

2 Pages on One Size Fits All



Introduction

*Why do (software) engineers constantly try to find THE RIGHT™ methodology? Well, for one thing, we want to be as efficient as possible. Nothing wrong with that, but what concerns me is when **we start believing in universal methodologies that simply do not exist**. The reasons why they don't exist are that 1) we have different needs and 2) the environments in which we act are different. Thus, how to organize and work in the best way clearly depends on who we are and where we are. Scrum, eXtreme Programming, RUP, and even that nasty old waterfall; any methodology must never be more than a framework from which we obtain a solid foundation to build from.*

Industrialization

I'll start with the ugly news. One of the best reasons for using ready-made methodologies and really doing them by the book is to **reduce the dependency on competent individuals**. For an organization, this seems to make very good sense – if your people are less important, you become less vulnerable to change (for example, when people get sick, or quit). Also, you can get by with hiring people that aren't super-smart, and thus cheaper.

Mind you, organizations seldom (consciously) enforce the use of certain methodologies for this reason. However, I think it is important to understand that when you are part of an industry that aims to produce **as much value as possible for as little money as possible** (this is obviously how to maximize profit), you are always going to be subjected to optimizations that have little or nothing to do with your individuality.

You, my dear ~~component~~ reader, are constantly being industrialized¹.

¹ Now get back in that line!

Reproducible Success

A much more noble argument for introducing strict adherence to a methodology is to enable **reproducible success**. When you find a good way of running projects, a clever idea is to try and *remember the good things*. You also want to *minimize the bad things* you do. That's what methodologies are about. As long as you remember that **the rationale for enforcing a certain methodology is to be able to reproduce success**, you will probably be fine. But if you start thinking that success is *caused* by the methodology, you're in big trouble. You see, **as long as the work we perform gets done by people, that's also where the success or failure come from**.

Methodology is only a framework that must be tailor-made to fit your people, not the other way around.

The Problem with One Size Fits All

When applying some methodology to a project without analyzing the people in the team, you will find that success is **not** reproduced between different projects and teams. People have different needs, and therefore different methodologies will work better or worse depending on the group. For example, people with authority problems will hate overly structured methodologies. People who need their creative freedom hate excessive specifications. People who don't want to make their own decisions will become paralyzed if there is no hierarchy.

The only reason I can see for anyone to try and apply a methodology without considering the people involved is to reproduce a *sufficiently efficient organization*² – totally disregarding the needs of the constituent individuals. You seek the ability to **clone** success. That thought was popularized by the military hundreds of years ago, and companies like McDonald's are happily reusing the idea. Of course, with this philosophy people are reduced to expendable and easily replaced parts. (Personally, I don't buy the idea that organizations are machines.)

Luckily, for almost all non-trivial (organizational) problems, applying a methodology by brute force is an algorithmically pathetic attempt at finding a solution. It might work, but it's not very efficient. And it's boring.

Individualization

A really cool (and effective) way of applying methodologies is taking into account the people involved, and clearly communicating that the **methodology is a framework that must be shaped and tuned**. Now, when you give the group/team/project the **power to self-organize** within the loose boundaries of a framework, group dynamics help shape the methodology to *fit the needs of individuals*. But be careful; this only works in functioning teams!

I always try to make sure that the needs of each individual are met when it comes to communication, specifications, meeting etiquette, feedback loops, and so forth, i.e., things that define the group's rules, values, and protocol. Then, I take into account the requirements from management and other stakeholders to find out what's missing from the group. Simply put, **individualization is the art of letting your people bend the rules**.

Thank you for reading,

Bjorn Karlsson

² One that is profitable and scales well.