

Agile@CA Development Process

Simon Cockayne

Misc Track



FOR INFORMATION PURPOSES ONLY

Terms of this presentation

This presentation was based on current information and resource allocations as of April 2013 and is subject to change or withdrawal by CA at any time without notice. Notwithstanding anything in this presentation to the contrary, this presentation shall not serve to (i) affect the rights and/or obligations of CA or its licensees under any existing or future written license agreement or services agreement relating to any CA software product; or (ii) amend any product documentation or specifications for any CA software product. The development, release and timing of any features or functionality described in this presentation remain at CA's sole discretion. Notwithstanding anything in this presentation to the contrary, upon the general availability of any future CA product release referenced in this presentation, CA will make such release available (i) for sale to new licensees of such product; and (ii) to existing licensees of such product on a when and if-available basis as part of CA maintenance and support, and in the form of a regularly scheduled major product release. Such releases may be made available to current licensees of such product who are current subscribers to CA maintenance and support on a when and if-available basis. In the event of a conflict between the terms of this paragraph and any other information contained in this presentation, the terms of this paragraph shall govern.

Certain information in this presentation may outline CA's general product direction. All information in this presentation is for your informational purposes only and may not be incorporated into any contract. CA assumes no responsibility for the accuracy or completeness of the information. To the extent permitted by applicable law, CA provides this presentation "as is" without warranty of any kind, including without limitation, any implied warranties or merchantability, fitness for a particular purpose, or non-infringement. In no event will CA be liable for any loss or damage, direct or indirect, from the use of this document, including, without limitation, lost profits, lost investment, business interruption, goodwill, or lost data, even if CA is expressly advised in advance of the possibility of such damages. CA confidential and proprietary. No unauthorized copying or distribution permitted.

Copyright © 2013 CA. All rights reserved. All trademarks, trade names, service marks and logos referenced herein belong to their respective companies. CA confidential and proprietary. No unauthorized copying or distribution permitted.

Simon Cockayne

CA Technologies.

Sr Principal Product Manager. Product Owner CA 2E/CA Plex

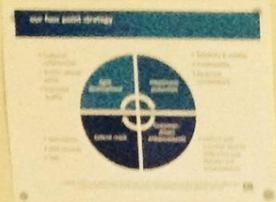
This session will discuss how both the CA 2E and CA Plex products are now developed using an Agile approach. There will be a special focus on the critical role that customers play in shaping the strategy and content of new releases, by collaborating with the Product Manager, Product Owner and the Scrum Teams. We'll also plan to have a rich discussion on what works well and where we can make the process even better.

Agenda

- Agile Manifesto and Principles
- Scrum
- Product Manager vs. Product Owner
- Collaboration and engagement
- Ideal Wall
- Roadmap
- Happiness Graphing
- Agile Delivery
- CA 2E and CA Plex
- Q&A

Agile software development

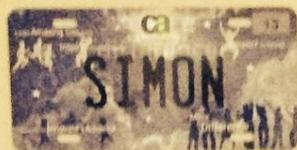
- Manifesto for Agile Software Development
 - <http://agilemanifesto.org>
- Principles behind the Agile Manifesto
 - <http://agilemanifesto.org/principles.html>



Meeting tips

- 1. Start on time
- 2. Have a clear agenda
- 3. Assign roles
- 4. Encourage participation
- 5. Stay on topic
- 6. End on time

BE PATIENT



agility
made possible™

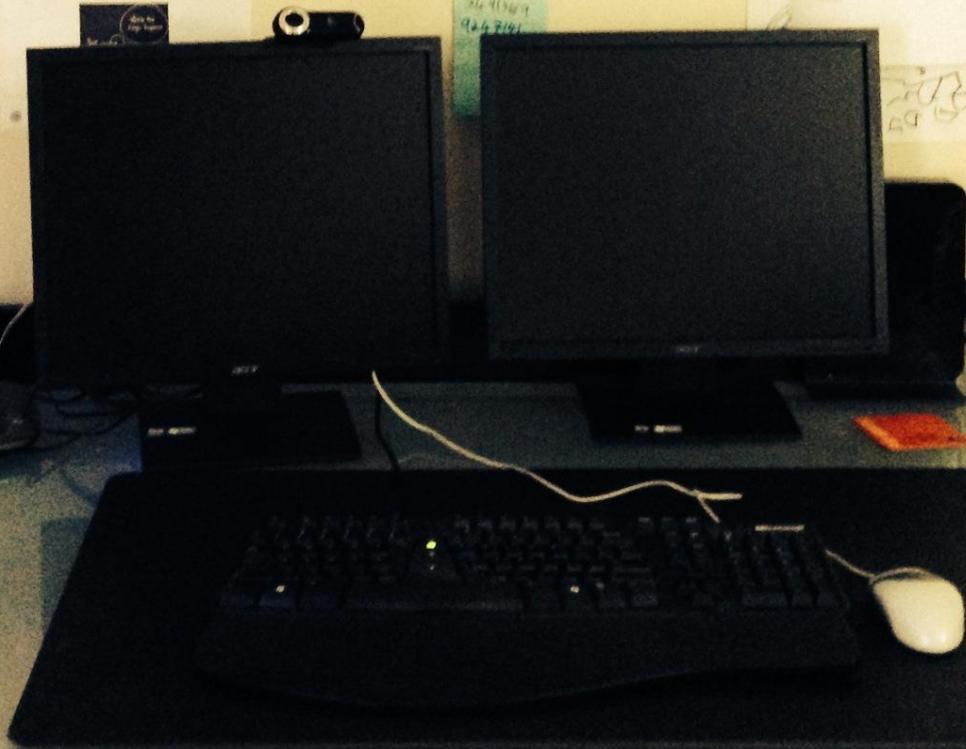
COURAGE



Member for Agile Software Development

Simon Cockayne

Member for Agile Software Development

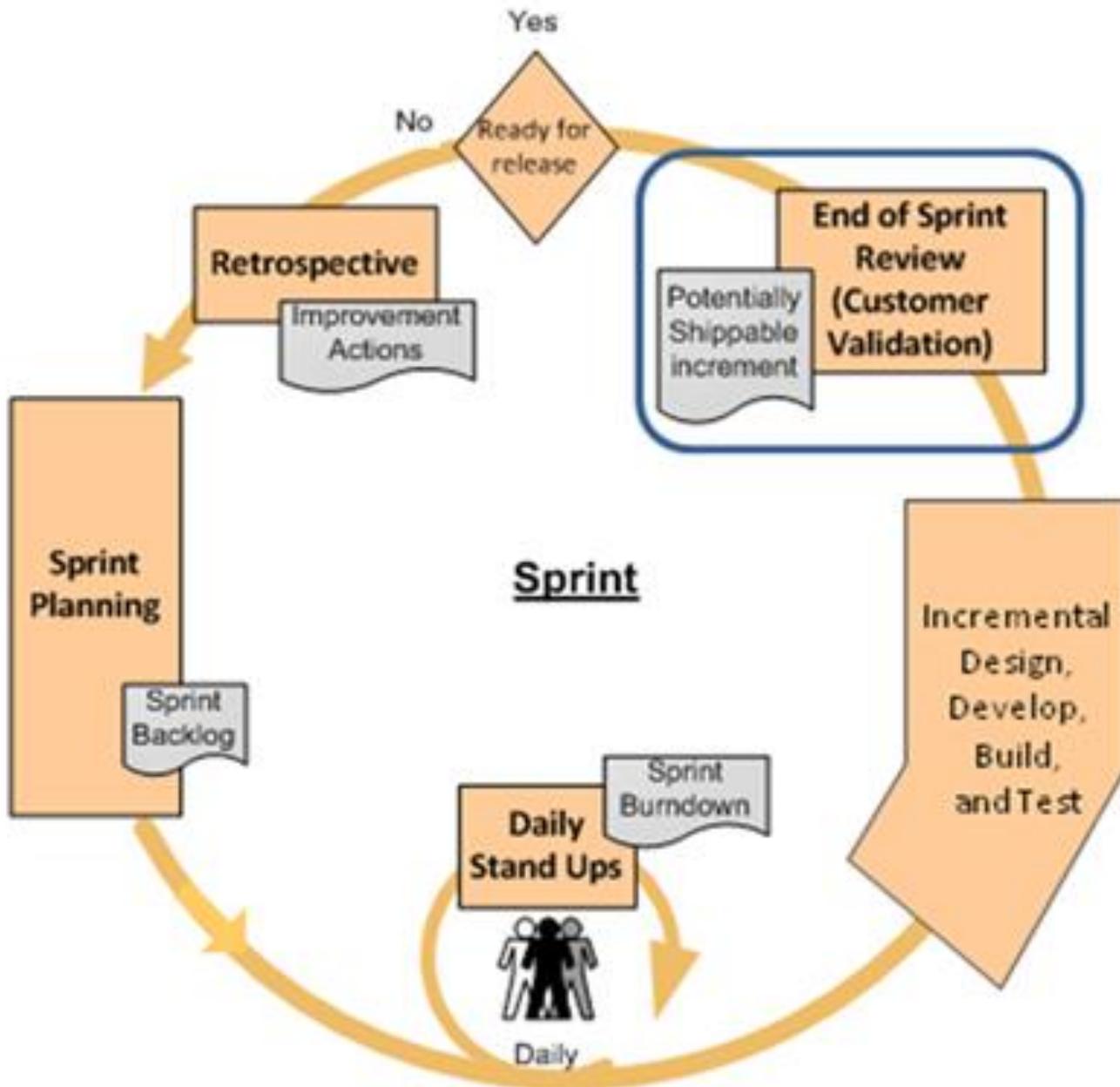


“**Scrum** is an iterative and incremental agile software development framework for managing software projects and product or application development. Its focus is on "a flexible, holistic product development strategy where a development team works as a unit to reach a common goal" as opposed to a "traditional, sequential approach".

[http://en.wikipedia.org/wiki/Scrum_\(development\)](http://en.wikipedia.org/wiki/Scrum_(development))

Agile Development

- Agile development - Scrum based.
- Maintain potential enhancements in a Product Backlog
 - Epics and Stories
- Release planning.
- Sprints ($1 > n$)
 - Sprint planning – Select from Product Backlog > Sprint Backlog
 - 4 week sprints.
 - Smaller self-contained chunks of work, a.k.a. **Story**
 - DEV/QA/DOC all done in parallel (Daily standups).
 - Demo of sprint output to eligible customers at end of sprint.
 - Regular feedback from customers – that we can react to before it is too late
- **No Beta (for Plex and 2E)**



Definition of Done – Applies to all stories.

- All tasks updated with accurate Done (hours), and To do (hours) 0 and Tasks marked as Completed.
- All acceptance criteria met.
 - Acceptance criteria are specific to a story.
- All QA automated (excluding Web Option GUI and WS GUI).
NOTE: Must include successfully run Test Set, in current Sprint folder.
- All QA passed. **No bugs.**
NOTE: A story cannot be closed if there are bugs of any severity.
- All doc must be complete.
NOTE: Including updating Release Notes AND updating Documentation Changes section in any modified manuals.

What if a story does not meet Definition of Done

- Ideally we will not update product or doc with part of a story, if it does not meet definition of done.
- We will not review/discuss stories that do not meet the definition of done.

End of Sprint Code

- End of Sprint code will be made available (to “accepted” customers only) – every four weeks.
- Do not use in production.
- Do not contact CA Support with questions.
- See next slide for feedback

- 2E r8.6 developed using Agile (Scrum)
 - Ongoing collaboration with selected customers during development
 - Pre-release features reviewed on a monthly basis
 - Extremely successful outcome
 - 3-month Beta release
- Future 2E and Plex releases...
 - Continue to use Agile (Scrum)
 - Building on our experiences with r8.6
 - Heavy customer collaboration and validation
 - No Beta release planned

Product Owner vs. Product Manager (PdM)

PdM vs. PO

- PdM and PO are a team.
- Jointly build and maintain a product backlog
- The PdM owns the overall strategy and does much of the business planning, market sizing, revenue planning
- The PO's overarching responsibility is to represent the voice of the customer during the development process.



development

Activity	Product Owner	Product Manager
Market research	Rarely	Core function
Work with UEX team to explore and define customer requirements	Core function	Core function
Document release plan; vision, 3-5 goals with outcomes and definition of success	Rarely	Core function
Write Themes & Epics for the Backlog	Sometimes	Core Function
Expand Epics into Stories	Core Function	As Needed
Work with R&D to size Epics/Stories	Core Function	As Needed
Work with Scrum team to schedule stories	Core Function	As Needed
Ensure Scrum team understands a story during a Sprint	Core Function	As Needed
Secure customers for customer validation	As Needed	Core function
Work with customers to validate functionality in Sprint reviews	Core Function	Rarely
Brief customers on product roadmap	Sometimes	Core Function
Define materials & campaigns with Marketing	Rarely	Core function
Talk with Analysts	Sometimes	Core function



Collaboration and engagement



Ideas



Image from gustavosarnarimkt.wordpress.com



Welcome back, SIMON Log out ca.com Contact

+ MyCA Welcome + **Communities** + Support + Projects

Welcome Message Board **Ideas** Calendar Documents Members Administration

Ideas

Ideas CA Plex CA 2E Global User Community

Leave



Simon_Cockayne

Recent Most Popular Top All-Time Most Commented

Have an inspirational idea that you'd like to share?

Post Idea

Search Ideas

Find by Status:

Default

Filter by Category:

- All
- CA Plex (71)
- CA 2E (103)**

Find by:

Promote Demote



Do not force 2E to have to use a collection

August 7, 2013 Posted By Crispin Status : New

5 Points

5 Votes

When generating DDL the 2E Compile Processor (YEXCSQL) requires hte target library to be an SQL Collection. This is not required to compile TABLE/INDEX/VIEW Objects. Would like this to be removed as a requirement from the YEXCSQL Processing.

Thanks,

Crispin.

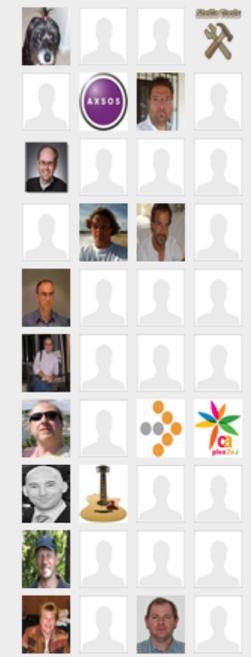
Edit Delete

0 comments

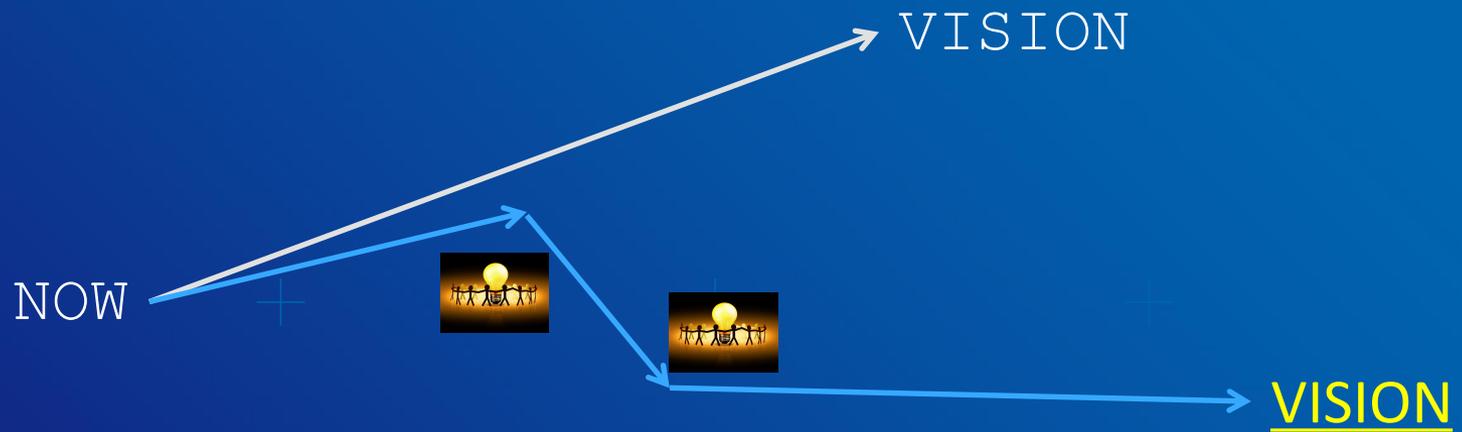
No comments yet. [Be the first.](#)

Categories: CA 2E

Recent Participants



Roadmap



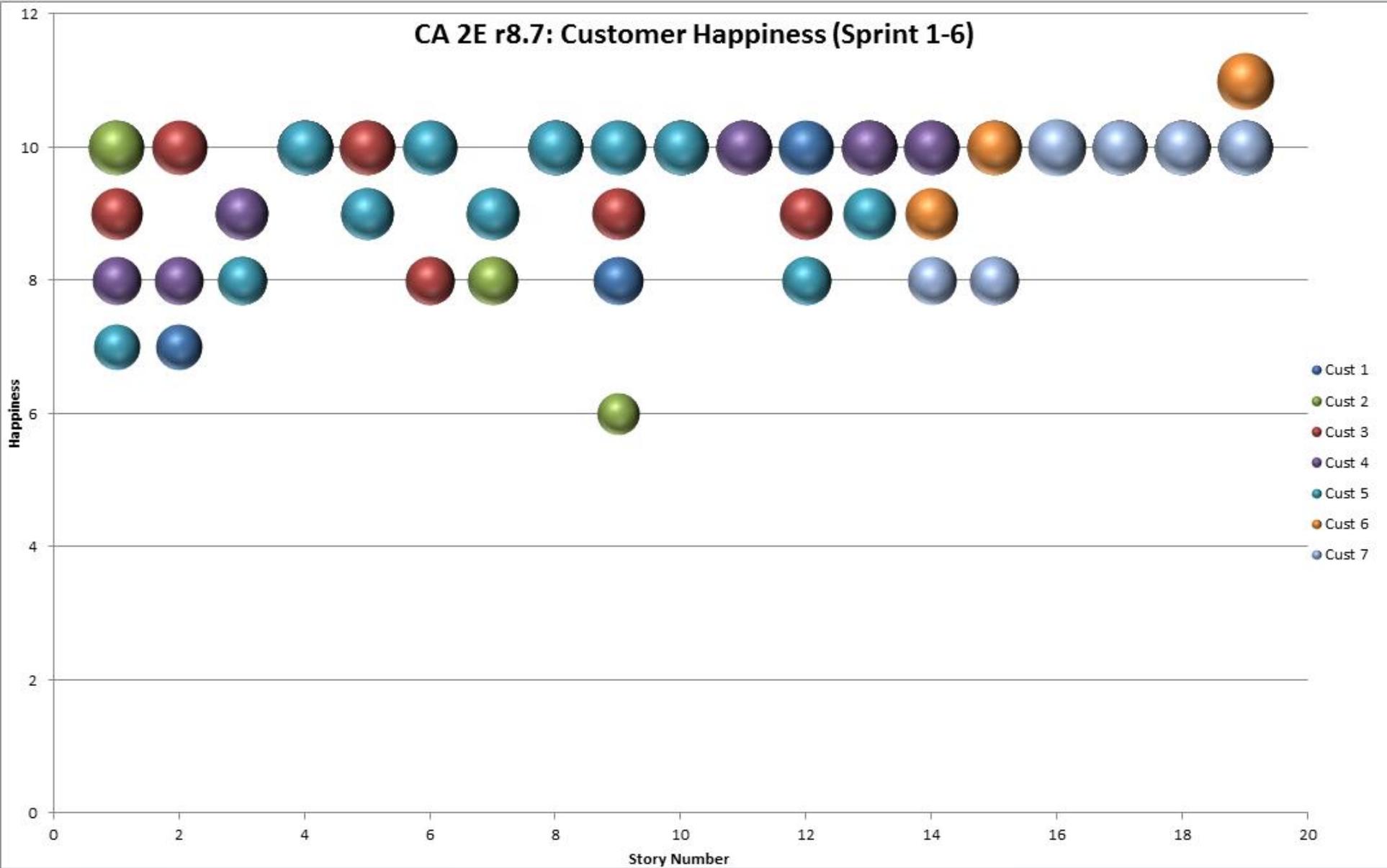
Happiness Graphing

An experiment: Graphing Your Happiness

- Our goal is to give you guys value, to help you be happy.
- Can we estimate and measure happiness?
- Can we show CA management how happy you are?
- How can we help you be happier?
- *Is our release backlog full of value adding things, in value order?*
- **Q:How happy does backlog item X make you?**
 - Super-happy (scores 9-10) – you love the backlog item. 😊
 - Ok - (score 7-8) – you are satisfied.
 - Unhappy (score 0-6) – you are unhappy (or miserable) about the backlog item. ☹
- Be Open. Be honest.
- Let's experiment...



CA 2E r8.7: Customer Happiness (Sprint 1-6)



Agile Delivery

CA Plex and CA 2E



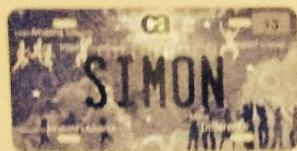
Summary



Meeting tips

- 1. Start on time
- 2. Have a clear agenda
- 3. Assign roles
- 4. Encourage participation
- 5. Stay on topic
- 6. End on time

BE PATIENT



agility
made possible™

COURAGE

Be brave. Be bold. Be fearless. Be you.



Member for Agile Software Development

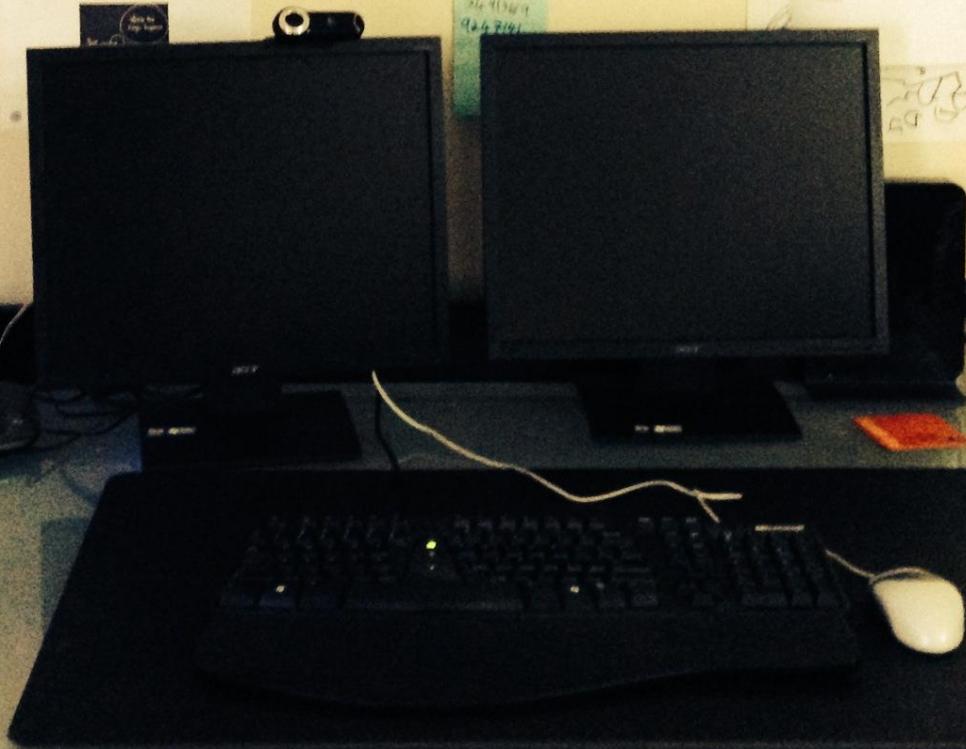
Agile software development is a mindset, not a methodology. It is a way of thinking and working that values individuals and interactions over processes and tools, working with customers and business partners, responding to change over following a plan, and collaborating with the team.

Simon Cockayne



Agile Manifesto

Individuals and interactions over processes and tools
Working with customers and business partners
Responding to change over following a plan
Collaborating with the team



Discussion