



AGILE IT - Fundamentals

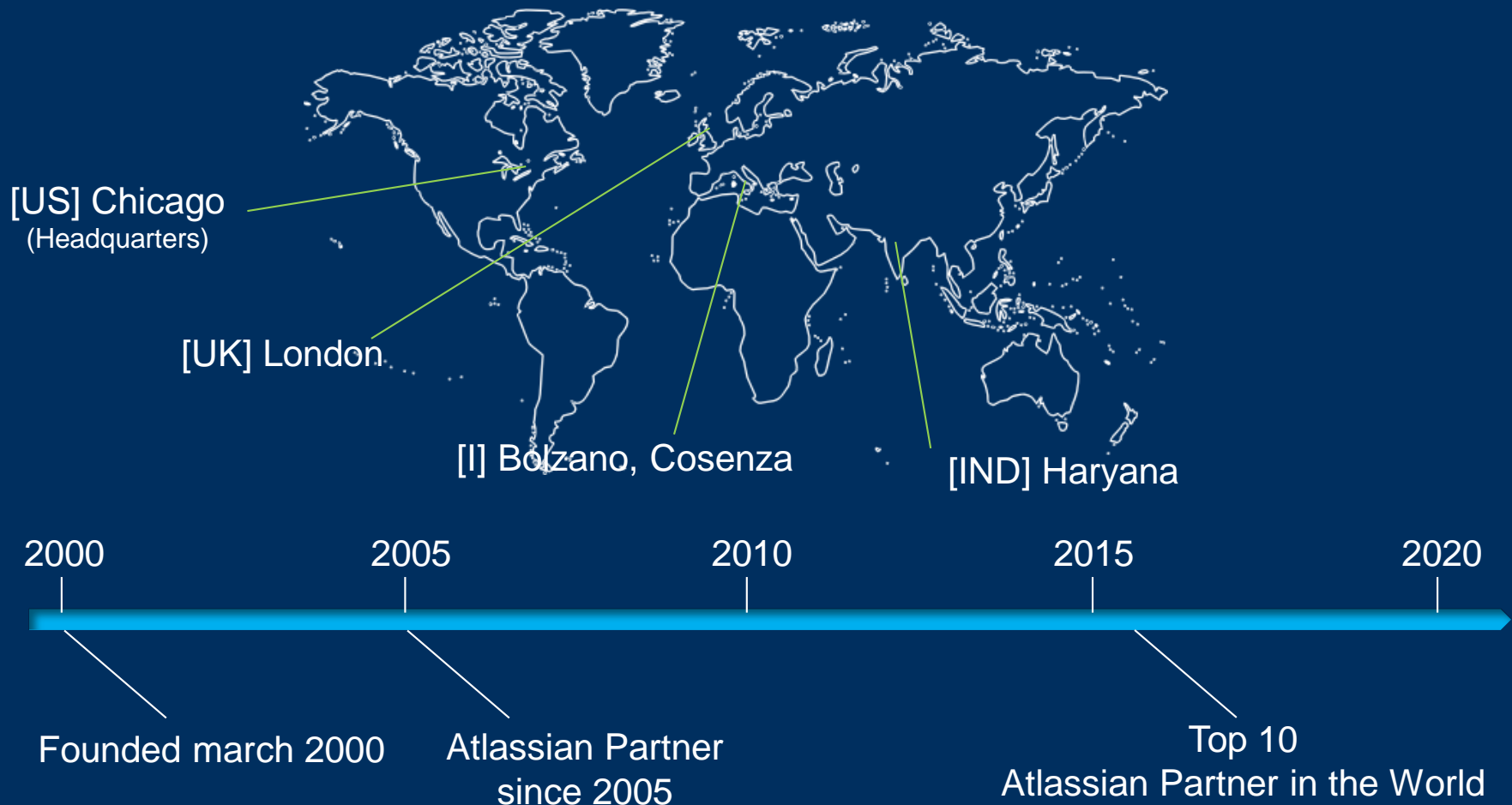


Agile IT Fundamentals

Agile Presentation
JIRA Software in action
Agile 3.0

Herzum Group: Agile and DevOps Leaders

International IT consulting group



Herzum Group: Agile and DevOps Leaders

International IT consulting group

United States | Italy | United Kingdom | India



2000

2005

2010

2015

2020

16+ years of agile transformations

- Small to very large adoptions
- Consulting practices for Agile adoptions, DevOps, and Atlassian
...and other practices
- Custom Development projects and helping setup software factories
- Our passion: Better IT thru Automation
Requirements, development, testing, release, and operations

Goals

An opportunity to increase our agile awareness



1

Deep insight into Agile Principles

2

How Agile Development works

3

What's next in Agile roadmap

Start of a Project

MANY QUESTIONS



WE NEED ANSWERS

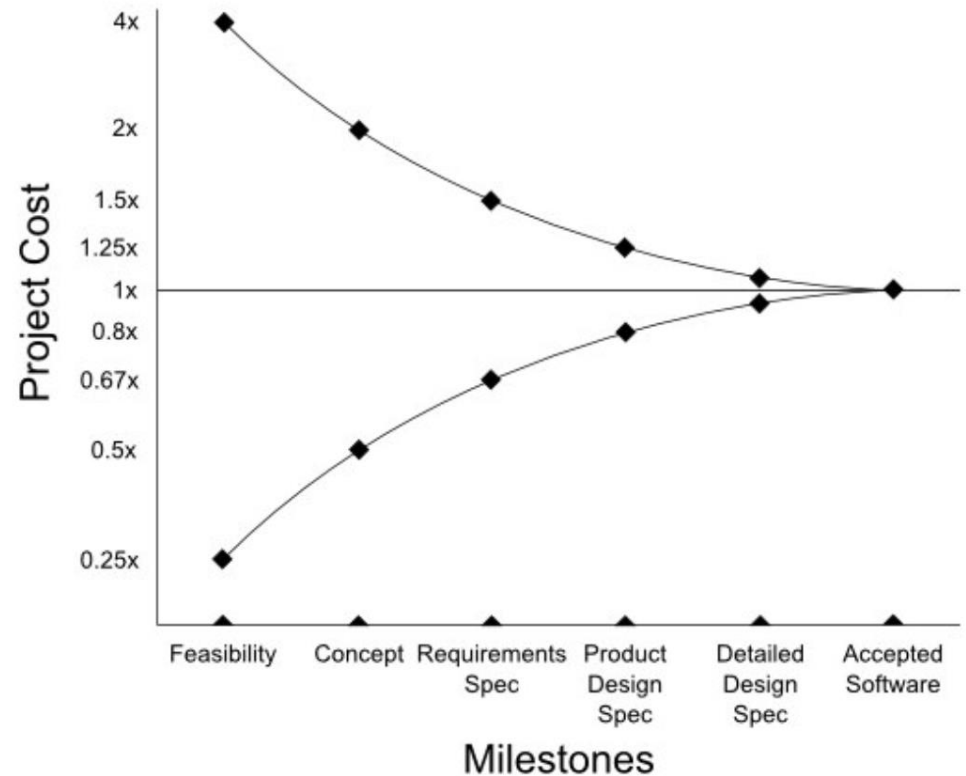
Pricing
Planning
Architecture

WHAT IS THE PROBLEM?

Uncertainty

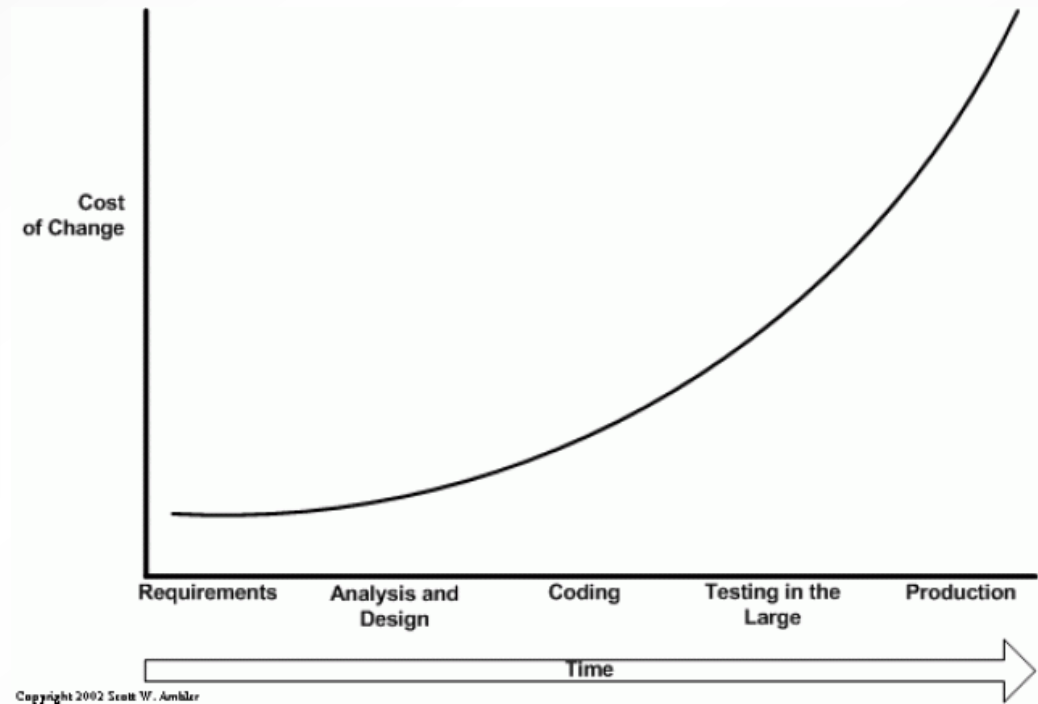
Estimates created early in the project are subject of high degree of error

Accuracy of Project Cost Estimates versus Milestones



Cost of Change

The larger the time buffer, the larger the budget buffer must be!



Waterfall like a Cannonball

Assumptions

the customer knows what he want
developers know how to build it
nothing will change along the way



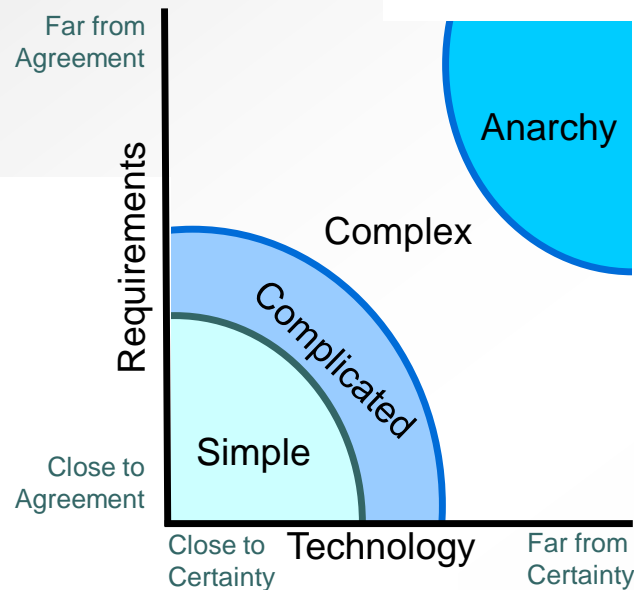
Agile like a Guided Missile

Facts

the customer discovers what he wants
developers discover how to build it
many things change along the way



Project Complexity Levels



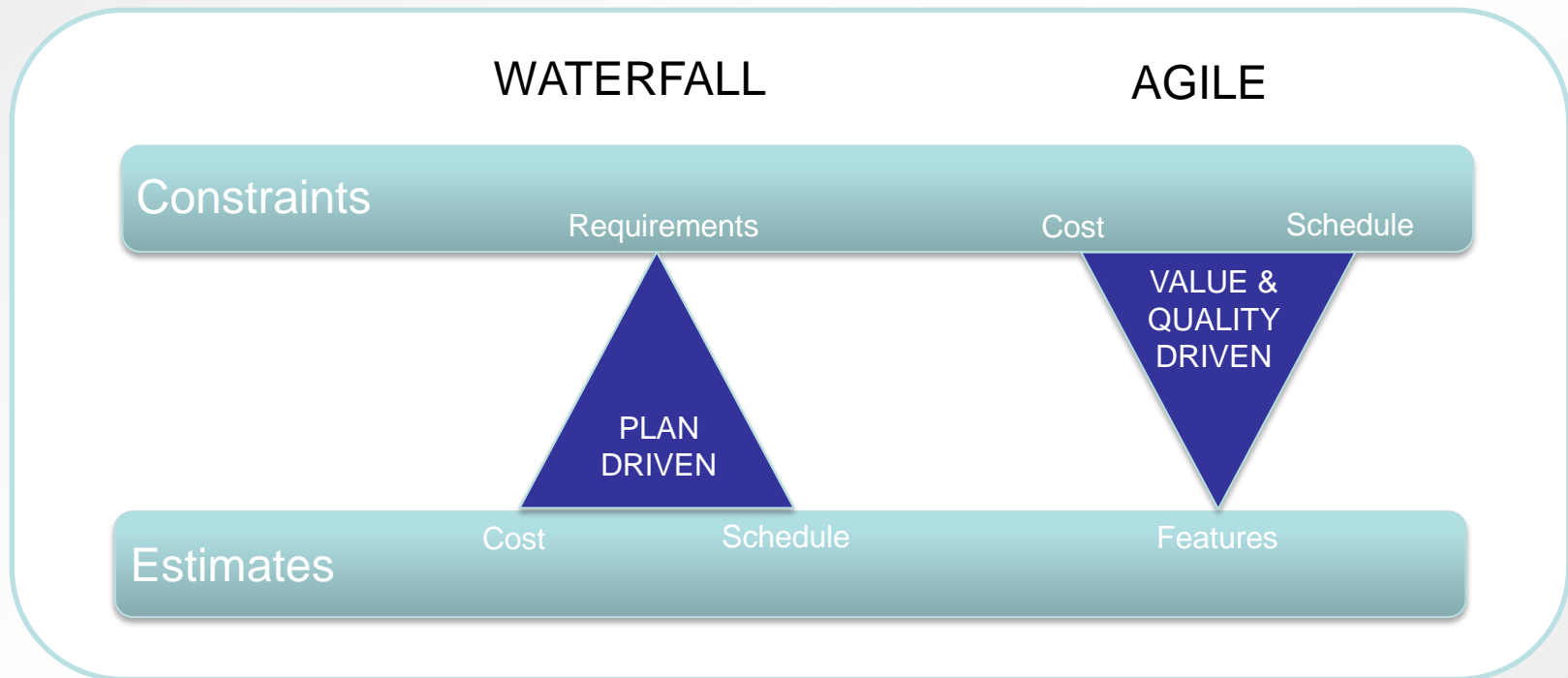
Fonte: *Strategic Management and Organizational Dynamics*, Ralph Stacey in *Agile Software Development with Scrum*, Ken Schwaber, Mike Beedle

TWO SHIPS PASS IN THE NIGHT

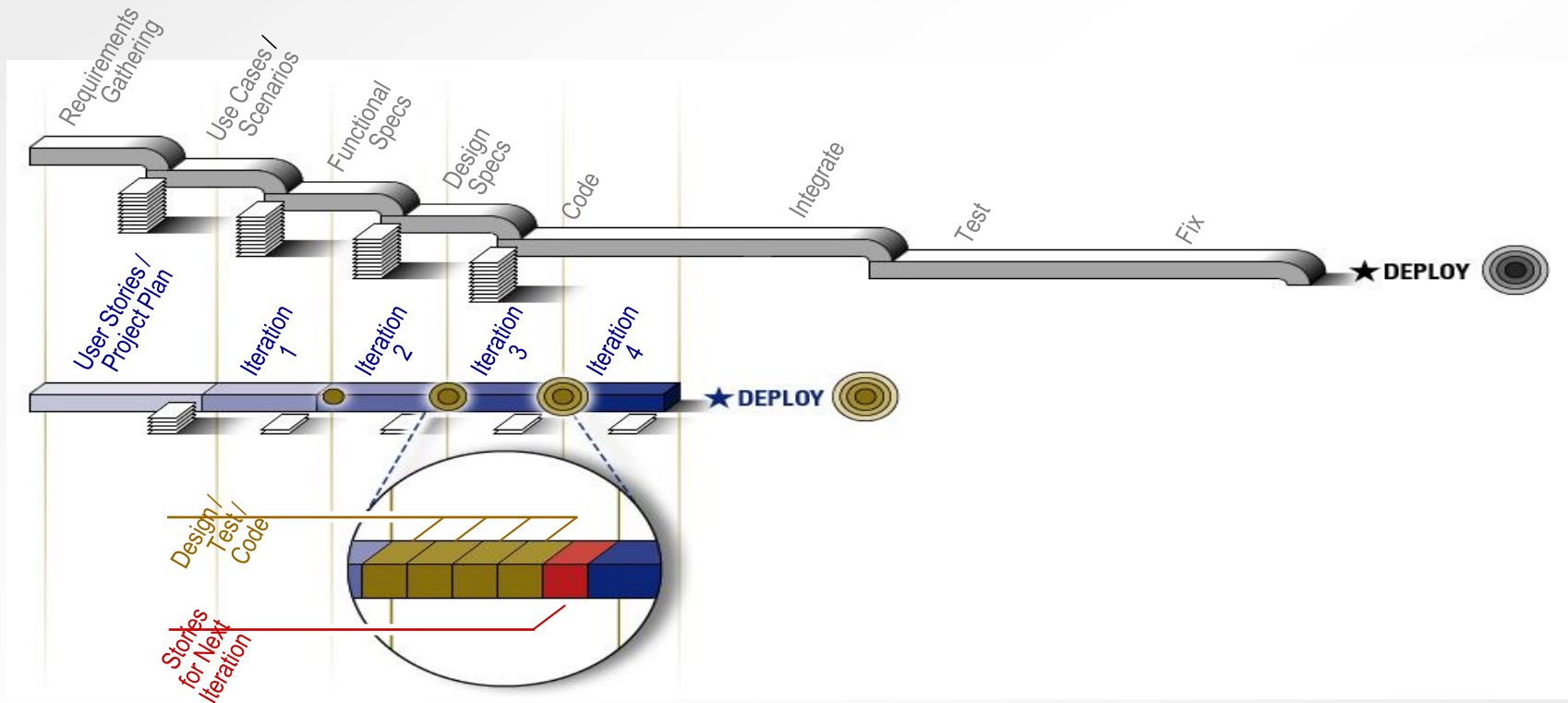
PMO

AGILE
TEAM

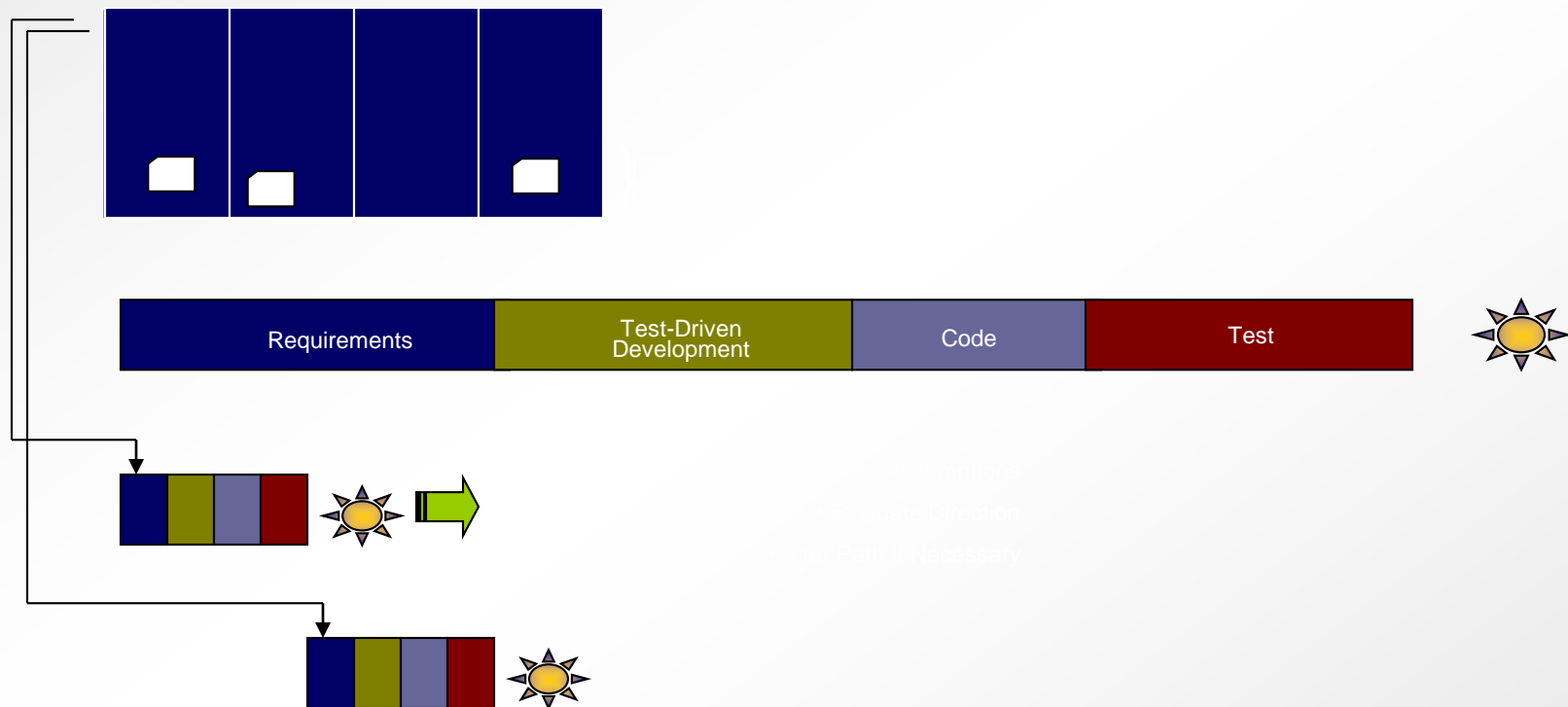
Value & Quality



AGILE VS. WATERFALL



How Sprints/Iterations Work



Manifesto for Agile Software Development

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

That is, while there is value in the items on the right, we value the items on the left more.

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

12 Principles behind the Agile Manifesto

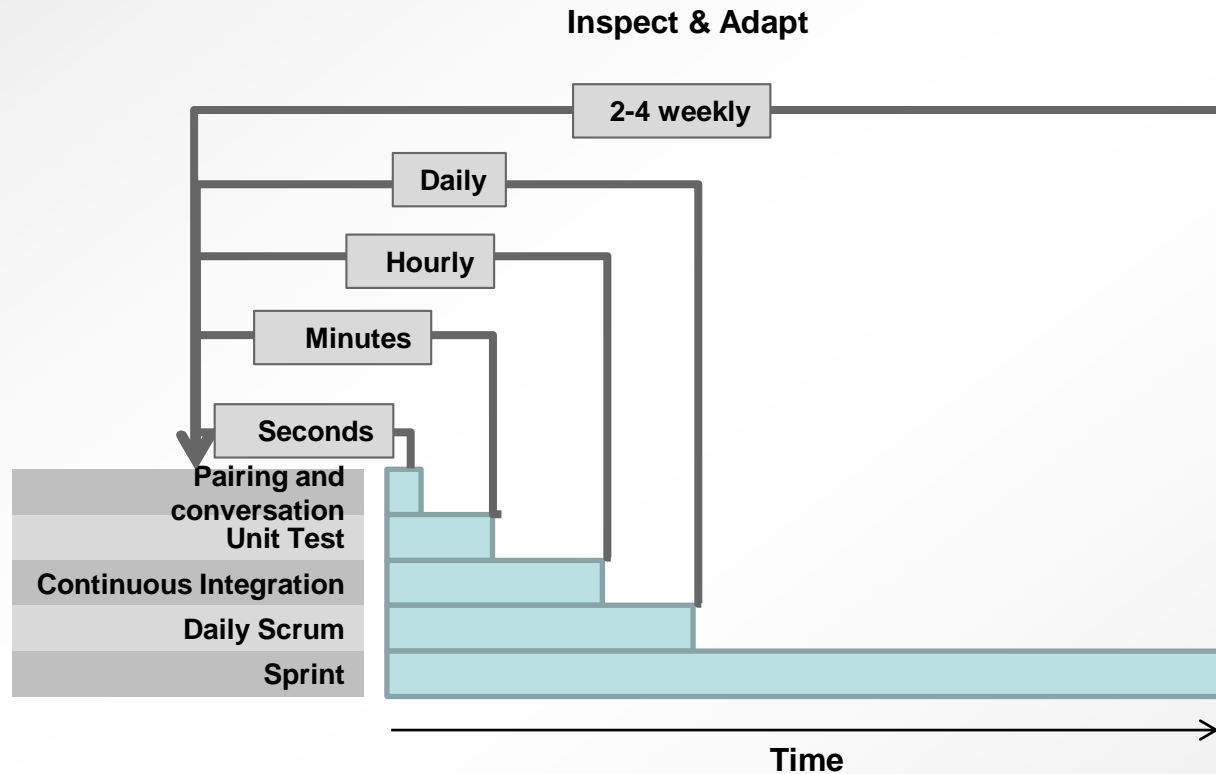
We follow these principles:

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

12 Principles behind the Agile Manifesto

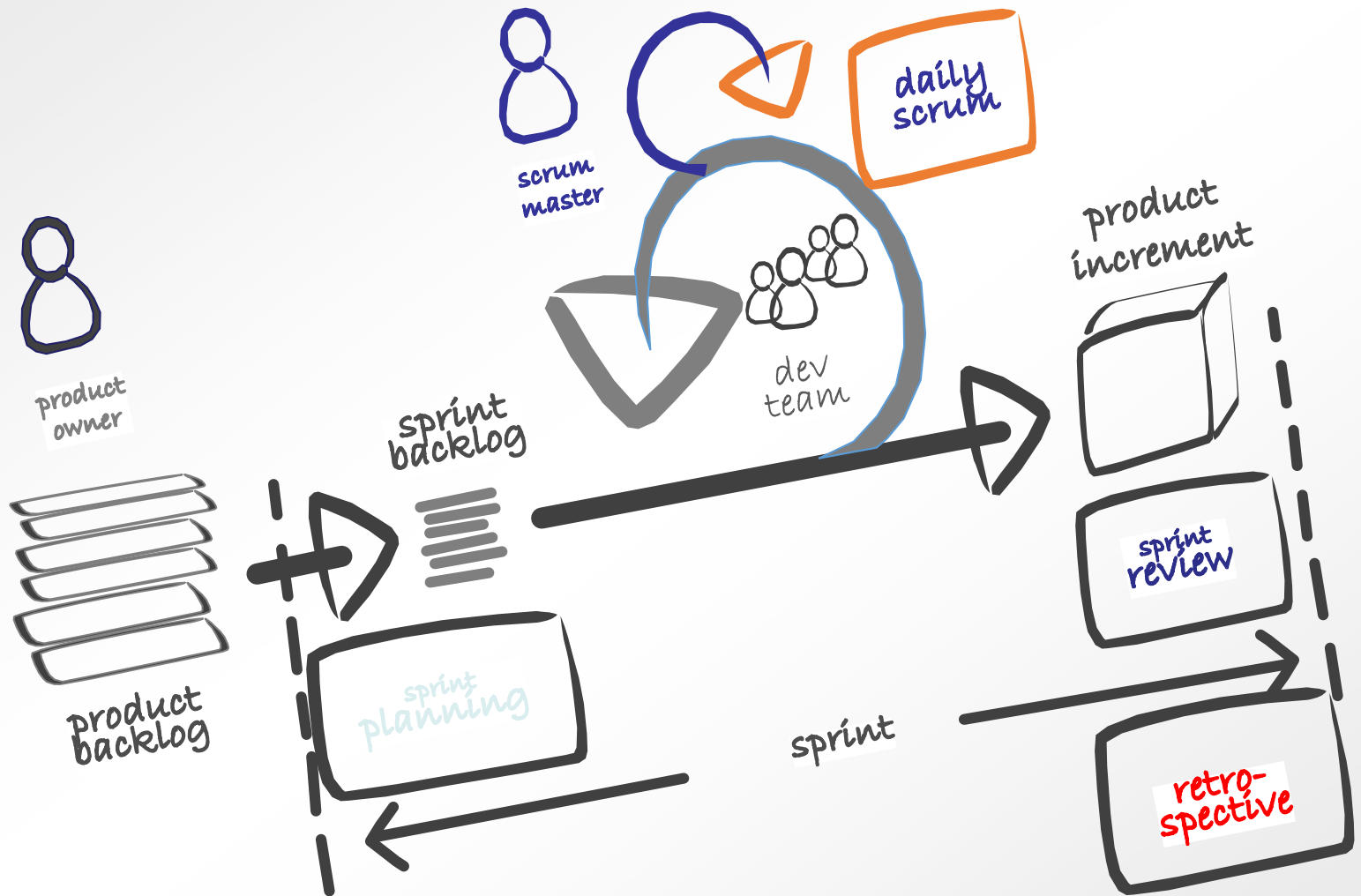
7. Working software is the primary measure of progress
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely
9. Continuous attention to technical excellence and good design enhances agility
10. Simplicity--the art 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

Feedback as soon as frequently possible



SCRUM





Origins

◆ Jeff Sutherland

- First uses of Scrum at Easel Corp in 1993
- IDX: 600 developers using Scrum
- From beginning, was applying to simple and complex projects

◆ Ken Schwaber

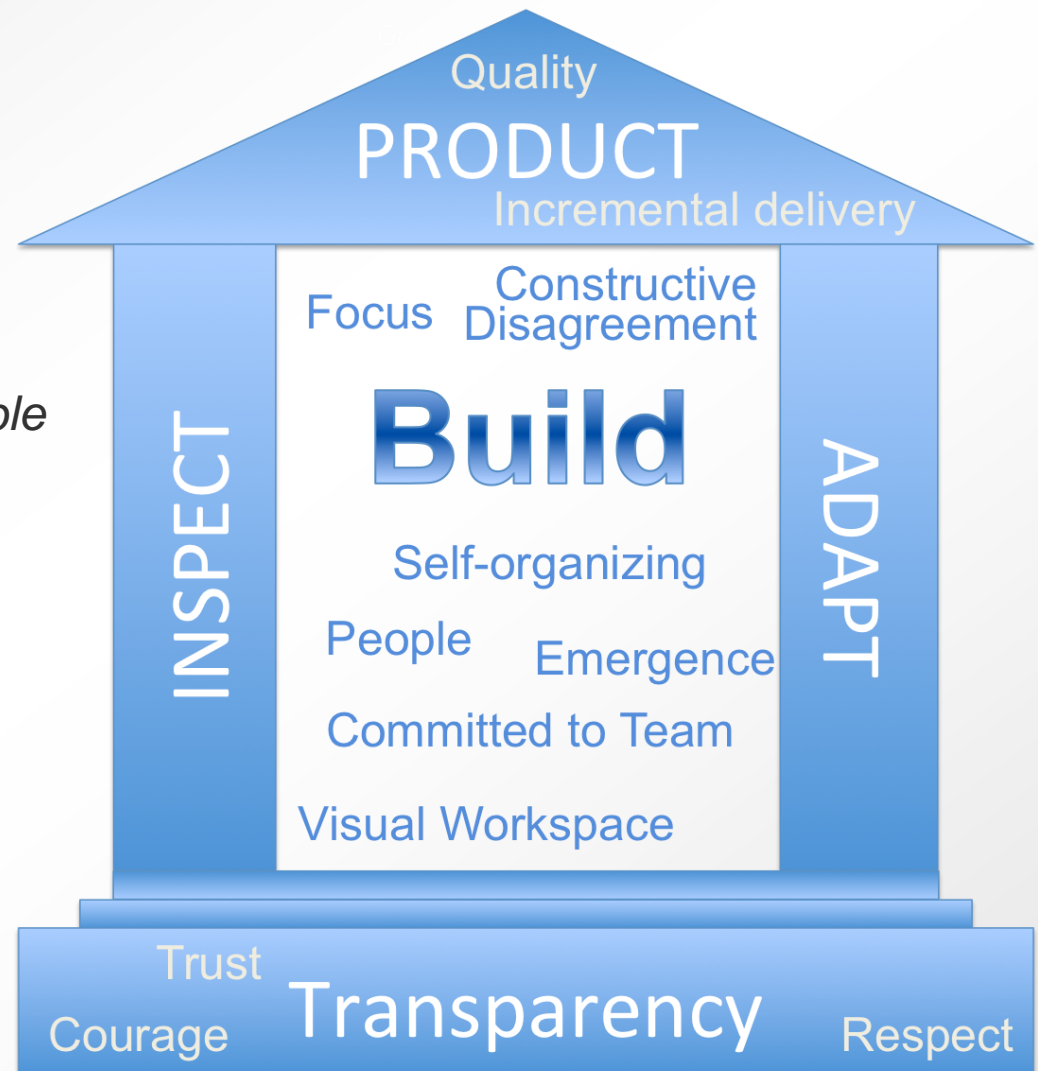
- Performed first known “conceptualization” of Scrum, for a presentation at OOPSLA 96 together with Sutherland

◆ Mike Beedle

- Presented Scrum patterns at PLOPD4

SCRUM

Scrum Is a Framework that helps teams to deliver customer value early and often in highly predictable manner

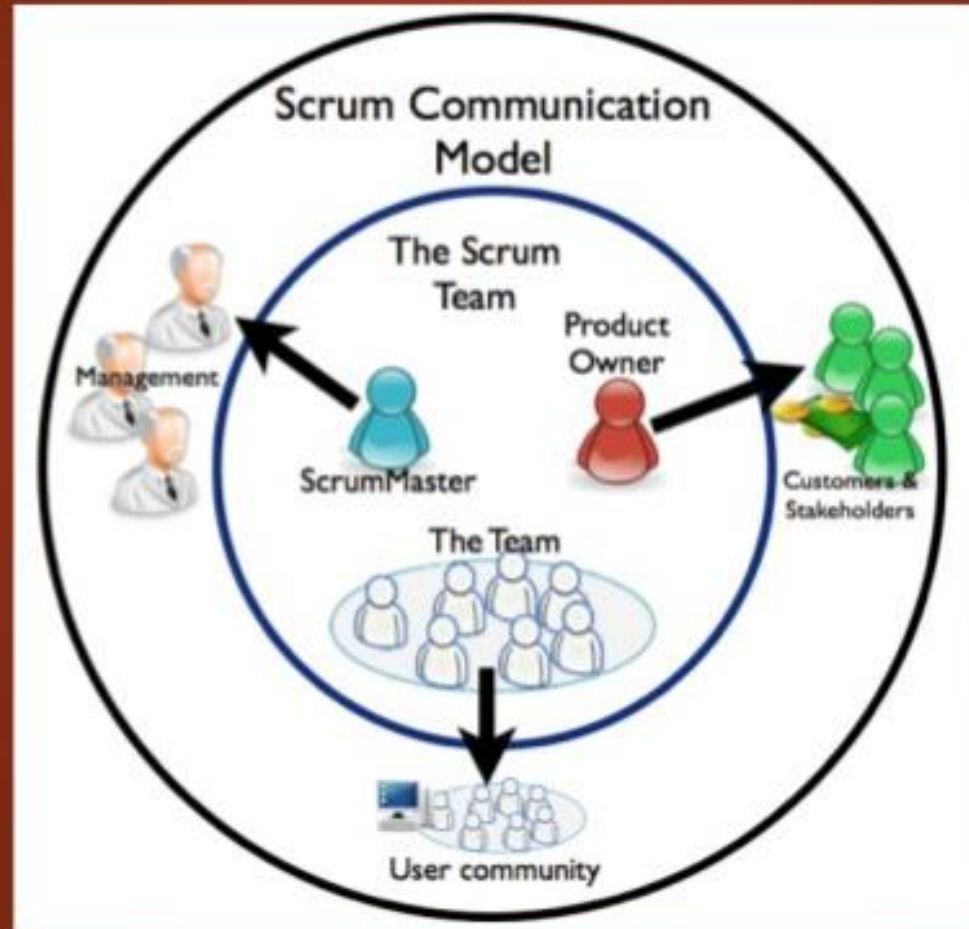


Original Characteristics

- ◆ Meant for “agile” projects
- ◆ Philosophical approach: self-organizing teams
- ◆ Projects organized through 1-month long “sprints”
- ◆ Requirements captured in a “product backlog”
- ◆ Non-prescriptive: no specific engineering techniques are “mandated”
 - You CAN use engineering techniques
- ◆ Uses principles to establish an “agile environment” for project development

Roles in “traditional Scrum”?

Scrum Communication



Day in Life of ScrumMaster

- ◆ Ensure everyone is doing what they have agreed to do
- ◆ Determine where Scrum is compared to where it could be and update your own Scrum impediment backlog
- ◆ A dead (fired) ScrumMaster is a useless ScrumMaster and,
- ◆ Use all of your senses, including common sense, and remember that you have no authority.



Product Owner



- Develops and maintains the Product Backlog
 - Orders the Product Backlog
- Empowered to make decisions for all customers and users
- Attends Sprint planning meeting and Sprint review meeting
 - Presents and explains Product Backlog to team
 - Manages the vision, ROI, and releases.

Team

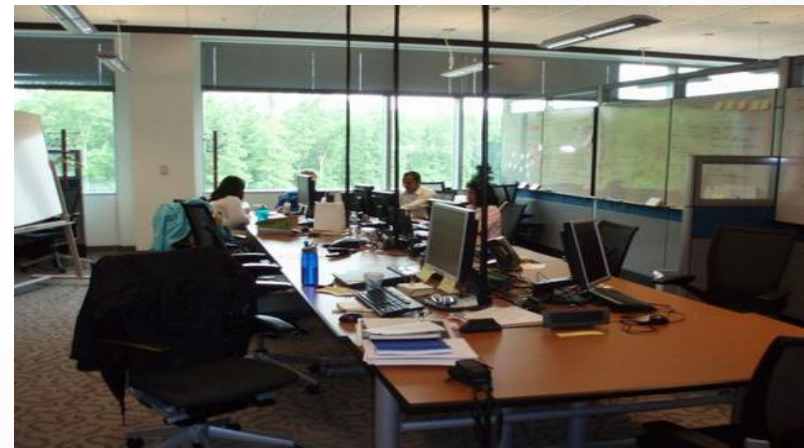


- Cross-functional with no roles
- Has the business and technical domain skills to build an increment of functionality
 - Commits to sprint goal / forecast work
 - Not for everyone
- Full autonomy and authority during a Sprint

Workspace

Everyone in same location
Open space without barriers



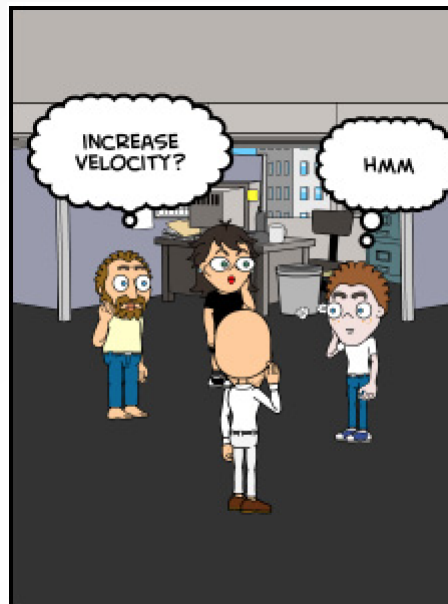


Sprint Planning

- ◆ 1 hour per part per week
- ◆ 1st – for team to select Product Backlog and sets goal with Product Owner
- ◆ 2nd - for team to define Sprint Backlog to build functionality
- ◆ Anyone can attend, but primary conversation and work is between team and Product Owner



ESTIMATE



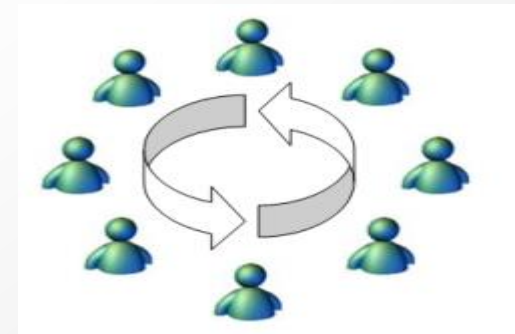
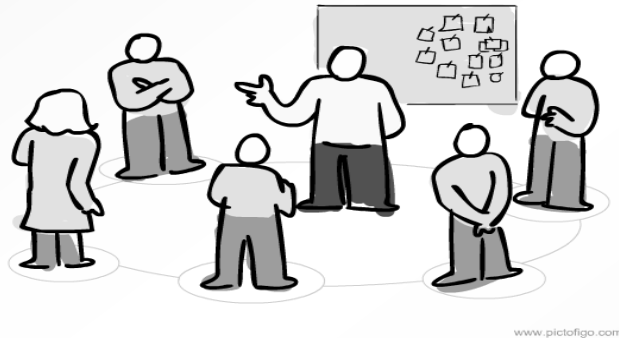
Daily Scrums

- ◆ Daily 15 minute meeting
- ◆ Same place and time every day
- ◆ Meeting room
- ◆ Three questions
 - What have you done since last meeting?
 - What will you do before next meeting?
 - What is in your way?

"If I had known how the questions from the Daily Scrum are used today I would have framed them differently, but it is too late to change it now"

Jeff Sutherland - April 2012

- Yesterday I helped the team by.....
- Today I will help the team by.....
- I am blocked from helping the team by.....





FAQ: Scrum meetings

◆ Why daily?

■ “How does a project get to be a year late?”

- “One day at a time.”

➤ Fred Brooks, The Mythical Man-Month

◆ Can a Scrum meeting be substituted by a written report?

■ No

- All team must see status each day
- The agreement taken in front of the rest of the team leads can motivate and put focus/pression on achievement of objectives

Sprint Review includes at least the following

- The Product Owner identifies what has been done and what hasn't been done.
- The Team discusses what went well during the Sprint and what problems it ran into, and how it solved these problems.
 - The Team then demonstrates the work that is done and answers questions.
- The Product Owner then discusses the Product Backlog as it stands. He or she projects likely completion dates with various velocity assumptions.
 - The entire group then collaborates about what it has seen and what this means regarding what to do next.

The Sprint Review provides valuable input to subsequent Sprint Planning meeting.

Sprint Retrospective

- ◆ Process improvement at end of every Sprint
- ◆ Facilitated by ScrumMaster (another Scrum Master)
- ◆ What went well, what could be improved.
- ◆ “Project Retrospectives,” *Norman Kerth*
- ◆ “Agile Retrospectives”, *Esther Derby – Diane Larsen*

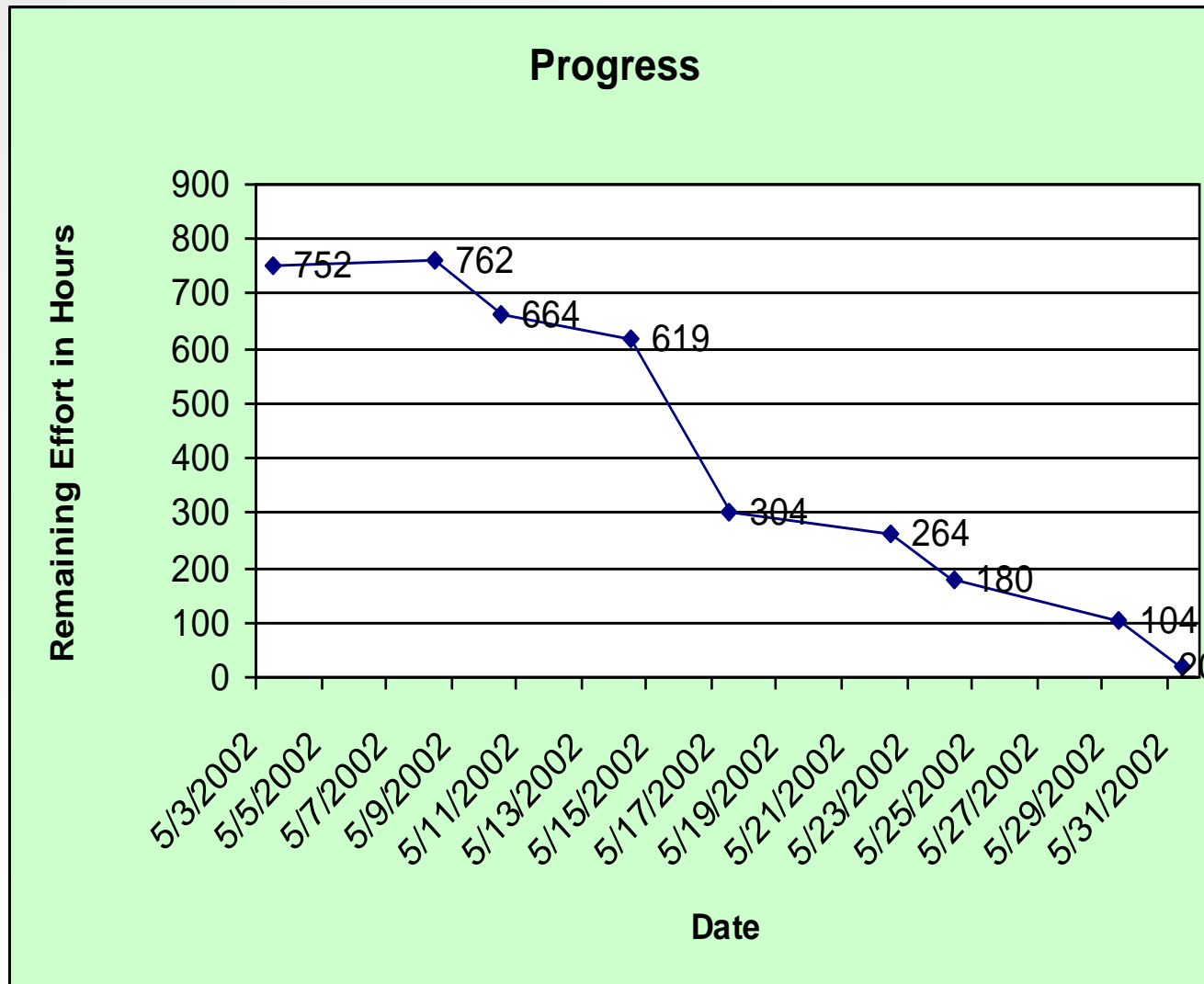
Use of Sprint Backlog during Sprint

◆ Changes to Sprint Backlog

- Team adds new activities deemed necessary to achieve the Sprint Goal
- Team can remove activities deemed unnecessary
- But: **Sprint Backlog can ONLY be updated by Sprint Team**

◆ Estimation can be updated when new information is obtained or discovered

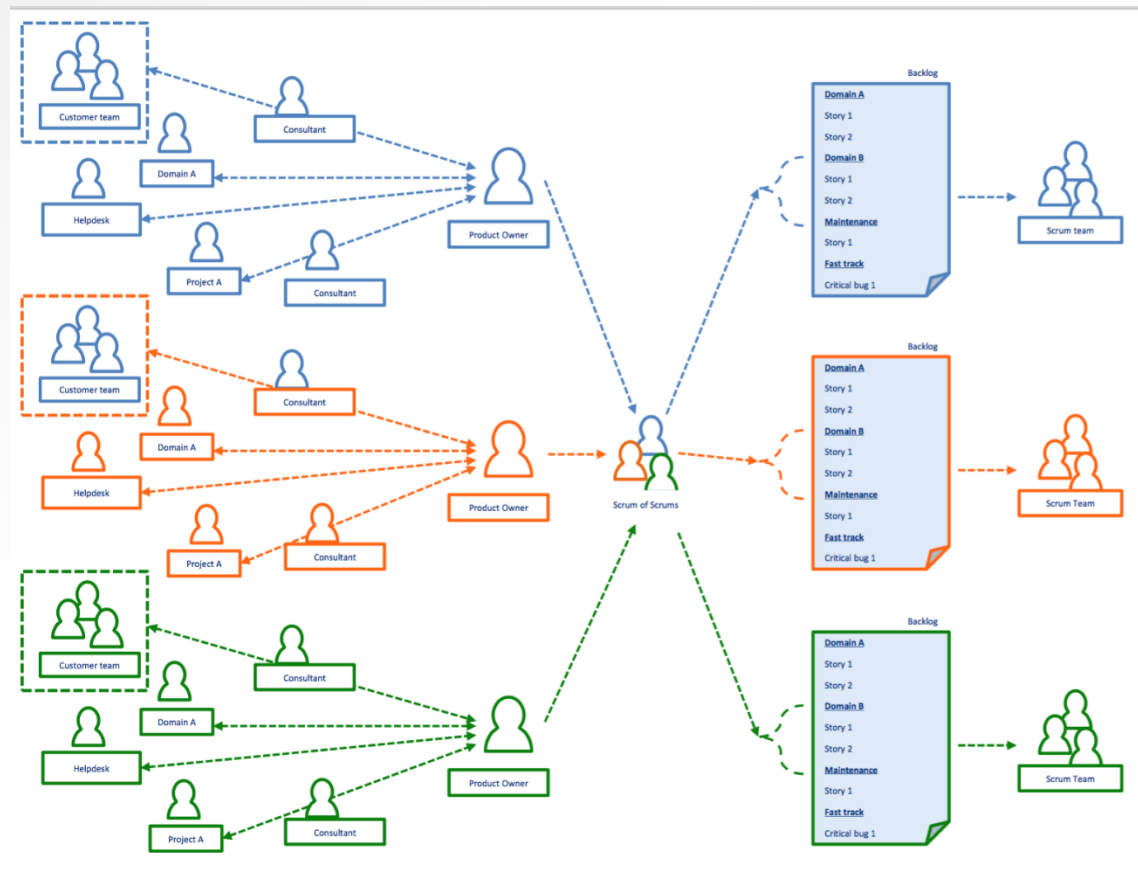
Sprint Burndown Chart



INVOLVED AND COMMITTED



SCRUM OF SCRUMS



A photograph of a Zen garden with a large grey rock in the center, surrounded by white sand with ripples, and green bushes in the background.

**simple is not
easy**

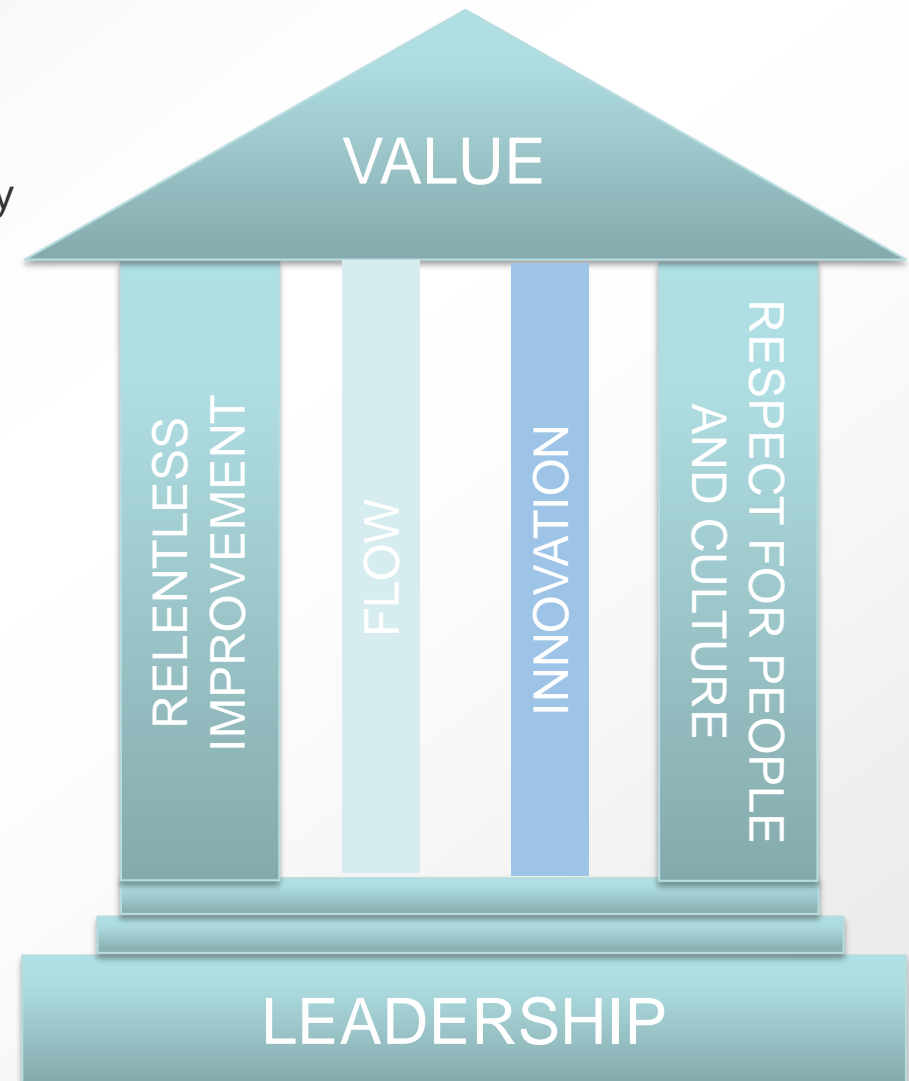
Lean Software Development

- ◆ Adaption of “lean manufacturing” to software development
- ◆ Lean principles
 - Eliminate waste
 - Amplify learning
 - Decide as late as possible
 - Deliver as fast as possible
 - Empower team
 - Build “integrity” in
 - See the whole

HOUSE OF LEAN

In Software Development, house of Lean provides the values organized around six key constructs:

- The goal: value with sustainably shortest lead time
- Respect for people and culture
- Flow
- Innovation
- Relentless improvement
- Leadership

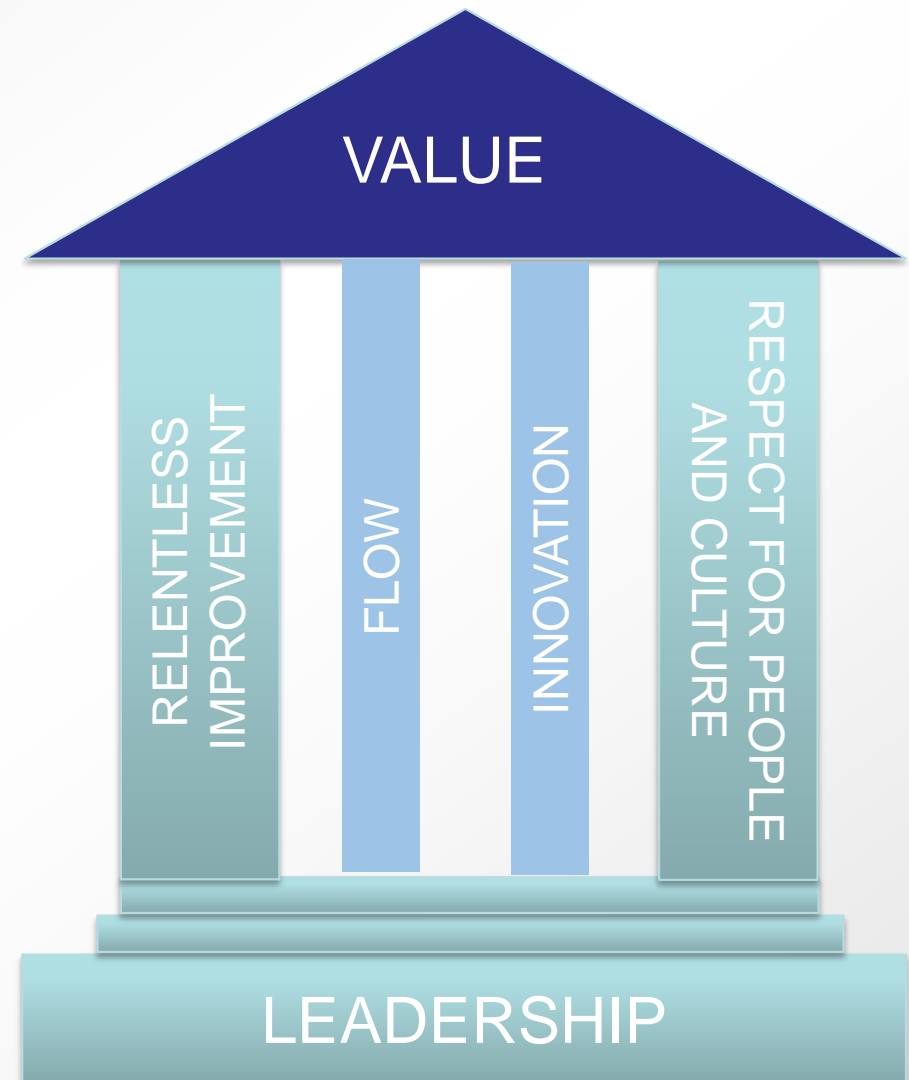


VALUE

All we are doing is looking at timeline, from when the customer gives us a order to when we collect the cash.

We are reducing the timeline by reducing the non value added wastes.

Taiichi Ohno



HOUSE OF LEAN

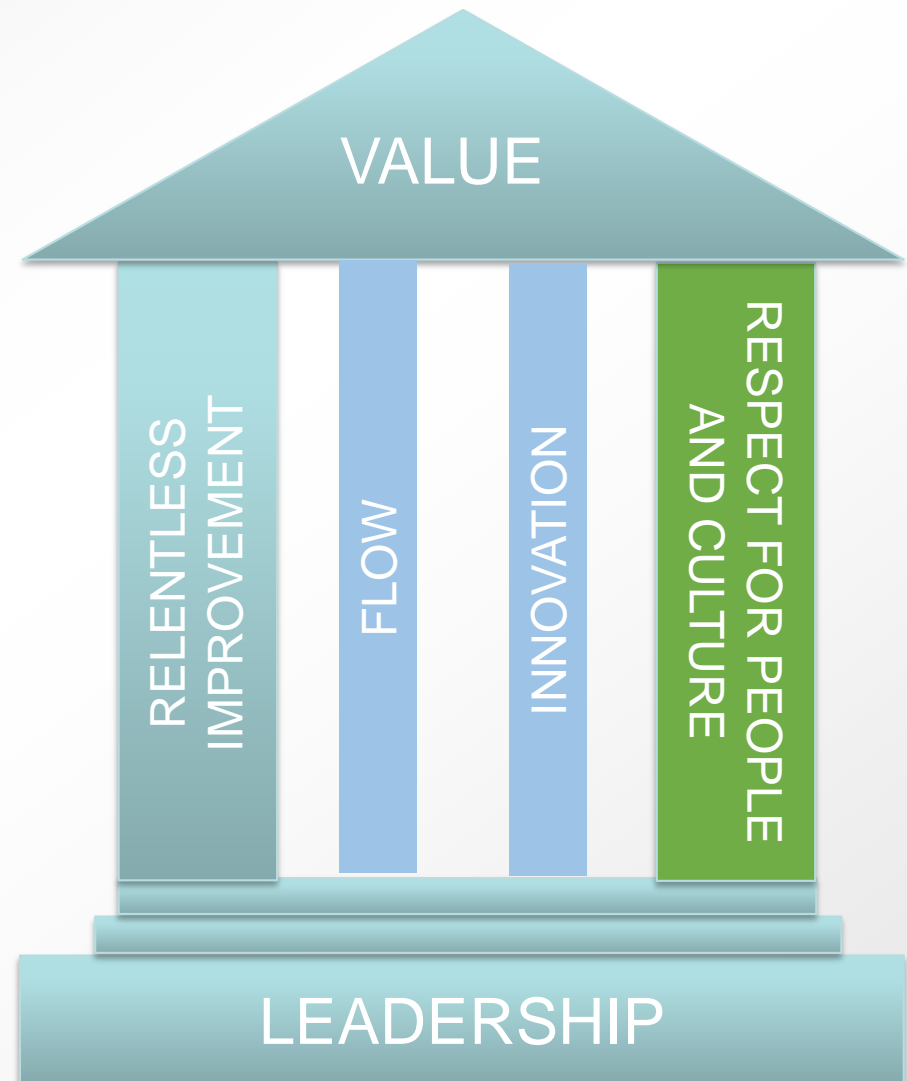
People do all the work

Customers is whoever consumes your work:

- don't overload and make them wait!
- don't force them to wasteful work
- don't impose wishful thinking

Build long term partnership based on trust

Culture drives behaviour



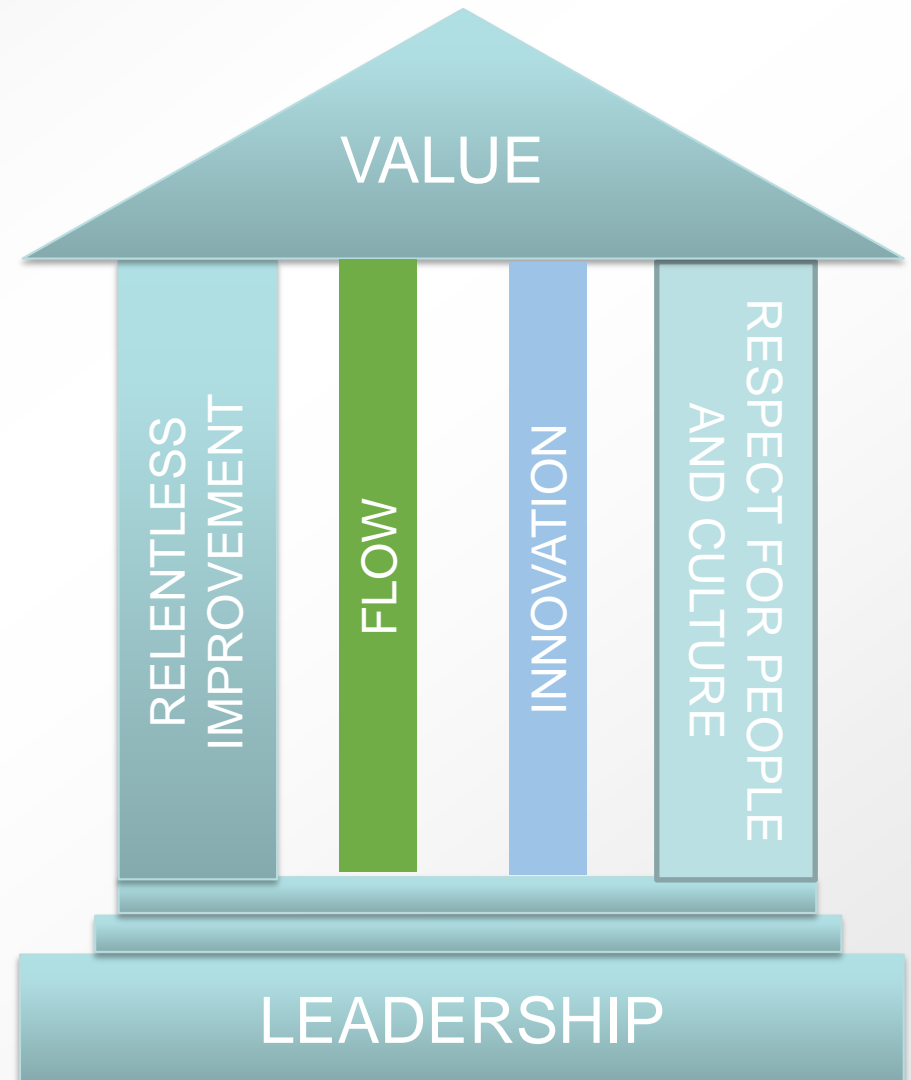
HOUSE OF LEAN

Implement a simple **Plan – Do – Check** cycle

Reduce delay and waste

Manage queues

Limit WIP



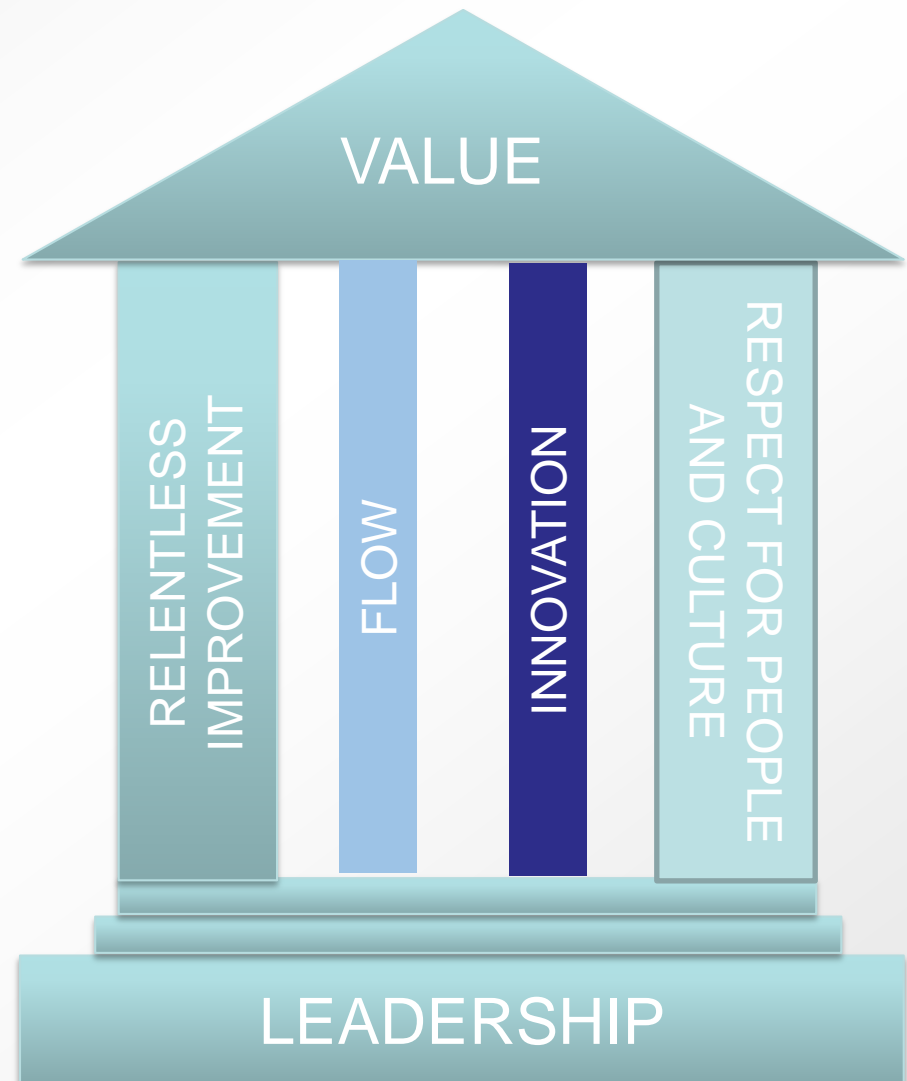
HOUSE OF LEAN

Producers innovate -> Customers validate

Go to Gemba!

Provide **time** and **space** for creativity

Pivot without mercy or guilt



HOUSE OF LEAN

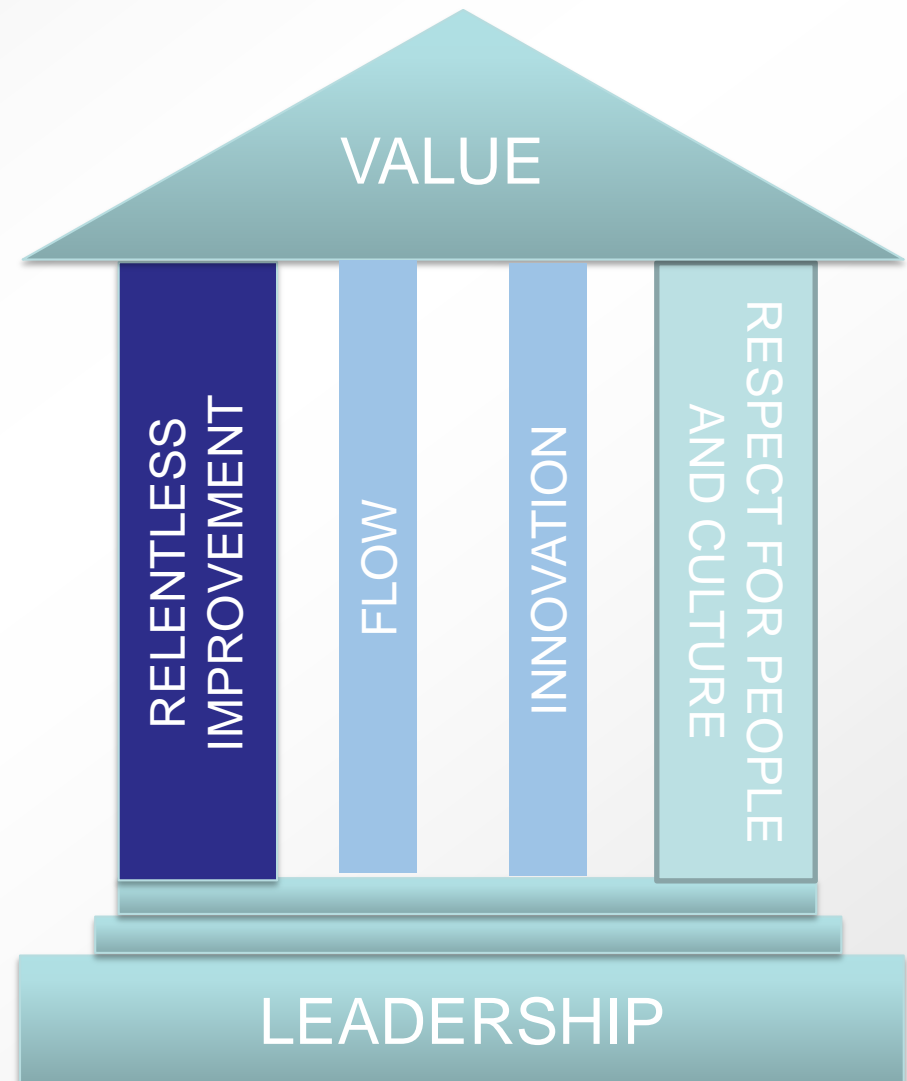
A constant sense of danger

Optimize the whole

Consider fact carefully then act quickly

Apply Lean tools and techniques to identify and address root causes

Reflect at key milestones to identify and address shortcomings



HOUSE OF LEAN

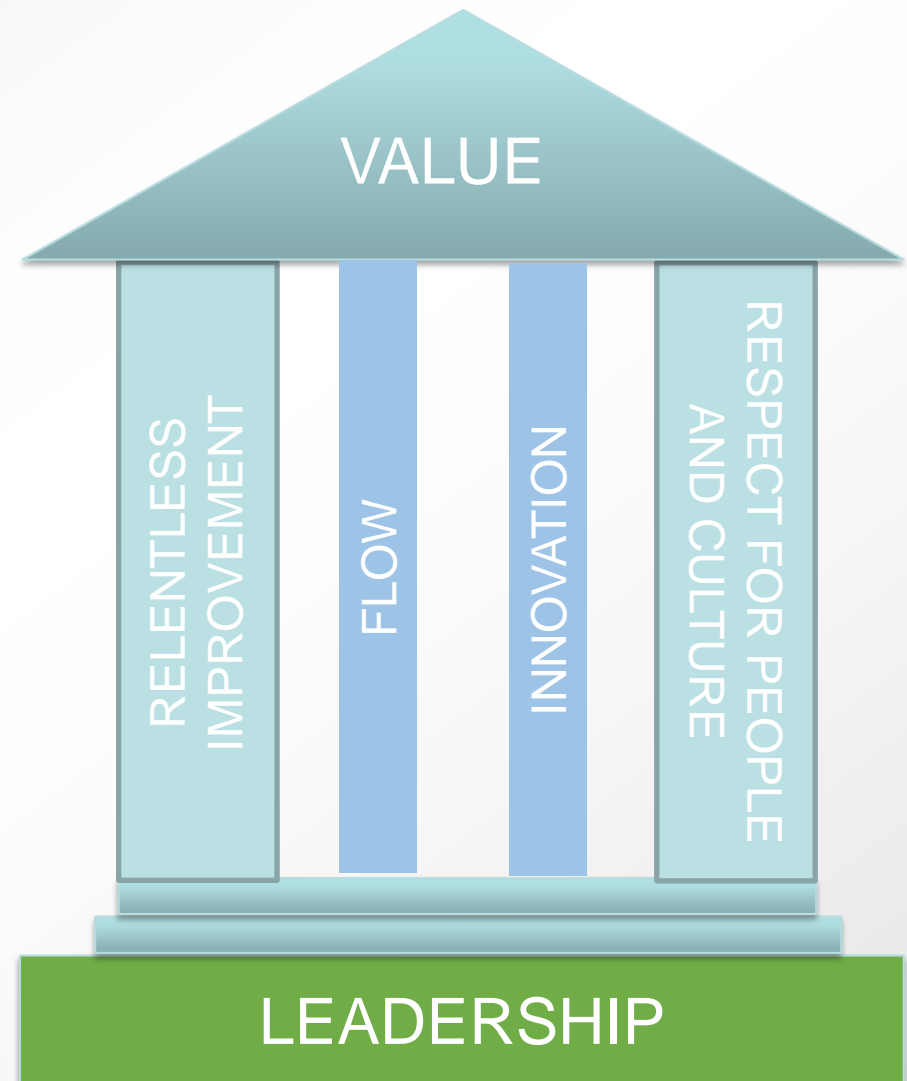
People are doing their best, the problems are with the system

Leader as developer of people

Inspire and align with mission and with minimum possible constraints

Emphasize life-long learning

Decentralize decision making



LEAN PRINCIPLE

There was a connection between **lean manufacturing** and **agile software** from the beginning in that many of the developers of the various agile methods were influenced by the ideas of lean manufacturing



LEAN & AGILE

AGILE FOR SOFTWARE DEVELOPMENT

Agile is about delivering the highest business value possibly faster by focusing on people and continuous improvement



LEAN

Lean Software Development is a translation of Lean Manufacturing and Lean IT principles and practices to the software development domain

LEAN & AGILE

Lean is a method for the elimination of waste (**Muda**) within a **manufacturing** system. **Lean** also takes into account waste created through overburden (**Muri**) and waste created through unevenness in work loads (**Mura**).

Agile is about delivering the highest business value possibly faster by focusing on people and continuous improvement

... but Agile has much less to say about how to connect the work of many different teams, and that's where Lean has a huge impact!

ANY
QUESTIONS
?



Thank you!



FABRIZIO TANI

DIRECTOR AGILE & ATlassian PRACTICE EMEA @ HERZUM