RECOMMENDATIONS TO SUCCESFULLY FAIL IN CFD

T. Poinsot

Université de Toulouse

Sometimes you cannot avoid being successful. But in CFD, if you are careful enough, you can make sure to fail totally. Not only for you but also for people working with you. This small text provides rules to make sure that anyone given a large CFD project will be reasonably sure to fail…

Rule 1:

Always change two things at the same time in your code. If you need to change something in the numerical method and some sub model, it is absolutely necessary to do the two changes at the same time. This way, when the code will fail, you will be unable to say why.

Rule 2:

Always avoid having a version of your code that works. Indeed, if your code works, and if you do not apply Rule 1, you can, every time you add something to it, determine right away whether it works or not. Very dangerous path towards success. Same idea if Mr Smith gives you his code: do NOT try to use it as it is to see if it works (why would you waste time on such a stupid test? This Smith has certainly done it). Change two or three models right away, add a few Python or C++ routines of your own before you test the code. By the way, do not forget to tell Mr Smith: 'Gee, your code was so old! How could you work with this ?!'

Rule 3:

Debugging is what CFD is about. 5 minutes to modify your code, 5 months to find why it does not work anymore. The very important thing is here: 'NEVER come back to a previous version!'. The reason why the code does not work now is NOT the lines you changed…. it is the lines you haven't changed yet

Rule 4:

The first three rules have no real interest if you save the versions of your code or worse, if you use CVS or SVN. Only the feeble CFD guys need this kind of safety net. The real artists don't. If you have a copy of your previous version and the present version does not work, you might be unable to apply rule 1 and compare the two versions with a *diff* line. And maybe even to find why the code does not work! On the other hand, if you never save previous versions and apply Rules 1 to 3 with enthusiasm, your codes will, without any doubt, become total messes! After a while, they will not even compile and the best destination for them will be the trashcan. Imagine Mr Smith when you will see him again after two months and say 'Hum, well, could you send me the version of the code you sent me two months ago. I would like to … hum… compare with my own present version…' This stupid Smith, who does not apply the present Rules, will still have a working version and a big smile: at this point, tell him you have multiple new ideas now and your next version will definitely be the best!

Rule 5:

Refuse all simple test cases. If someone asks you to compute a laminar flow in a tube, say: 'no need to do this, we have an analytical solution for this! Why waste CPU time?'. Indeed, you should NEVER compare your code results with analytical solutions. If you do, there will be no excuse for your code if results do not match. Worse, you might improve your code. It is better to avoid this and to stick only to complex turbulent flows where you can say when CFD results do not match experiments (we all agree that this will be the case, no?): 'hum, probably, this is due to the fact that the smallest scales of turbulence were not really isotropic!'.

Rule 6:

CFD guys are very lucky: they can fill any machine by refining the mesh they use. So please do it: never do CFD with small meshes. Start right away with the largest mesh that can fit on the machine. This has MULTIPLE advantages: first, the code will take much more CPU time so that tests will become very difficult, second, this will also slow down the machine so that the other users will also have problems to work. When this happens, find ways to block the others by requesting special batch queues for you so you can completely block the machine with your runs. Explain that you need the whole machine because you are the only one doing serious CFD. If enough users behave this way, you cannot only be sure to fail but also help the whole team to fail! Think also of the discussions at the coffee break when you will tell Smith that he has to wait for you to finish your runs before having access to the computer. Awesome!

Rule 7:

Plot everything in colour, even y=f(x) curves. Never use symbols on your curves. If asked, say 'black and white + symbols is for old guys, colour is everywhere now'. And never add labels or legends or titles. Normally, these nice pictures (especially green and yellow) will look great on your screen. And only on your screen: after photocopy, the curves will disappear. This way, discussions with those who could help you because they understand CFD will be impossible. And it will also avoid publication in journals in cases where you would have forgotten to apply the other rules of this text.

Rule 8:

Never use any debugging tool. Never check anything. If you change something, start from the assumption you did it right the first time and do not check. Do not add new diagnostics in your code (like checking that you conserve global mass or enthalpy): if asked why you do not do it, say 'well, if it was useful, the guys before me would have done it!'. All this information would only help you to understand what your code is really doing. Which you do not need because you know what it should do, isn't it?

Rule 9:

Do not invest money in post processing softwares. Do not write scripts that make post processing fast or simple. Use different zooms and scales on each figure. If possible, never watch the fields produced by your codes. This could show you a big problem at an unexpected place and ways to fix the problem.

Rule 10:

Do not organize the directories where you run the code. Better: use only ONE directory for all files. Do not write anywhere what you do or why or what the results are. The unique directory is great: this way, the results of run N erase or even better get mixed with the results of run N-1, N-2, etc. Finding out what is going on becomes really exciting. And please, do not use names like 'Velocity_at_point_1'. The name A000001 is much better. And imagine how fun it is to have almost the same names for all files. You can also use 'TOTO' as a name but I recommend 'NEW'. This is a good one!

Rule 11:

Never leave your screen. Do NOT read any paper. This is lost time. What is important is your NEXT run because this is the one that will save the day!