I was So Naive When It Came to Buying CFD Software

I saw the ads. CFD that's so simple I can be solving real world fluid/thermal problems in no time. That's the software I want to buy. But will it really do as they say?

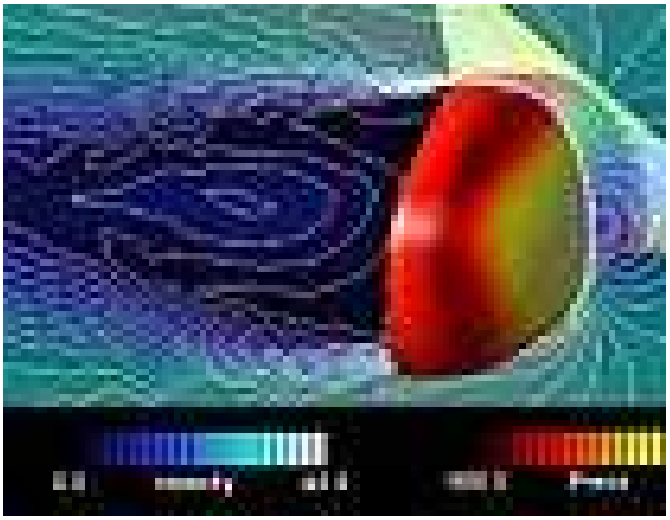
Throughout my career as an engineer, I've had the opportunity to purchase more than a few CFD software packages for different applications and different companies. More than 20 years ago when I made my first purchase, I didn't know much about CFD. I learned almost everything from my first software supplier. Over the years, I accumulated more experience. With more experience, I started to think I knew what was going on when it came to selecting and buying CFD software.

I didn't realize it at the time, but it seemed like no matter what software we used, whether the early versions with primitive interfaces or the latest products on the market today which are very user-friendly, the learning curve was always about the same. We always seemed to have our share of headaches with a few twists here and there. Over time (usually several months) we finally figured out how to make the software do what we wanted. It was only then that we were able to get off and running. This happened so often I thought this was part of the job.

Now I know that my experiences agree with observations made by others. I've recently read, on-line and in magazine articles, about others who have had this same experience.

...as CFD codes become more user friendly, the tendency is to think that you can just sit down and start using them right away to solve your engineering problems.

Namely, it takes a lot longer to become a proficient CFD user than what you initially expect (no matter what the CFD sales people tell you). This warning should be especially heeded today, because as CFD codes become more user friendly, the tendency is to think that you can



just sit down and start using them right away to solve your engineering problems. In fact, more than a few different CFD software companies claim they can have you running their codes with just ½ day of training. It's that easy. The truth is that, yes, you might be able to run their code for a very simple problem with just ½ day of training. But it's more likely that it'll take you weeks, if not months, to get to the point where you have enough expertise to start solving your real engineering problems.

Why is it that it takes so long to get up to speed in spite of all the advances that make CFD software so much easier to use today (better user interfaces, more robust translators, automatic grid generation, solver wizards)? I think there are at least four answers to this question.

1. **Since we can do more, we expect more.** I'd be remiss to imply that the difficulties we have with modeling and simulating today are the same problems we had 20 years ago. They can be related, but the order of complexity is much higher today. For example, 20 years ago I struggled to build a two dimensional model of a gas turbine combustor using a Cartesian grid and then perform a cold flow analysis. Today, the industry builds three dimensional reacting flow models, includes all the injection holes, fuel injection, and nozzle swirl, and performs conjugate heat transfer. The learning curve required to get a final solution 20 years ago (limited by what the software could do then) compared to what can be achieved today may

be close to the same. But the models today are much more sophisticated.

2. **In spite of all the advances that make CFD software easier to use, CFD is still a very, very complicated subject.** Many decisions are required to properly model, set-up, solve, and analyze a problem. Many of these can't be made automatically by the software but require the user dictate the direction. What are the right boundary conditions for your model? Can you take greater advantage of symmetry to reduce the size of your model? How much can you simplify the model geometry without losing accuracy? There's still a lot of knowledge that must be acquired to build good models. And some of this knowledge may only be learned by trial and error.

3. **The CFD software company is probably not familiar with the problem you're trying to solve and will oversimplify any demonstration they show you.** This is a paradox because most of the engineering problems we're all trying to solve are so complicated they require some simplification before they can be reasonably modeled. Your value, as the engineer, is knowing how much to simplify the problem without sacrificing the necessary accuracy. However, the CFD Company probably doesn't have this level of expertise. When they build a demonstration model to show how their software can be used to model your application, they will likely over simplify the problem. When you see their demonstration you may think the software is fully capable. But, in fact, it may be lacking critical capabilities that you may not even realize until you try to solve the real problem.

4. **CFD salespeople oversell the capabilities of the software.** If you're new to CFD you'll be much more susceptible to this happening. But, to be quite honest, even experienced CFD users can be easily swayed by impressive color graphics and simulation video. In addition, many of us are optimistic and are eager to find a solution.

Most CFD experts agree that one of most frustrating things about learning CFD, at least initially for new users, is the amount of time it takes to become reasonably proficient using the software. However, there are ways to get up the learning curve quicker. We'll talk about some of these next month.