Introduction

As the transportation industry continues to leverage the advancements in software technology made over the past decade, we often lose sight of the core ideals and values that have been critical to the success of our practice for decades. The more that our fingertips become savvy to the clicks of a mouse or the clacks of a keyboard rather than the grind of a pencil, the more we place an abundant amount of trust into software programs that promise to make our lives easier and more efficient. But as engineers, it is easy to forget the power of influence that virtual automation has had on our basic core principles; and as it goes, “with great power comes great responsibility.” Having gone through grade school in the late 90s and high school in the early 00s, I saw firsthand the shift from remembering your times tables to remembering to pack your graphing calculator. That is the current shift that our industry faces today. The reliance on machines to yield the solutions that were once done by hand is taking us by storm. The time it once took to perform the analysis and design of a simple span bridge has been cut by ten-fold. Thus we are now able to analyze much more structure in far less time. With this advancement comes many complex challenges. However, these challenges can be solved with simple solutions that date back to a time before we looked at the computer for answers. Having managed several large-scale, complex finite element models for major bridge projects over the past decade, offered herein are some tips on how to avoid succumbing to the choke hold of our computers and rising above the machines to make sure that we are in control of the software to provide us with the results we desire.

I. Clear Goals & Tangible Expectations

Often, when overseeing a design modeling task, I find that some engineers frequently lose sight of the sole purpose of the problem at hand. These engineers’ likely first step in beginning a design task in front of them is rushing to the computer to scatter nodes and connect beams without first understanding why they are doing so. When building a model in a software program, whether it is a simple span beam or a large-scale finite element model, a clear set of expectations as to what results are going to be produced is crucial. We sometimes find ourselves creating problems instead of solving them when we let the programs take hold of the gray matter that sits between our ears. Having a clear goal as to what the model’s sole purpose is will help us successfully determine the type of model we need, the type of elements it will contain, and the parameters that are critical to define. Clear goals and expectations will prevent falling into the “analysis paralysis” trap, especially with large complex problems.

II. Knowledge of Software

Much like the research executed before buying a new car or house, we must also research the programs in our tool set before fully delving into them. Imagine buying a new pair of shoes before first trying them on; it may be too late before you discover they do not fit. Reading the tips-and-tricks section on the vendor’s website or the fancy marketing brochure you picked up at the last conference is not enough to properly
vet the product. Working closely with the vendor as much as possible so that we can understand not only what the software can do, but to also understand its limitations is critical to the successful implementation of the product. I recently finished work on a three-mile twin river crossing structure. The global analysis of the structure, along with the BIM requirements the contract had, required the team to perform intensive research far in advance of design to make sure the right software was in place to perform the analysis and meet the requirements. Every software product will always have restrictions. Some products will offer what the other doesn’t, and vice-versa. If we are trying to find that perfect, all-encompassing product, we will emerge empty handed. As long as the product meets our key goals and expectations, and we understand its limitations, we can efficiently and successfully deliver innovative solutions. Our answer for this particular project was found at Bentley Systems in an application they offer called RM Bridge. We worked feverishly with Bentley to properly vet RM Bridge to make sure it was the right product. Ultimately, it turned out to be the right solution and we completed the global analysis using RM Bridge.

III. Model Organization

We often believe that building the framework of the model is the first priority and organization of results is the second. This assumption violates the first previously stated rule, which is to make sure we understand the goal of our model and the expectation for our results. Without proper model organization, no clear or direct way exists to ensure that we have efficiency in: erecting the model, troubleshooting issues, and organizing results. Every element of the model needs to have a unique identifier, one in which the user can easily recognize just by looking at the numerical name. On the recent project mentioned above, the team spent weeks developing a unique element labeling approach prior to opening up the software package. This work went hand in hand with working with the Bentley team to understand the best way to approach both erecting and organizing the model. RM Bridge easily lent itself to organizing the elements of the model, which combined with the unique element labeling technique the team developed, helped tremendously with making sure we could effectively manage this large-scale model. One of the most significant decisions to be made before beginning model development is to develop this organization approach. This strategy will vary from project to project but will always yield the same result: effective management of models both big and small.

IV. Core Knowledge

Many times we find ourselves 90 percent complete with a model and realize that the last 10 percent is the hardest part. As the model increases in size and complexity, it becomes more difficult to troubleshoot issues and/or evaluate and understand results. A fine line exists between bright engineers using a model as a tool to strengthen their grasp on a concept and farsighted engineers who are relying on the model to solve all of their questions without a true understanding of the basic principles. We must be confident enough to push the model aside, revisit its original scope, and review what our understanding of the model is. Without a core foundation of the basic principles of
engineering, troubleshooting a model and understanding its results can prove extremely difficult. This paradox yields one simple ideal: utilize the model to amplify our ability to apply the basic principles of engineering to large, complex problems we can’t efficiently complete by hand. We should be using a model only because it will take far too long to analyze it by hand, not because we don’t know the answer, or at least know what the answer could yield. Of course, the model will tell us whether something is going to pass or fail. However, if we lean on the model to tell us the complete solution then we are merely at the mercy of the computer.

The engineering field, like most professions today, is immersed in the utilization of technology to improve efficiency and quality. Most would agree that the knowledge and understanding of such technological advancements offered by finite element analysis software is a necessity to succeed in the 21st century. A strong foundation in the core engineering theories and the confidence to use what you know are both necessary to properly implement this technology and as a result, become a truly relevant, valuable resource to the industry.