

Your tool provider won't tell you that, nor will those fancy methodology books, but verification is not all about mastering technical skills. True, those will help you very much in your daily work but verification is first and foremost TEAM WORK. But not only that, there are several key factors, or qualities if you will, that really distinguish the great verification teams from the good ones. Here there are:

Resourcefulness - This is the single most important quality that a verifier should possess. Verifiers live in a turbulent environment, and if there's one thing certain about verification is that things never go according to plan. This turbulence is caused by two groups of factors - exogenous factors and endogenous factors. Exogenous factors could be vague specifications, evolving requirements and so on. Something that keeps shaking the ground beneath you along the way. Endogenous factors could be debugging complexity, tool problems, and so on. One way or another, these things affect predictability and quite often verification team members get stuck. The path they're on is blocked and they need to find a new path to proceed. Being able to always come up with new approaches and steer the verification ship forward is paramount.
Executive Tip: Try to predict in advance what could go wrong at any stage of the verification process and always have a Plan B.

Excitement - Verifiers, let's face it, don't necessarily enjoy much glory in their work. Their job is often (wrongly) perceived as boring and second in importance (not true either). Being able to constantly generate excitement is a key factor. Different team members find excitement in different things. For instance, some verifiers might find it exciting to reveal bugs. They like the feeling of finding a critical bug in the design, and you can't blame them. Other team members might get a kick out of building complex environments " they enjoy software engineering challenges and once they manage to put together a really nice verification environment they're very satisfied.
Executive Tip: Identify what excites your team members, find out what tasks they're really good at and try to assign each member a matching task, to the extent possible.

The Pareto Principle - As with everything else, the Pareto principle can be applied here as well to optimize the efforts of the verification team. A great verification team knows that 20% of their efforts account for 80% of their achievements. For example “ in Coverage Driven Verification it is a known fact that achieving 80% of the coverage goals is the relatively easy part. Getting to 100% takes much more effort and time.

Executive Tip: The sooner you identify what 20% will generate 80% of your verification goals the better - focus on that!

Planned Mode vs. Combat Mode - Verification projects typically have a work plan. The project is divided into groups of tasks and sub-tasks of various types along the project's time-line. Experience shows that there are stages in the verification process where the team should stick to the plan at all costs. Typical stages “ during the development of verification IP, or during the process of analyzing the Spec and generating verification plans. There might be, however, periods of times when the plan ought to be tossed away and the team should switch to combat mode. Once things start to stabilize again it's time to go back to the Gantt chart and readjust the work plan. A good combat mode example is the regression phase which is usually the last phase of any verification project. In this phase the team is focused on debugging failing tests from the nightly or weekly regression. It's hard to estimate the duration of this phase. In most cases reality dictates schedule. Another example - integrating verification IP or design IP. It doesn't matter how much time such tasks *should* take, in reality those tasks cannot be forced into completion. They need their time. Code integration is therefore a classic combat mode phase where plans should be left aside, replaced by a big to do list.

Executive Tip: Plans are nothing; planning is everything (Dwight. D Eisenhower)

Educate Your Company - Great verification teams not only outperform the good ones, but they also try to educate their colleagues along the way. Why? Because people who have no hands-on verification experience [don't understand it](#) (hey- what's the big deal, just write a bunch of tests and run them. Now how hard could that be right?). Ignorance and wrong perceptions inevitably lead to wrong decision making. The times are changing, and the sooner chip designers and managers realize that verification is much less bug hunting and much more system engineering than they think, the better.

Executive Tip: Make a short presentation on how verification is done today “ concepts, architecture, methodology and tools. Your colleagues will thank you.