

Table of Contents

Prateek Singh - Retrospective: Moving From a Subjective to Objective World	1
Premise	1
Summary	1
Action / Learning	1
Presentation	2
Notes	2

Prateek Singh - Retrospective: Moving From a Subjective to Objective World

Premise

Traditionally Agile teams run retrospectives to facilitate learning, development, collaboration and improvement of the team. Teams have tried multiple different formats to get the right outcomes. Issues of all sorts seem to pop up many times - lack of participation, finger pointing, irrelevant topics, lack of prioritization of issues. Are these or similar problems hindering the effectiveness of your retrospectives? What format of retrospective can best promote the culture of collective vision and collaboration in your team? How can we get the best outcomes from the team by having the most collaborative retrospectives? Recently numerous teams have moved to objective retrospectives from a completely subjective form of retrospective as a solution. Objective retrospectives use the team's data to conduct the retrospective, to create talking points and to see the effects of action items taken by the team since the last retrospective. We will go over examples of patterns seen in metrics and graphs to assist with identification of talking points and examples of resultant action items. We will also do a deeper dive into facilitation of techniques for facilitating both subjective and objective retrospectives. We will also talk through figuring out the best format to choose for

Learning Outcomes: Attendees are expected to learn which retrospective techniques work best for teams of different maturity levels. They will be able to pick up subtleties of different types of subjective retrospectives. They will be guided towards the best formats of retrospectives for the particular nuances of their team. They will gain a generic knowledge of how to look at scatterplots and Cumulative Flow Diagrams and deduce information from these. Examples of patterns to look for and the lessons that have been learned by other teams when they have looked at these patterns will also be presented as a starting guide to Objective retrospectives. This would help attendees both conduct and participate effectively in whichever style of retrospective the team chooses. The target audience is the leaders of teams that are looking for effective retrospective styles, with participants of these retrospectives being a secondary audience.

Summary

- Content rating (0-no new ideas, 5 - a new ideas/approach, 9-new ideas): 3
- Style rating (0-average presentation, 5 - my level, 9-I learned something about presenting): 3

Action / Learning

- Write about this in comparison to Scott Downey? This is another approach to making sure we have

the data we need for retrospectives.

Presentation

Attachments: [Retrospectives.pdf](#)

Speakers Prateek Singh - Manager of Software Engineering, Ultimate Software @singhpr

Notes

Post technical - recovering developer

Traditional retrospective Good, not so well, what better - works well in ideal conditions

Sometimes not work Not talk about tough stuff Rose colored glasses Negative nelly Kindergarten syndrome - blame Lack of measurable outcomes / impacts Wasted coaching opportunity

New voice in the room - team data Quantitative Have things changed since last time

Provide context to the team itself

Trad metrics Velocity Burnup Can be used - eg why so much work remaining,

Not enough

More tools - cumulative flow diagrams Work item count Process state

Look for bulges Looks out of the ordinary What happened here Represents disruption in the flow of work

Look for flat area Probably mean didn't get anything done Why How do we stop this happening again

Eg Goal was to go twice as fast but this team was already high performance, so not expected to double Most time spent in development done Now operating twice as fast as before

What did they do Put strict limit on development done column - have to help test Start doing more automated testing

Can even apply to high performing

Cfd for ideal scrum team (is this for the ideal team?) Gain the productivity in the areas of complete What mods can we make

More tools - scatter plots

Number of days it takes to get something done Can talk about each and every item

Can be done at the project level How did we do overall? Plot all teams Look at things like how long something takes

Why is it taking longer (happened a couple of times) Not limiting wip Have a queue for testing - because there was dependent stories (test couldn't start on one item without starting another)

Their example has lot like story leakage one sprint to the next

Thruput went up by about 20%

Dependent story Don't create them Do one to completion before starting dependent item

Started to get complacent. A couple of outliers

When you have variability screws up entire system Instability propagates

More than just graphs

PO saw velocity anomaly 3 pt stories taking longer than 5 pt stories

Followup Note identified issues and behaviors Ident actions items avoid repeat actions Look at data directly to see if changes happened in right direction.

A lot of teams are analytical / shy Data helps for these people We have to make this look better Helps with creating targets

Team members should remember blocks as they do story

Is it enough just to have data Probably not - need "what else do we need to talk about?" So mix approaches

3 types of retrospective Autocratic - sm says what we work on - not good except at beginning Guided - elicit tops, and add in topics Team centric -

[Conference](#), [Collaboration](#), [Retrospective](#)

From:
<http://www.hanssamios.com/dokuwiki/> - **Focussed Agile / Hans Samios Wiki**

Permanent link:
http://www.hanssamios.com/dokuwiki/prateek_singh_-_retrospective_moving_from_a_subjective_to_objective_world

Last update: **2016/07/03 13:38**

