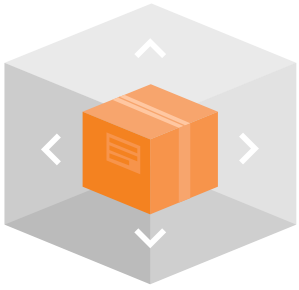
# THE IMPACT OF AGILE
# QUANTIFIED

## SWAPPING INTUITION FOR INSIGHT®

## KEY FINDINGS TO IMPROVE YOUR SOFTWARE DELIVERY

EXTRACTED BY LOOKING AT REAL, NON-ATTRIBUTABLE DATA FROM 9,629 TEAMS USING THE RALLY PLATFORM

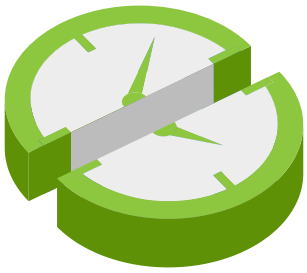


### DOUBLE YOUR PRODUCTIVITY

PAGE 5


### IMPROVE QUALITY BY 250%

PAGE 8



### CUT TIME TO MARKET IN HALF

PAGE 10

### BALANCE YOUR TEAM PERFORMANCE

PAGE 12

Larry Maccherone with Mark Smith and Michael Dellanoce. Contributions from Jennifer Maccherone, Kevin Chabreck, and Eric Willeke.

Special thanks to the Software Engineering Institute (SEI) at Carnegie Mellon, especially Jim McCurley and Sarah Sheard. Finally, thank you to the Rally teams who were big supporters of this work.


Go Agile. Go Rally.®

## **Larry Maccherone**
### Director of Research and Analytics

Larry Maccherone is the Director of Analytics for Rally Software. He has been with Rally since 2009, first as an Agile coach and then as a Product Owner of the analytics team based in our Raleigh office. Prior to his current role, Larry served as a software process coach focusing primarily on Scrum and the SEI's Team Software Process (TSP). He obtained qualifications and certifications in a number of software process and quality practices including the SEI's Personal Software Process (PSP) instructor, TSP launch coach, CMMI Assessor, ISO-9000, Rational Unified Process, and Certified Scrum Master (CSM).

Prior to Rally, Larry served as Chief Engineer and CEO of Comprehensive Computer Solutions, a systems integrator for factory floor automation, and was founder of QualTrax, software for measurement and management for ISO-9000 and other standards compliance.

Larry is currently finishing work on his Ph. D. in Software Engineering at Carnegie Mellon. His research focuses on Agile measurement, analysis, and visualization for software and systems engineering.

# Background

## About the Findings

Though people have made Agile recommendations for many years, we have never been able to say how accurate they actually are, or how much impact a particular recommendation might make.

The findings in this document were extracted by looking at non-attributable data from 9,629 teams using Rally's Agile Application Lifecycle Management (ALM) platform. Rally is in the unique position to mine this wealth of SaaS (cloud-based) data, and uncover metrics-driven insights.

These insights give you real-world numbers to make an economic case for getting the resources you need, and getting your people to commit to change. That's the underlying motivation of this work.

"These insights give you real-world numbers to make an economic case for getting the resources you need, and getting your people to commit to change.

### The Software Development Performance Index

The SDPI framework includes a balanced set of outcome measures. These fall along the dimensions of Responsiveness[A5], Quality[A6], Productivity[A7], Predictability[A8], as well as softer aspects such as employee engagement, customer satisfaction, and what we think of as a "Build-the-right-thing" metric.

You will eventually be able to use the entire SDPI framework to get feedback on your own teams and organization. In this document, we share useful insights based on analysis of existing data utilizing the first four SDPI dimensions – the ones we can extract automatically from our data set.

## Correlation: Not Necessarily Causation

The findings in this document are extracted by looking for correlation between "decisions" or behaviors (keeping teams stable, setting your team sizes to between 5 and 9, keeping your Work in Process (WiP) low, etc.) and outcomes as measured by the dimensions of the SDPI. As long as the correlations meet certain statistical requirements[1] we report them here. However, correlation does not necessarily mean causation. For example, just because we show that teams with low average WiP have ¼ as many defects as teams with high WiP, doesn't necessarily mean that if you lower your WiP, you'll reduce your defect density to ¼ of what it is now. The effect may be partially or wholly related to some other underlying mechanism.

## This Is Just the Beginning

Further research is underway to add an additional 30 decisions and aspects of context to the analysis. Our goal is to gather enough information about the context under which particular relationships hold to build a predictive regression model for success with Agile projects.

## About These Four Dimensions of Performance

### Responsiveness[A5]

Based on Time in Process (or Time to Market). The amount of time that a work item spends in process.

### Quality[A6]

Based on defect density. The count of defects divided by man days.

### Productivity[A7]

Based on Throughput / Team Size. The count of user stories and defects completed in a given time period.

### Predictability[A8]

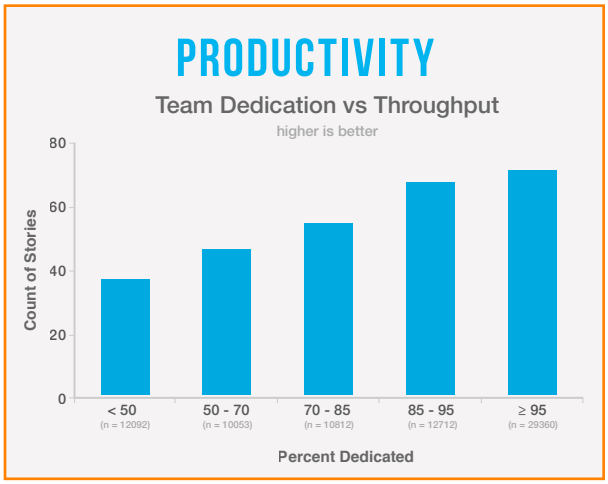
Based on throughput variability. The standard deviation of throughput for a given team over 3 monthly periods divided by the average of the throughput for those same 3 months.

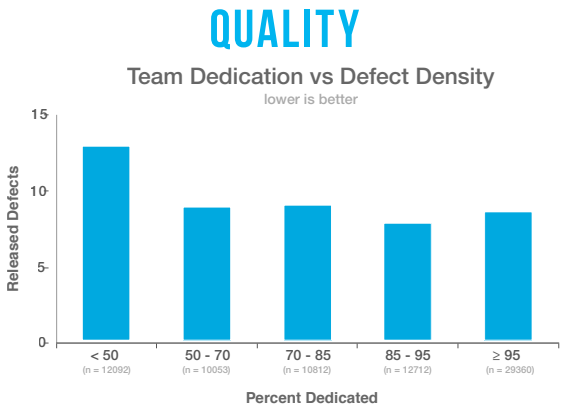
  

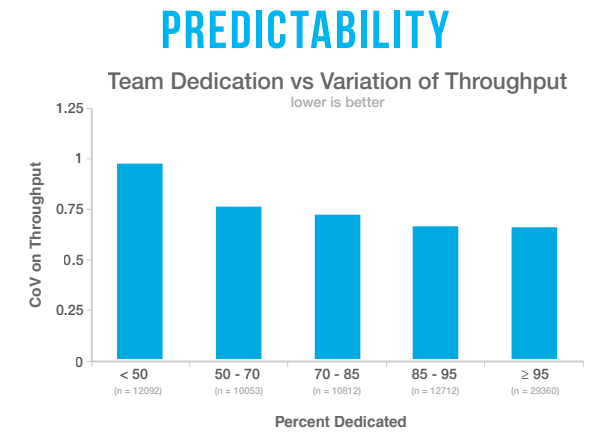
# DOUBLE YOUR PRODUCTIVITY

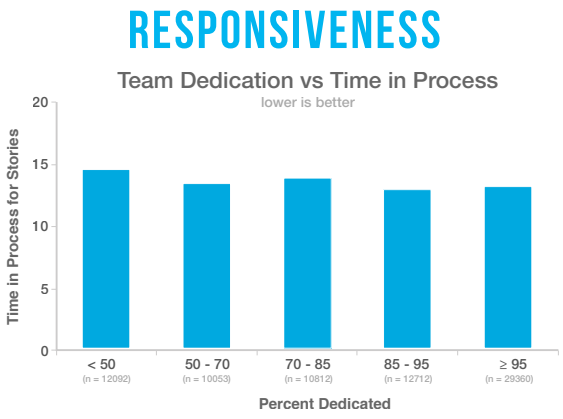
If people are dedicated to only one team rather than multiple teams or projects, they stay focused and get more done, leading to better performance. But which aspect of performance is impacted most?

The answer is Productivity. We can see that there is almost a 2:1 difference in throughput between teams that are 95% or more dedicated compared with teams that are 50% or less dedicated. Dedicating people to one team also has an impact on Predictability and Quality, but mostly in the extreme. You can see from the charts showing the variability of throughput and defect density, the effect is most prominent for the < 50% dedicated group.



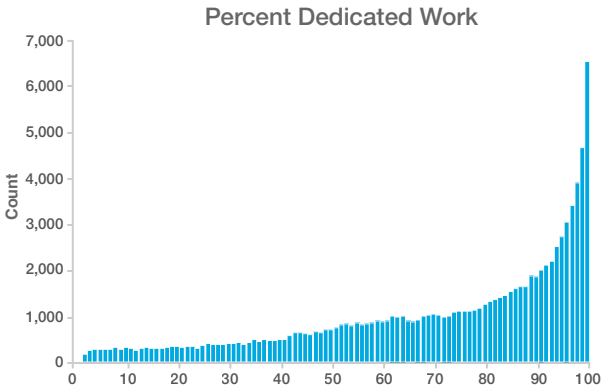



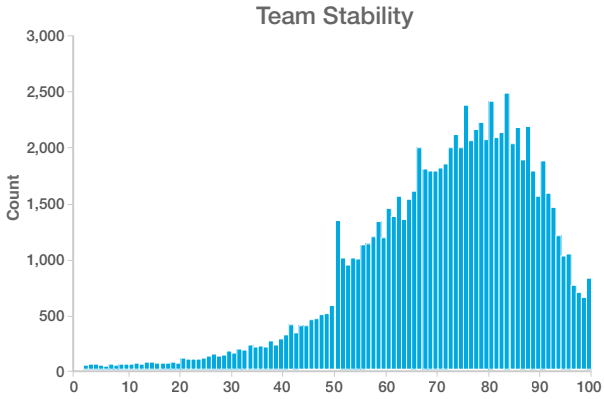
> We can see that there is almost a 2:1 difference in throughput between teams that are 95% or more dedicated compared with teams that are 50% or less dedicated.

On a positive note, the recommendation that we dedicate people to one team is widely followed. You can see in the histogram that the highest spike is in the far right. This is the count of the number of team-quarters where 99% or better of the work was done by people who are dedicated to this one team. The next bar to the left is the 98%-99% group and it's the second highest. This histogram shows that we are consistently dedicating people to one team.




People are mostly dedicated to one team

However, the story is not so good for the Agile recommendation of keeping teams stable. The stability metric measures how many of the team members stay the same from one quarter to the next. This histogram shows that very few teams actually have 100% stability. The median of this data is at 74.8% which means that roughly 1 out of 4 people on these teams change every 3 months. Teams are very unstable.




One out of four team members change every three months

## Key Findings

Stable teams result in up to:

- 60% better Productivity
- 40% better Predictability
- 60% better Responsiveness

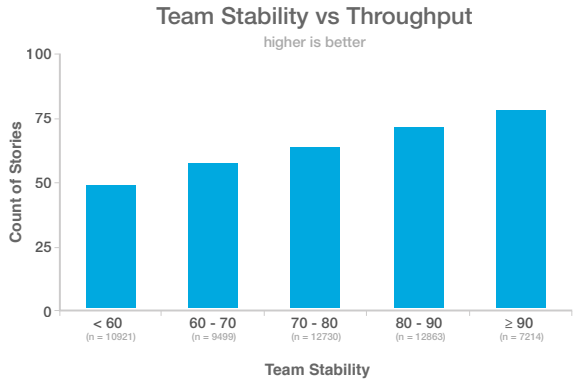
  

Having unstable teams is hurting performance, which makes sense. If we shift the teams around, we have to train new team members. While we are ramping them up, we're not getting work done. Again, Productivity (throughput effect of up to 60%) is most impacted. But Predictability (variability of throughput effect of up to 40%) and Responsiveness (time in process effect of up to 60%) also show a significant effect.

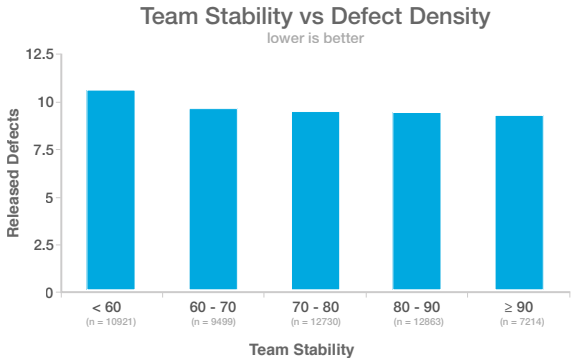
## Recommendations

- Dedicate people to a single team
- Keep teams intact and stable

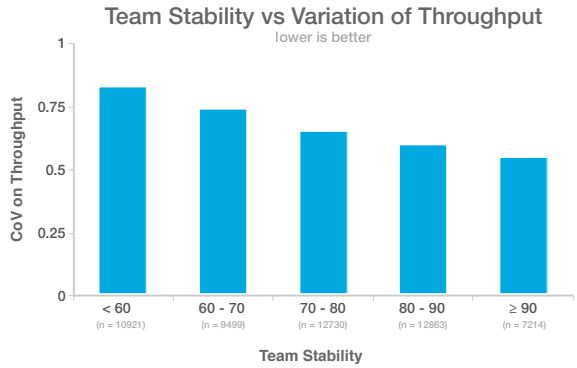
## PRODUCTIVITY



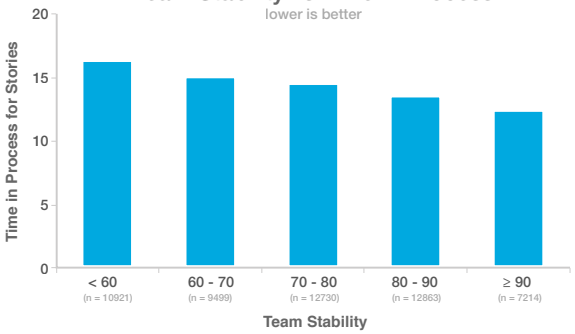

## QUALITY




## PREDICTABILITY




## RESPONSIVENESS




# IMPROVE QUALITY BY 250%

We looked at teams that followed four different estimating processes. The first group, which is only 3% of our teams, did no estimating even though 90% or more of their work was put into iterations.

The second group is doing Full Scrum. They are consistently putting story points on their stories before adding them to iterations, and they are also consistently breaking those stories down into tasks and making task hour estimates. This group represents the vast majority of our teams: 79%.

The third group we have labeled Lightweight Scrum and it represents 10% of the teams in the study. Some Agile coaches suggest that mature teams may be able to skip task breakdown and task hour estimating without hurting performance. Let's see if the data bears this out.

The fourth and last group is teams that are not doing story point estimation but are doing task hour estimates. They do all of their estimating in hours. We were a bit surprised to see that 8% of the teams in the study were doing this because we know of no Agile coaches who recommend this process. We believe that

> "... teams that follow the Full Scrum process perform better than most alternatives but Lightweight Scrum is actually better overall.

| Process Type | % of Teams |
|---|---|
| No Estimates | 3% |
| Full Scrum<br>Story points and task hours | 79% |
| Lightweight Scrum<br>Story points only | 10% |
| Hour-oriented<br>Task hours only | 8% |

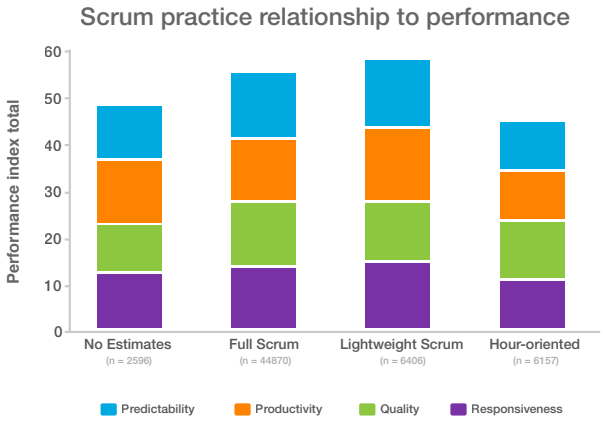
## Key Findings

- Teams doing Full Scrum have 250% better Quality than teams doing no estimating
- Lightweight Scrum performs better overall, with better Productivity, Predictability and Responsiveness

these are teams that have come from a pre-Agile world and started using Rally with little or no coaching. They did their estimates in hours before they started using Rally and that's what they are used to.

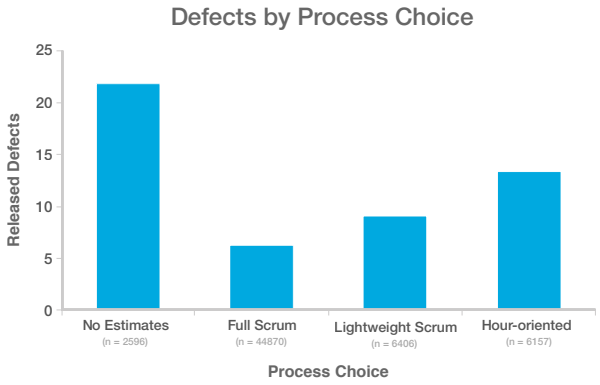
What we found when we compared these various process choices, is that teams that follow the Full Scrum process perform better than most alternatives but Lightweight Scrum is actually better overall. This chart shows a "score" for each of the four dimensions added together.[A2]




It's interesting to note that the group that we believe has received the least coaching (task-hour estimates only) performs the worst and the coaching recommendation for mature teams (Lightweight Scrum) performs best.

There is one dimension where Full Scrum outperforms Lightweight Scrum, and that is the dimension of Quality. There is a 250% difference in defect density between the best and worst process choices so that's pretty dramatic.[1]

## QUALITY




## Recommendations

- Experienced teams may get best results from Lightweight Scrum
- If new to Agile, or focused strongest on Quality, choose Full Scrum

# CUT TIME TO MARKET IN HALF

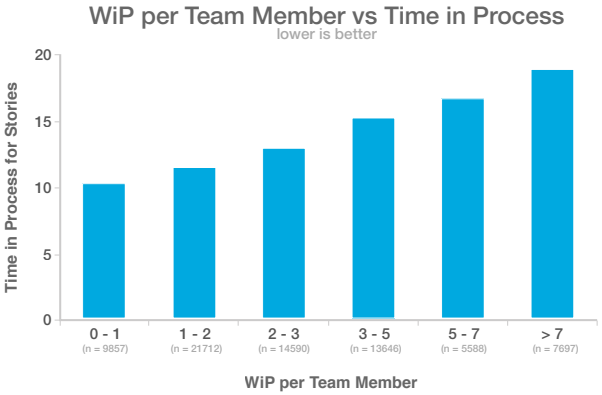
Coaches tell you that lower WiP is always better. Is that really true?

Work in Process (or WiP) is the measure of the number of simultaneous work items that are “In process” at the same time.

Let’s look at the relationship of WiP per team member and Time In Process. The group on the far left is very good at controlling their WiP. They have, on average, less than 1 work item per team member in process. The group on the far right is not controlling WiP very well at all. They have 7 or more work items per team member in process at the same time. So a team of 5 would have a WiP of 35 or more.

Queuing theory (Little’s Law in particular) predicts that there will be a linear relationship between WiP and Time in Process (TiP), and sure enough we see these results. The Time in Process for teams that poorly control their WiP is up to two times as long as teams that control their WiP very well. This makes intuitive sense. The more focused you are on a few things, the quicker you’ll get each one done.

## RESPONSIVENESS




Fewer things in process means that each gets done faster

## Key Findings

Teams that aggressively control WiP:

- Cut time in process in half
- Have ¼ as many defects
- But have 34% lower Productivity

We discovered a huge effect on Quality for teams that have low WiP. Teams with the lowest WiP have 4 times better Quality than teams with the highest WiP.[1]

Queuing theory also predicts that if you lower WiP too much, you'll have a negative impact on Productivity. This too makes sense. If some work gets blocked, there is not enough other work to do. The two groups here on the left have pushed their WiP so low they have negatively impacted their throughput. In fact, teams with very low WiP have 34% lower Productivity.

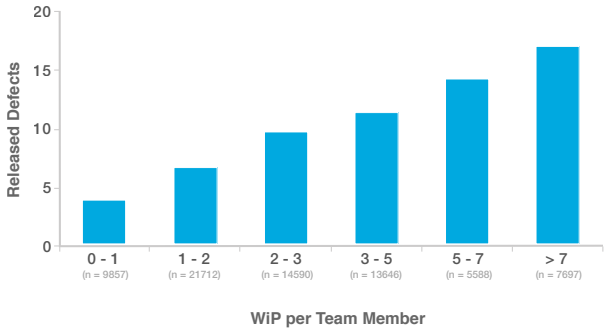
In summary, if your WiP is already high, then by all means drive it lower. However, if your WiP is already low, consider your economic model before you decide to drive it lower. If you're at risk for missing a market window, then drive your WiP as low as possible by focusing on just a few things. But if Productivity is the primary driver of your economic model, don't push your WiP to extremely low levels because if work gets blocked, you won't have any Productivity.

## Recommendations

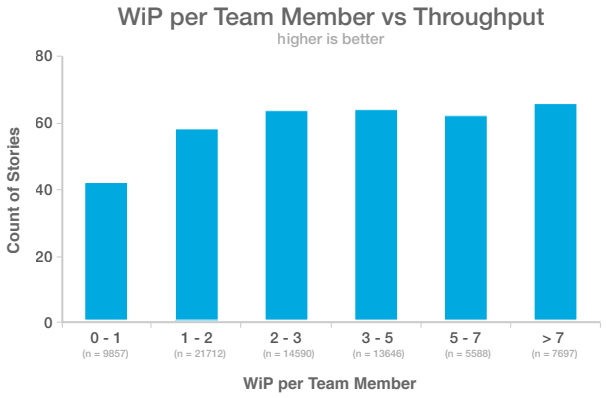
- If your WiP is high, reduce it
- If your WiP is already low, consider your economic drivers
  - If Productivity drives your bottom line, don't push WiP too low
  - If time to market drives your bottom line, push WiP as low as it will go

> "Teams with the lowest WiP have 4 times better Quality than teams with the highest WiP.[1]

### QUALITY

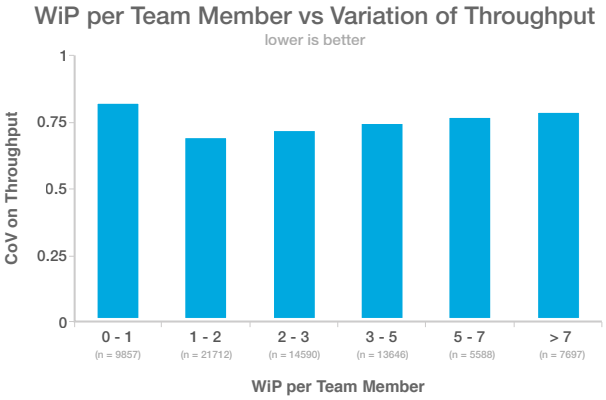



### PRODUCTIVITY




More work in process = more defects.

### PREDICTABILITY

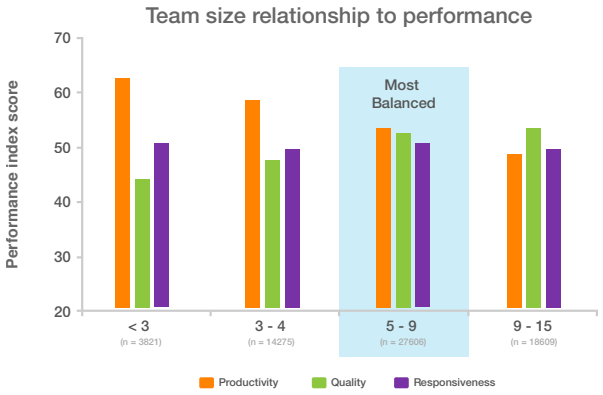


# BALANCE YOUR TEAM PERFORMANCE

Agile recommends that the ideal team size is 7 ±2. How ideal is this when we actually look at the data?

Teams that are smaller than the recommended size tend to have better Productivity, but also tend to have worse Quality. There is little effect on Responsiveness.




## Does Organization Size Matter?

Yes and no. It turns out that organizations of different sizes tend to make different choices. Smaller organizations tend to have a higher proportion of smaller teams, which makes sense.

Larger organizations tend to choose Full Scrum more than smaller organizations. These choices explain most of the differences we see in the variation in performance between larger and smaller organizations.

## Key Findings

Small teams (of 1-3 people) have

- 17% lower Quality
- But 17% more Productivity

Than teams of the recommended size (5-9)

## Recommendations

- Set up team size of 7±2 people for the most balanced performance
- If you are doing well with larger teams, there's no evidence that you need to change

# Support Smart Decisions across Your Enterprise

## Rally's Agile & Lean Metrics Design Workshop Helps You:

- Co-craft an optimal measurement strategy, data mining, rollout, and training plan
- Build a plan for evaluating Measurements, Insights, Decisions, and Outcomes
- Configure Rally to generate and display dashboards and visualizations

**Start Today.** Agile and Lean practices can produce 4X improvements in productivity, quality, value, and time to market if you also adopt a coordinated Agile and Lean measurement strategy.

# Learn More

www.rallydev.com/metrics-workshop

## End notes

1. Our data set is made up from the change records recorded from users working in the Rally application lifecycle management platform. We conduct this research by extracting three types of higher-level measurements from these low-level change records: 1) context (example: is this “Project” entity a real team or meta-team or a project), 2) decisions or behaviors (team size, estimating process, WiP, etc.), and 3) outcomes (the SDPI dimensions from defect density, time in process, etc.). We then use a technique called “analysis of variation” (ANOVA) to determine if differences in the mean of an outcome measurement (type 3) for various alternative decisions (type 2) under a particular context (type 1) are statistically significant. If the ANOVA p-value is less than 5%, then it is highly unlikely that the effect of the decision upon the outcome is due to chance, and as long as a few other requirements are met, the finding is included in this report. We have such a large data set that in most cases the p-value for these findings is much less than 1% indicating a very low likelihood that the finding is from chance.

2. In course of doing this research, we had two particularly dramatic findings, both correlations with high Quality, which led us to wonder if the correlation we were seeing was not causal: 1) the correlation of high Quality with the process choice of Full Scrum, and 2) the correlation of high Quality with low WiP. One plausible theory is that there could be some underlying mechanism (high discipline, for example) that both leads to higher Quality and causes people to choose Full Scrum and low WiP. However, if this were true, then we should see a correlation between the choice of Full Scrum and low WiP, so we took a look at that. We did not find a strong correlation. So, while it’s still possible that there is some underlying mechanism that causes both high Quality and choosing Full Scrum... and another underlying mechanism that causes both high Quality and low WiP, the lack of a correlation between low WiP and Full Scrum is evidence that it is not the same underlying mechanism. That largely rules out, “high discipline” as the underlying mechanism for both findings.

## Appendix: Useful Definitions

1. **Time Buckets**
   Each metric is calculated for a particular time bucket. The charts in this document are all shown for time periods of 3 months in length.

2. **Percentile Scoring**
   Each raw metric has a unique distribution and for some metrics higher is better whereas lower is better for others. To make it easier to interpret the metric and enable the aggregation of dissimilar units into a single index, raw metrics are converted into a percentile score across the entire distribution of all similar metrics. Higher is always better for percentiles.

3. **Calculating the Index**
   The SDPI is made up of several dimensions. Each raw metric is percentile scored and one or more of those are averaged to make up a particular dimension. To calculate the overall SDPI, we take the average of the contributing dimensions' scores.

4. **Team Size**
   We heuristically extract the team membership by looking at who is working on what items and who is the owner of those work items, along with which Rally project/team those work items are in. We then determine what fraction of each team member's time is dedicated to each team. The team size is the sum of these fractions.

5. **Responsiveness Score from Time in Process (TiP)**
   Time in Process (TiP) is the amount of time (in fractional days) that a work item spends in a particular "state." Weekends, holidays, and non-work hours are not counted. We attribute a work item to the bucket where it left that state. You can think of this as the time bucket where work completed. We then take the median TiP of all the work items in that time bucket. While other parameters are possible, we only look at the TiP of user stories and we define "in Process" as ScheduleState equals "In-Progress" or "Completed."

6. **Quality Score from Defect Density**
   Defect density is the count of defects divided by man days, where man days is team size times the number of workdays in that time bucket. This results in a metric that represents the number of defects per team member per workday.

   We look at both the defects found in production as well as those found in test and other areas as indicated by the "Environment" field in Rally. We sense whether or

not defects are typically being recorded in Rally for each of these types for each team over a time period and only use it if it passes this test. We'll take either as the Quality score or the average of the two if both are reliably recorded.

7. **Productivity Score from Throughput / Team Size**
   Throughput is simply the count of user stories and defects completed in a given time period. The Productivity score is the percentile scoring of this Throughput normalized by the team size. While defects are shown in the drill down charts, currently only user stories contribute to the Productivity score.

8. **Predictability Score from Throughput Variability**
   Throughput variability is the standard deviation of throughput for a given team over 3 monthly periods divided by the average of the throughput for those same 3 months. This is referred to as the Coefficient of Variance (CoV) of throughput. Again, we only look at user stories for this Predictability score.