User Stories

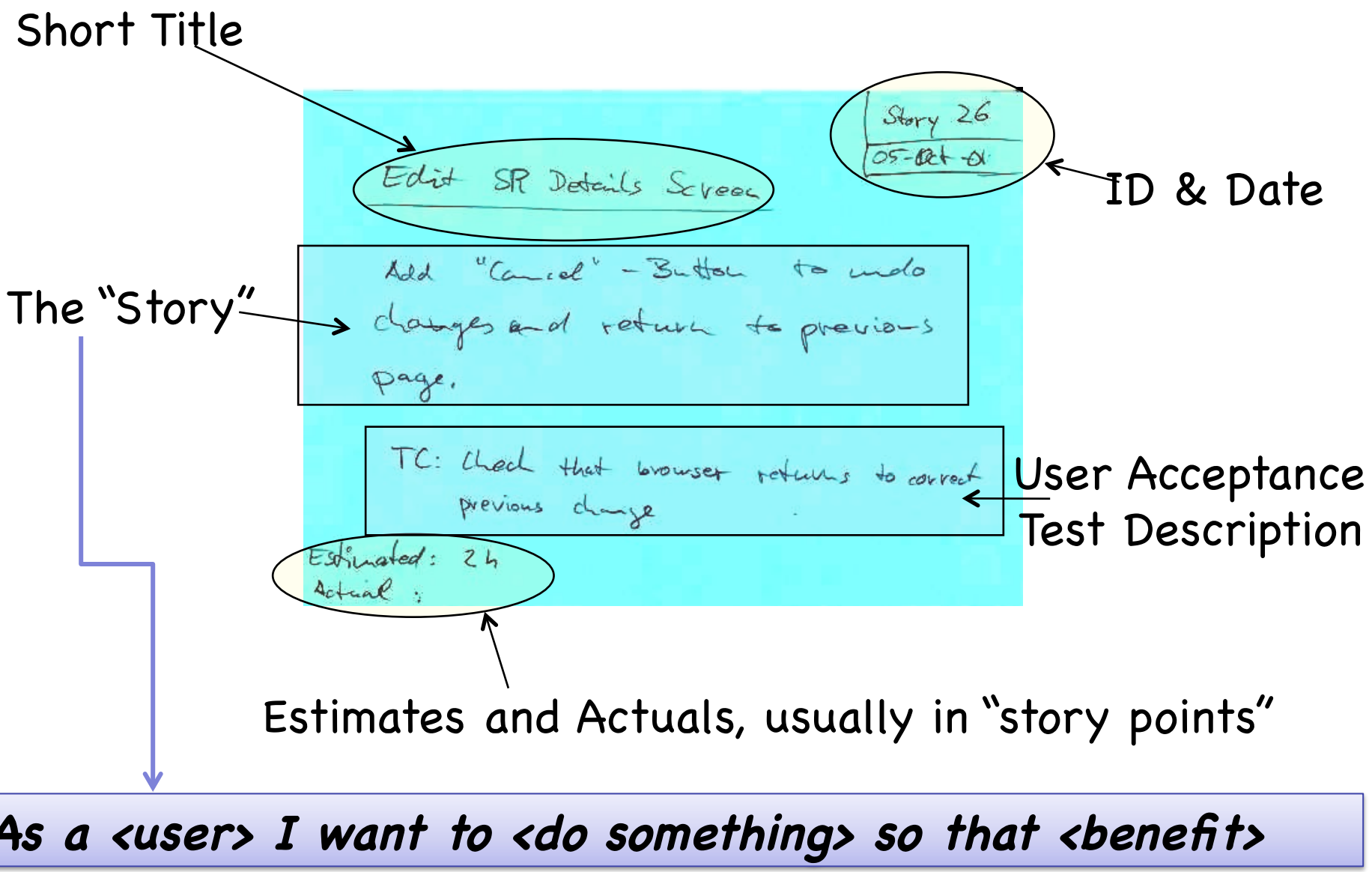
User Stories

What are User Stories?

- Written on 4 x 6 index cards
  - If your card is too crammed, consider breaking out detail into additional cards with sub-stories
- Focus on capturing a conversation
  - User Stories are meant to be torn up
  - “Give the user what they need, not what was written down”
- Broaden by identifying User Roles & Acceptance Tests
  - User Roles: Who is the conversation with?
    - What are their skillsets? Proficiencies with the system? etc.
  - Acceptance Tests: How do I know what done looks like?
    - User Stories intentionally at a high-level
    - Tests allow developers to know when they are done

“User Stories are a placeholder for a conversation”

Anatomy of a User Story



User Story Best Practices

Ron Jeffries 3 Cs*:

1. Card: "token representing the requirement"
2. Conversation: "*exchange* of thoughts, opinions, and feelings"
3. Confirmation: through acceptance tests

*http://ronjeffries.com/xprog/articles/expcardconversationconfirmation/

How do you get the User Story? *Have the conversation*

- Use proper elicitation techniques: Interviews, Surveys, Workshops, …
  - *"Elicitation"* a bad word to XP folks, implies user is "hiding" requirements from us
  - *"Trawling for requirements"* – Using the right size net, the proper requirements may be "sifted out"
- Ask the right questions
  - Open-ended and Context-free

Don't lead or constrain the user, or point toward an *a priori* solution. Let her/him tell a story!

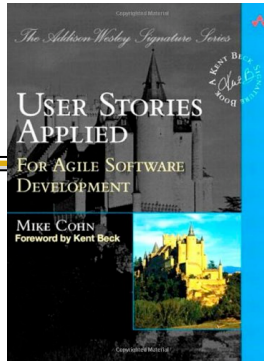
The INVEST Heuristics

Characteristics of good User Stories *(INVEST [Wake 2003])*:

- **I**ndependent: does not depend on other cards
- **N**egotiable: Biz and Dev will negotiate over these
- **V**aluable: to Users, Biz, or Dev. To somebody!
- **E**stimate-able: Dev can assign IET to it
- **S**mall: large stories are complex and incomplete, hiding assumptions. Consider splitting if it is too large
- **T**estable: goal is to define acceptance tests

IMHO, last one particularly important, as it provides a signoff point with the user when they come back and see what you have done

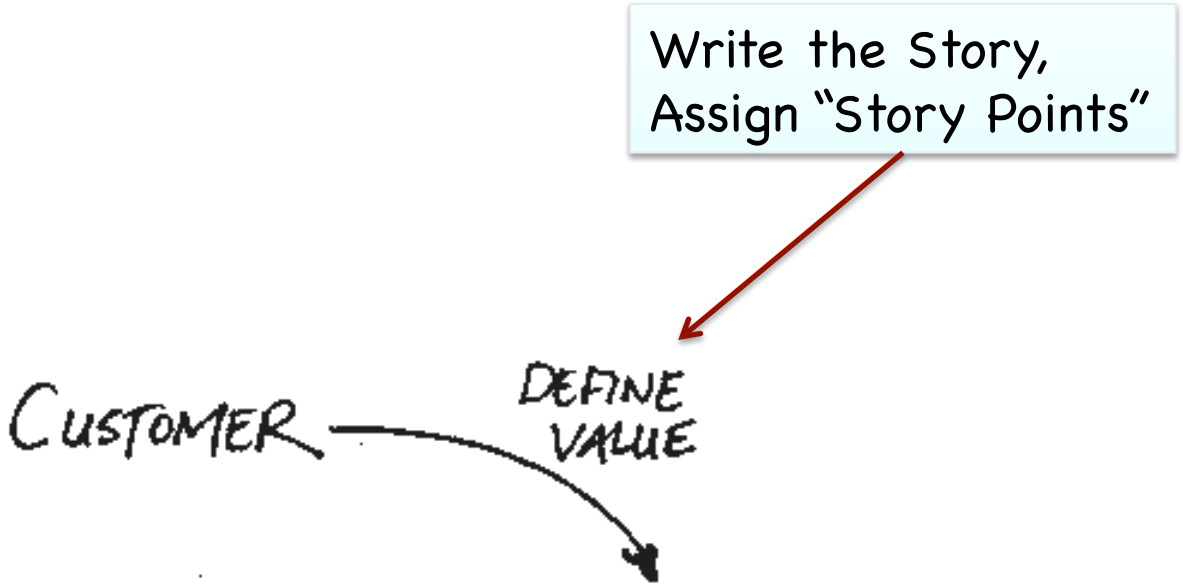
User Story Best Practices



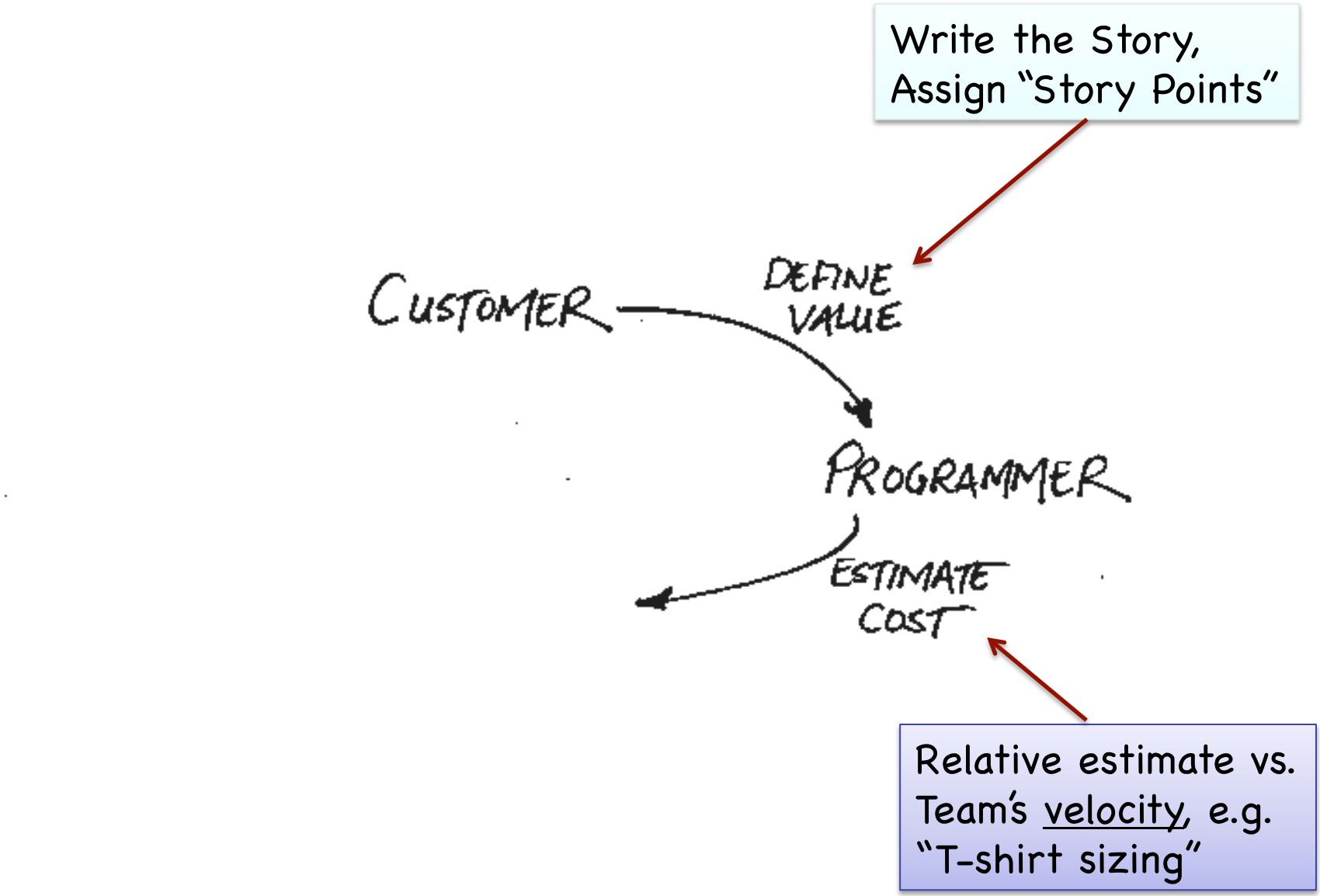
Guidelines for Writing Good User Stories (Cohn, 2004)

1. Start with Goal Stories
2. Slice the Cake – end-to-end functionality; resist desire to decompose
3. Put Constraints on Cards – analogous to UC “biz rules” or NF reqs
4. Size the Story to the Horizon – More detail the closer to impl
5. Keep the UI out as long as possible
6. Some things aren’t Stories
7. Include User Roles in Stories – helps specificity
8. Write for One User – not to a user “class”; eliminates ambiguity
9. Write in Active Voice – e.g. “A Job Seeker can post a resume”, not “A resume can be posted by a Job Seeker”
10. Customer Writes

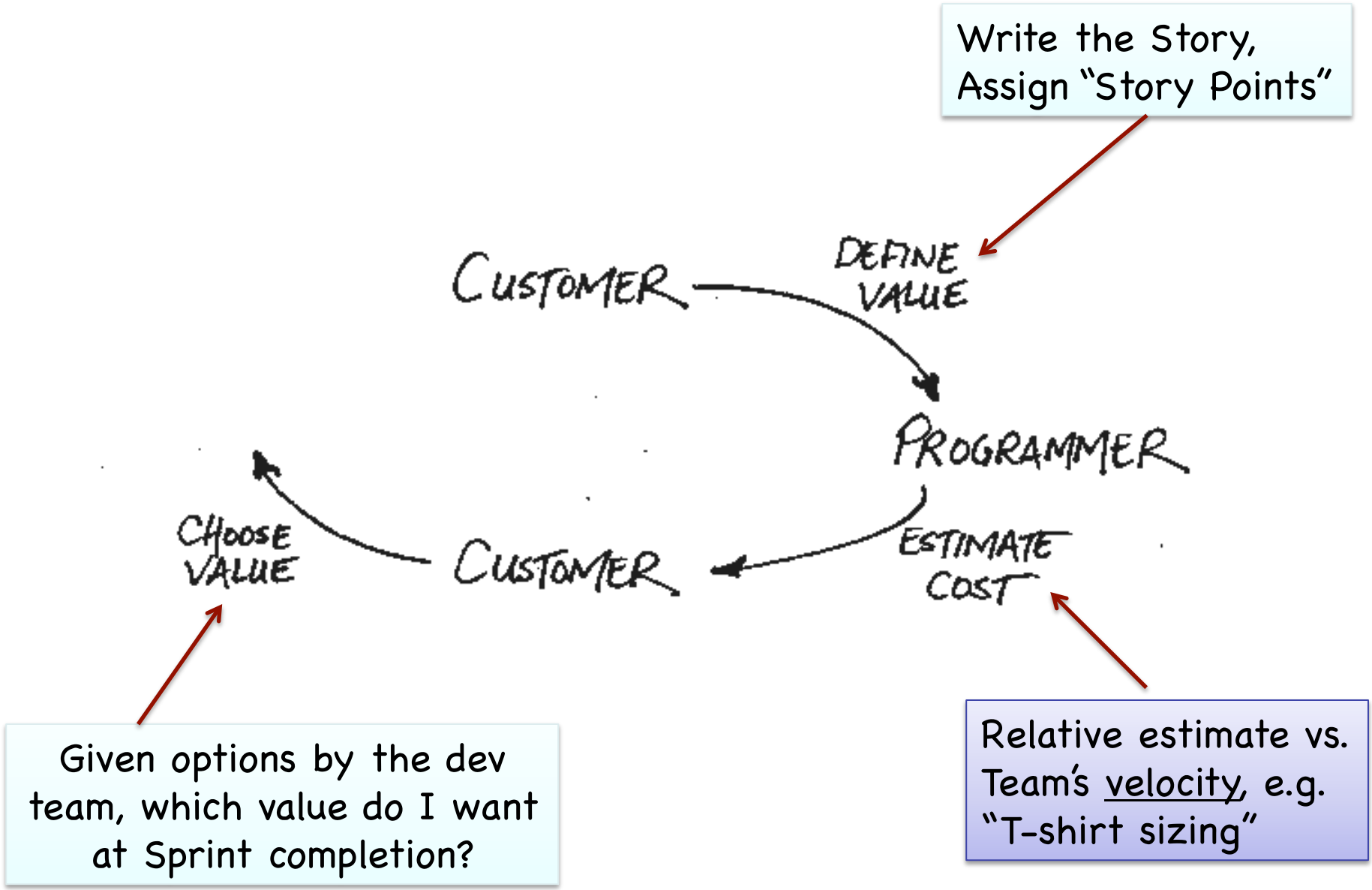
User Stories and the Agile Relationship



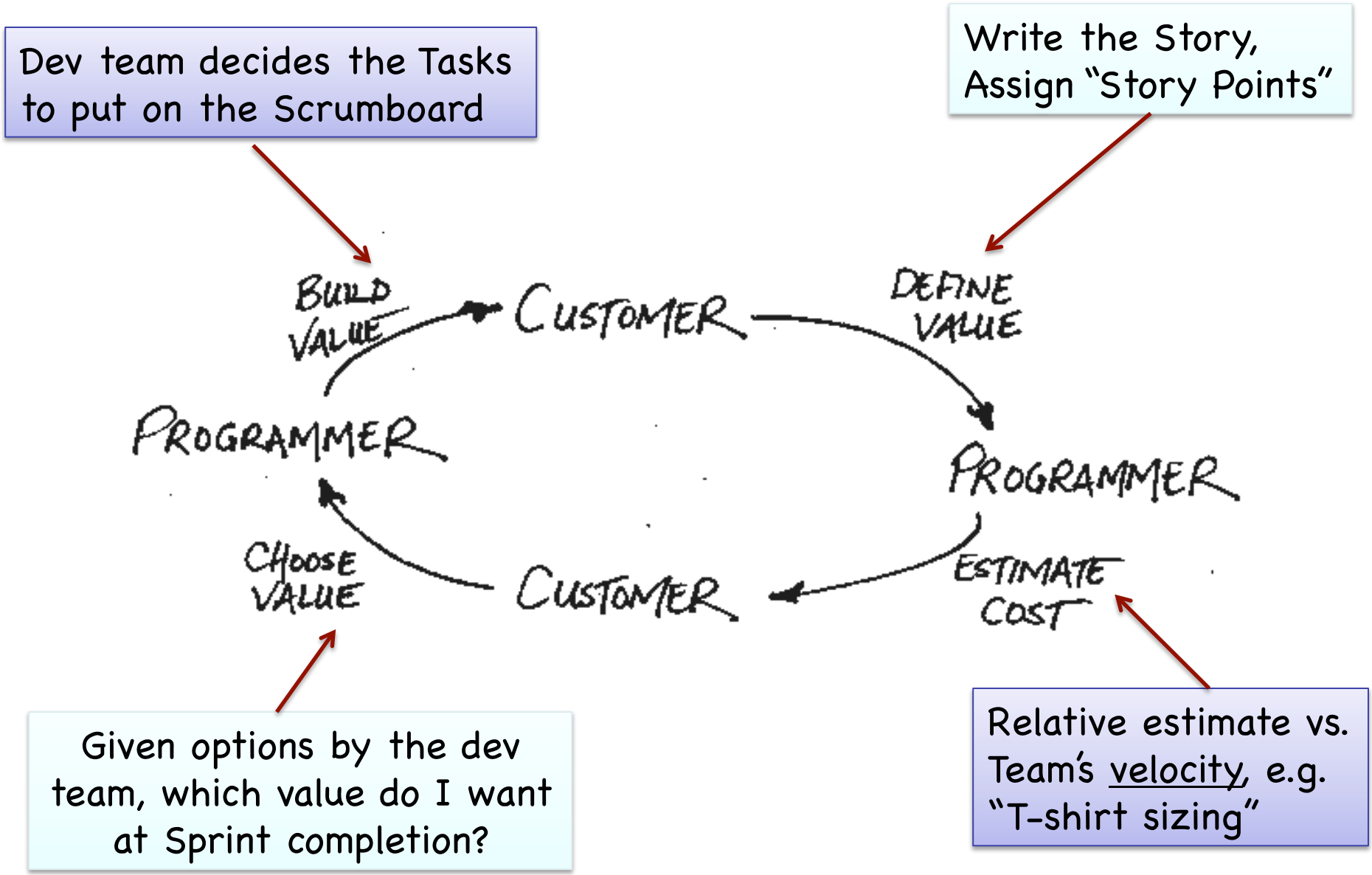
User Stories and the Agile Relationship



User Stories and the Agile Relationship



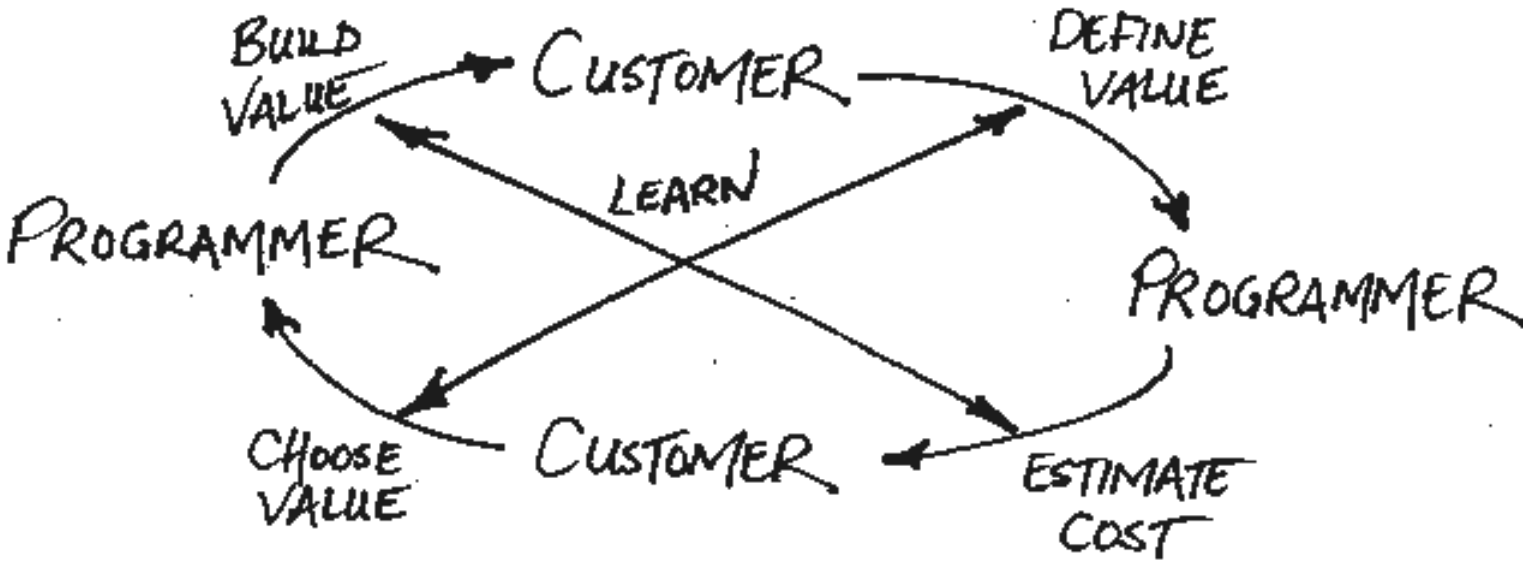
User Stories and the Agile Relationship



User Stories and the Agile Relationship

Business stakeholders are responsible for Scope, Priority, and the Composition of releases – whether a release meets an acceptance criteria

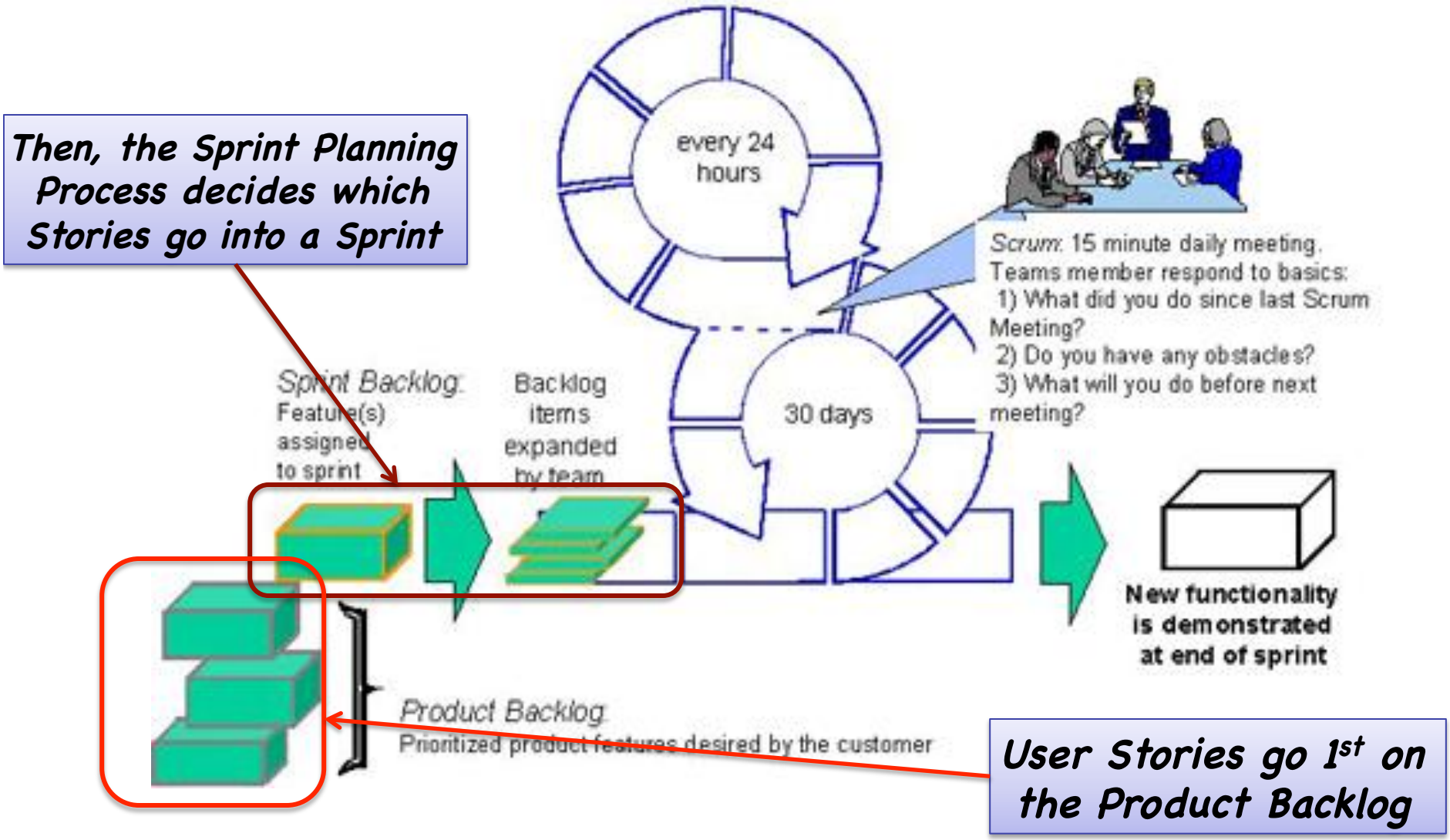
Developers are empowered – they do estimates, determine tradeoffs, decide on processes. Accept responsibility for a deliverable but not how to do it!



So how do the User Stories, once negotiated, fit into our SCRUM model?

Recall our Scrum Model

"Lather, Rinse, Repeat"
Every Sprint Planning



End Result: Now the Dev Team Builds Value!

A Scrumboard with User Stories and planned Tasks:

CONTINUOUS ASSESSMENT PLATFORM V2 SPRINT 1 14 JAN 2016-29 JAN 2016

9% 56 total points 5 completed points 11 open tasks 0 closed tasks 0 iocaine doses

USER STORY	NEW	IN PROGRESS	READY FOR TEST	CLOSED
#5 AS AN INSTRUCTOR, I CAN APPLY A RECIPE TO THE STUDENTS AND TEAMS IN A PARTICULAR COURSE 10 points	Unassigned #22 UI modifications			
#21 As a User I must authenticate to CAssess to protect my privacy 8 points	Unassigned #23 Modify httpd to include auth module Unassigned #24 Support redirect to CAssess via token Unassigned #25 CAssess validate token against auth module			
#7 As an Instructor, I can save the recipe I am configuring and come back to it later 8 points	Unassigned #26 Create DB schema for record Unassigned #27 Persist recipe state per transition Unassigned #28 Retrieve stored state Unassigned #29 UI Modifications			

User Story Summary

User Stories are a *placeholder for a conversation*

- User Stories are the currency of the Backlogs
  - User Stories are assigned relative value as story points and prioritized on the Product Backlog
  - User Stories are what is negotiated over during Sprint Planning to put on a Sprint Backlog

Good User Stories:

- are written in as active and goal-driven to ensures we know who it is for and why it is valuable
- have an Acceptance Test so we know what Done looks like

Copyright © 2003 United Feature Syndicate, Inc.