

# Intro: Agile over lunch

April 2007

Joe Little



# How much do you know about Agile?



Almost  
Nothing

Expert in  
Agile

# Attributions

- W Ken Schwaber and Jeff Sutherland for Scrum and some of the slides below.
- W Many friends and colleagues for ideas about how to present this information, especially my friends at CC Pace and Capital One.
- W Bill Wake for Scrum on a page.
- W The Poppendiecks for some of the Lean ideas.
- W Kent Beck, Ron Jeffries and Ward Cunningham for most of the XP ideas.
- W Mike Cohn and Bill Wake for INVEST and related things.
- W Alistair Cockburn, Esther Derby and Diana Larsen.
- W The Agile community which includes people in Lean and people in other fields who may not use the word “agile” at all.



# WIIFM (What's in it for me?)



- w More business value
- w Faster Time To Market (TTM)
- w Better work & sustainable pace
- w Better working relationships
- w More visibility/truth (less opacity and obfuscation)
- w More fun (“I wish I could be on that kinda team again.”)
- w More confidence  
*(but still no silver bullet <sigh>)*



# Agenda



- w The Basics
- w Tying with Lean
- w Principles
- w Your Questions



# The Basics

# Scrum Roles



Product Owner

- w Defines the features of the product, decides on release date and content
- w Is responsible for the profitability of the product (ROI)
- w Prioritizes features according to market value
- w Can change features and priority every 30 days
- w Accepts or rejects work results



ScrumMaster

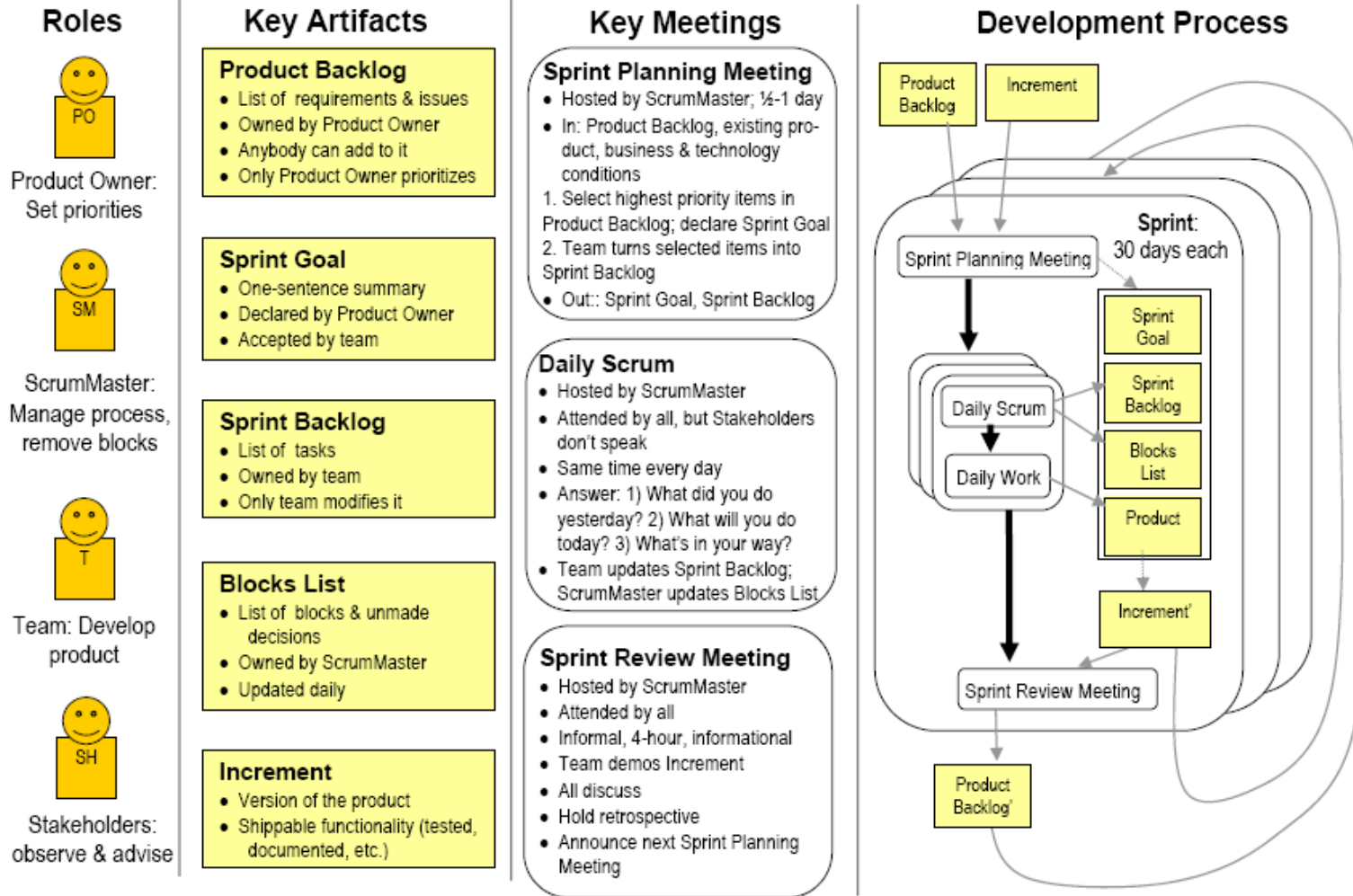
- w Ensures that the team is fully functional and productive
- w Enables close cooperation across all roles and functions and removes barriers
- w Shields the team from external interferences
- w Ensures that the process is followed. Invites to daily scrum, iteration review and planning meetings



Team

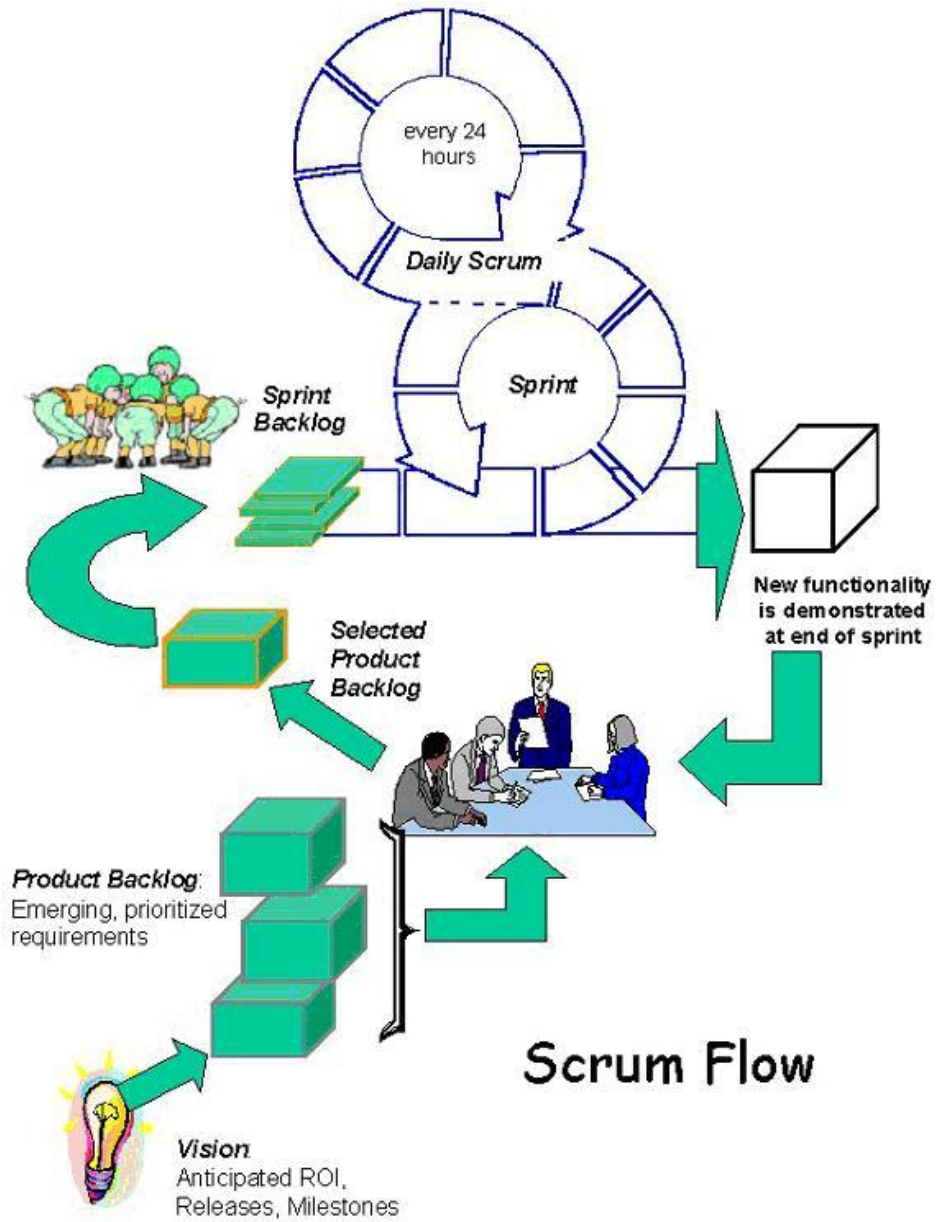
- w Cross-functional, seven plus/minus two members
- w Selects the iteration goal and specifies work results
- w Has the right to do everything within the boundaries of the project guidelines to reach the iteration goal
- w Organizes itself and its work
- w Demos work results to the Product Owner

# Scrum Development Process



Copyright 2004, William C. Wake, William.Wake@acm.org, [www.xp123.com](http://www.xp123.com)  
 Free for non-commercial use. 1-25-04







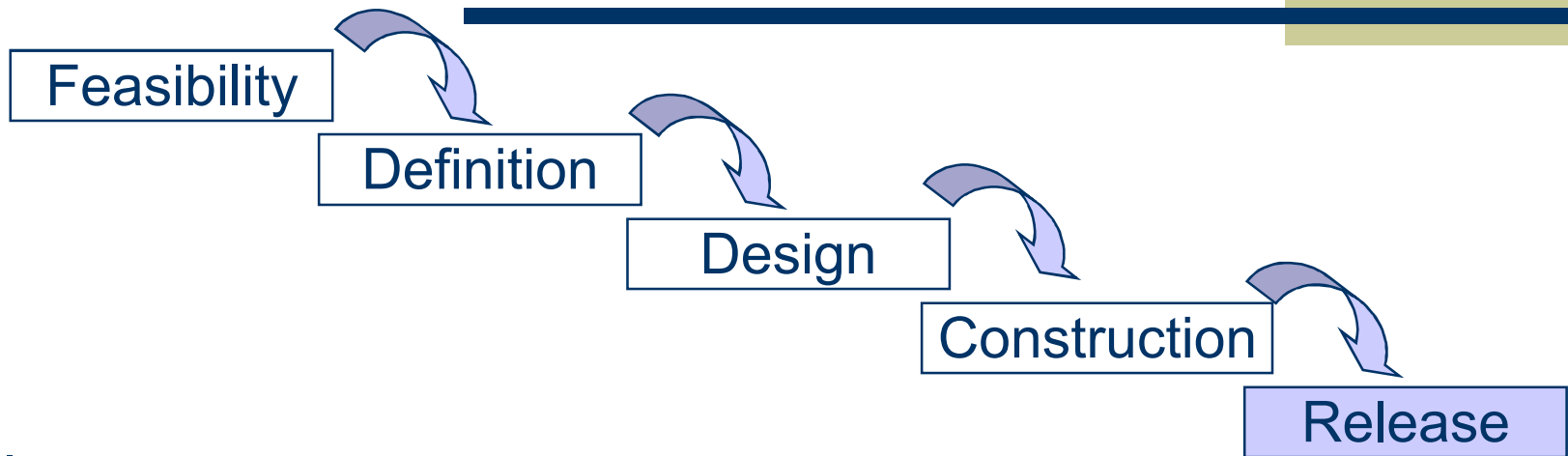
# Additional Practices



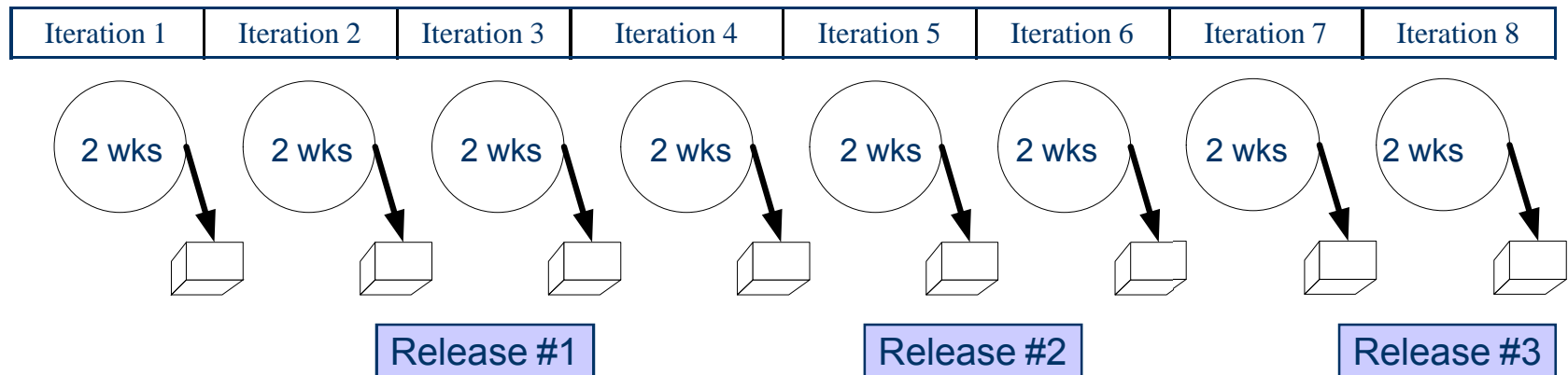
- w** Real Customer Involvement
- w** Sit Together
- w** Informative Workspace
- w** Stories
- w** Agile Estimating & Planning

# Waterfall vs. Agile – Delivering Incremental Value

Waterfall



Agile





# Tying with Lean

# 8 Forms of Waste (Muda)

## Manufacturing

1. In-process Inventory
2. Over-production
3. Extra processing
4. Transportation
5. Motion
6. Waiting
7. Defects

## Software Development

1. Partially done work
2. Extra features
3. Relearning
4. Handoffs
5. Task switching
6. Delays
7. Defects

Based on: Tom & Mary Poppendieck

# Lean Improvement Tools

5 S - Process for creating and maintaining an organized, clean, safe, high performance work place. Sort, Set in order, Shine, Standardize, Sustain

Mistake Proofing - Method which either makes it very, very difficult to produce defective work or makes the defect immediately obvious.

Visual Management - Practice of making process priorities and performance visually available to all performers in the process on a rapid, regular basis.

Work Balancing - Method to review the current number and grouping of process tasks so that they each step in a process takes a similar amount of time.

Pull Systems - Method to control the inventory & WIP in a system by only starting new work when work has been removed from the system by the customer.

Kanban – Card Signal. Method and system to insure the immediate availability of standard materials which minimizes inventory supports doing the next piece of work.

Kaizen Blitz Event - Focused improvement effort where a team will stop normal work and restructure their work and workplace in an intensive, dedicated, and structured way over a short period (eg, 5 days) with senior management support.

5 Why's - Looking for the root cause of any problem (not accepting the first answer).



# Principles

# The Agile Manifesto

*“We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:*

- n **Individuals and interactions** over processes and tools
- n **Working software** over comprehensive documentation
- n **Customer collaboration** over contract negotiation
- n **Responding to change** over following a plan

*“That is, while there is value in the items on the right,  
we value the items on the left more”*

Signed by: Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, Dave Thomas

© 2001, the above authors

this declaration may be freely copied in any form, but only in its entirety through this notice.  
[www.agilemanifesto.org](http://www.agilemanifesto.org)





# Principles behind the Agile Manifesto

---

---

1. *Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.*
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. *Business people and developers must work together daily throughout the project.*
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

# Principles behind the Agile Manifesto - 2

7. *Working software is the primary measure of progress.*
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. *Continuous attention to technical excellence and good design enhances agility.*
10. *Simplicity--the art of maximizing the amount of work not done--is essential.*
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



# Agile Engineering Practices



- w** Incremental requirements
- w** Incremental design
- w** Incremental planning
- w** Pairing
- w** Refactoring
- w** Define tests first
- w** Automated testing
- w** Continuous integration
- w** Test driven development



# Questions



---

# What are your Questions?

---

1. Are these things true about Agile? (The myths.)
2. What is the number one reason not to use Agile?
3. Other common questions

# Are these things true about Agile? NO!

It is a myth that Agile....:

- w Is a silver bullet
- w Will solve all my people issues
- w Will solve all my [X] issues
- w Agile won't work if...
- w Has no planning / documentation / architecture / <insert pet peeve>
- w Doesn't build on my previous experience & expertise
- w Is undisciplined or a license to hack
- w Creates quality issues
- w Is new and unproven
- w Is not being used by industry leaders





# Top 10 Reasons not to use Agile

---

---

10. No one knows what the business value is.
9. You don't have a team.
8. You don't have a good team.
7. The business does not wish to collaborate.
6. No one is willing to remove impediments.
5. Your area is not able to deal with the truth about projects.
4. A project manager needs to have command and control.
3. Nothing will change during the project.
2. You can't identify a good enough Product Owner.
1. Too many players don't want to play Agile.

# Common Questions - 1

- w Agile vs. Offshoring?
- w Can Agile be done with remote or offshore resources? Yes, if...
- w Which projects are the best candidates for Agile?
  - n Where the players are motivated to use it
  - n Where there is clear business value
  - n Where there is “change” or learning
- w How do I become a Product Owner or find one?
- w How allocated does the Team need to be? 100% is a good place to start. Really!





# Common Questions - 2



**w** What about “modified” Agile?

**w** Is Agile too loopy-goopy?

**w** Individual Performance Measurement vs  
Team Dedication

# Where Can I Get More Information?

- W Yahoo Discussion Groups:  
<http://groups.yahoo.com/group/scrumdevelopment/>  
<http://tech.groups.yahoo.com/group/extremeprogramming/>  
<http://finance.groups.yahoo.com/group/agileprojectmanagement/>
- W *Agile Project Management with Scrum*, Schwaber
- W *Agile Software Development with Scrum*, Schwaber and Beedle
- W *Lean Software Development*, Mary and Tom Poppendieck
- W *Crystal Clear*, Alistair Cockburn
- W *Extreme Programming Explained (2<sup>nd</sup> Ed.)*, Kent Beck
- W *Agile Project Management*, Jim Highsmith

# Where Can I Get More Information? Web resources

- W *Wikipedia: [http://en.wikipedia.org/wiki/Agile\\_software\\_development](http://en.wikipedia.org/wiki/Agile_software_development)*
- W *[www.agilealliance.com](http://www.agilealliance.com) articles: <http://agilealliancebeta.org/library>*
- W *[www.controlchaos.com](http://www.controlchaos.com) Ken Schwaber*
- W *[www.Scrumalliance.org](http://www.Scrumalliance.org)*
- W *[www.martinfowler.com/articles/newMethodology.html](http://www.martinfowler.com/articles/newMethodology.html)*
- W *[www.poppendieck.com](http://www.poppendieck.com) Mary & Tom Poppendieck*
- W *<http://www.estherderby.com/>*
- W *<http://www.futureworksconsulting.com/diana.html> Diana Larsen*
- W *<http://jeffsutherland.com/scrum/index.html>*
- W *<http://www.xprogramming.com/> Ron Jeffries*
- W *<http://www.threeriversinstitute.org/> Kent Beck*
- W *Google “agile” (30 million hits)*



# Appendices

# Scrum – Product Backlog

ID	Priority	Story	Cust	Est. pts	Acceptance Criteria
A	High	<b>Ongoing Support Story:</b> As a customer I require ongoing support for the following items: Defect Management Support, Change Control Support, TDD's, and Routing Support.	Marcus Williams	10	<p>Story A: Acceptance Criteria – Acceptable when:</p> <ul style="list-style-type: none"> <li>· A VRU resource attends Fraud/ProphIT meeting and fields questions from Lead BSA (Thursday)</li> <li>· A VRU resource checks-in with routing team to ensure support is adequate</li> <li>· The BAU production build is completed successfully</li> <li>· Call Flows accurately reflect what's in the VRU/Test Director(English)</li> </ul>
C	High	<b>Builds &amp; Deploy VRU Release to INT testing</b> - As a customer I request that a new build be created and deployed to the INT test environment during week 2 of the Sprint. At the end of each Sprint Cycle; and during the Sprint as it makes sense.	Chris Reilly	0	
D	High	<b>Support Defect Fixes and Issue Resolution</b> - As a customer I would require support for Defect fixes and issue resolution.	Marcus Williams	5	<ul style="list-style-type: none"> <li>* VRU team has identified the root cause and fix for the core dumping issue, Spanish language changes</li> <li>* Peer reviews are conducted</li> <li>* Code is integrated with call flows and regression testing is successfully performed (all test cases pass)</li> <li>• ICM is able to see Variable 1 with authenticated acct number from the VRU</li> </ul>
E	High	<b>Address Audit findings from Helix Req Plan audit</b> - As a customer I would like to have the VRU team to provide impact assessment of Audit findings and develop resolution plan	Cameron Peek	0	
H	High	As a customer I would like resources from the VRU team to Evaluate fortnight contingency plans for feasibility (provided by business) Size impact and effort, and determine what pre-work needs to be done Enter phrase request for any messages	Brian Cerasuolo		The deliverable of this story will be a Contingency Plan Document which will list what steps will be taken under what circumstances and estimates of how quickly the prescribed measures could be implemented. This doc should be reviewed with customer.
I	Medium	<b>Execute Spanish Testing</b> - As a customer I want the VRU team to execute Helix test cases in Spanish.	Marcus Williams	8	All Spanish test cases for SAC and Statement Tab have been executed and pass (using 800-955-7070 and the Spanish direct number) Test Cases in Test Director are updated to accurately reflect what's in the call flow/VRU and to reflect use of Spanish # during testing Customer Walkthrough is performed.
J	Medium	<b>Update NMS Card to play phrases</b> - As a customer, I want the VRU team to update the NMS card to play each of the new formats provided by Walsh so that a final format for phrases can be identified.	Rck Duggan		Acceptable when the VRU plays each of the Walsh formats, an assessment of the feasibility of playing each of the formats is conducted, and a final format selected and given to Walsh for conversion.

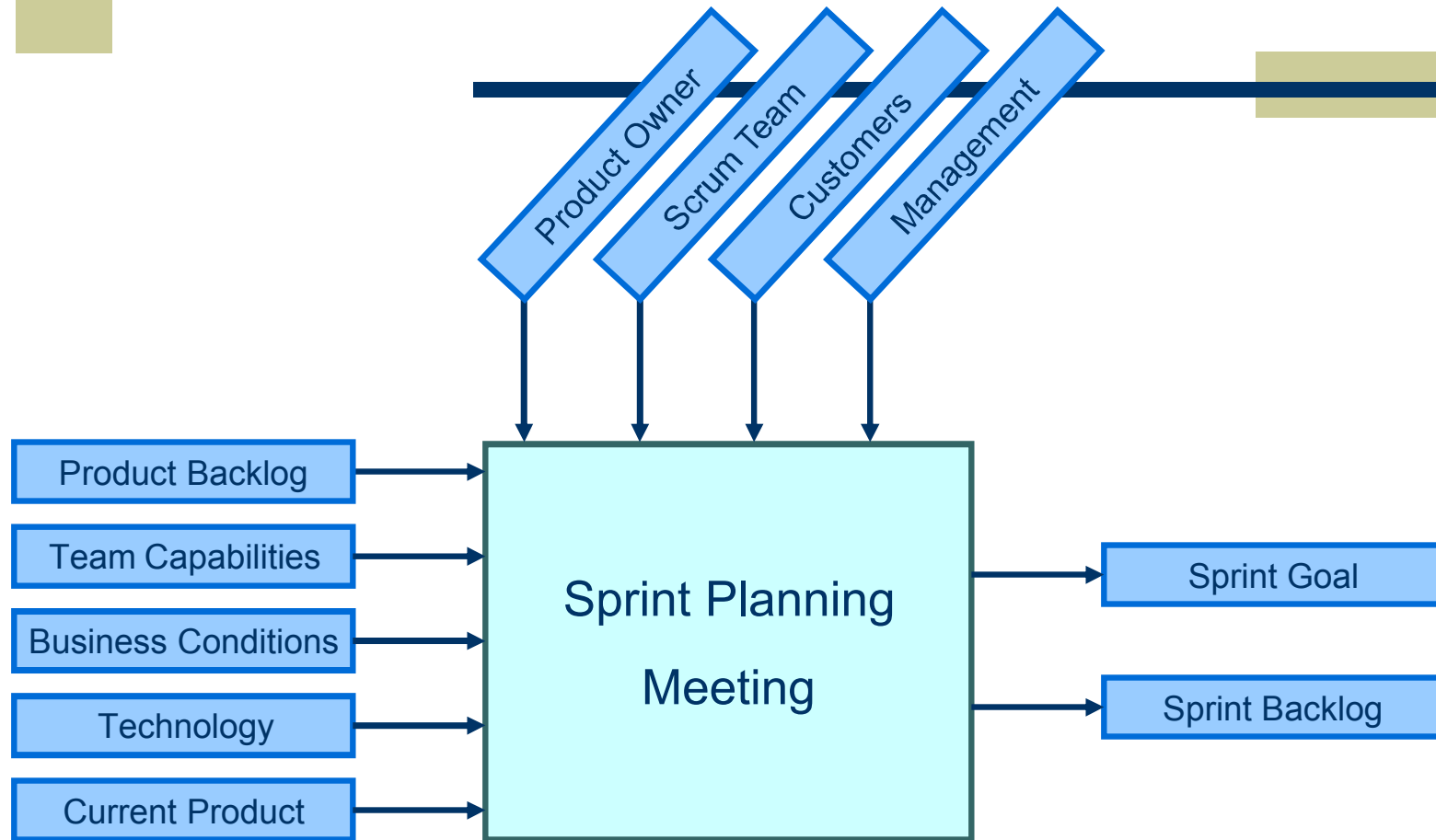


# Requirements – User Stories



- ∅ Independent
- ∅ Negotiable
- ∅ Valuable
- ∅ Estimatable
- ∅ Sized-right
- ∅ Testable

# Sprint Planning Meeting



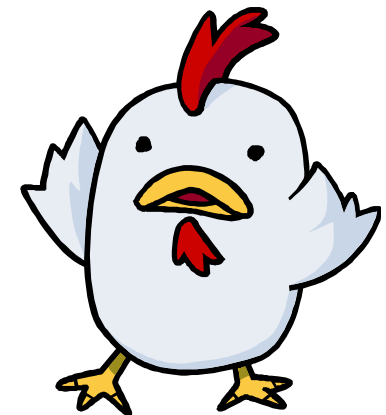
# Daily Scrum Meetings (Stand Up)

## W Parameters

- n Daily
- n 15-minutes timebox
- n Participants stand
- n Story and Task focused
- n Not for problem solving

## W Three questions:

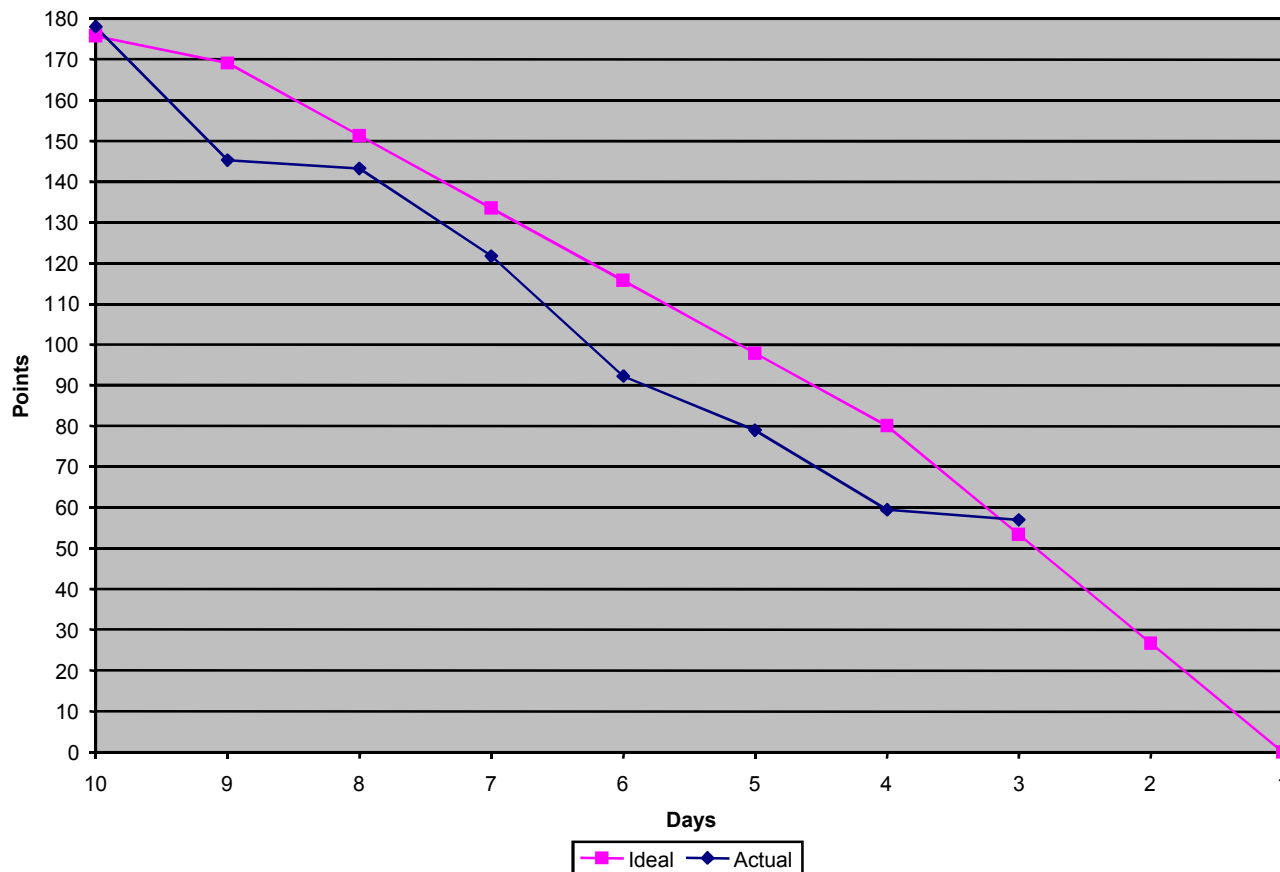
1. What did you do yesterday?
2. What will you do today?
3. What obstacles are in your way?



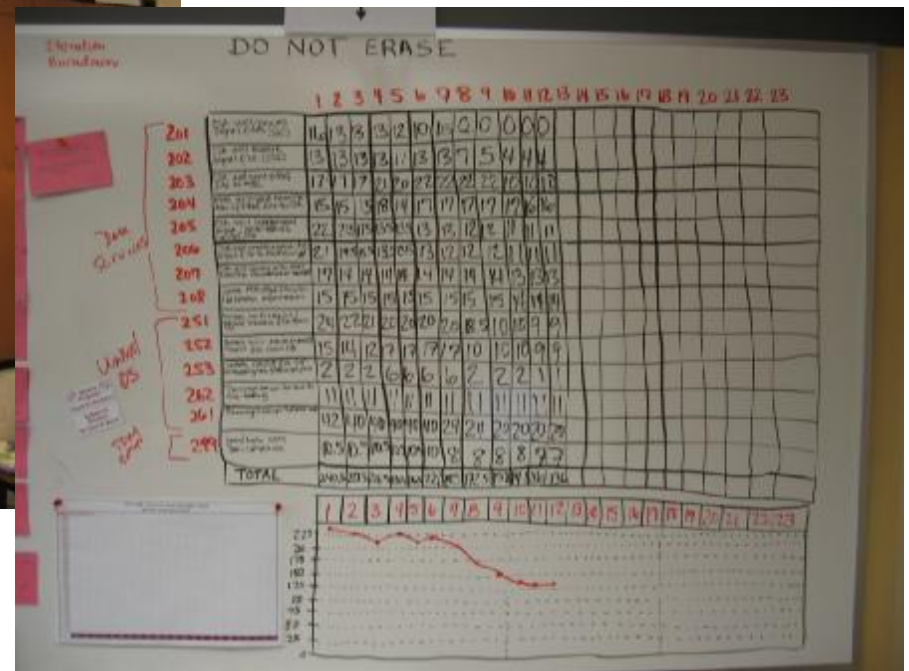


# Scrum – Sprint Burndown

Sprint '15' Burndown Chart

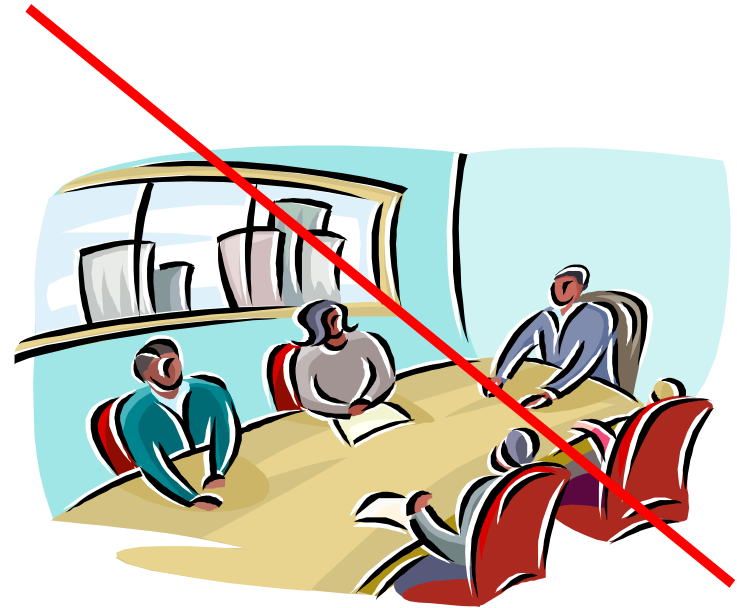


# Visual Workspaces and Information Radiators



# Sprint Review Meeting

- w Team presents what it accomplished during the sprint
- w Typically takes the form of a demo of new features or underlying architecture
- w *Informal*
  - n *No prep for ceremony (maybe for content)*
- w Participants
  - n Team (incl. Product Owner)
  - n Customers
  - n Management
  - n Other Interested Parties





# Sprint Retrospective Meeting

- W** Team discusses opportunities to improve process
- W** Promotes continuous improvement, adaptive learning
- W** Informal
  - n** 1.5 hours after a 2 week iteration
- W** Participants
  - n** Team (incl. Product Owner)

# Glossary of Terms

- W** Agile: A conceptual framework for undertaking software engineering (typically) projects. Agile methods are a family of development approaches. For example, a Team can use Scrum (one approach) and Extreme Programming (another approach) together.
- W** Burndown Charts: A burndown chart is a simple visual tool for measuring and managing project progress. Visually, a burndown chart is nothing more than a line chart representing remaining work over time. Burndown charts are used to measure the progress of an agile project at both a micro and macro level.
- W** Daily Standup: Each day the Scrum Master leads the team in the Daily Scrum Meeting. Fifteen-minute time-boxed meeting. Makes visible the status of the work in the iteration. Each team member speaks to three questions: what did I do yesterday, what did I do today, and what impediments got in my way?
- W** Product Backlog: Acts as a repository for ‘work orders’ targeted for release at some point. Many firms use user stories. These are typically high level requirements with high level estimates provided by the product owner and team.
- W** Product Owner: The person who represents the customers to the Team is termed the product owner. This person is knowledgeable of the business and can provide the “Why” behind the work. Ideally the Product Owner should be able to communicate the requirements in detail, have all the capabilities needed for decision-making (eg, prioritization) of the Product Backlog, and be able to evaluate success.

# Glossary of Terms – Cont'd

- W** Retrospective: It is analogous to lessons learned and is based on the principles of applying the learning from the previous sprint to the upcoming sprint or later projects. Agile projects aim to use Retrospectives during the life of the project rather than after catastrophic failure!
- W** Scrum: Scrum is an agile (adaptive) method for software development (and project management), considered to be less burdensome than heavy SDLCs or methodologies (aka waterfall) and at the same time more rigorous in the ways we need. Scrum is named after the Scrum in rugby.
- W** ScrumMaster/Agile Coach: The team facilitator is called the ScrumMaster. His/her job is to implement and manage the Scrum process in the project, to manage impediments, and to assist the team in achieving higher productivity (releasing more business value). The ScrumMasters is like a coach in sports, working with the Team but not playing the game. Aside from being the referee on Scrum rules, they have no “authority”.
- W** Sprint / Iteration: The Sprint is where the software development takes place. A Sprint includes all the activities needed to produce working software. The duration of the sprint varies, and can range from 1 week to 4 weeks.

# Glossary of Terms – Cont'd

- W** **Sprint Review:** Each Sprint is followed by a Sprint review. The working software developed in the previous Sprint is reviewed and if necessary product backlog items are added/modified. The team shows the Product Owner what they have done. Others may attend (stakeholders, managers, customer types). Other work is also reviewed.
- W** **Sprint Backlog:** Starting each sprint, the team has sprint planning, which produces a sprint backlog. This includes the stories (requirements) they promise to complete, along with task assignments and LOEs for each task. These are the items that the team will "Burndown" against throughout the duration of the sprint
- W** **Sprint Planning:** Is attended by the Product Owner, Scrum Master, the entire Team. (Additional parties may provide information.) During the meeting the Product Owner describes the highest priority features to the team. There is a set of dialogues between the team and the Product Owner. The team promises to deliver a set of Product Backlog items in that Sprint. And creates the Sprint Backlog of tasks that will enable that delivery.
- W** **Team Velocity:** A relative number which describes how much work (eg, stories) the team is likely to get done in the next iteration(s)
- W** **User Stories:** A user story is a very high-level definition of a requirement. It is a placeholder for a series of conversations between the Product Owner and the Team, and a token of the work involved (eg, stories are used for planning).

# Who is Joe Little?

- w** An Agile coach and project witch doctor. CSP.  
20+ years doing projects of many types.
- w** A businessman and consultant with an MBA.
- w** <http://www.kittyhawkconsulting.com/>
- w** <http://agileconsortium.blogspot.com/>
- w** <http://agileconsortium.pbwiki.com/>
- w** [Jhlittle@kittyhawkconsulting.com](mailto:Jhlittle@kittyhawkconsulting.com)
- w** 917-887-1669 or 704-376-8881