “Y”” as opposed to “if it is a long part it shall be “X” and if it is short it shall be “Y”” thus making an opinion the deciding factor.

Once all of the cost drivers for a product have been identified and have been deemed to be actual features of the product void of Bias, the drivers should then be sorted left to right in terms of impact.

3. Testing

The code should be tested against historical products and compared/adjusted accordingly in order to create an accurate representation for its intended purpose.

Summary

CLOC is a system that while takes time and effort to create and deploy pays dividends allowing a whole company to benefit and use it. The ability to have everything classified creates a common language that takes data driven activity to a whole new level.