Our Estimates are Terrible!
Hans Samios
Who Am I?

• Agile Coach / Scrum Master at Intergraph SG&I
  – Responsible for the rollout of Scrum / agile to Intergraph
  – Previously Certified Scrum Master and Professional

• Contact:
  – Email: hans.samios@intergraph.com
  – Twitter: hanssamios
  – Blog: hanssamios.com
We Run 60 Teams in 8 Countries with Scrum

PRODUCT BACKLOG → SPRINT PLANNING → SPRINT BACKLOG

ScrumAlliance

2-4 WEEK SPRINT

POTENTIALLY SHIPPABLE PRODUCT INCREMENT
Sprint Planning 1
• Preliminary Plan based on story points estimates and velocity history

We have hours left over so we need more stories!

Sprint Planning 2
• Committed Plan based on hour estimates and total team capacity

Starting list of stories to work on

Result: Over-commitment!
Product Development Process / ... / How do we improve our point based estimates and resultant velocity?

**What estimation practices can we try?**

Added by Samios, Hans-peter, last edited by Samios, Hans-peter on Jul 14, 2014 (view change)

Here is a list, in no particular order of things that someone has tried or suggested as a way of improving something about the estimation process:

18 Child Pages

- Approach - Establish a "done" definition for a story to be READY for the team
- Approach - Establish a Keystone User Story.
- Approach - Establish known size to known definition of done for all work
- Approach - Establish points guidelines for each point value
- Approach - Establish things that matter matrix
- Approach - Give default estimates for classes of work and work exceptions
- Approach - If it involves work, then it should have a non-zero estimate
- Approach - Just because Jira doesn't track information does not mean we should not figure out a way to track it
- Approach - Only use Fibonacci sequence numbers
- Approach - Review all the 5s to see that the represent the same relative size of work
- Approach - Split Every Story to a Common Small Relative Size
- Approach - Story Points represent reality based on current knowledge of the relative size of the work
- Approach - Team has right to reject stories for estimation that are not ready
24 times where we estimated a 3 point story and it actually was a 3 point story.
BOOTLEGGERS

Hand crafted in small patches since 2012

180 proof software
I've had a number of conversations with teams recently where people start to realize that their estimates are systematically off. But do we really have a problem?

One team, the Bootleggers, were concerned enough that they wanted to analyze the issue. So, @Jackson, Jeanne G (Jen) prepared a data set of estimates and results. The chart shows the estimate provided by the team and the subsequent actual story points recorded.

If this was "Gaussian" (normal) distribution, the number of items below -100 should actually have 26 cases (or 13%).

Interestingly, majority of the errors (42%) band around 0 relative error (i.e., a lot of estimates are exactly right). Implication we have long tail distribution.

Long tails deprive systems of consistent predictability; Waterfall (serial workflows) are more robust in such situations.

Long tails imply we have a "complex" system (cynefin) where appropriate action is probably not to try and "fix" the estimate, but to utilize the insights gained from the errors.
Questions?