



March 17, 2021

# Key Metrics

## PART 3: PRODUCTIVITY

*Written by: Kirk Hoaglund, Chief Executive Officer*

I've written about the challenges of using well-known business metrics which are trailing indicators. Developing forward-looking metrics can be a key in effective management of projects and of your enterprise. I've written about several that we use such as "[average days late](#)", "[revenue per headcount](#)", and "[churn](#)". This time I'll write about a class of metrics that are predictive of productivity.

Productivity measures can be very dependent on what your business does. In the days when I was still writing software, we measured "lines of code per day". A truck driver might be measured by miles driven per day. Perhaps these sound reasonable on the surface, but are they? A driver could get lost and drive many miles while not approaching his desired destination. Good software is not always complex software. Lots of lines of code might be the opposite of what you want.

There are ways to separate from the specifics of the job, to gather metrics that are predictive of productivity. One we use to great effect is Estimate Variance.

Regardless of what outcome a project is producing, that project involves people completing work over time. Projects are planned and timelines are formed through estimation of that effort. Upon examining the things that must be done, the doers make a claim about how much effort will be required to complete the work. The only way to create accurate estimates is to break the work down into easily understood parts with a well-defined proper outcome that can be accomplished within a relatively short time.

When this is done properly, the resulting components will, over time, converge on a common size with small variance. For instance, an experienced team will often describe components in such a way that none require more than one day to complete. We see this behavior with our teams all the time. The best teams have the smallest amount of variance in estimated sizing. A large variance indicates a poor understanding of the components and will always lead to reduced productivity for a team.

Successful and productive teams will begin to self-police using this mechanism. And they do it almost unconsciously. Team members will call out estimates that fall outside the range that has become the norm for their team.

For instance, in agile methods, estimation is often begun by assigning “story points” to a “user story”. A user story represents a statement of requirement for a project. Story points are unitless numbers that define the relative effort size for meeting the needs expressed in the user story. They are unitless on purpose and are designed to serve as relative sizing comparison points. Commonly the sizing numbers follow the Fibonacci sequence of 1, 2, 3, 5, 8, 13, 21, 36, and so on. Also by design, the difference between the numbers increases as the size increases. This carries a sort of built uncertainty factor: the bigger the number the less confidence it shows.

A new team might apply a wide range of story points in their estimation exercise. Some 3, some 13, some 36. Over time, however, these teams begin to converge on a much smaller set. For many of our teams, it is as simple as 3, 5, or 8. You could say “small”, “medium”, and “large”. A user story assigned a 13, 21, or 36 score is one the team has not yet fully analyzed and does not yet understand it well enough to decompose it into executable chunks. When the team is confidently and routinely assigning 3, 5, or 8, they developed the habit of requirement decomposition, their estimates are much more accurate, and their productivity reaches its peak.

For us, we know that a team has gelled and is humming at top efficiency when their story sizing consistently produces a narrow range of numbers that are not too big and not too small. Life is good.

CONTACT US



***“Good software is not always complex software. Lots of lines of code might be the opposite of what you want.”***

*- Kirk Hoaglund*