Let's say that there's a [reproducible](https://en.wikipedia.org/wiki/Reproducibility) fair test with the following specifications:

1. The variable to be tested is A
2. All the other variables as a set B is controlled to be K
3. When A is set as X, the test result is P
4. When A is set as Y, the test result is Q

Then can you always safely claim that, X and Y must universally lead to P and Q respectively, and A is solely responsible for the difference between P and Q universally?

If you think it's a definite yes, then you're probably oversimplifying [control variables](https://en.wikipedia.org/wiki/Control_variable), because the real answer is this: **When the** [**control variables**](https://en.wikipedia.org/wiki/Control_variable) **are set as K, then X and Y must lead to P and Q respectively.**

Let's show you an example using software engineering(**Test 1**):

1. Let's say that there's a [reproducible](https://en.wikipedia.org/wiki/Reproducibility) fair test about the difference of impacts between procedural, object oriented and functional programming paradigms on the performance of the software engineering teams, with the other variables, like project requirements, available budgets, software engineer competence and experience, software engineering team synergy, etc, controlled to be the same specified constants, and the performance measured as the amount and the importance of the conditions and constraints fulfilled in the project requirements, the budget spent(mainly time), the amount and the severity of unfixed bugs, etc.
2. The result is that, procedural programming always performs the best in all the project requirement fulfillment, budget consumption, with the least amount of bugs, all being the least severe, and the result is reproducible, and this result seems to be scientific right?
3. So can we safely claim that procedural programming always universally performs the best in all those regards? Of course it's absurd to the extreme, but those experiments are indeed [reproducible](https://en.wikipedia.org/wiki/Reproducibility) fair tests, so what's really going on?
4. The answer is simple - The project requirements are always(knowingly or unknowingly) controlled to be those inherently suited for procedural programming, like writing the front end of an easy, simple and small website just for the clients to conveniently fill in some basic forms online(like back when way before things like Google form became a real thing), and the project has to be finished within a very tight time scope.
5. In this case, it's obvious that both object oriented and functional programming would be overkill, because the complexity is tiny enough to be handled by procedural programming directly, and the benefits of both of the former need time to materialize, whereas the tight time scope of the project means that such up front investments are probably not worth it.

If the project's changed to write a 3A game, or a complicated and convoluted full stack cashier and inventory management software for supermarkets, then I'm quite sure that procedural programming won't perform the best, because procedural programming just isn't suitable for writing such software(actually, in reality, **the vast majority of practical projects should be solved using the optimal mix of different paradigms**, but that's beyond the scope of this example).

This example aims to show that, even a [reproducible](https://en.wikipedia.org/wiki/Reproducibility) fair test isn't always accurate when it comes to drawing universal conclusions, because the contexts of that test, which are the [control variables](https://en.wikipedia.org/wiki/Control_variable), also influence the end results, so **the contexts should always be clearly stated when drawing the conclusions, to ensure that those conclusions won't be applied to situations where those conclusions no longer hold.**

Another example can be a [reproducible](https://en.wikipedia.org/wiki/Reproducibility) fair test examining whether proper up front architectural designs(but that doesn't mean it must be [waterfall](https://en.wikipedia.org/wiki/Waterfall_model)) are more productive than counterproductive, or visa versa(**Test 2**):

1. If the test results are that it's more productive than counterproductive, then it still doesn't mean that it's universally applicable, because those project requirements as parts of the [control variables](https://en.wikipedia.org/wiki/Control_variable) can be well-established and being well-known problems with well-known solutions, and there has never been abrupt nor absurd changes to the specifications.
2. Similarly, if the test results are that it's more counterproductive than productive, then it still doesn't mean that it's universally applicable, because those project requirements as parts of the [control variables](https://en.wikipedia.org/wiki/Control_variable) can be highly experimental, incomplete and unclear in nature, meaning that the software engineering team must first quickly explore some possible directions towards the final solution, and perhaps each direction demands a [PoC](https://en.wikipedia.org/wiki/Proof_of_concept) or even a [MVP](https://en.wikipedia.org/wiki/Minimum_viable_product) to be properly evaluated, so proper architectural designs can only be gradually emerged and refined in such cases, especially when the project requirements are constantly adjusted drastically.

**If an universally applicable conclusion has to be reached, then one way to solve this is to make even more fair tests, but with the** [**control variables**](https://en.wikipedia.org/wiki/Control_variable) **set to be different constants, and/or with different variables to be tested, to avoid conclusions that actually just apply to some unstated contexts.**

For instance, in **Test 2**, the project nature as the major part of the [control variables](https://en.wikipedia.org/wiki/Control_variable) can be changed, then one can check if the following new [reproducible](https://en.wikipedia.org/wiki/Reproducibility) fair tests testing the productivity of proper up front architectural designs will have changed results; Or in **Test 1**, the programming paradigm to be used can become a part of the [control variables](https://en.wikipedia.org/wiki/Control_variable), whereas the project nature can become the variable to be tested in the following new [reproducible](https://en.wikipedia.org/wiki/Reproducibility) fair tests.

Of course, that'd mean a hell lot of [reproducible](https://en.wikipedia.org/wiki/Reproducibility) fair tests to be done(and **all those results must be properly integrated, which is itself a very complicated and convoluted matter**), and the difficulties and costs involved likely make the whole thing too infeasible to be done within a realistic budget in the foreseeable future, but it's still better than making some incomplete tests and falsely draw universal conclusions from them, when those conclusions can only be applied to some contexts(and those contexts should be clearly stated).

**Therefore, to be practical while still respectful to the truth, until the software engineering industry can finally perform complete tests that can reliably draw actually universal conclusions, it's better for the practitioners to accept that many of the conclusions there are still just contextual, and it's vital for us to carefully and thoroughly examine our circumstances before applying those situational test results.**

For example, JavaScript(and sometimes even TypeScript), is said to suck very hard, partly because there are simply too many insane quirks, and writing JavaScript is like driving without any traffic rules at all, so it's only natural that we should avoid JavaScript as much as we can right?

However, to a highly devoted, diligent and disciplined JavaScript programmer, JavaScript is one of the few languages that provide the amount of control and freedom that are simply unthinkable in many other programming languages, and such programmers can use them extremely effectively and efficiently, all without causing too much [technical debts](https://en.wikipedia.org/wiki/Technical_debt) that can't be repaid on time(of course, it's only possible when such programmers are very experienced in JavaScript and care a great deal about code qualities and architectural designs).

The difference here is again the underlying context, because those blaming JavaScript might be usually working on large projects(like those way beyond the 10M LoC scale) with large teams(like way beyond 50 members), and it'd be rather hard to have a team with all members being highly devoted, diligent and disciplined, so the amount of control and freedom offered by JavaScript will most likely lead to chaos; Whereas those praising JavaScript might be usually working alone or with a small team(like way less than 10 members) on small projects(like those way less than the 100k LoC scale), and the strict rules imposed by many statically strong typed languages(especially Java with checked exceptions) may just be getting in their way, because those restrictions lead to up front investments, which need time and project scale to manifest their returns, and such time and project scale are usually lacking in small projects worked by small teams, where short-term effectiveness and efficiency is generally more important.

Do note that these opinions, when combined, can also be regarded as [reproducible](https://en.wikipedia.org/wiki/Reproducibility) fair tests, because the amount of coherent and consistent opinions on each side is huge, and many of them won't have the same complaint or compliment when only the languages are changed.

**Therefore, it's normally pointless to totally agree or disagree on a so-called universal conclusion about some aspects on software engineering, and what's truly meaningful instead is to try to figure out the contexts behind those conclusions, assuming that they're not already stated clearly, so we can better know when to apply those conclusions and when to apply some others.**

**Actually, similar phenomenons exist outside of software engineering.**

For instance, let's say there's a test on the relations between the number of observers of a knowingly immoral wrongdoing, and the percentage of them going to help the victims and stop the culprits, with the entire scenes under the watch of surveillance cameras, so those recordings are sampled in large amounts to form [reproducible](https://en.wikipedia.org/wiki/Reproducibility) fair tests.

Now, [some researchers](https://www.newscientist.com/article/2207693-bystander-effect-famous-psychology-result-could-be-completely-wrong/) claim that the results from those samplings are that, the more the observers are out there, the higher the percentage of them going to help the victims and stop the culprits, so can we safely conclude that the [bystander effect](https://en.wikipedia.org/wiki/Bystander_effect) is actually wrong? It at least depends on whether those bystanders knew that those surveillance cameras did exist, because if they did know, then it's possible that those results are affected by [hawthorne effect](https://en.wikipedia.org/wiki/Hawthorne_effect), meaning that the percentage of them going to help the victims and stop the culprits could be much, much lower if there were no surveillance cameras, or they didn't know those surveillance cameras did exist(but that still doesn't mean the [bystander effect](https://en.wikipedia.org/wiki/Bystander_effect) is right, because the truth could be that the percentage of bystanders going to help the victims has little to do with the number of bystanders).

**In this case, the existence of those surveillance cameras is actually a major part of the** [**control variables**](https://en.wikipedia.org/wiki/Control_variable) **in those** [**reproducible**](https://en.wikipedia.org/wiki/Reproducibility) **fair tests, and this can be regarded as an example of the** [**observer's paradox**](https://en.wikipedia.org/wiki/Observer%27s_paradox)(whether this can justify the more and more numbers of surveillance cameras everywhere are beyond the scope of this article).

Of course, this can be rectified, like trying to conceal those surveillance cameras, or finding some highly trained researchers to regularly record places that are likely to have culprits openly hurting victims with a varying number of observers, without those observers knowing the existence of those researchers, but needless to say, these alternatives are just so unpragmatic that no one will really do it, and they'll also pose even greater problems, like serious privacy issues, even if they could be actually implemented.

Another example is that, when I was still a child, I volunteered into a research of the sleep quality of children in my city, and I was asked to sleep in a research center, meaning that my sleeping behaviors will be monitored.

I can still vaguely recall that I ended up sleeping quite poorly at that night, despite the fact that both the facilities(especially the bed and the room) and the personnel there are really nice, while I sleep well most of the time back when I was a child, so such a seemingly strange result was probably because I failed to quickly adapt to a vastly different sleeping environment, regardless of how good that bed in that research center was.

While I can vaguely recall that the full results of the entire study of all children volunteered was far from ideal, **the changes of the sleeping environment still played as a main part of the** [**control variables**](https://en.wikipedia.org/wiki/Control_variable) **in those** [**reproducible**](https://en.wikipedia.org/wiki/Reproducibility) **fair tests**, so I still wonder whether the sleep qualities the children in my city back then were really that subpar.

To mitigate this, those children could have been slept in the research center of many, instead of just 1, nights, in order to eliminate the factor of having to adapt to a new sleeping environment, but of course the cost of such researches to both the researchers and the volunteers(as well as their families) would be prohibitive, and the sleep quality results still might not hold when those child go back to their original sleeping environment.

Another way might be to let parents buy some instruments, with some training, to monitor the sleep qualities of their children in their original sleeping environment, but again, the feasibility of such researches and the willingness of the parents to carry them out would be really great issues.

The last example is the famous [Milgram experiment](https://en.wikipedia.org/wiki/Milgram_experiment), does it really mean most people are so submissive to their perceived authorities when it comes to immoral wrongdoings? There are some problems to be asked, at least including the following:

1. Did they really think the researchers would just let those victims die or have irreversible injuries due to electric shocks? After all, such experiments would likely be highly illegal, or at least highly prone to severe civil claims, meaning that it's only natural for those being researched to doubt the true nature of the experiment.
2. Did those fake electric shocks and fake victims act convincing enough to make the experiment look real? If those being researched figured out that those are just fakes, then the meaning of the whole experiment would be completed changed.
3. Did those being researched(the "teachers") really don't know they're actually the ones being researched? Because if those "students" were really the ones being researched, why would the researchers need extra participants to carry out the experiments(meaning that the participants would wonder the necessity of some of them being "teachers", and why not just make them all "students" instead)?
4. Assuming that the whole "teachers" and "students" things, as well as the electric shocks are real, did those "students" sign some kind of private but legally valid consents proving that they knew they were going to receive real electric shocks when giving wrong answers, and they were willing to face them for the research? If those "teachers" had reasons to believe that this were the case, their behaviors would be really different from those in their real lives.

**In this case, the majority of the** [**control variables**](https://en.wikipedia.org/wiki/Control_variable) **in those** [**reproducible**](https://en.wikipedia.org/wiki/Reproducibility) **fair tests are the test setups themselves, because such experiments would be immoral to the extreme if those being researched truly did immoral wrongdoings, meaning that it'd be inherently hard to properly establish a concrete and strong causation between immoral wrongdoings and some other fixed factors, like the submissions to the authorities.**

Some may say that those being researched did believe that they were performing immoral wrongdoings because of their reactions during the test and the interview afterwards, and those reactions will also manifest when someone does do some knowingly immoral wrongdoings, so the [Milgram experiment](https://en.wikipedia.org/wiki/Milgram_experiment), which is already [reproduced](https://www.sciencedaily.com/releases/2017/03/170314081558.htm), still largely holds.

But let's consider this [thought experiment](https://en.wikipedia.org/wiki/Thought_experiment) - You're asked to play an extremely gore, sadistic and violent VR game with the state of the art audios, immersions and visuals, with some authorities ordering you to kill the most innocent characters with the most brutal means possible in that game, and **I'm quite certain that many of you would have many of the reactions manifested by those being researched in the** [**Milgram experiment**](https://en.wikipedia.org/wiki/Milgram_experiment)**, but that doesn't mean many of you will knowingly perform immoral wrongdoings when being submissive to the authority, because no matter how realistic those actions seem to be, it's still just a game after all.**

The same might hold for [Milgram experiment](https://en.wikipedia.org/wiki/Milgram_experiment) as well, where those being researched did know that the whole thing's just a great fake on one hand, but still manifested reactions that are the same as someone knowingly doing some immoral wrongdoings on the other, because **the fake felt so real that their brains got cheated and showed some real emotions to some extent despite them knowing that it's still just a fake after all, just like real immense emotions being evoked when watching some immensely emotional movies.**

It doesn't mean the [Milgram experiment](https://en.wikipedia.org/wiki/Milgram_experiment) is pointless though, because **it at least proves that being submissive to the perceived or real authorities will make many people do many actions that the latter wouldn't normally do otherwise**, but whether such actions include knowingly immoral wrongdoings might remain inconclusive from the results of that experiment(even if authorities do cause someone to do immoral wrongdoings that won't be done otherwise, it could still be because that someone really doesn't know that they're immoral wrongdoings due to the key information being obscured by the authorities, rather than being submissive to those authorities even though that someone knows that they're immoral wrongdoings).

Therefore, to properly establish a concrete and strong causation between knowingly immoral wrongdoings and submissions to the perceived or real authorities, we might have to investigate actual immoral wrongdoings in real life, and what parts of the perceived or real authorities were playing in those incidents.

**To conclude, those making reproducible fair tests should clearly state their underlying control variables when drawing conclusions when feasible, and those trying to apply those conclusions should be clear on their circumstances to determine whether those conclusions do apply under those situations they're facing, as long as the time needed for such assessments are still practical enough in those cases.**