

# Development Velocity Is a Bogus Metric That Needs to Die

Seth Purcell, CTO, Signpost  
Georgian Partners Portfolio Conference  
October 2018

## Annoying Fact

There is **no** practical way to measure software development productivity in the wild.

Sorry.

# Measuring Software Development Productivity

[www.construx.com](http://www.construx.com)



@stevemconstrux



Steve  
McConnell

# What is Development Velocity?

“Velocity is the average **amount of work** a scrum team completes **during a sprint**, measured in either **story points or hours**, and is very **useful for forecasting**.”

## Problem 1

# The Name

## Problem 2

# The Definition

Productivity =

Output / Input

# Dimensional Analysis

A. Story points *per sprint*

$\Rightarrow$  points/time

B. Hours *per sprint*

$\Rightarrow$  time/time!



## Problem 3

# The (Ab)use

Found on the Web:

“Velocity is a key feedback mechanism for the Team. It helps them measure whether process changes they make are improving their productivity or hurting it.”

from <https://www.scruminc.com/velocity/>

# Getting Bonkers Now

“While a Team's velocity will oscillate from Sprint to Sprint, over time, a well-functioning Scrum Team's velocity should steadily trend upward by roughly 10% each Sprint.”

(same source)

# Recap

- We have a real management need
- With no corresponding measure
- But with a metric that looks *a lot* like a measure...

# The Core Problem



“If you can’t measure it,  
you can’t manage it.”



You can’t measure  
dev productivity.



“I’m gonna need you  
to maximize our dev  
team’s productivity.”

## Pick Two.

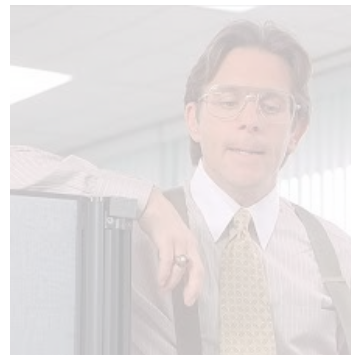
# The Core Problem



“If you can’t measure it,  
you can’t manage it.”



You can’t measure  
dev productivity.



“I’m gonna need you  
to maximize our dev  
team’s productivity.”

## 1. The Hands-Off Approach

# The Core Problem



“If you can’t measure it,  
you can’t manage it.”



You can’t measure  
dev productivity.



“I’m gonna need you  
to maximize our dev  
team’s productivity.”

## 2. The Velocity = Productivity Approach

# The Core Problem



“If you can’t measure it,  
you can’t manage it.”



You can’t measure  
dev productivity.



“I’m gonna need you  
to maximize our dev  
team’s productivity.”

## 3. The Right Approach



## The Solution

We can't actually measure developer  
and dev team productivity –  
but we can *judge* it.

# The Actual Quote!



**“It is wrong to suppose that if you can’t measure it, you can’t manage it — a costly myth”**

## What Does This Mean in Practice?

“Study after study has shown that motivation probably has a larger effect on productivity and quality than any other factor” [McConnell, 3]

“Organizational influences define the productivity envelope within which a team performs.” [McConnell, 4]

# Takeaways

- Velocity does *not* measure productivity
- Because *nothing* measures productivity
- But it's commonly abused as a measure of productivity
- Owing to its name, its definition, and a painful need
- So it needs to die and be replaced by something else
- But we can *judge* productivity and manage it effectively by focusing on efficiency and motivation

# Thank you!

Please send feedback to [spurcell@signpost.com](mailto:spurcell@signpost.com)

## References

1. <https://www.atlassian.com/agile/project-management/metrics>
2. Steve McConnell. *Measuring Software Development Productivity*. ACM Talk.
3. Steve McConnell. *Rapid Development*. Microsoft Press, 1996.
4. Steve McConnell, *Making Software*. O'Reilly, 2010.