Visual Studio LIVE!
EXPERT SOLUTIONS FOR .NET DEVELOPERS
Anaheim

★ CAMPAIGN FOR CODE 2016 ★

PSD

Professional Scrum Development Using Visual Studio 2015

Richard Hundhausen
Consultant/Trainer
Accentient

About Me

- From Boise, Idaho, USA
- President of Accentient
- Microsoft Regional Director
- Microsoft MVP (Visual Studio ALM)
- Professional Scrum Developer
- Professional Scrum Trainer
- Co-creator of the Nexus
- richard@accentient.com

@rhundhausen

Microsoft

Professional Scrum Development with Microsoft Visual Studio 2012

Richard Hundhausen

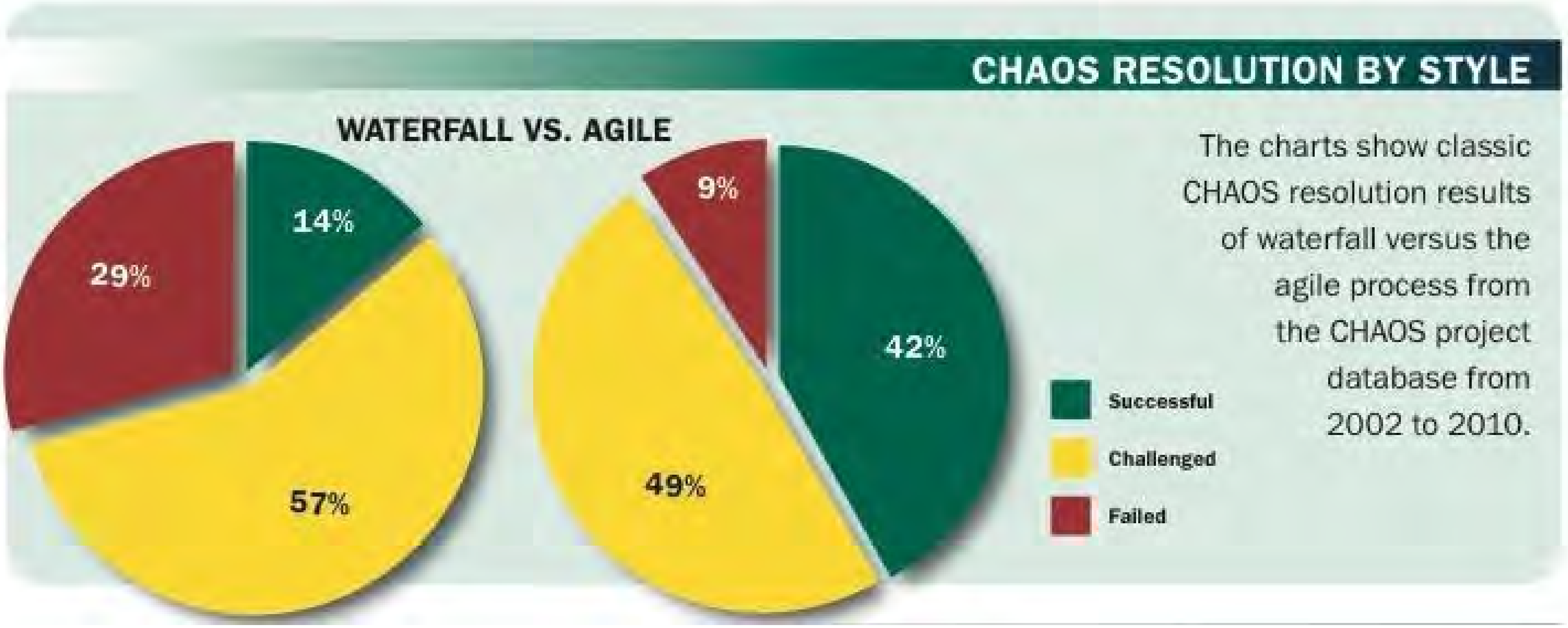
Session Backlog

- Why Scrum and what is it anyway?
- How Visual Studio supports Scrum
- Setting up a Scrum team project
- Planning and managing work

“We need to figure out a way to deliver software so fast that our customers don’t have time to change their minds.”

- Mary Poppendieck

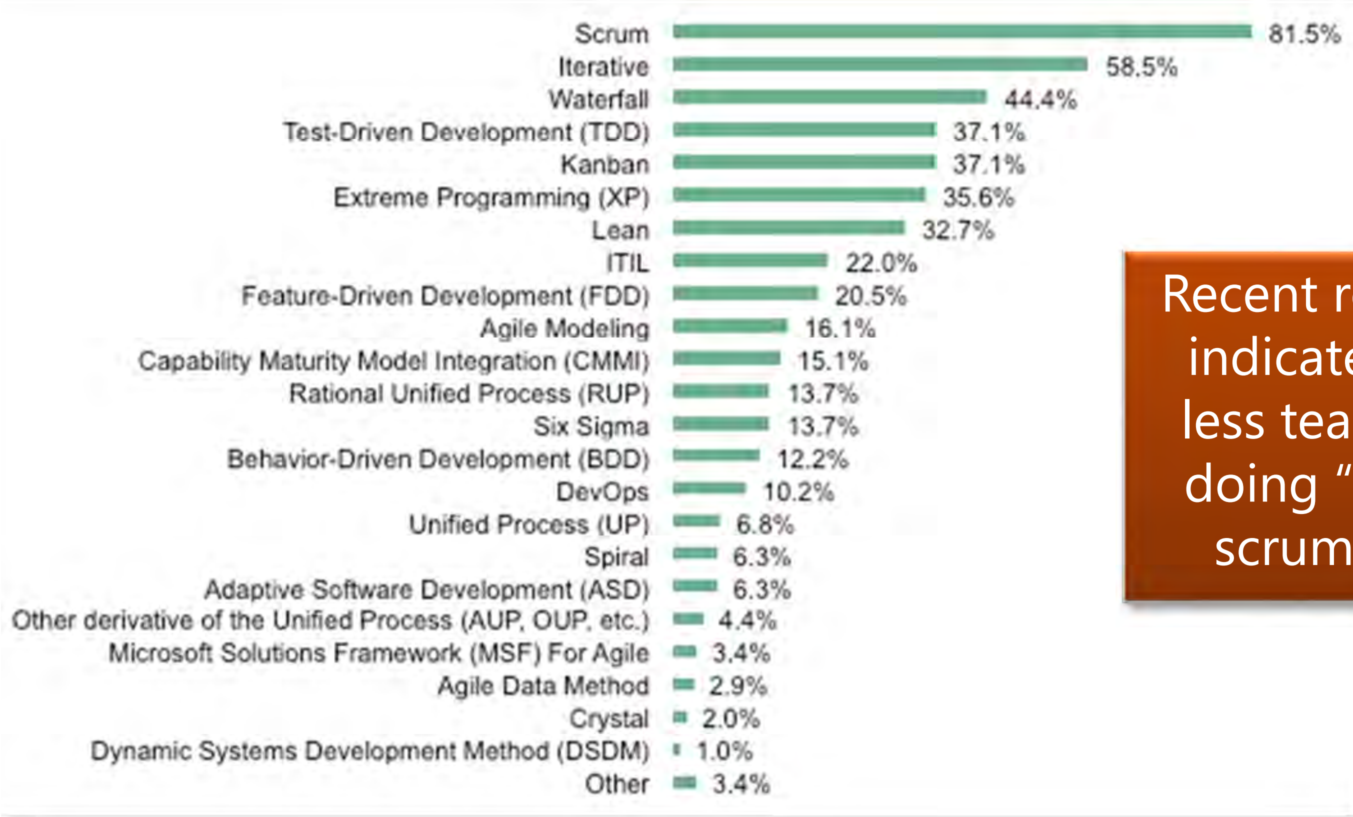
The Process Wars Are Over: Agile Won



CHAOS RESOLUTION BY STYLE

The charts show classic CHAOS resolution results of waterfall versus the agile process from the CHAOS project database from 2002 to 2010.

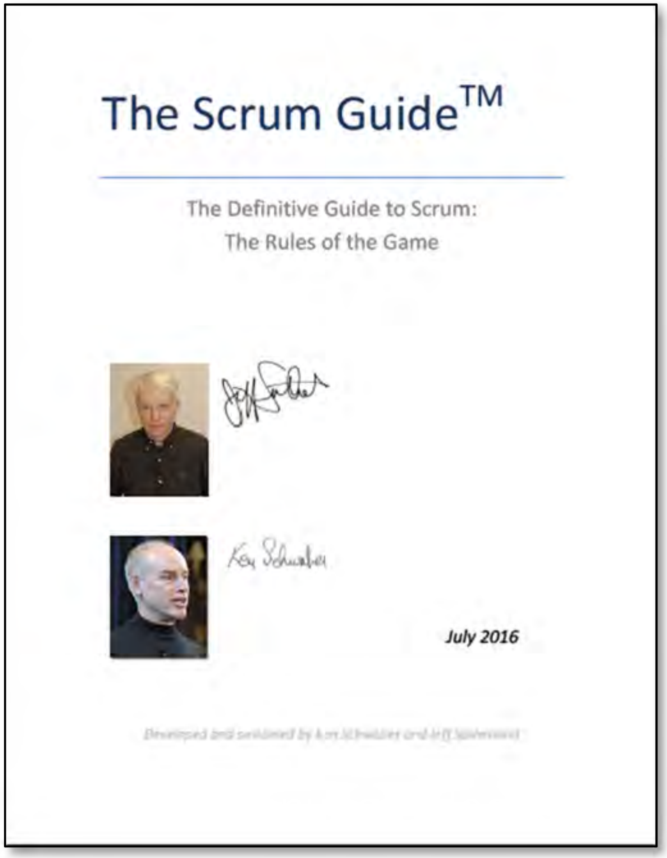
And By Agile, I Mean Scrum



Recent research indicates that less teams are doing "water-scrum-fall".

What is Scrum?

Scrum is a framework within which people can address complex problems, and productively and creatively deliver products of the highest possible value.



The Scrum Guide™

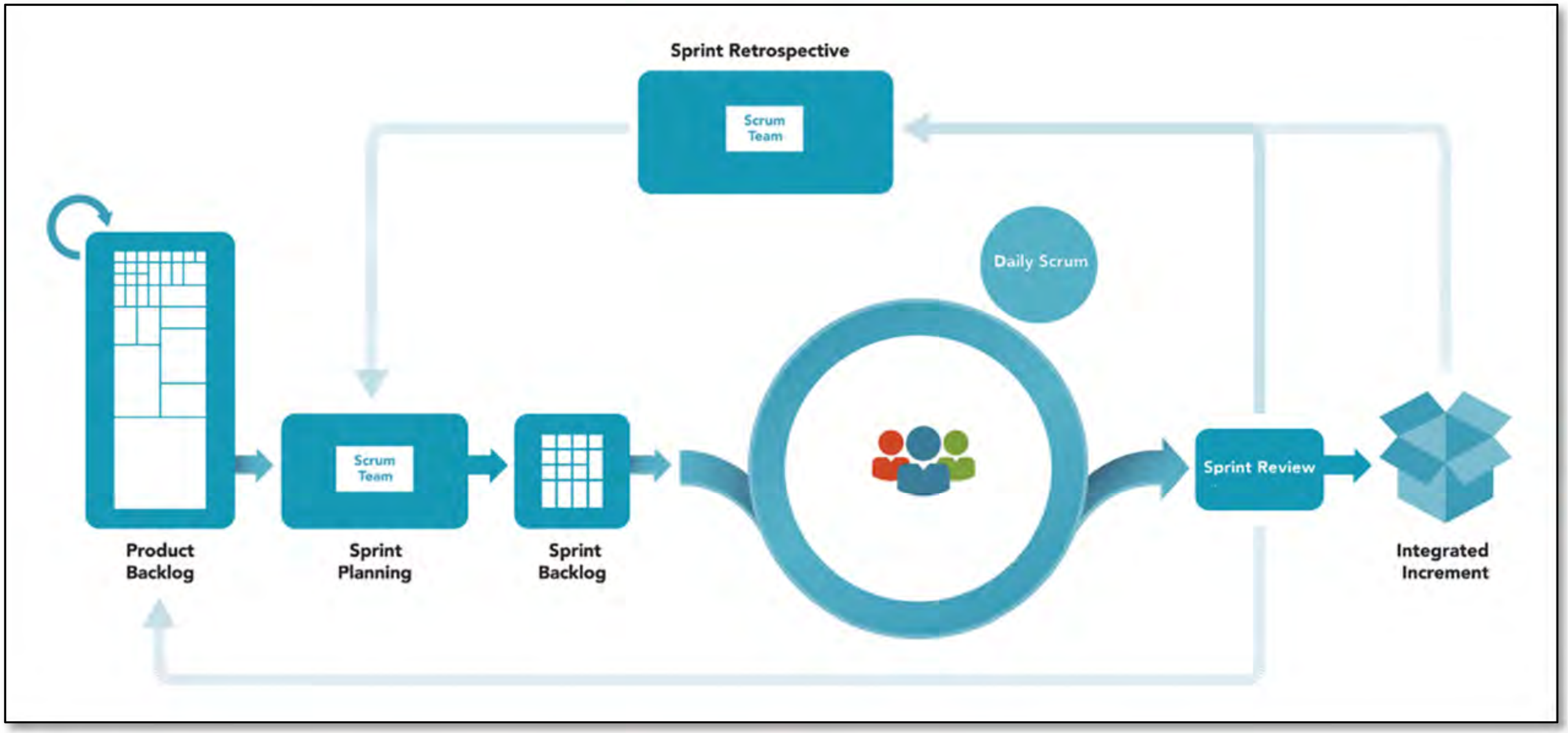
The Definitive Guide to Scrum:
The Rules of the Game

Ken Schwaber

July 2016

Developed and sustained by Ken Schwaber and Jeff Sutherland

Scrum in 60 Seconds



The Product Backlog

Set Up a New Product

- ✓ Create the team project
  - – Use the Visual Studio Scrum process template
- ✓ Configure security groups and permissions
- ✓ Configure version control folders and add files
  - – Don’t create branches unless and until they are necessary
- ✓ Setup dashboard (DoD, DoR, etc.)
- ✓ Setup work item areas and iterations (sprints)
- ✓ Migrate/create work items for the Product Backlog

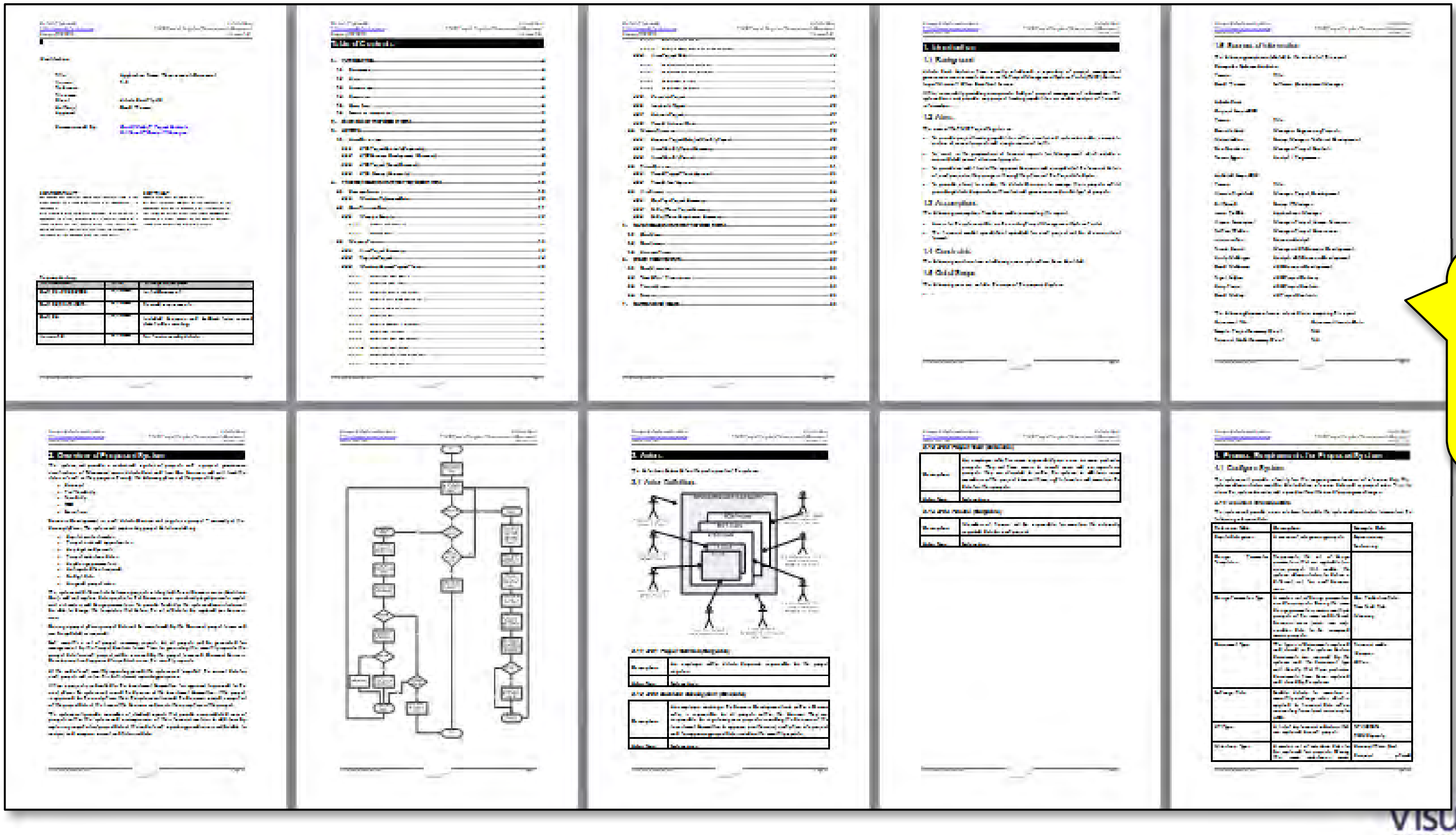
Refine the Product Backlog

- ✓ Refine the Product Backlog regularly
  - – Ready items should be adequately described, estimated, and ordered
- ✓ The Product Owner owns (is accountable for) the Product Backlog content – the “what”
  - – The Development Team owns the estimates

Valid Product Backlog Items

- Features
- Bugs
- Behaviors
- User Stories
- Requirements
- Use Cases
- Desirements
- Quality Attributes (NFRs)

But Not This …



Who's going to read this thing?

Enter the User Story

- Represents a user’s need
- Planning item
- Incomplete by design
- Mechanism for deferring the conversation

Twitter Feed

As a visitor to the home page,

I want to see recent tweets,

So that I can find a sweet deal

The Sprint

Planning a Sprint

- ✓ Ensure that the sprint iteration node exists
- ✓ Modify the *Current Sprint* queries
  - ✓ Or use @CurrentIteration
- ✓ Review your team’s velocity
- ✓ Edit the PBI work items being forecasted
  - – Set the Iteration Path (drag and drop works)
  - – Set the State to *Committed*
- ✓ Create and link initial task and test case work items

Sprinting

- ✓ Create and link task work items as needed
- ✓ Take ownership of tasks from the Sprint Backlog
- ✓ Update State and Remaining Work estimates at least daily
- ✓ Complete work according to your team’s DoD
- ✓ Create and manage Impediment work items as needed
- ✓ Assess progress (via boards, charts, conversation)
- ✓ Refine the Product Backlog when appropriate

Completing a Sprint

- ✓ Set the State of the PBI to Done
  - ✓ This can be done at any time during the Sprint
- ✓ Create, update, or split PBI work items for any undone work or new work identified during Sprint Review
- ✓ Capture the Sprint Retrospective feedback

The Sprints Ahead

Getting *Done*

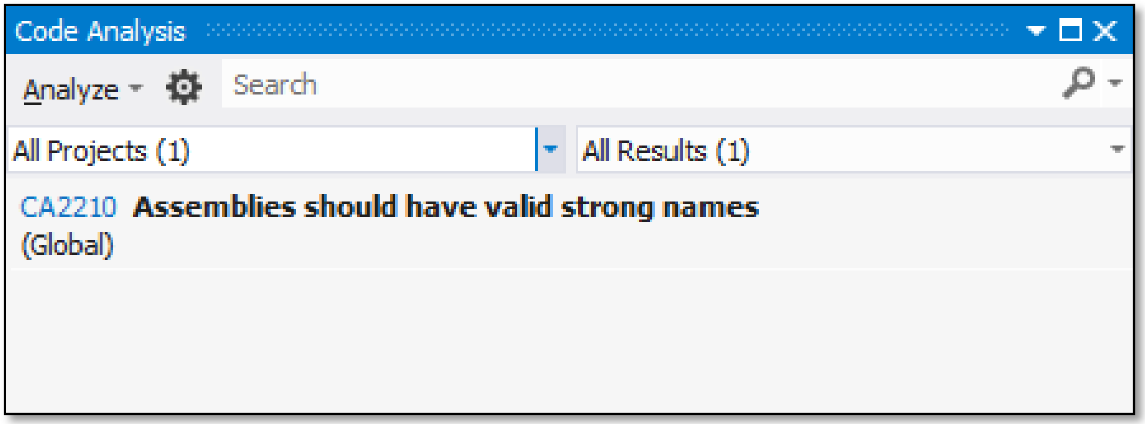
- *Done* is the state when a PBI becomes “releasable”
- A PBI is done when
  - ✓ It is fit for purpose
  - ✓ Meets the Product Owner’s acceptance criteria
  - ✓ Accepted by the Product Owner
  - ✓ Meets the Development Team’s DoD
- Cutting quality to get “Done” is not an option
  - – The DoD is there for a reason: to keep quality high
  - – The Development Team’s forecasts will improve over time

Handing Undone Work

- Undone work is a reality
- Handling undone work
  - Don’t release or demo undone work
  - Move PBIs back to the Product Backlog
  - Feature toggles, single-piece flow, or branching practices
- Over time, the Development Team will improve
  - Better estimation and planning will emerge

What Would You Do?

Your Development Team believes that they have met their Sprint Goal and the Product Owner is quite happy with the functionality. One of the PBIs wasn't completed according to your Definition of Done, however. It seems that a Code Analysis warning remains:



CA2210 **Assemblies should have valid strong names**
(Global)

Don’t Be Flaccid

- In early 2009, Martin Fowler described flaccid Scrum
  - http://martinfowler.com/bliki/FlaccidScrum.html
- Scrum Teams were using the nouns, but not doing the verbs
- Flaccid Scrum Teams believe in magic
  - And so do their customers
- Scrum is not a “silver bullet”

Don’t Change Scrum

- Scrum is just a set of rules put forth in the Scrum Guide
  - This makes it comparable to the game of chess
  - You can use Scrum according to its rules, or you can cheat
  - Cheating only provides short-term “wins”
  - Improving as a team only occurs if you play within Scrum’s rules
- Every Scrum role, rule, and event is designed to provide the desired benefits and address predictable recurring problems

A Roadmap to Consider

✓ Adopt Scrum

  – Keep improving how you play the game

✓ Use VSTS to plan and track your work

✓ Assemble your Product Backlog

✓ Work with Development Teams to size the items

✓ Refine the Product Backlog regularly

✓ Build, Release, Profit

✓ Repeat

DONE IS
BETTER
THAN
PERFECT

Done();

(thank you)

richard@accentient.com | @rhundhausen