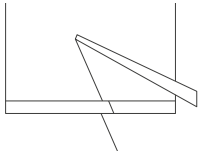
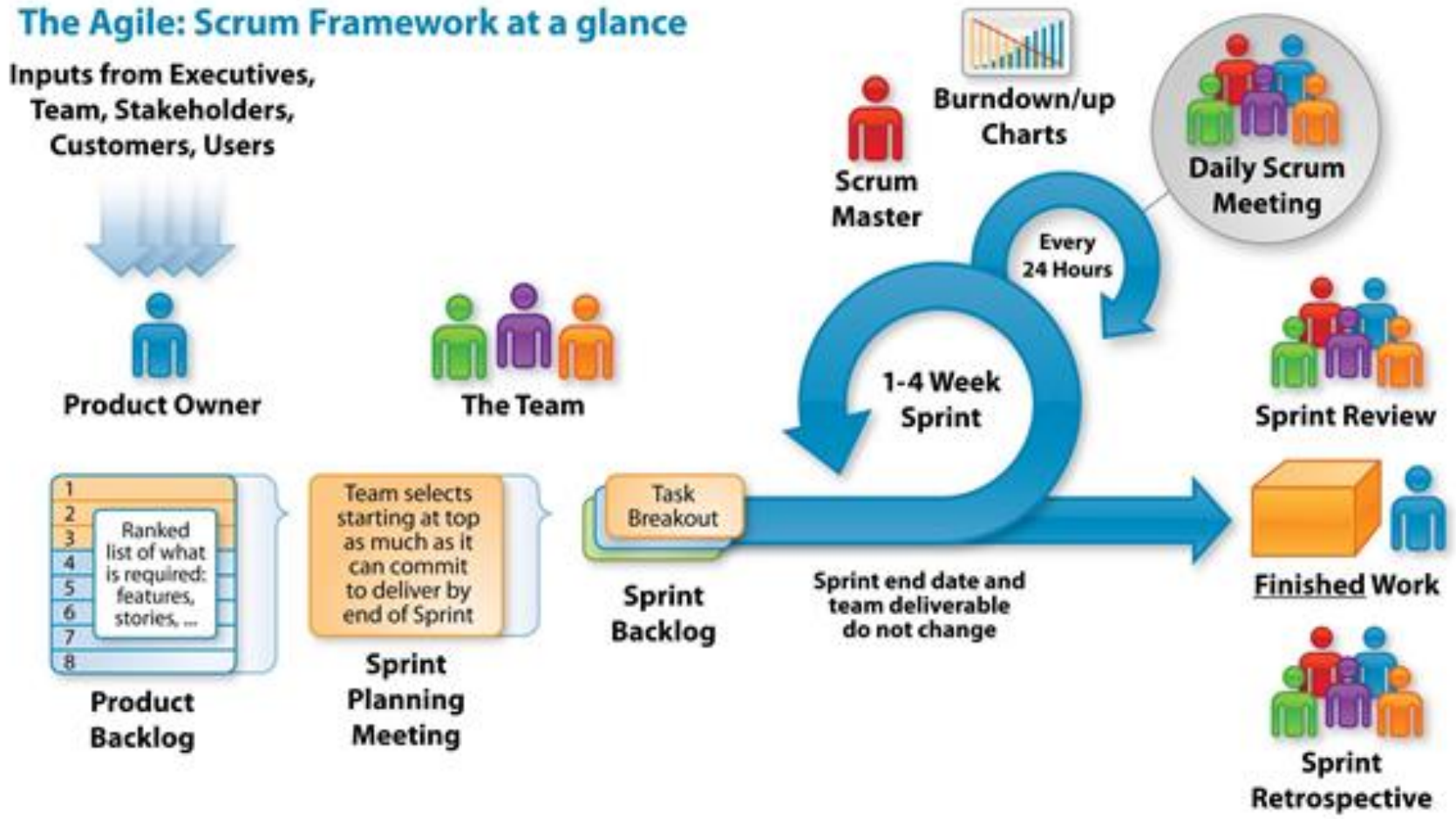
Bring ideas to life
VIA University College



SCRUM and Unified Process

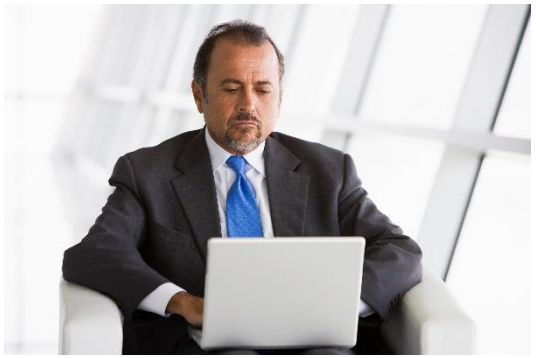
SWE1

What is SCRUM



Source: http://www.ness-tech.co.il/en/solutions/testing-qa-v-ness/Scrum-Roles

External SCRUM Roles



Manager

Customer

User

External SCRUM Roles

Customer

- Has the vision and idea for the product/project

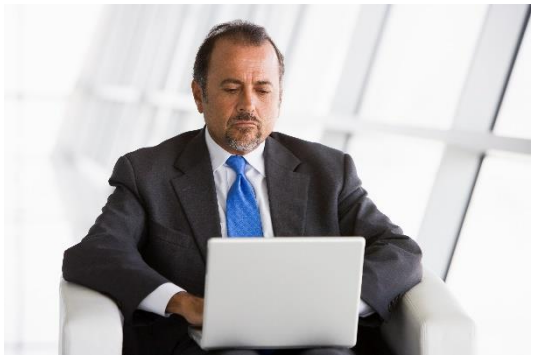
Manager

- Sponsor for the project
- Resources, equipment etc.

End Users

- Uses the final product

Internal SCRUM Roles



Manager

SCRUM Master

Customer

Product Owner

Team

User

Internal SCRUM Roles

Product Owner

- The interface to the Customer
- Find the requirements
- Manage and prioritise Product

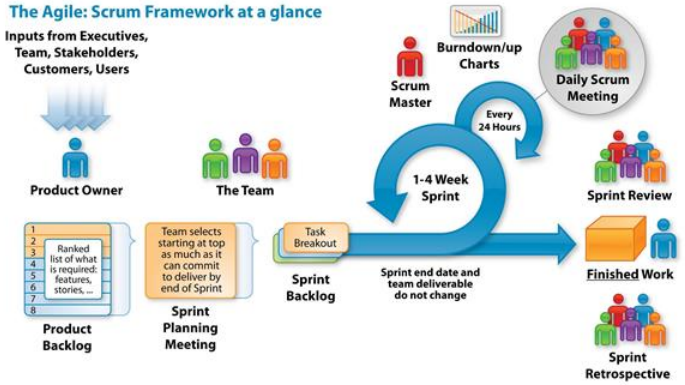
SCRUM Master

- Process/Team facilitator
- Removes obstacles
- Works with the Product Owner

SCRUM Team

- Implements increments of deliverable software
- Estimates size of Backlog Items
- Accountable to Product Owner for delivering as promised

Product Backlog



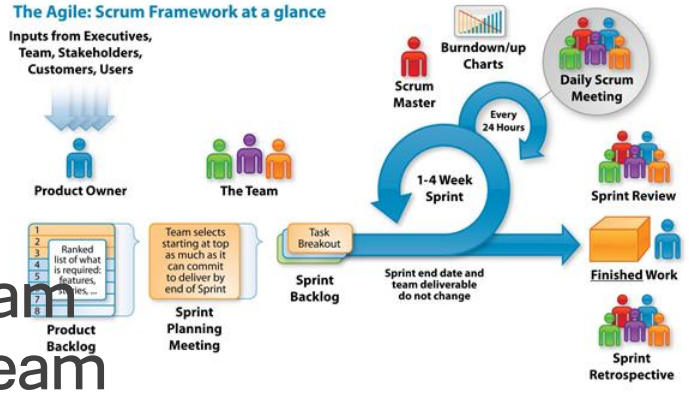
Backlog Items/Stories

Format:
"As a *user-role*[1] I want a *feature*[2] so that I get *business value/benefit*[3]."

These are the three most important things to mention!

Items/Stories **must never** include design ideas – only describe wanted functionality!!

Sprint Planning Meeting



Participants

- Product owner, Scrum Master and the entire Scrum team
- Outside stakeholders may attend by invitation of the team

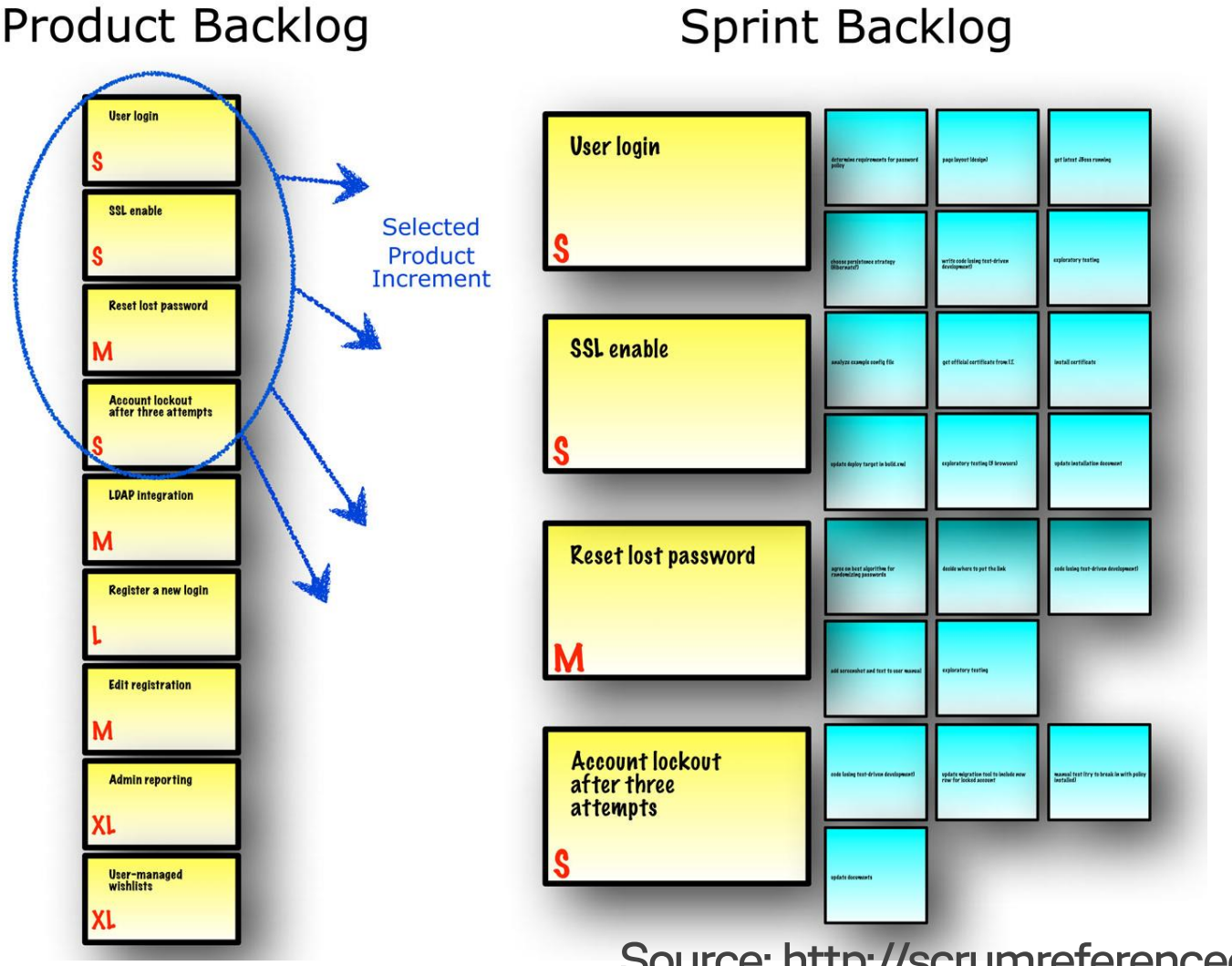
Plan your next Sprint

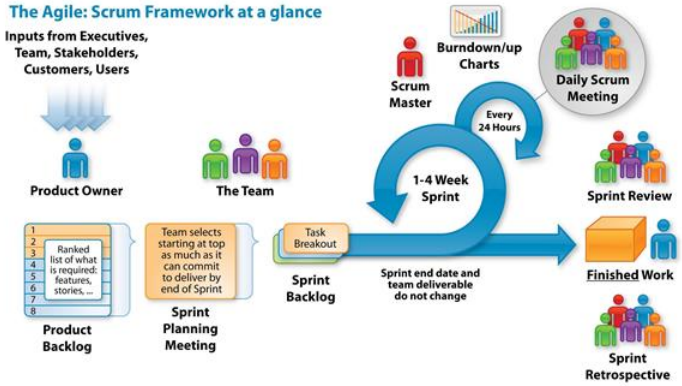
- Product owner describes the highest priority features to the team.
- The team asks enough questions that they can turn a high-level user story of the product backlog into the more detailed tasks of the sprint backlog
- Pick tasks from the backlog to be completed in your next Sprint
- The team decides what tasks to include in this Sprint
- And who will be responsible for them (not always)

Result

- Define Sprint goal
- Sprint Backlog

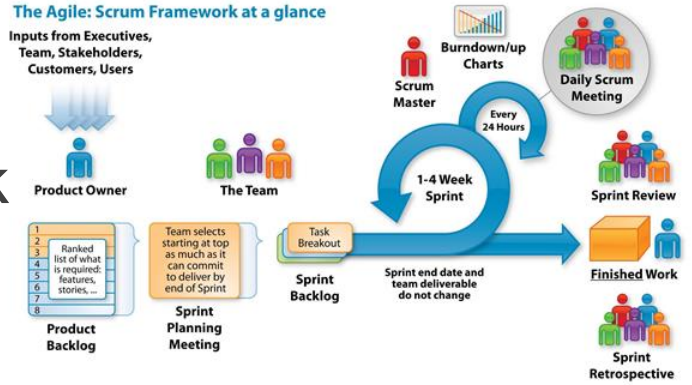
Sprint Planning Meeting





Source: http://scrumreferencecard.com/scrum-reference-card/

In the Sprint



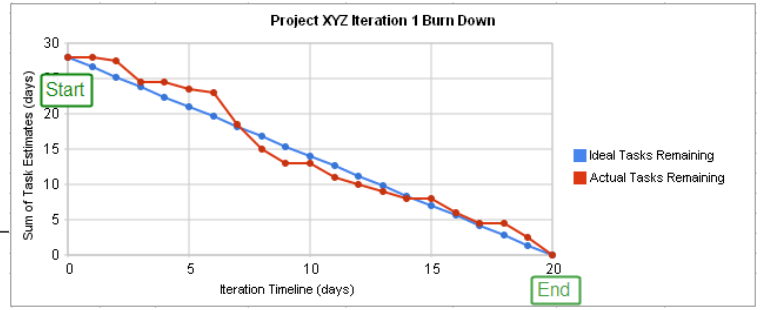
Team members picks tasks to work on and starts the work

Daily Scrum Meeting (max 15 minutes!!!)

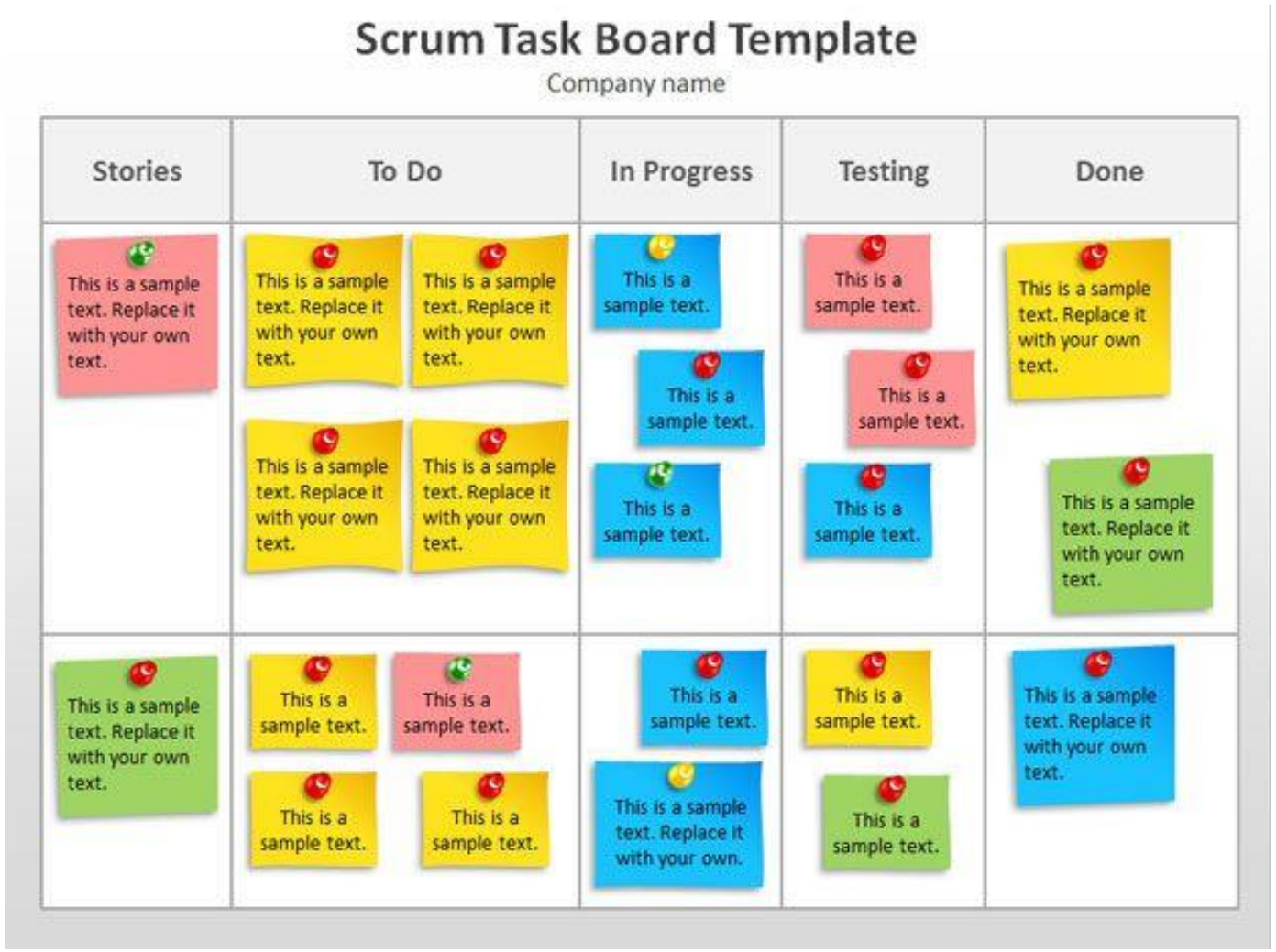
- Each member answers **these** three Questions
  1. What has he/she accomplished since the last daily Scrum meeting?
  2. What is he/she is going to accomplish until the next Scrum meeting?
  3. What are the impediments/problems that prevent him/her from accomplishing his/her tasks?
- The answers are **commitments** from the team members!!

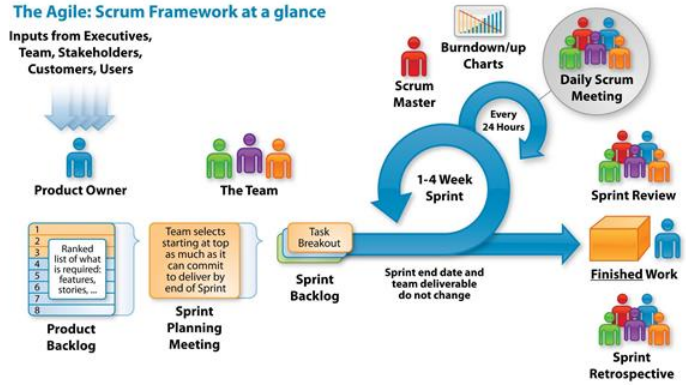
Each member is responsible for daily estimating remaining work on current task

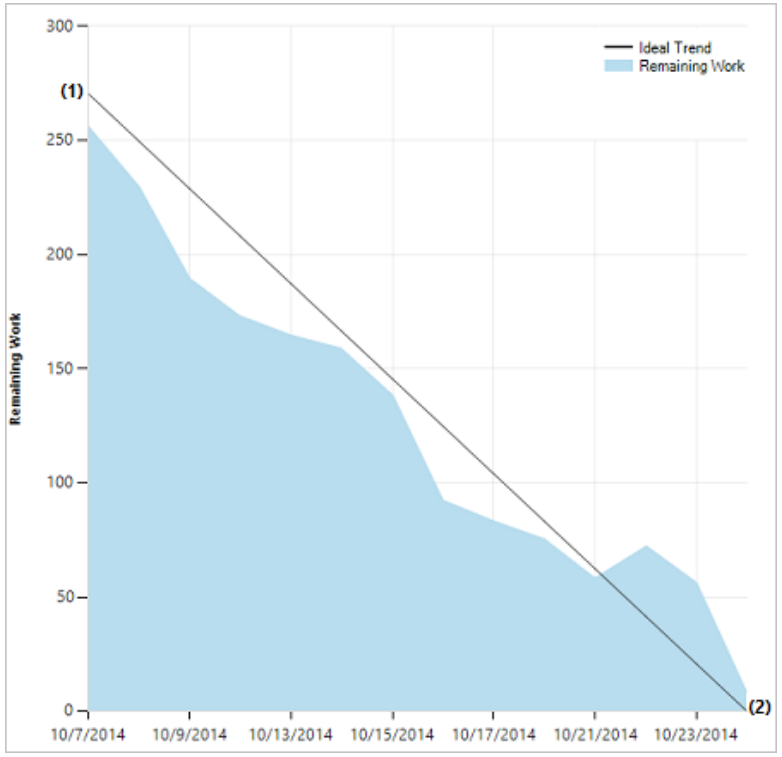
- That updates the Burn-down Chart
- Tool-examples:
  https://www.visualstudio.com/team-services/
  https://trello.com/



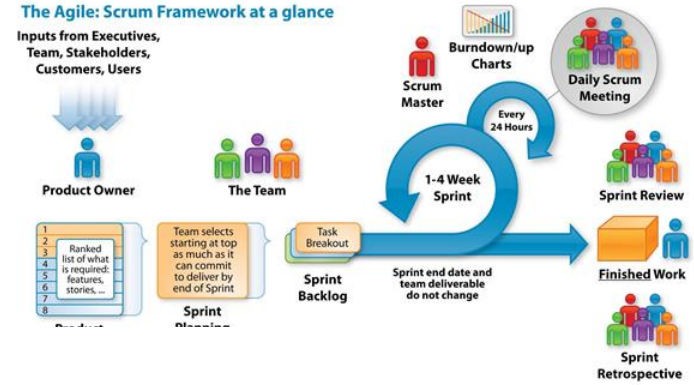
In the Sprint



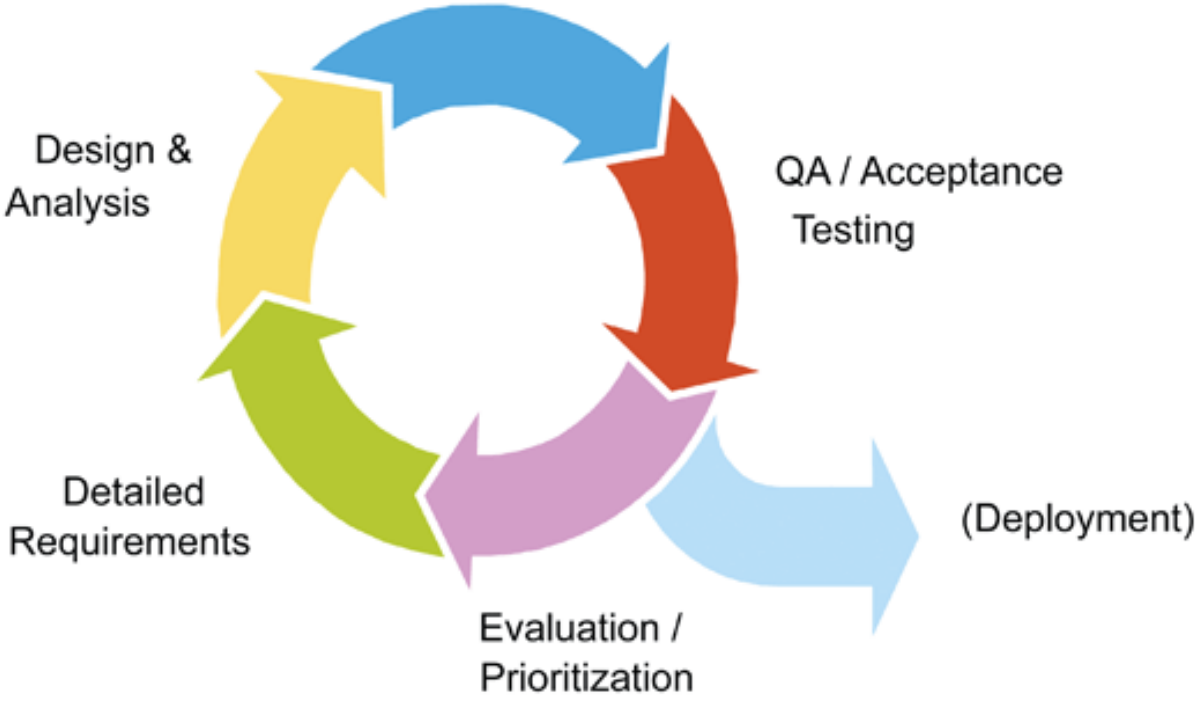




In the Sprint

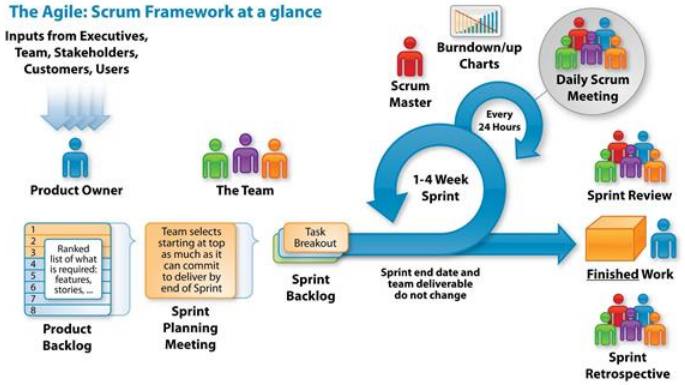


Iteration Detail



Source: http://scrumreferencecard.com/scrum-reference-card/

Sprint Review



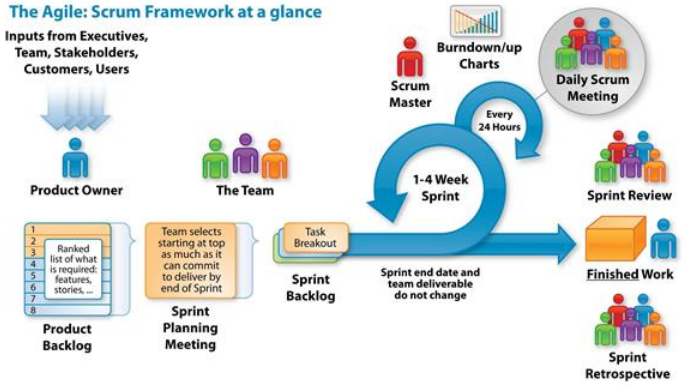
Participants

- Product owner, the Scrum team, the Scrum Master, management, customers and developers from other projects

At the end of each sprint, the team has produced a analysed, designed, coded, tested documented and usable piece of software

The Scrum team shows what they accomplished during the sprint. Typically this takes the form of a demo of the new features

Sprint Retrospective up to an hour!



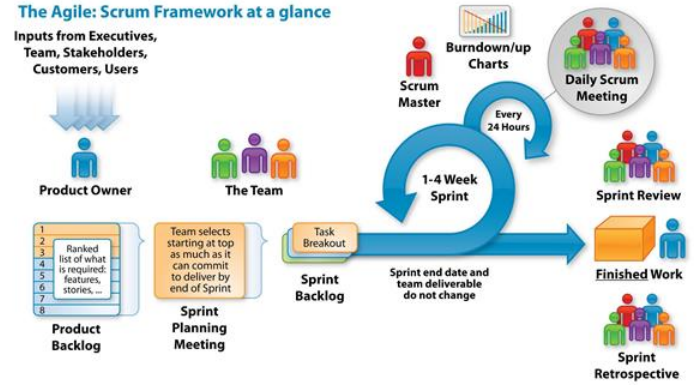
Participants

- The entire team, including both the Scrum Master and the product owner should participate

Each team member is asked to identify specific things that the team should

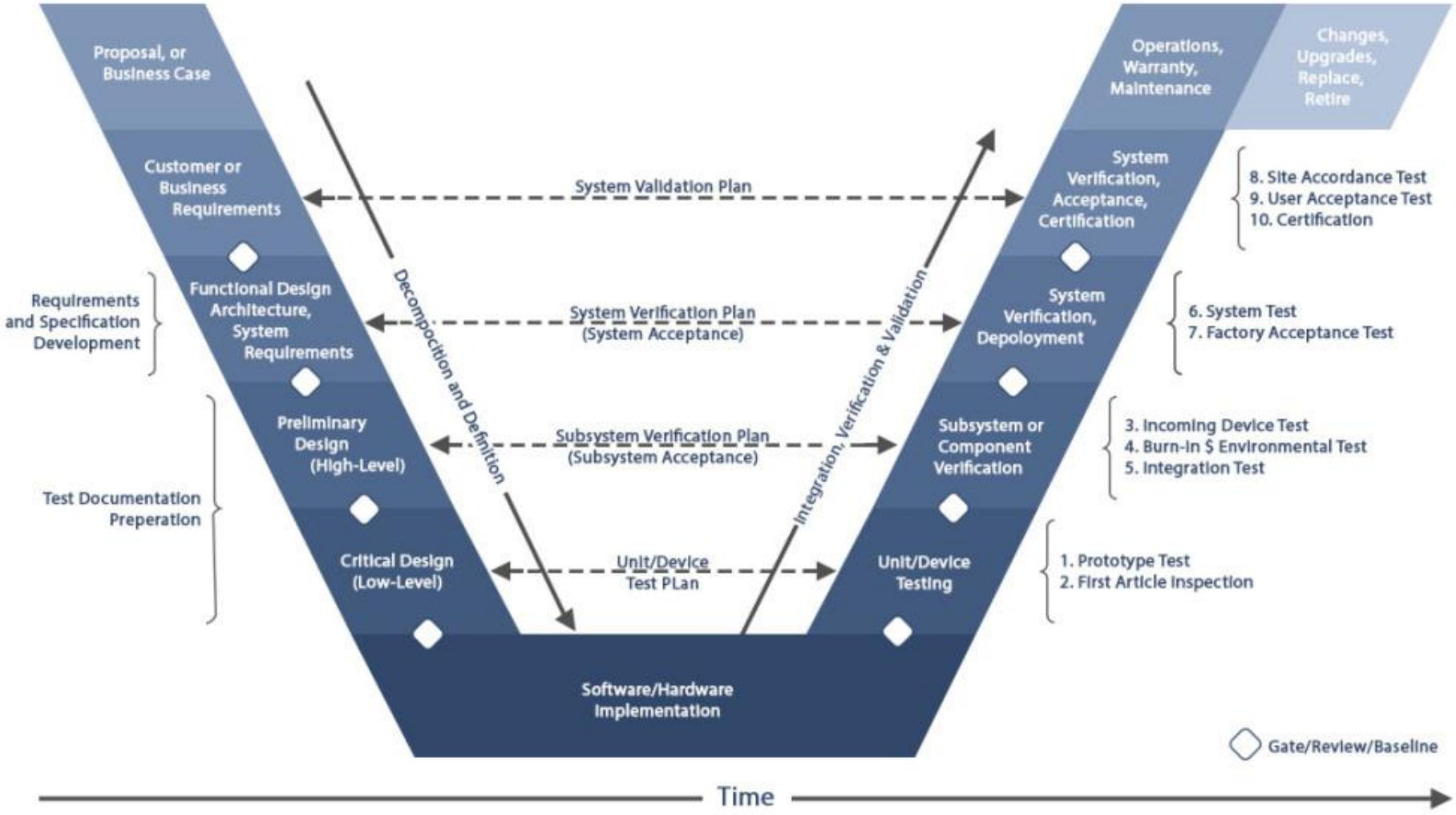
- Start doing
- Stop doing
- Continue doing

What is missing in SCRUM?



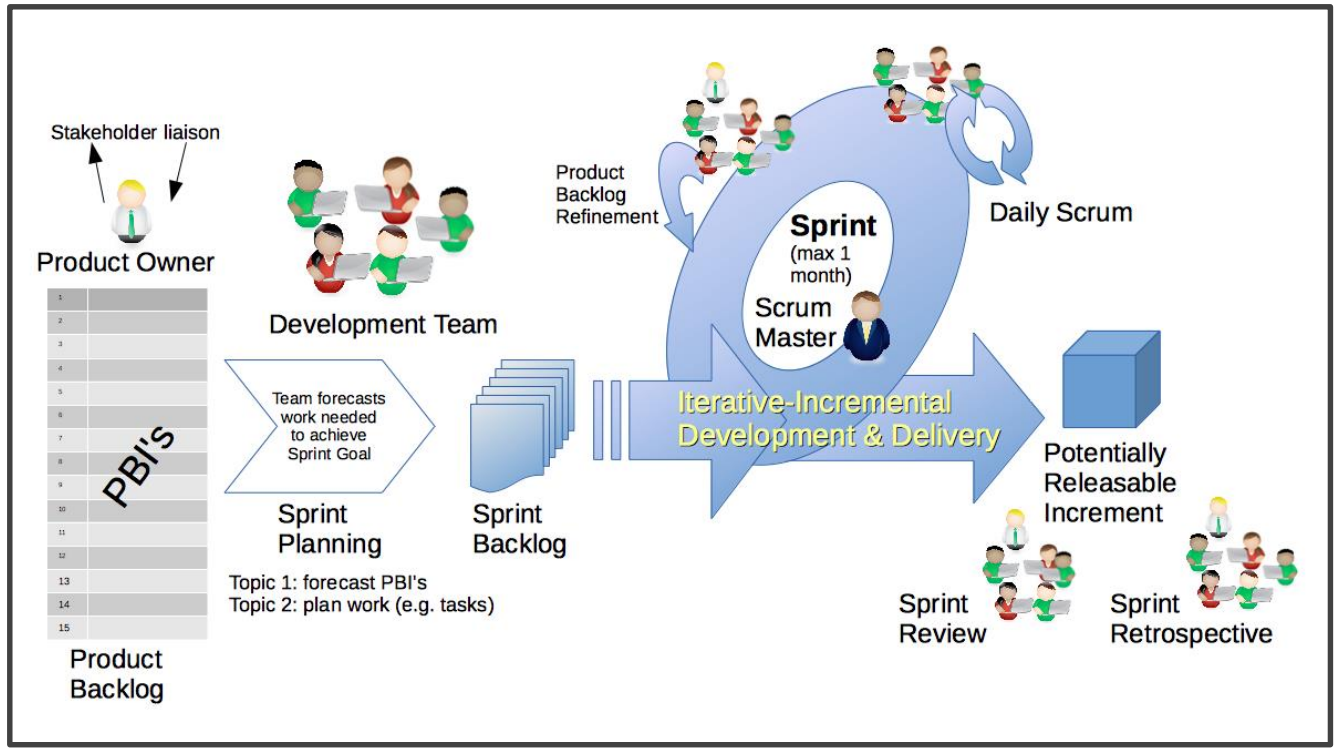
Discuss two and two in 5 minutes if there is anything you feel is missing in SCRUM

V-Model & Test


Proposal, or Business Case
Customer or Business Requirements
Functional Design Architecture, System Requirements
Preliminary Design (High-Level)
Critical Design (Low-Level)
Software/Hardware Implementation
Unit/Device Testing
Subsystem or Component Verification
System Verification, Depoloyment
System Verification, Acceptance, Certification
Operations, Warranty, Maintenance
Changes, Upgrades, Replace, Retire
Requirements and Specification Development
Test Documentation Preperation
Decompcition and Definition
Integration, Verification & Validation
System Validation Plan
System Verification Plan (System Acceptance)
Subsystem Verification Plan (Subsystem Acceptance)
Unit/Device Test PLan
8. Site Accordance Test
9. User Acceptance Test
10. Certification
6. System Test
7. Factory Acceptance Test
3. Incoming Device Test
4. Burn-in $ Environmental Test
5. Integration Test
1. Prototype Test
2. First Article Inspection
Gate/Review/Baseline
Time

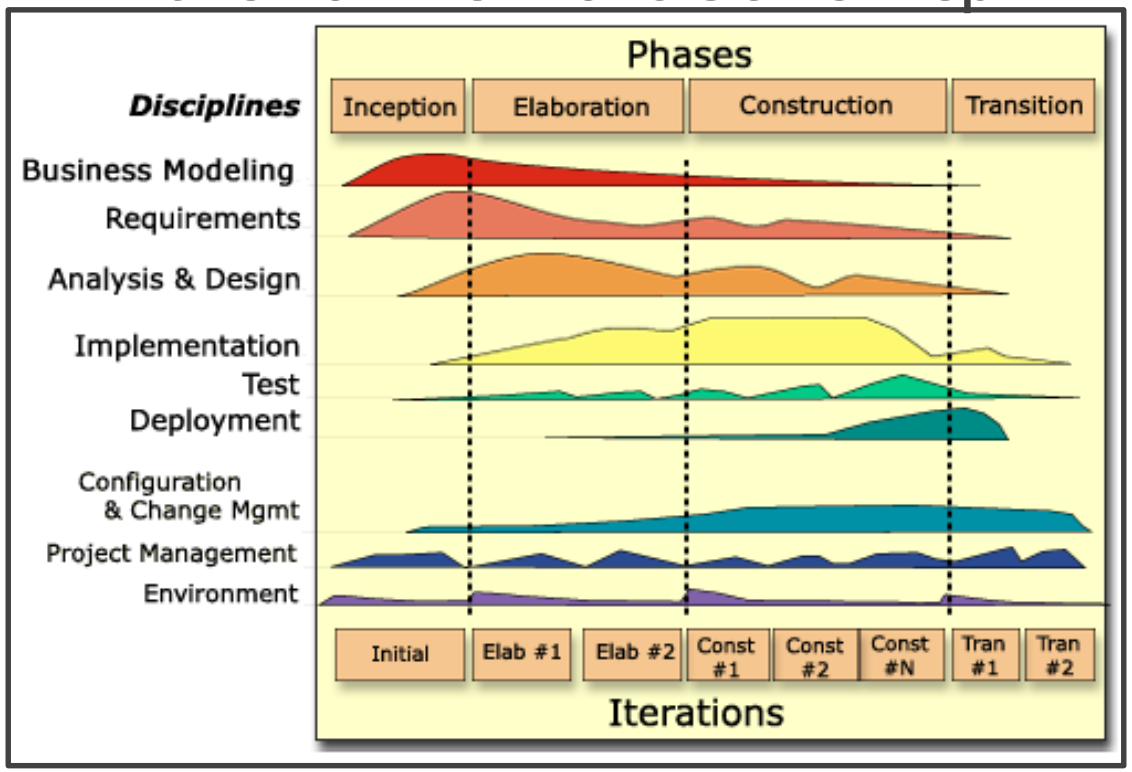
SCRUM vs Unified Process

Mainly Process



Process, disciplines and artifacts

- Elaboration, Construction and Transition fits in one SCRUM sprint



Why making Artifacts?

The point of creating artifacts and documents

- **Is not** the diagram or the document itself
- **It is** the thinking, analysis and proactive readiness!!

The purpose of modelling is to improve the understanding!!

"In preparing for battle I always found that plans are useless, but the planning indispensable"

- General Eisenhower

Inception

We must be able to answer these questions after the inception phase

- What is the vision and business case for this project?
- Is it feasible?
- Buy and/or build? – Are there already something on the market?
- Rough unreliable range of cost: Is it $10K-100 or in millions?
- Should we proceed or stop?

The goal is only to: Decide if the project is worth a serious investigation (during elaboration) – not to do the investigation!

Inception Artifacts

Artifact	Comment
Vision and Business Case	Describes the high-level goals and constraints, the business case, and provides an executive summary.
Use-Case Model	Describes the functional requirements. During inception, the names of most use cases will be identified, and perhaps 10% of the use cases will be analyzed in detail.
Supplementary Specification	Describes other requirements, mostly non-functional. During inception, it is useful to have some idea of the key non-functional requirements that have will have a major impact on the architecture.
Glossary	Key domain terminology, and data dictionary.
Risk List & Risk Management Plan	Describes the risks (business, technical, resource, schedule) and ideas for their mitigation or response.
Prototypes and proof-of-concepts	To clarify the vision, and validate technical ideas.
Iteration Plan	Describes what to do in the first elaboration iteration.
Phase Plan & Software Development Plan	Low-precision guess for elaboration phase duration and effort. Tools, people, education, and other resources.
Development Case	A description of the customized UP steps and artifacts for this project. In the UP, one always customizes it for the project.

Artifacts are only partly finished in inception!!

[Larman, 2005] Table 4.1

What requirements to focus on?
Inception

10% - 20% of the Architectural significant, Risky and High Business-value requirements

Inception vs. SCRUM

Often a time boxed discipline where we get a better understanding of the problem domain and the problem at hand

Here you write the Project Description!!

Typically done before any SCRUM sprints are started

Elaboration

We must be able to answer these questions after the elaboration phase

- Have the vision been refined to our new knowledge?
- Is the iteratively build core architecture good for the rest of the system?
- Are the main part of the requirements identified?
- Is it possible to create a more realistic estimate for the project?

This is the phase where the core architecture of the system is established, and high risk issues are mitigated!

After the elaboration phase there is typical a milestone where it is decided if the project should be stopped or continued

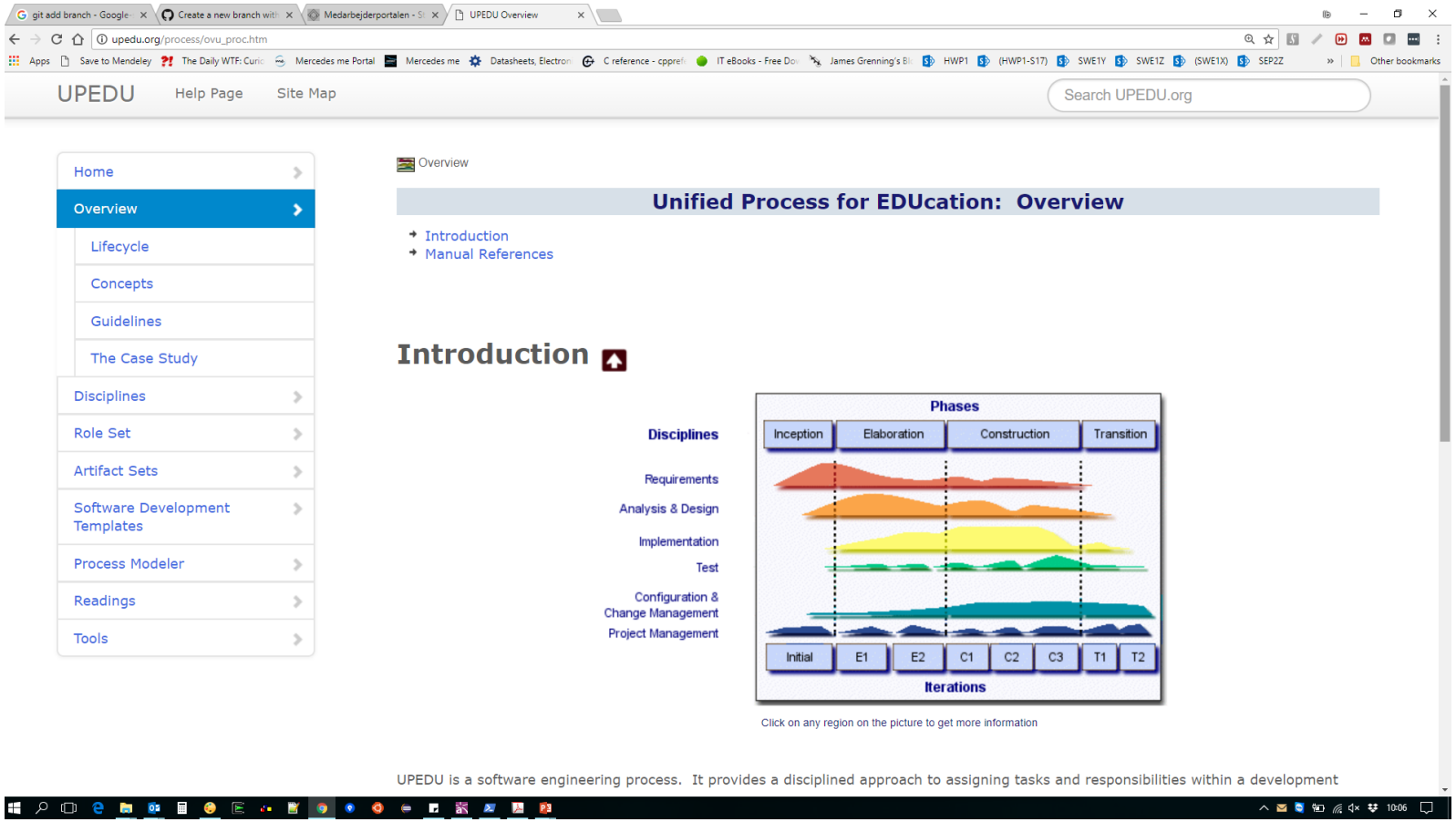
Elaboration Artifacts

Artifact	Comment
Domain Model	This is a visualization of the domain concepts; it is similar to a static information model of the domain entities.
Design Model	This is the set of diagrams that describes the logical design. This includes software class diagrams, object interaction diagrams, package diagrams, and so forth.
Software Architecture Document	A learning aid that summarizes the key architectural issues and their resolution in the design. It is a summary of the outstanding design ideas and their motivation in the system.
Data Model	This includes the database schemas, and the mapping strategies between object and non-object representations.
Use-Case Storyboards, UI Prototypes	A description of the user interface, paths of navigation, usability models, and so forth.

[Larman, 2005] Table 8.1

Unified Process for Education

http://upedu.org/



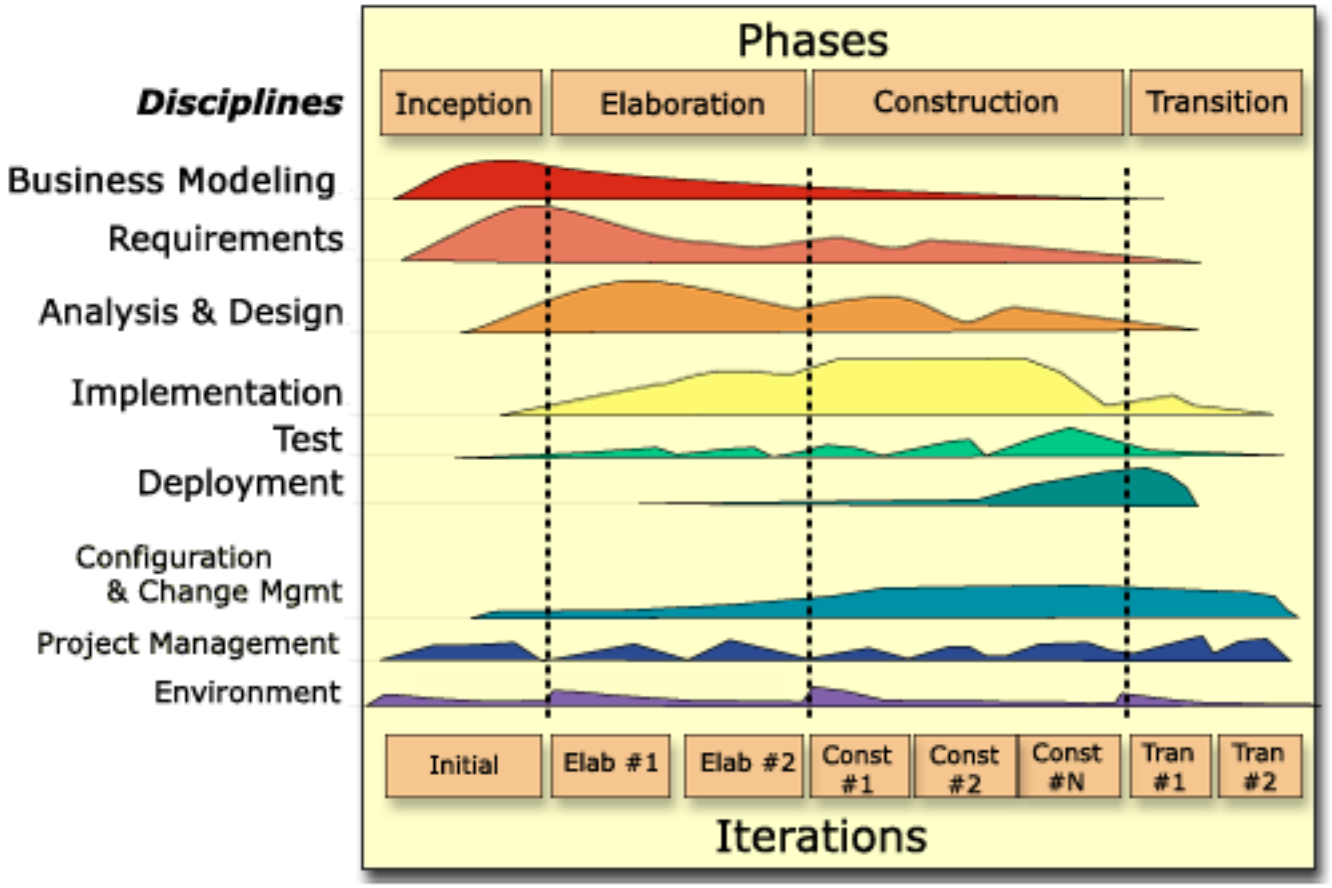
The Work in the SCRUM Sprint

In each Sprint you have to **analyse, design, implement, test** and **document** one or more Backlog Items/User Stories

In each sprint you typically follow the Unified Process phases: Elaboration, Construction and Transition

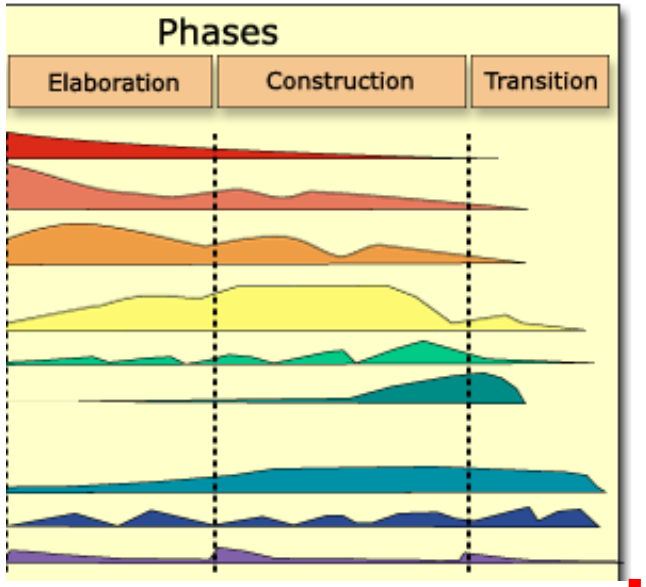
Even in an Agile environment you will be doing

- Business Modeling
- Requirements
- Analysis
- Design
- Implementation
- Test
- Etc.



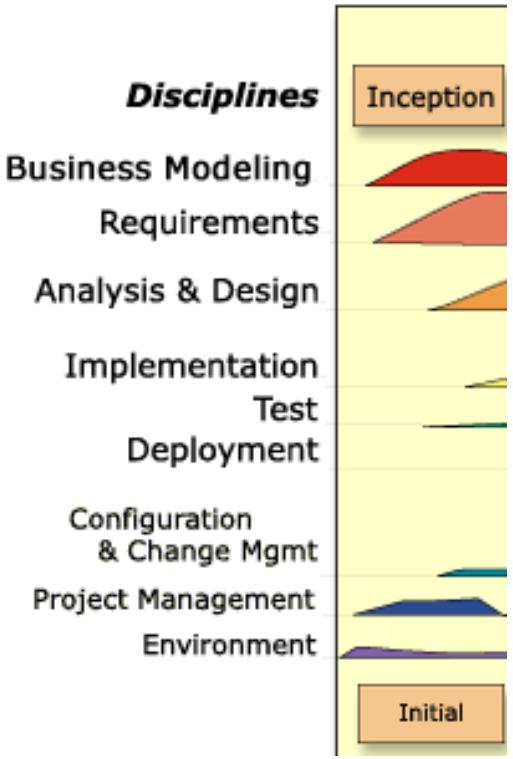
Conclusion SCRUM with UP

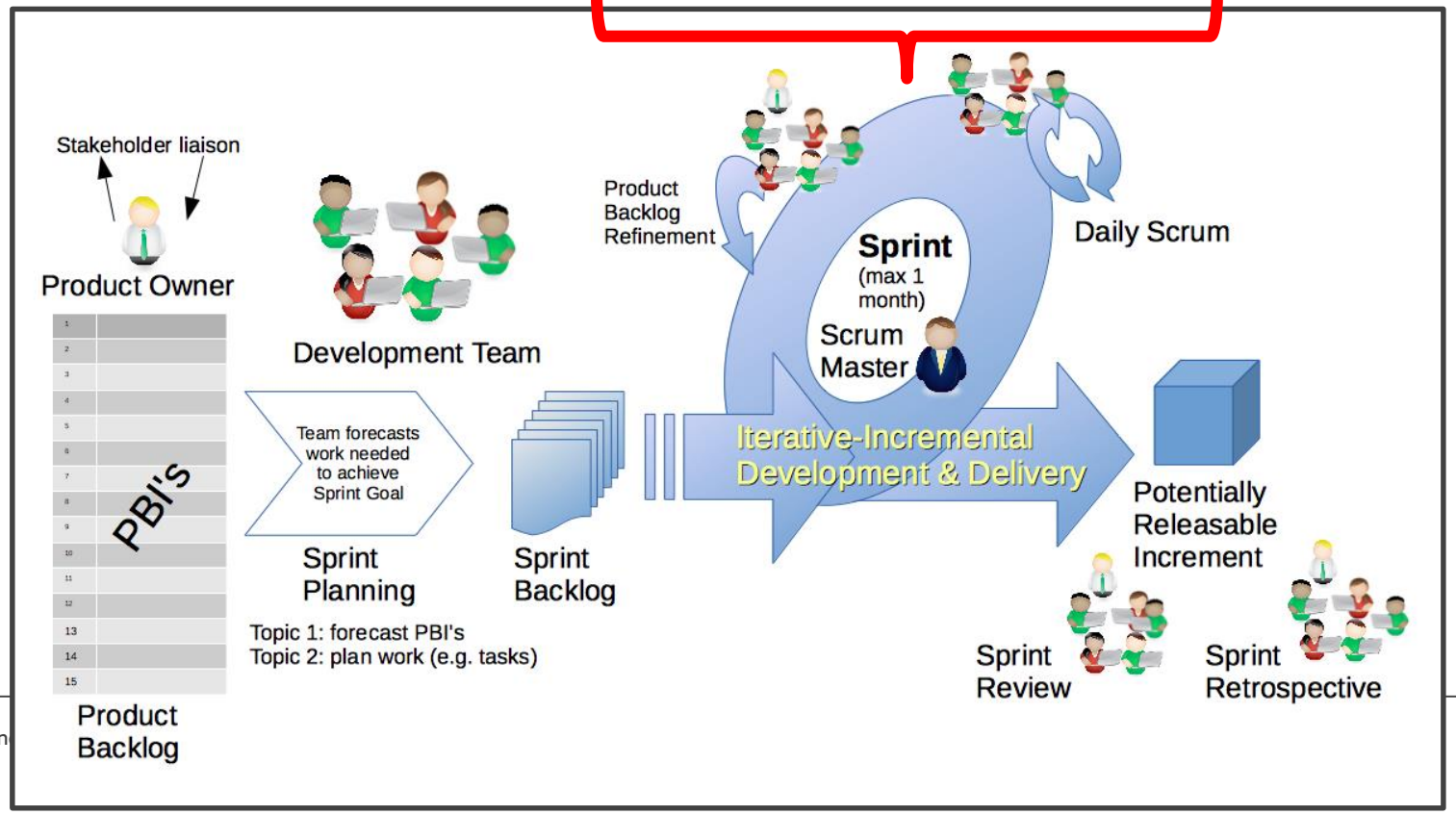
In each Sprint



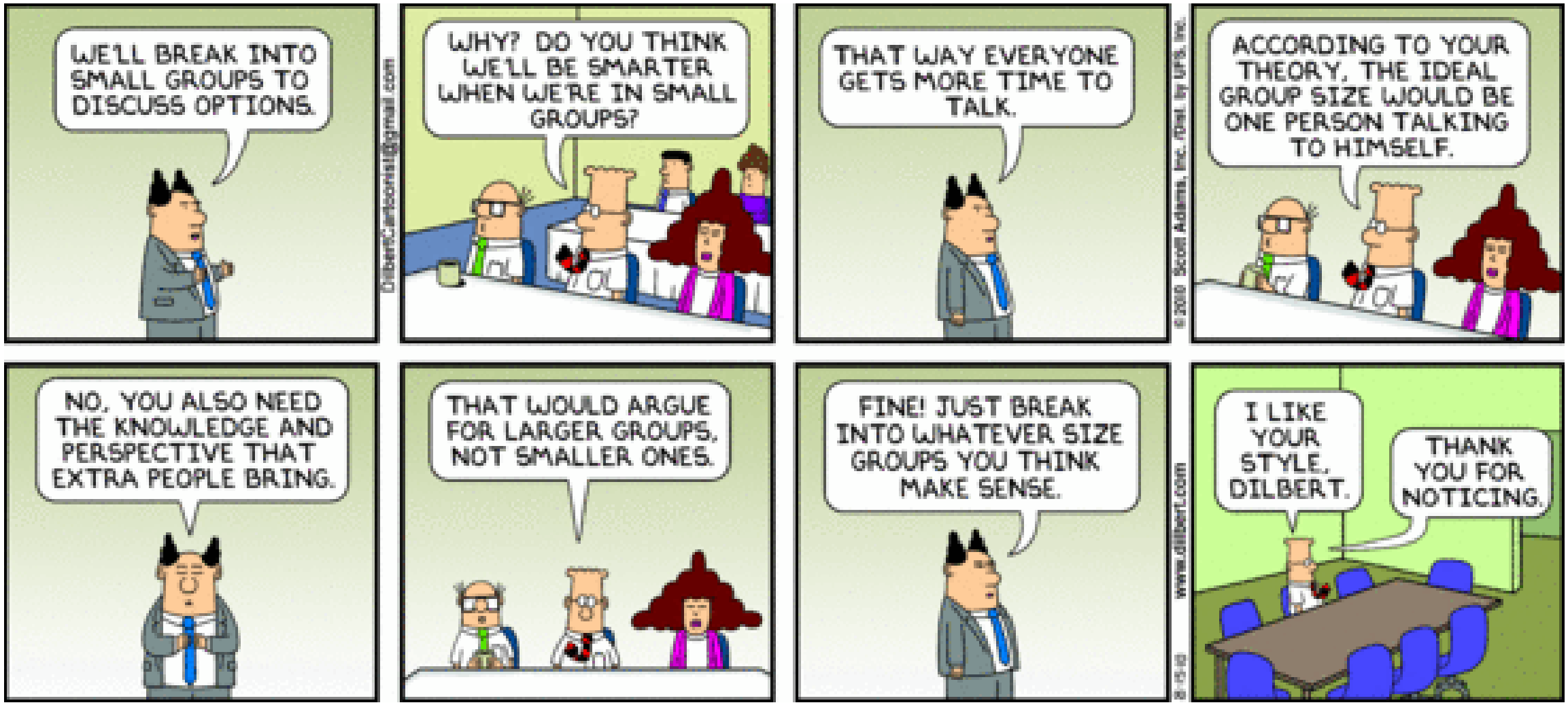
Stop/Go

Before SCRUM





Project Description is written here!!


WE'LL BREAK INTO SMALL GROUPS TO DISCUSS OPTIONS.
WHY? DO YOU THINK WE'LL BE SMARTER WHEN WE'RE IN SMALL GROUPS?
THAT WAY EVERYONE GETS MORE TIME TO TALK.
ACCORDING TO YOUR THEORY, THE IDEAL GROUP SIZE WOULD BE ONE PERSON TALKING TO HIMSELF.
NO, YOU ALSO NEED THE KNOWLEDGE AND PERSPECTIVE THAT EXTRA PEOPLE BRING.
THAT WOULD ARGUE FOR LARGER GROUPS, NOT SMALLER ONES.
FINE! JUST BREAK INTO WHATEVER SIZE GROUPS YOU THINK MAKE SENSE.
I LIKE YOUR STYLE, DILBERT.
THANK YOU FOR NOTICING.
DilbertCartoonist@gmail.com
© 2010 Scott Adams, Inc. /Dist. by UFS, Inc.
www.dilbert.com