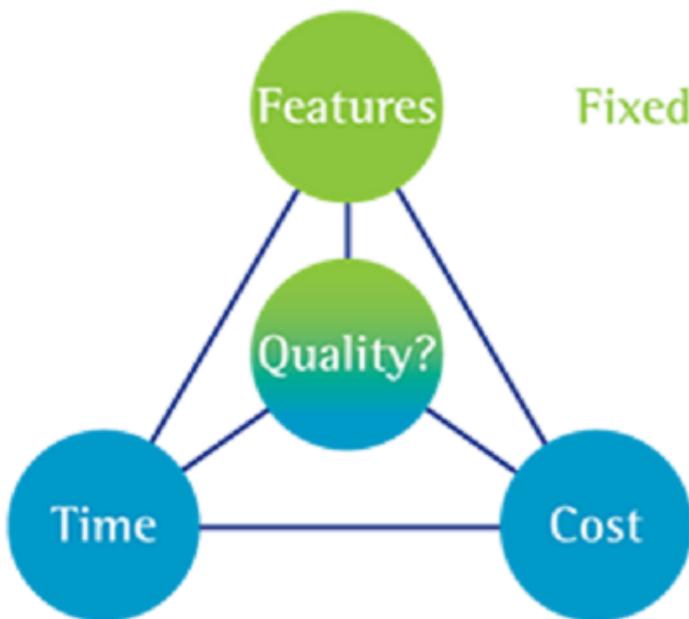


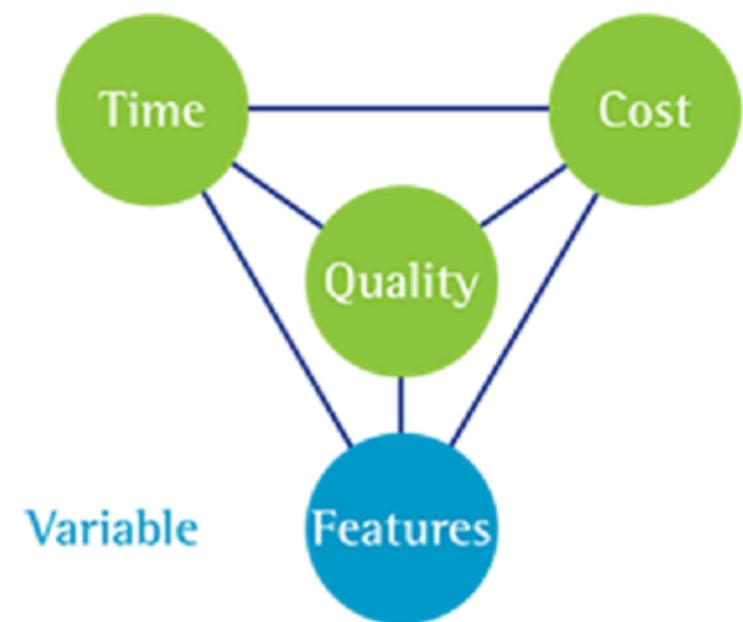
Agile development

Approaches to software development

Traditional Approach



Altern Approach



The Agile Manifesto <http://agilemanifesto.org/>

- “We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:
 - **Individuals and interactions** over processes and tools
 - **Working software** over comprehensive documentation
 - **Customer collaboration** over contract negotiation
 - **Responding to change** over following a plan
- “Whilst there is value on the right, we value the left more”

Kent Beck et al (2001)

12 Principles of Agile

- Customer satisfaction by rapid delivery of useful software
- Welcome changing requirements, even late in development
- Working software is delivered frequently (weeks rather than months)
- Working software is the principal measure of progress
- Sustainable development, able to maintain a constant pace
- Close, daily co-operation between business people and developers
- Face-to-face conversation is the best form of communication (co-location)
- Projects are built around motivated individuals, who should be trusted
- Continuous attention to technical excellence and good design
- Simplicity- The art of maximizing the amount of work not done - is essential
- Self-organizing teams
- Regular adaptation to changing circumstances

Philosophy a little at a time

- The plan changes a little at a time,
 - Design changes a little at a time
 - The team changes a little at a time
-
- Don't design, plan or code more than is needed at the moment so that options for the future remain open.

Philosophy part I

- As each ‘piece’ is added developers learn what works and what doesn’t.
- Customer learns what value the system offers and what features are needed next.
- Do the simplest thing that works and meets current needs, don’t build in extra complexity ‘in case’.
- The team should take pride in delivering high quality.

<https://vimeo.com/25121889>

Agile Methods

- Group of methods based on highly iterative and incremental development.
 - Test Driven Development (TDD)
 - Write test cases for our requirements (before coding)
 - Write enough code to just pass the test cases
 - Refactor to improve code quality
 - XP & Pair programming
 - <https://www.agilealliance.org/glossary/xp/>
 - Scrum



SCRUM – A framework for managing software development

- for developing and sustaining complex products.
- iterative and incremental for managing projects in less controlled environments.
- Within which people can address complex adaptive problems, whilst productively and creatively delivering products of the highest possible value.
- Scrum is
 - Lightweight
 - Simple to understand
 - Difficult to master
- Scrum is NOT a process or a technique for building products; rather it is a frame work within which you can employ various processes and techniques.
- The scrum framework consists of Scrum Teams and their associated **roles**, **events, artifacts** and **rules**. The rules bind them all together.

Scrum Theory

- Founded on empirical control theory, or empiricism.
- Which asserts that knowledge comes from experience and making decisions based on what is known.
- Scrum uses an iterative, incremental approach to optimize predictability and control risk.

Definition of DONE

- Transparency
 - Common standard so observers share a common understanding of what is being seen
- Those doing the work and those accepting the work must share a common understanding of ‘the definition of ‘DONE’’
- Each product must reach the definition of done in order to be considered completed
 - Usually assessed at the end of the development process
- Can include aspects both visible and invisible to the client
 - Accessibility, cross-browser uniformity
 - Unit tests passed, comments approved, code reviewed
- Helps to ensure effort to develop product is accurate.
- As Scrum Teams mature, it is expected that their definition of ‘Done’ will expand to include more stringent criteria for higher quality.

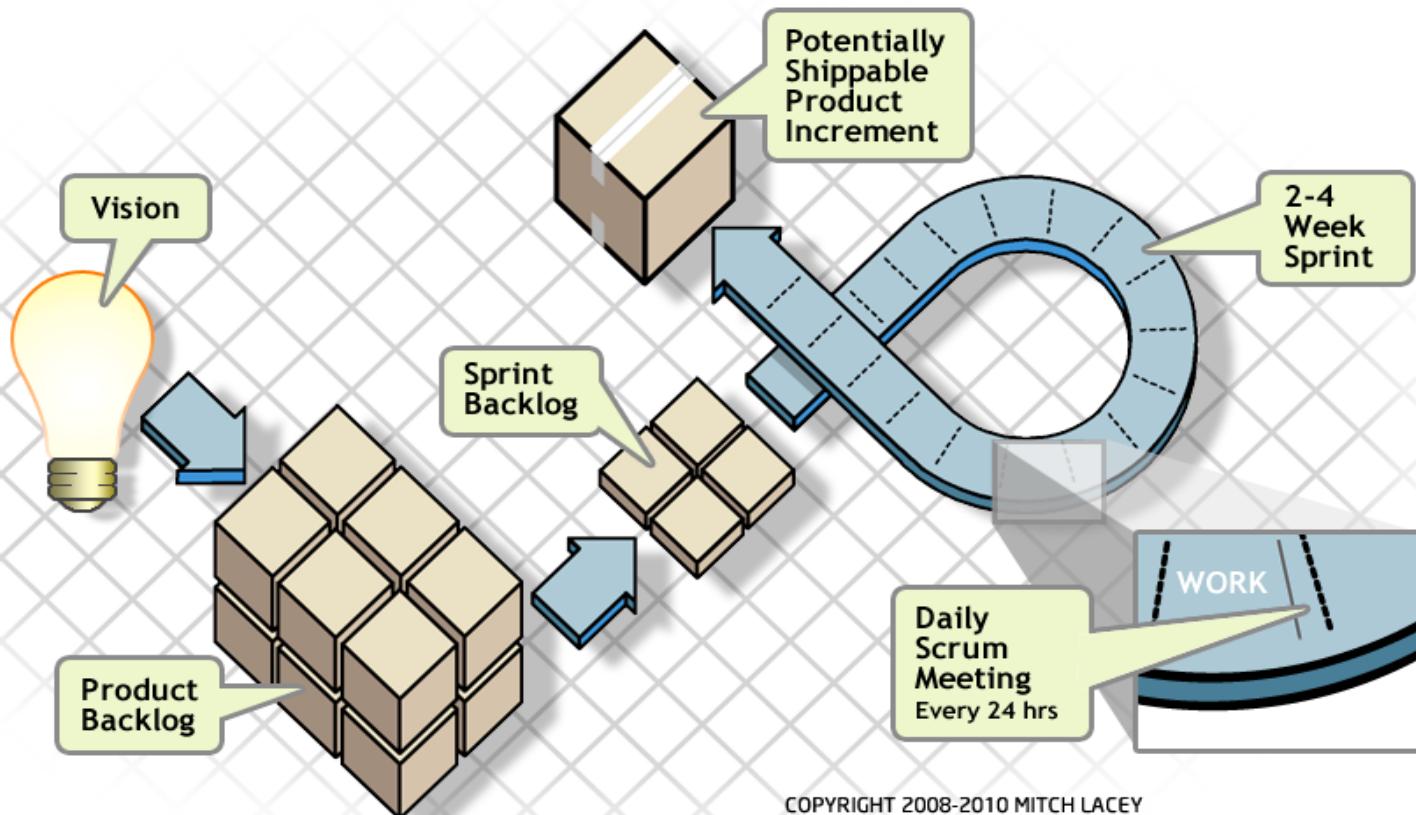
The SCRUM Team (Roles)

- Product Owner – responsible for maximising the value of the product and the work of the development team.
 - Represent stakeholders
 - Is one person not a committee
 - Their decisions must be respected by the entire organisation.
 - Creates user stories and Manages backlog
 - Development team isn't allowed to act on what anyone else says
 - Ensure team is delivering value
- Development Team
 - self organising and cross functional
 - Only members of the Development Team create the increment.
 - Accountability belongs to the team as a whole
 - The only job title is developer and there are no sub teams
 - Between 4 and 8 developers
- Scrum Master - Facilitate the scrum process

Scrum Events

- Prescribed events are used to create regularity, and to minimise the need for meetings not defined in Scrum.
- All events are time-boxed events, such that every event has a maximum duration.
- Once a Sprint begins, its duration is fixed.
- Every event in Scrum is a formal opportunity to inspect and adapt something.

The SCRUM Process



COPYRIGHT 2008-2010 MITCH LACEY

[HTTP://WWW.MITCHLACEY.COM](http://WWW.MITCHLACEY.COM)

Breaking the work down

- Initially a “Vision” is defined in terms of high level user stories (EPIC’s)
- The Product Owner and clients arrange the EPIC’s into priority order
- The EPIC’s are them broken down into user stories
- This creates an ordered, sized product backlog

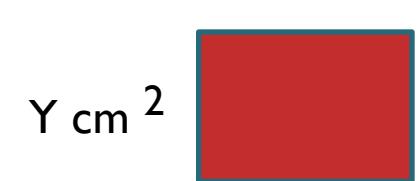
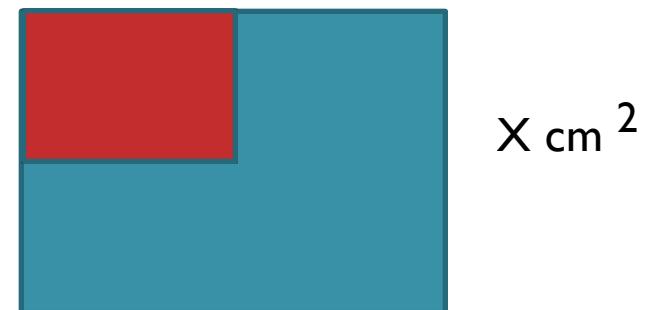
The Product Backlog

- The Product Backlog is dynamic, prioritized and ordered list of everything
 - features, functions, enhancements, fixes to past releases.
 - Product Backlog items are usually user stories - have the attributes of a description, estimate and value.
- Product Backlog refinement is the act of adding detail, estimates, and order to items in the Product Backlog.
 - The Product Owner is responsible for the Product Backlog, including its content, availability, and ordering.
- Product Backlog items that can be actioned by the Development Team within one Sprint are deemed “Ready” for selection in a Sprint Planning.

Estimating Story Points

- A story point is a relative measure of complexity instead of effort (hours etc.) of a user story in agile estimation and planning.
- Planning poker (SCRUM)
 - Cards of 0, $\frac{1}{2}$, 1, 2, 3, 5, 8, 13, 21, 50 – used by the team for discussing the size

We're pretty good at estimating relatively

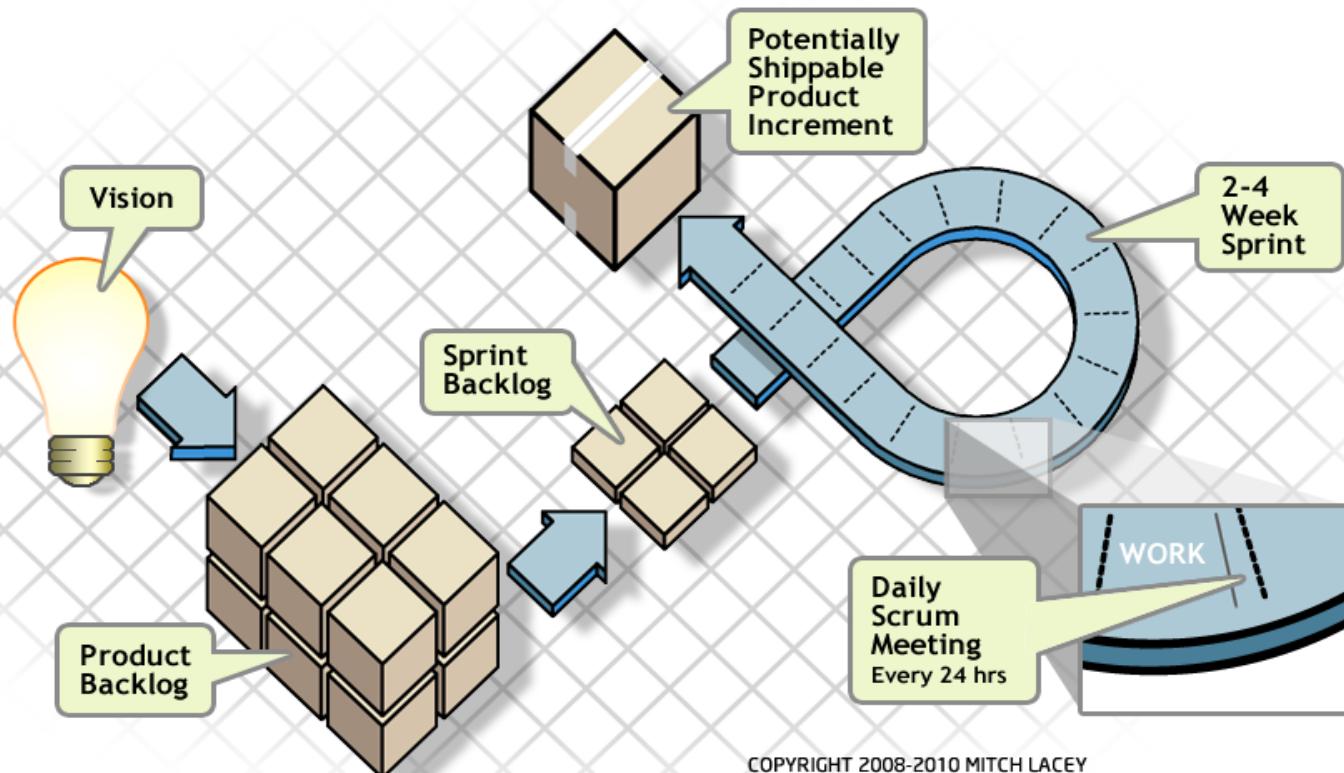


SCRUM Ceremonies

4 formal events for inspection & adaptation;

- **Sprint planning**
 - the team meets to choose a set of work to deliver during a sprint
- **Daily scrum**
 - the team meets each day to share struggles and progress
- **Sprint reviews**
 - the team demonstrates to stakeholders what it has completed during the sprint
- **Sprint retrospectives**
 - the team looks for ways to improve the product and the process.

The SCRUM Process



COPYRIGHT 2008-2010 MITCH LACEY

[HTTP://WWW.MITCHLACEY.COM](http://www.mitchlacey.com)

The Sprint – the heart of Scrum

- A time-boxed period for delivering a collection of completed user stories – where a “Done”, useable, and potentially releasable product Increment is created.
- Sprints have consistent durations throughout a development effort.
- Sprints contain and consist of the Sprint Planning, Daily Scrums, the development work, the Sprint Review, and the Sprint Retrospective.

Sprint Planning

- This plan (Sprint backlog) is created by the collaborative work of the entire Scrum Team.
- The Development Team considers the functionality that will be developed during the Sprint.
- The team velocity is used to estimate how much can be undertaken in a sprint
- The team velocity is calculated from the average number of story points completed over the previous sprints
- Sprint Planning is time-boxed to a maximum of eight hours for a one-month Sprint. For shorter Sprints, the event is usually shorter.
- The Scrum Master ensures that the event takes place, that attendants understand its purpose and it is kept to within the time-box.
- Sprint Planning answers the following:
 - What can be delivered in the Sprint?
 - How will the work be achieved?

Daily Scrum – what happens

- The Daily Scrum is a 15-minute time-boxed event to synchronize activities and create a plan for next 24 hours.
- Work since the last Daily Scrum and forecasting the work that could be done before the next one. Inspects how progress is trending toward completing the work in the Sprint Backlog. Optimizes the probability of meet the Sprint Goal.
- Development Team members explain:
 - What did I do yesterday that meet the Sprint Goal?
 - What will I do today?
 - Do I see any impediment that prevents me from meeting the Sprint Goal?
- The Daily Scrum is held at the same time and place each day to reduce complexity.

Daily Scrum – improve

Communication
Quick decision-making
Identify impediments
Improves level of knowledge
Eliminate other meetings

Sprint Review

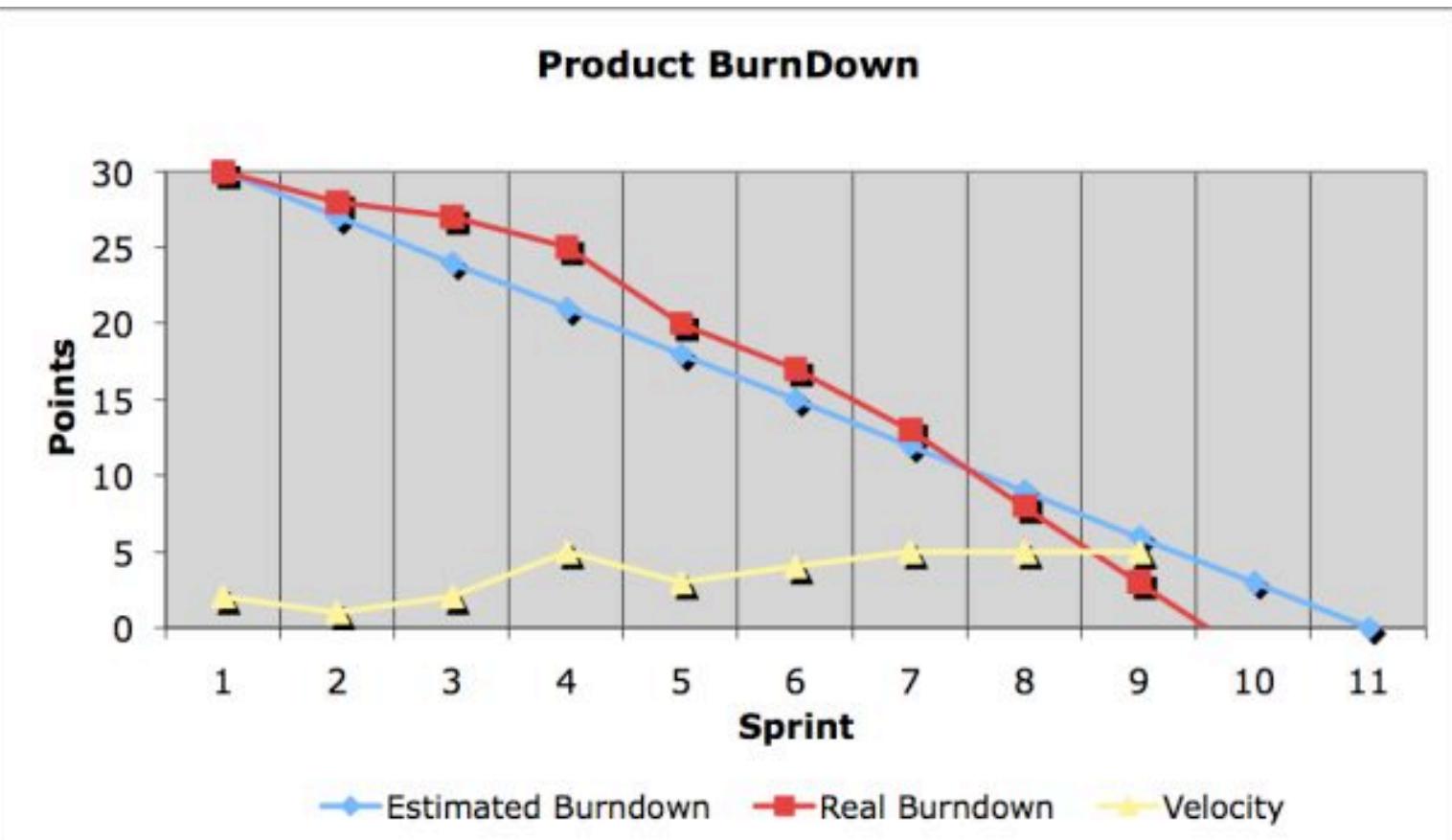
- A Sprint Review is held at the end of the Sprint, informal meeting, intended to elicit feedback and foster collaboration.
- Team demonstrates the work that it has “Done” and answers questions about the Increment; adapt the Product Backlog if needed.
- Product Owner discusses projects likely completion dates based on progress to date (if needed);
- Collaborates on what to do next, so that the Sprint Review provides valuable input to subsequent Sprint Planning;
- Review of the timeline, budget, potential capabilities and how marketplace or potential use of the product might have changed
- Consider the next things that could be done to optimize value.
- This is a four-hour time-boxed meeting for one-month Sprints. For shorter Sprints, the event is usually shorter.

The result of the Sprint Review is a revised Product Backlog that defines the probable Product Backlog items for the next Sprint.

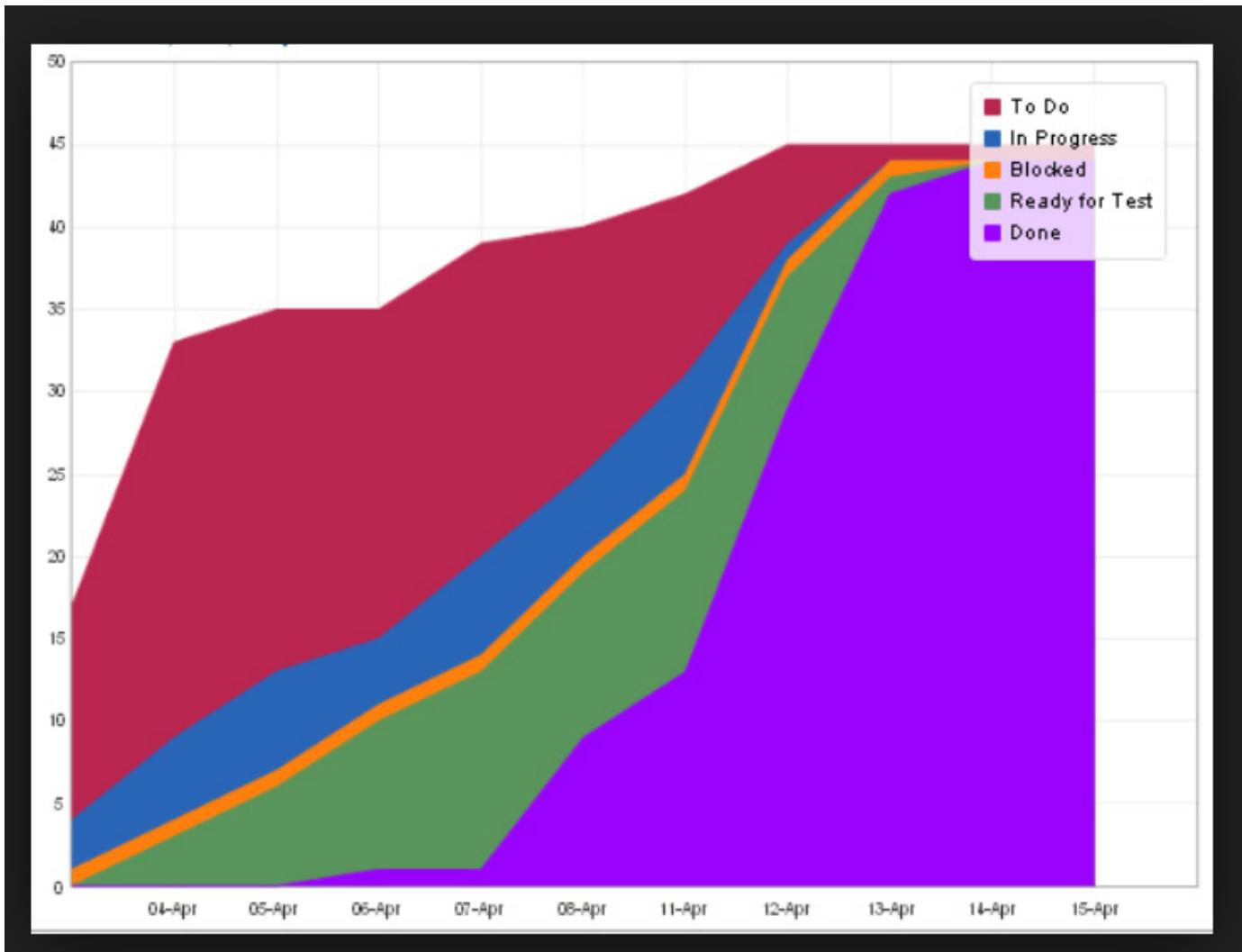
Sprint Retrospective

- The Sprint Retrospective occurs after the Sprint Review and prior to the next Sprint Planning.
 - Inspect how the last Sprint went with regards to people, relationships, process, and tools;
 - Identify and order the major items that went well and potential improvements; and,
 - Create a plan for implementing improvements to the way the Scrum Team does its work for the next sprint
 - The Scrum Team plans ways to increase product quality.
- The Scrum Master encourages the Scrum Team to improve, its process and practices to make it more effective and enjoyable for the next Sprint.
- This is a three-hour time-boxed meeting for one-month Sprints. For shorter Sprints, the event is usually shorter.

Burn down chart



Cumulative Flow Diagram

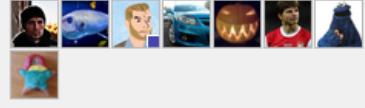


A Sprint Board on Trello

Trello  Help  Notifications  Boards 

BGR-100 Dev To-Do Delta Force 

To Do	Doing	Done
Overview page  0/1	PDF generation  4/8	Add a card...
Data Service Updates  0/1	Error Handling -> area information  1/7 	
Add a card...	Map Updates And Design changes  1   2/6 	
	Water Hardness Display  4/5 	
	Add a card...	

Members


[Add Members...](#)

Board
[Options](#) 
[Add List](#)
[Filter Cards](#)

Activity [View all...](#)

 Martin Jackson on PDF generation

- left
- completed Update Service to fix bug with other datafields being

Problems with SCRUM

- Small team methodology
- Co-location
- Doesn't fit well with all business processes
- Iteration planning is difficult
- Lack of clear development steps
 - In progress column on a scrum board does not give enough detail about what is happening to a task at any given point.
- Uncompleted Work
 - SCRUM focus is on delivering a batch of tasks or stories in one chunk
- Difficult to work on simultaneous projects within a team - Task switching