AGILE Methodology: SCRUM

Dinesh Anantwar

Agenda

1 Agile – Myth Bursting, Understanding Philosophy, Mindset change

2 Scrum : Roles

3 Scrum : Practices

4 Scrum : Value Proposition, Benefits & Constraints

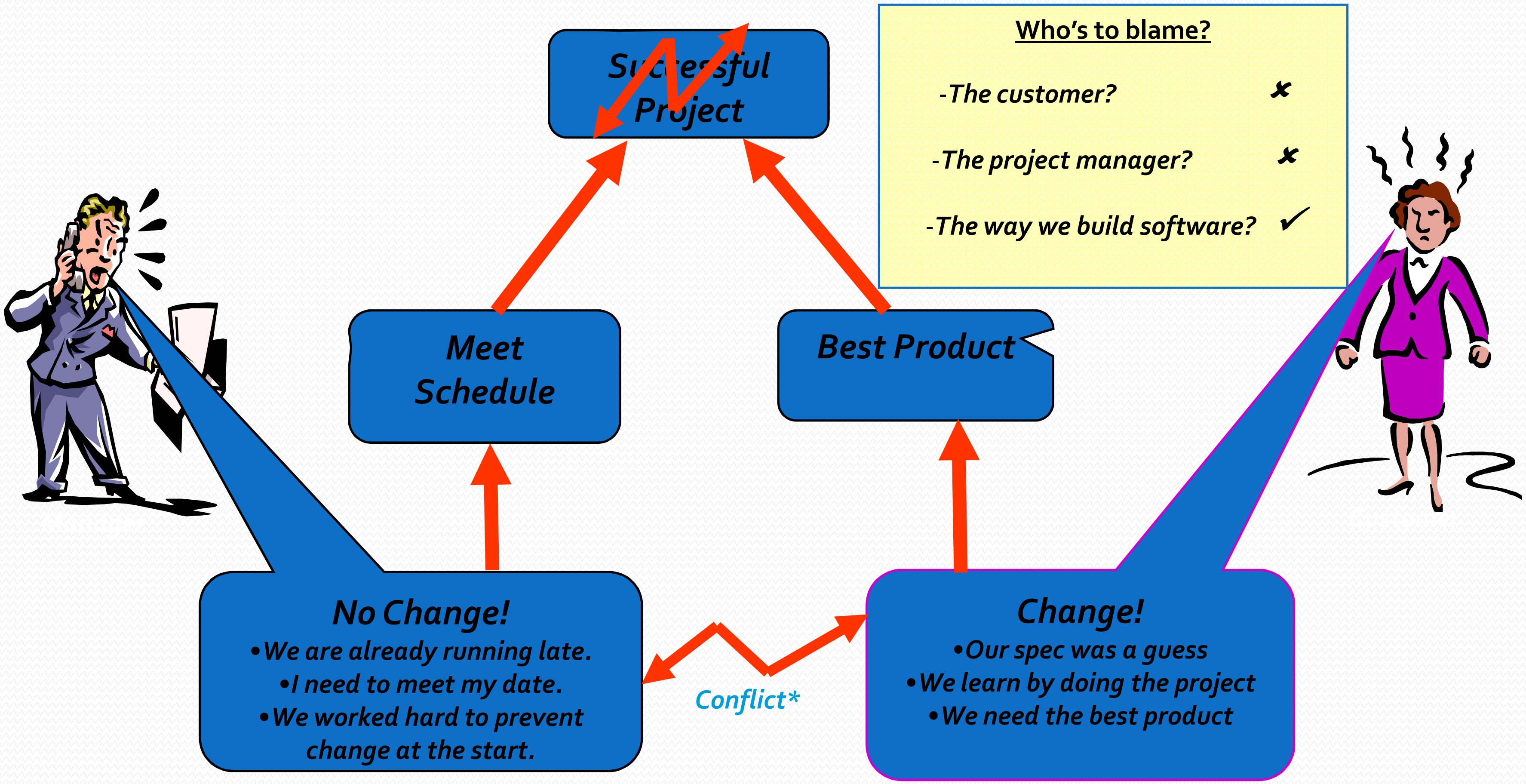
5 OGILE - Distributed Agile

6 Large Projects: SCRUM of SCRUMs

Agile Mythbusters

Myth	Reality
No Documentation	Agile Documentation
Undisciplined	Requires great discipline
No Planning	Just-in-time (JIT) planning
Not Predictable	Far more predictable
Does not Scale	It does Scale
Is a Fad	It‘s quickly becoming the norm
Silver Bullet	It requires skilled people
RUP isn‘t agile	RUP is as agile as you make it
Not Fixed Price	Agile provides stakeholders control over the budget, schedule, and scope

Successful Project: A Distant Dream



Successful Project

Meet Schedule

Best Product

Who's to blame?

-The customer? ✗

-The project manager? ✗

-The way we build software? ✓

No Change!
- We are already running late.
- I need to meet my date.
- We worked hard to prevent change at the start.

Conflict*

Change!
- Our spec was a guess
- We learn by doing the project
- We need the best product

What Do We Mean By “Agile?”

- According to the Merriam-Webster on-line dictionary “agile” means:
  - “1: marked by ready ability to move with quick easy grace;”
  - “2: having a quick resourceful and adaptable character.”
- In agile software development, “agile” tends to mean “the ability to respond to change.”

Change In Projects

- Changes From Requirements
  - Customers Learn from the Solution
  - Business Environment and Conditions Change
  - Business Processes are Re-engineered
- Changes From Technology
  - Tools/Platform Release New Versions
  - Actual Tool/Platform Capabilities May Vary from Plans
- Changes From People
  - Interactions are Complex
  - Individual Behavior is Unpredictable ##

What’s Really Different About “Agile?”

- “Ad-Hoc” Processes
  - Still the Most Common Process
  - Not Necessarily Chaotic, Just Not Consistent
- “Defined” Processes
  - Creation of Comprehensive Activity-Based Plans
  - Execution of Defined Activities
  - Management by Controlling Activities to Conform to Plan
- “Adaptive” Processes
  - Setting of Goal-Based Objectives for Deliverables
  - Opportunistic, Local Planning and Execution of Activities
  - Management by Retrospection, Learning, Adaptation

Characteristics of Agility

- Empowered, self-organizing teams
- Multi-discipline, cross-functional teams (whole team culture)
- Project- and product-centric focus, minimal organizational focus
- Shared responsibility, role-based accountability
- Shared vision of standards of excellence
- Close, continuous collaboration, direct communication ##

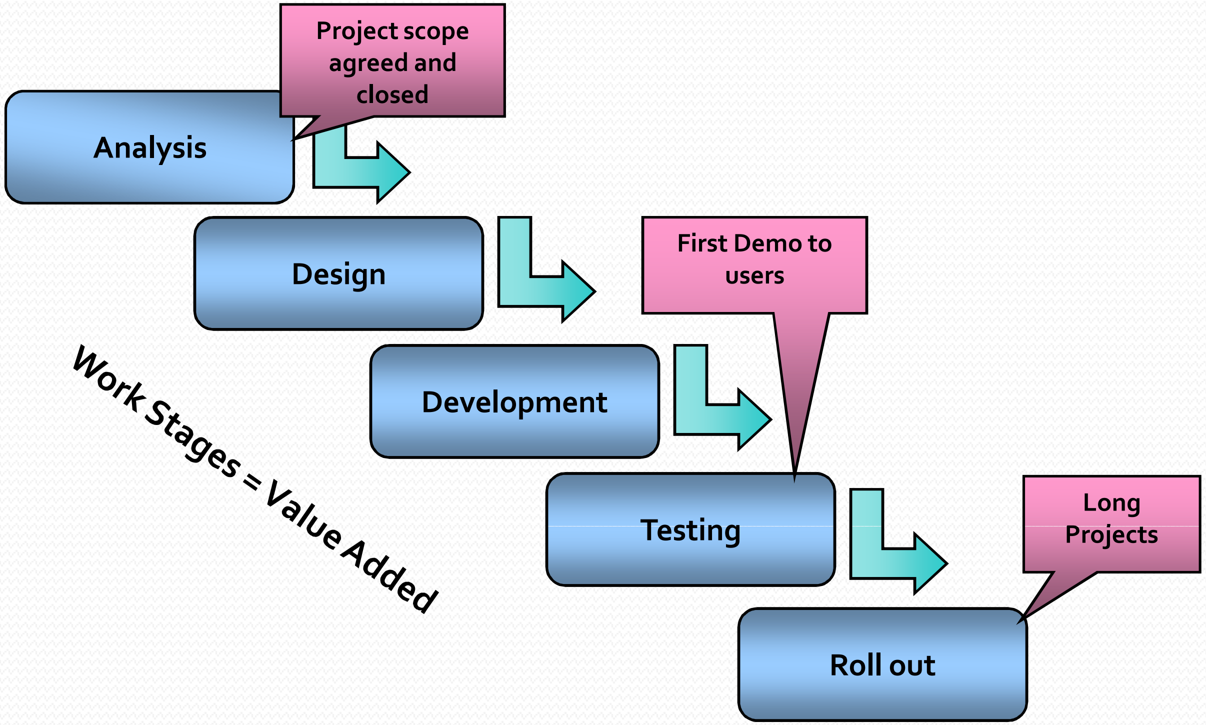
Characteristics of Agility, contd…

- Early, frequent, and continuous demonstration of progress through concrete deliverables
- Rapid feedback, reflection, learning, adjustment
- Small work batch sizes, minimal specialization, reduced queuing delays
- Just in time production, minimize production of artifacts not immediately (or ever) consumed
- Low friction – Simplicity, Minimalism, Practicality
- Sustainable, Constant, Predictable pace ##

Goals & Potential Benefits of Agility

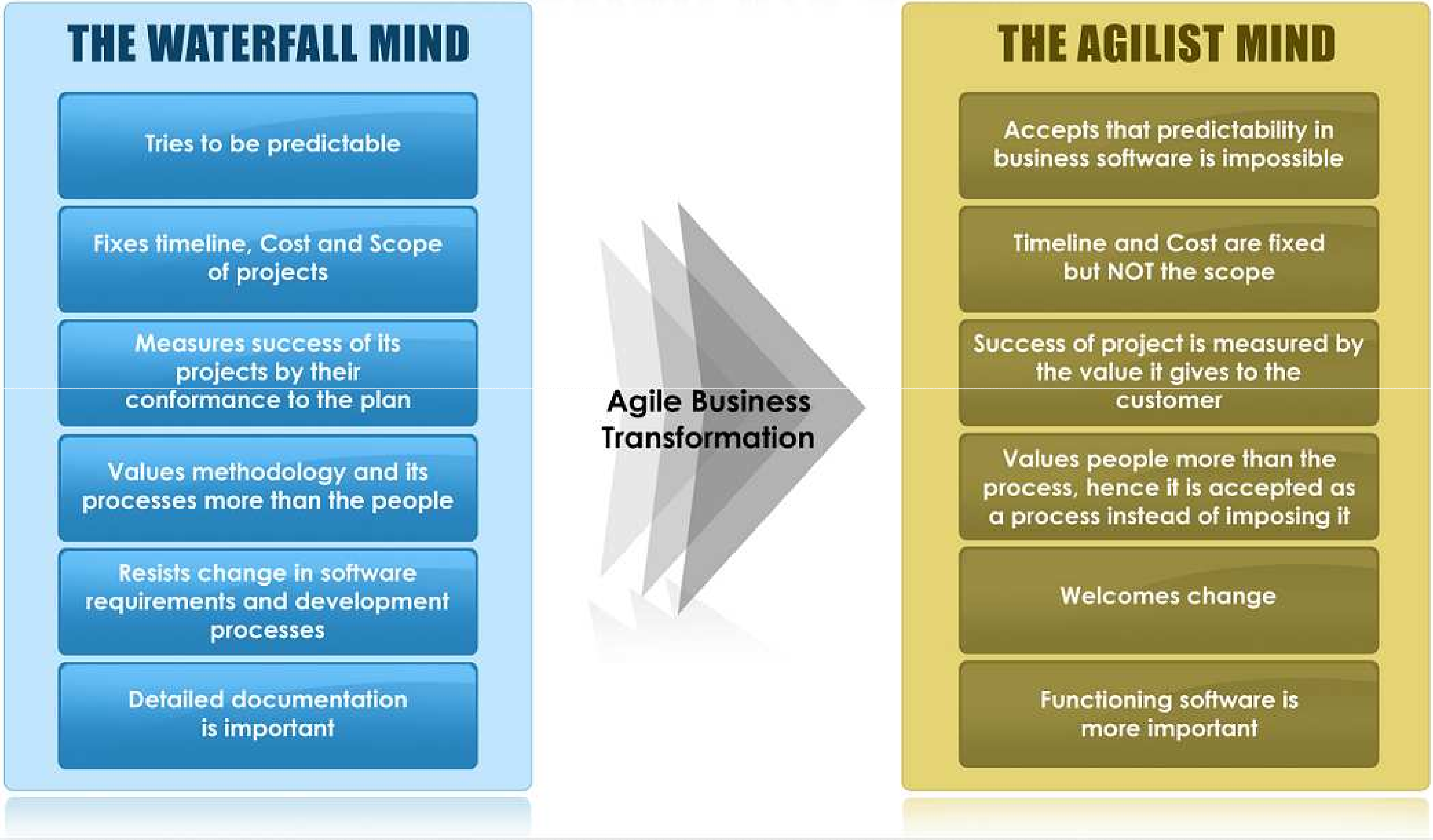
- Delivering the most value to the business, efficient use of resources, maximize ROI and time-to-ROI
- Faster development, higher productivity
- Flexibility to respond to change and leverage learning
- Better quality solutions, more enduring systems
- More fulfilling development culture

Waterfall Methodology & Constraints



- ✓ Scope is fixed at very early phase. Changes cost rework
- ✓ Poor visibility for Business Users about progress & the actual Product
- ✓ Longer duration projects. Late returns after initial investment

Agile Business Transformation; Change in the Mindset



THE WATERFALL MIND

- Tries to be predictable
- Fixes timeline, Cost and Scope of projects
- Measures success of its projects by their conformance to the plan
- Values methodology and its processes more than the people
- Resists change in software requirements and development processes
- Detailed documentation is important

Agile Business Transformation

THE AGILIST MIND

- Accepts that predictability in business software is impossible
- Timeline and Cost are fixed but NOT the scope
- Success of project is measured by the value it gives to the customer
- Values people more than the process, hence it is accepted as a process instead of imposing it
- Welcomes change
- Functioning software is more important

Agile Vs Traditional Methodologies

Agile methodology vs. traditional software development methodology on various parameters:

Parameter	Traditional	Agile
Requirements	Fixed	Evolve
Time & People	May vary	Fixed
Customer Involvement	Before, After	During
Negotiable	Estimates	Schedule
Testing	After code	Integrated
Feedback	After	During
Concentration on	Processes; reviews	Workable software
Focus	Plan driven	Value driven
Stages	Requirements, Design, Code, Test, Feedback	(Plan-do-adapt)*

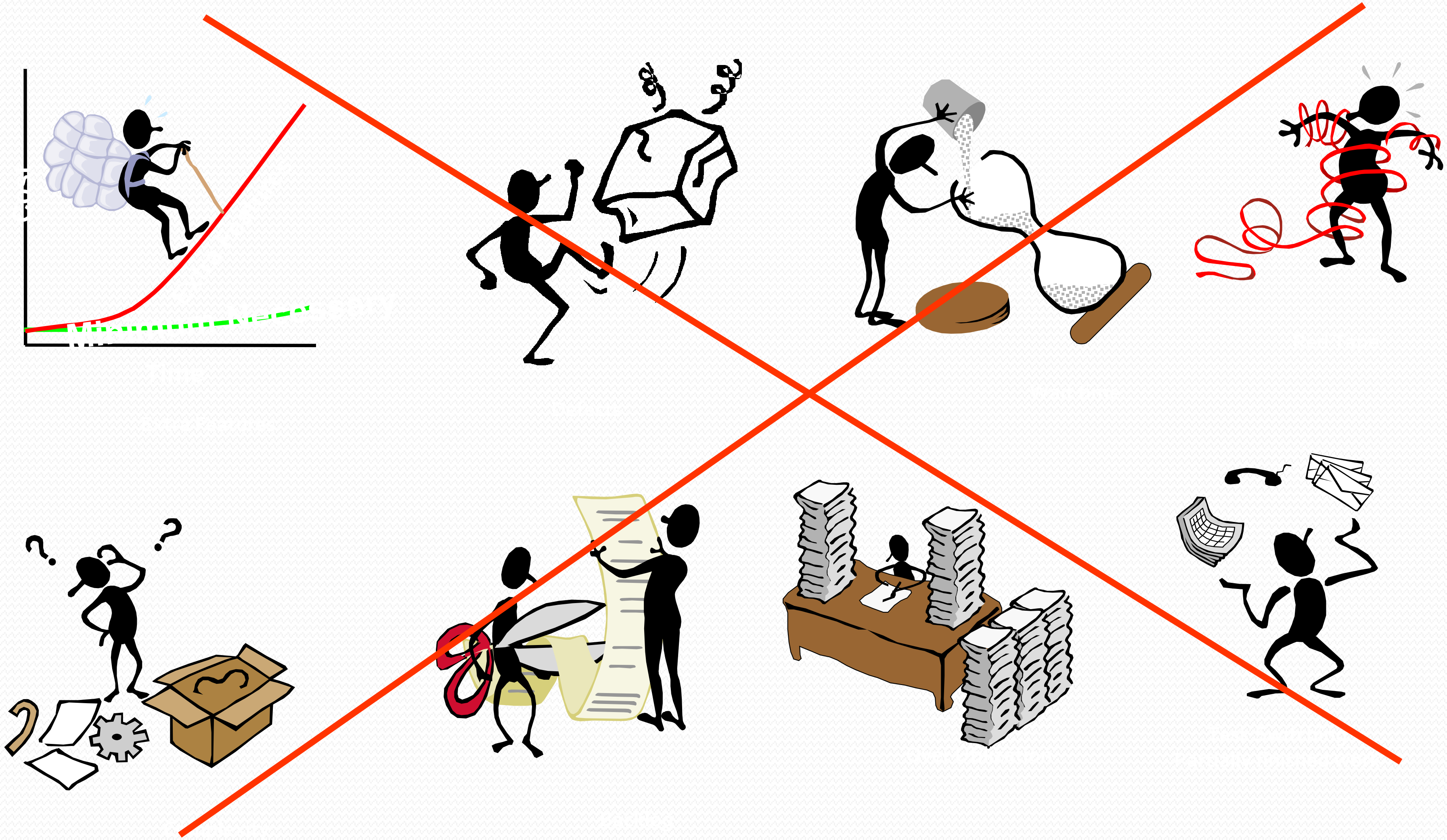
Agile in 30 seconds

Agile is just the use of continuous stakeholder ***feedback***
to deliver ***high quality*** *consumable code*
through use cases and a series of ***short time-boxed iterations***

1. at the end of every iteration you have stable code
   - that means it actually works, no sev 1s, sev 2s, few sev 3s.
   It's done, done, Done!
2. we have meaningful stakeholder interaction
3. we trust the team to do the work they need to
4. the team works at a sustainable pace

That's all it is!

Eliminate Waste



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:

Individuals and interactions **over processes and tools**

Working software **over comprehensive documentation**

Customer collaboration **over contract negotiation**

Responding to change **over following a plan**

www.agilemanifesto.org

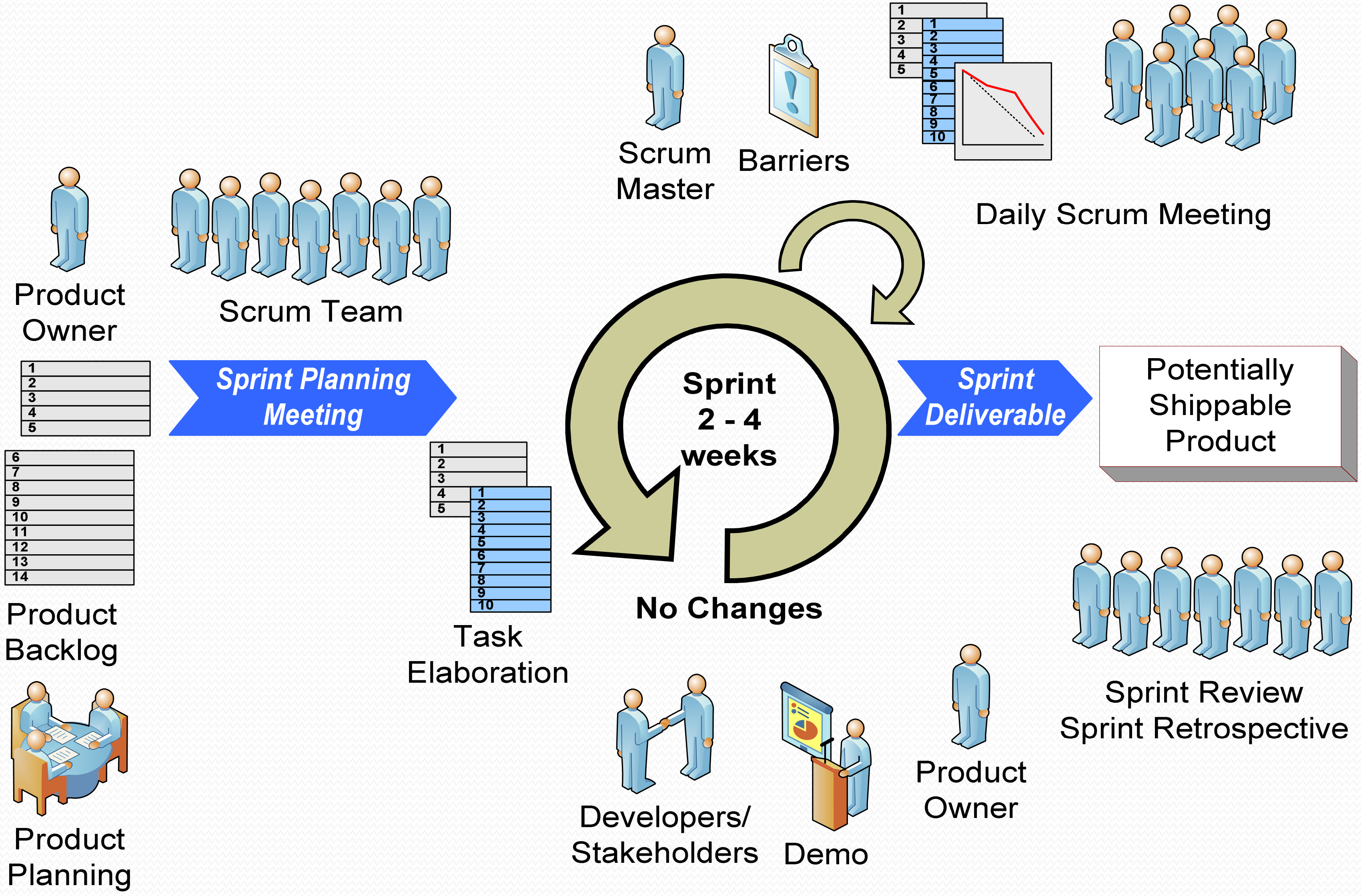
www.agilemanifesto.org/principles.html

Agile Methodology : Types

There are many modeling techniques for Agile approach but the most commonly used are

- Scrum
- eXtreme Programming (XP)
- Feature Driven Development (FDD)
- Dynamic Systems Development Method (DSDM)
- Unified Process ##

Typical Scrum Process



Scrum Terminology

Roles

- ✓ Product Owner
- ✓ Scrum Master
- ✓ Scrum Team – Developers and QA Testers
- ✓ Stakeholders and SMEs

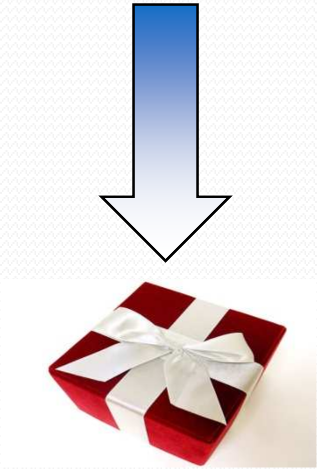
Practices

- ✓ Sprints and Sprint Planning Meeting
- ✓ Story, Task & Story Points
- ✓ Product Backlog (Back Burner)
- ✓ Sprint Backlog (Front Burner)
- ✓ Daily Scrum
- ✓ Sprint Review and Retrospective
- ✓ Burn Down Chart

Role - Product Owner

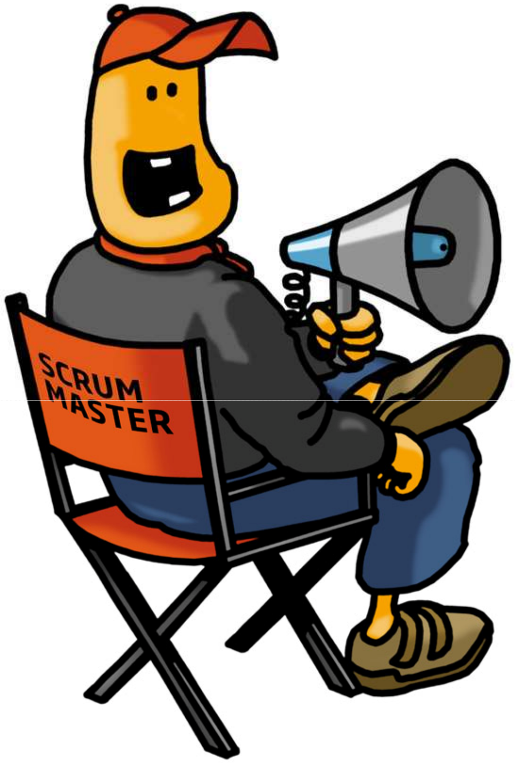
Voice of customer

- ✓ Define the features of the product
- ✓ Decide on release date and content
- ✓ Be responsible for the profitability of the product
- ✓ Prioritize features according to market value
- ✓ Accept or reject work results.
- ✓ Adjust features & priority every iteration, as needed
- ✓ Ensures that the Scrum Team works with the right things from a business perspective. ##



Shippable Product

Role - Scrum Master



- ✓ Coaches Development Team
- ✓ Meets with the team every day
- ✓ Removes possible impediments
- ✓ Ensure that the team is fully functional and productive
- ✓ Acts as SPOC for outsiders ##

Constantly works to ensure that the team has the best possible circumstances for realizing the goals fixed for the Sprint.

Role - Scrum Team

- ✓ Consists of 5-9 people
- ✓ Self-organized team
- ✓ Self directed team
- ✓ Members have a joint responsibility for the results
- ✓ The team members decide how the work is arranged and how assignments are distributed. ##

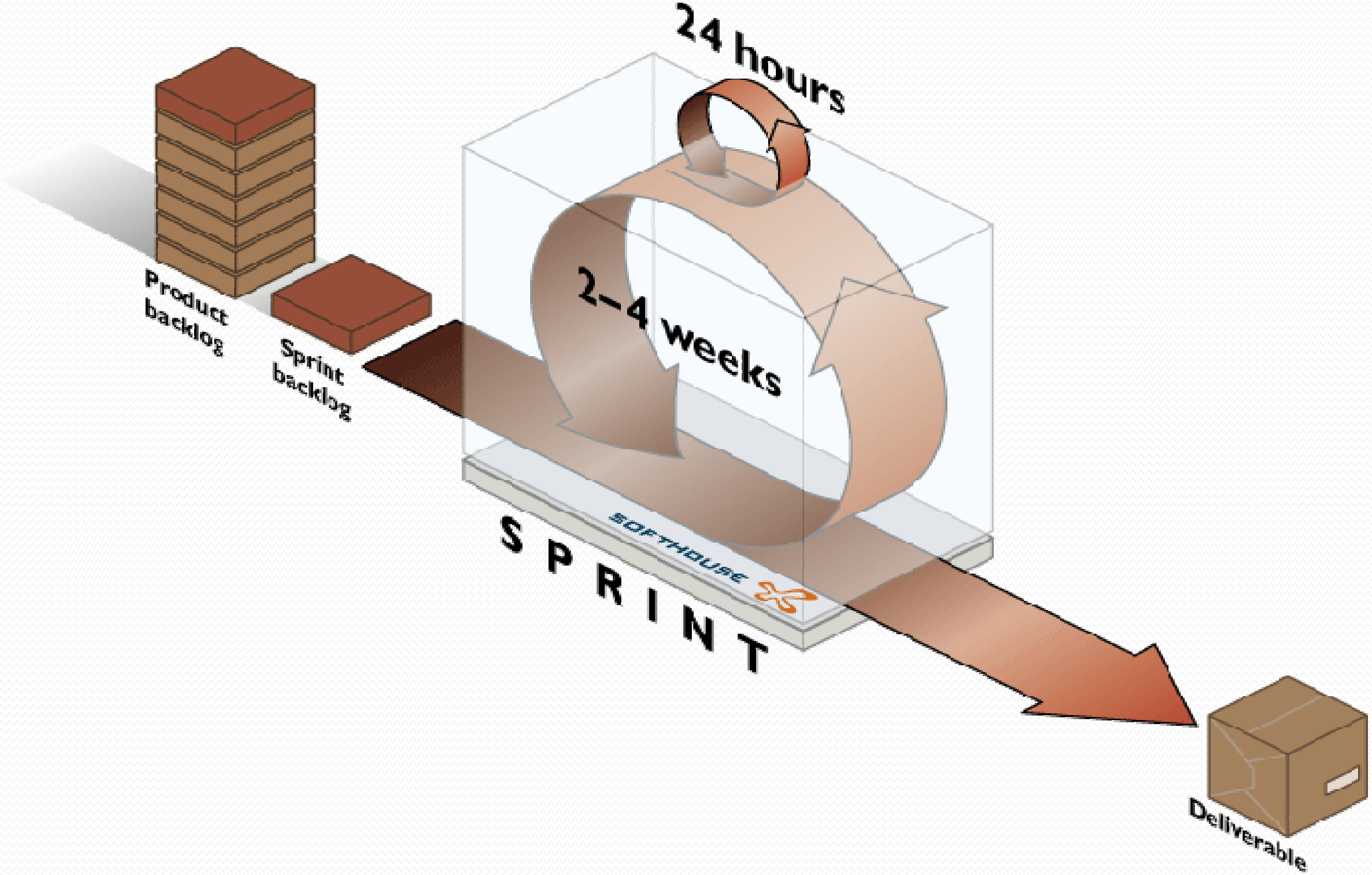
There are no set project roles
+everyone should be able to swap tasks with another member.

- Prevent individual members from being experts in a field.

Practice - Sprint

- Each small iteration in Scum is known as Sprint.
- Duration of sprint can be 1 week and maximum 4 week.
- For each sprint capacity (number of resources and their availability hours) is considered. Capacity will help to calculate how many points can be completed in a sprint.

Practice : Sprint



Practice - Sprint Planning Meeting

- Responsibility : Product Owner
- Entry point : Product Backlog
- Objective : To discuss and develop a detailed plan for the iteration
- Duration : Around 4 hrs.
- Product Owner, SME, Scrum Master and Scrum Team are the participants of the meeting
- Product Owner reviews the vision, the roadmap, the release plan and the Product Backlog with the Scrum team
- The team decides and commits how much work it can successfully take into the sprint based on team size, available hours and level of team productivity. ##

Practice – Story, Task & Story Points

- Story –

  Each feature introduced in sprint is know as Story.

- Task –

  To complete a story different activities needs to be completed. Each activity related to story is known as task. For each task developer needs to assign estimated hours required to complete.

- Story Points –

  Considering Total Detailed Estimated hours and complexity of story Points are identified. ##

User Stories: Components

- A User Story describes functionality that will be useful to a stakeholder of the system.

- User Stories are comprised of three things
  - A brief description of the story used for planning
  - Conversations about the story
  - Tests that convey and document details

User Story – the 3 Cs

Card:

- Stories are traditionally written on note cards
- May be annotated with notes, estimates etc.

Conversation:

- Details behind the story come out during conversations with product owner

Confirmation:

- Acceptance tests confirm the story was coded correctly

User Stories: Card: A brief description

ser Story takes the form of:

As a <Role>, I want to <Goal> so I can <Business value>.

Examples:

Title: Set up AGILE support structure and process

Description: As Fund-TECH **Delivery Manager**, I want a **small team to be staffed** to help projects with AGILE development**, how to get started, client communication, pitfalls**, etc.

Practice - Product Backlog (Back Burner)

- A to-do list that is constantly prioritized – Backlog
- Prepared by Product Owner
- List of customer requirements prioritized by business value
- Should include all features visible to the customer, as well as the technical requirements needed to build the product
- The highest priority items in the Product Backlog need to be broken down into small enough chunks to be estimable, testable and deliverable ##

Practice - Sprint Backlog (Front Burner)

- An artifact of the Sprint Planning Meeting
- A list of the specific development tasks required to implement a feature
- Scrum Team has selected and committed to deliver a set of top priority features from the Product Backlog, the Product Backlog's features are broken down into a Sprint Backlog
- These tasks are broken down into pieces that will require less than two days (or sixteen developer-hours) of work. When the Sprint Backlog is complete, the total work estimated is compared with original estimates from the Product Backlog. If there is a significant difference, the team negotiates with the Product Owner to get the right amount of work to take into the Sprint with a high probability of success ##

Practice - Daily Scrums

- Each day the Scrum Master leads the team in the Daily Scrum Meeting. This is a fifteen-minute meeting designed to clarify the state of the Scrum

- The goal is to get a global snapshot of the project, discover any new dependencies, address any personal needs of committed individuals, and adjust the work plan in real time to the needs of the day

- Three questions:

  - ✓ What have you done since last meeting?
  - ✓ What will you do between now and the next meeting?
  - ✓ Is there anything preventing you from doing what you have planned?

- Only Scrum Master and team members allowed to speak ##

Practice - Sprint Review & Retrospective

Sprint Review –

- Demonstration of functioning software at the end of sprint
- Presented to a larger group, possibility including users, PO and management
- Basis for decision to turn to production #

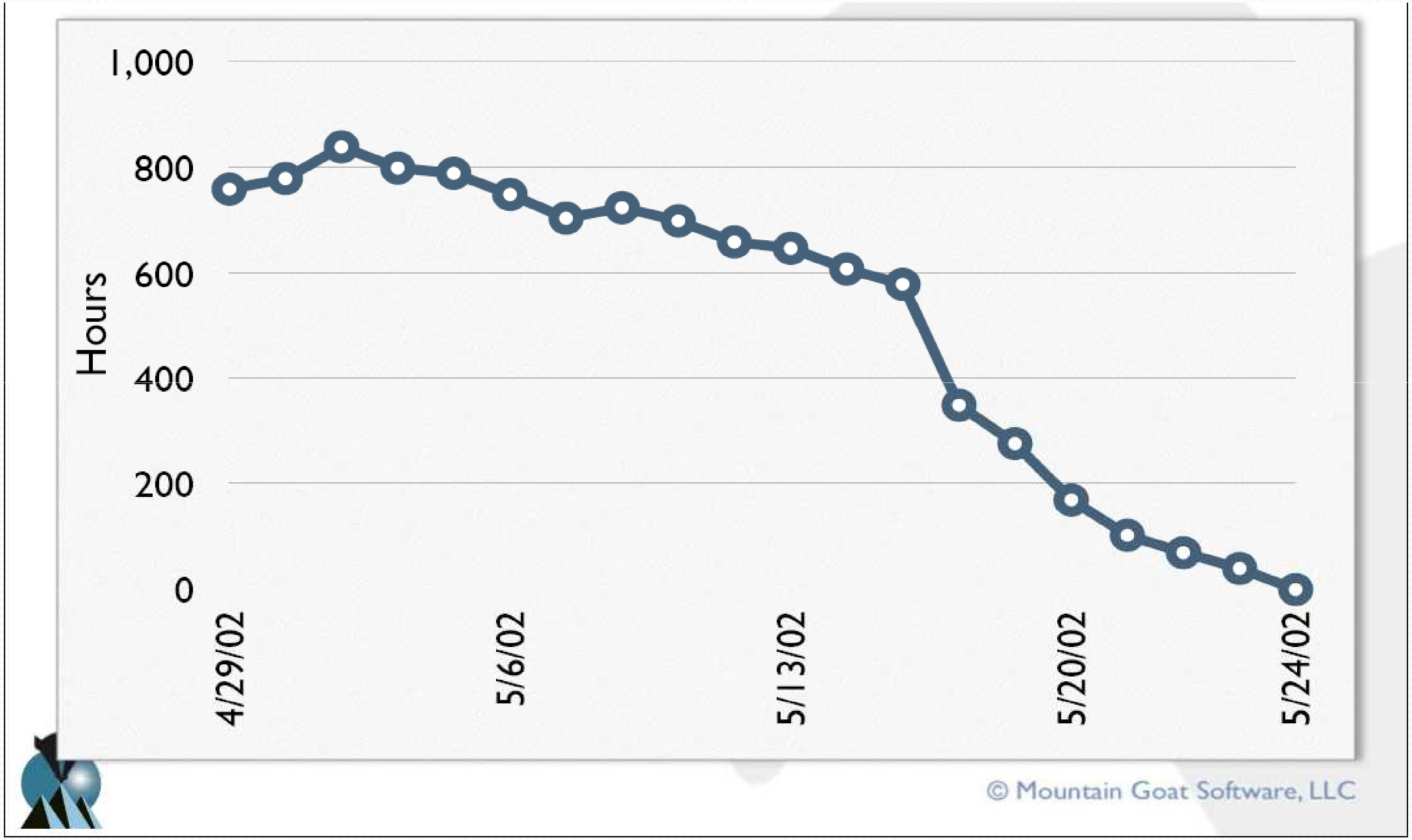
Retrospective –

- What is good and bad about this sprint
- Which are the areas where improvements are required
- Feedback for next Sprint ##

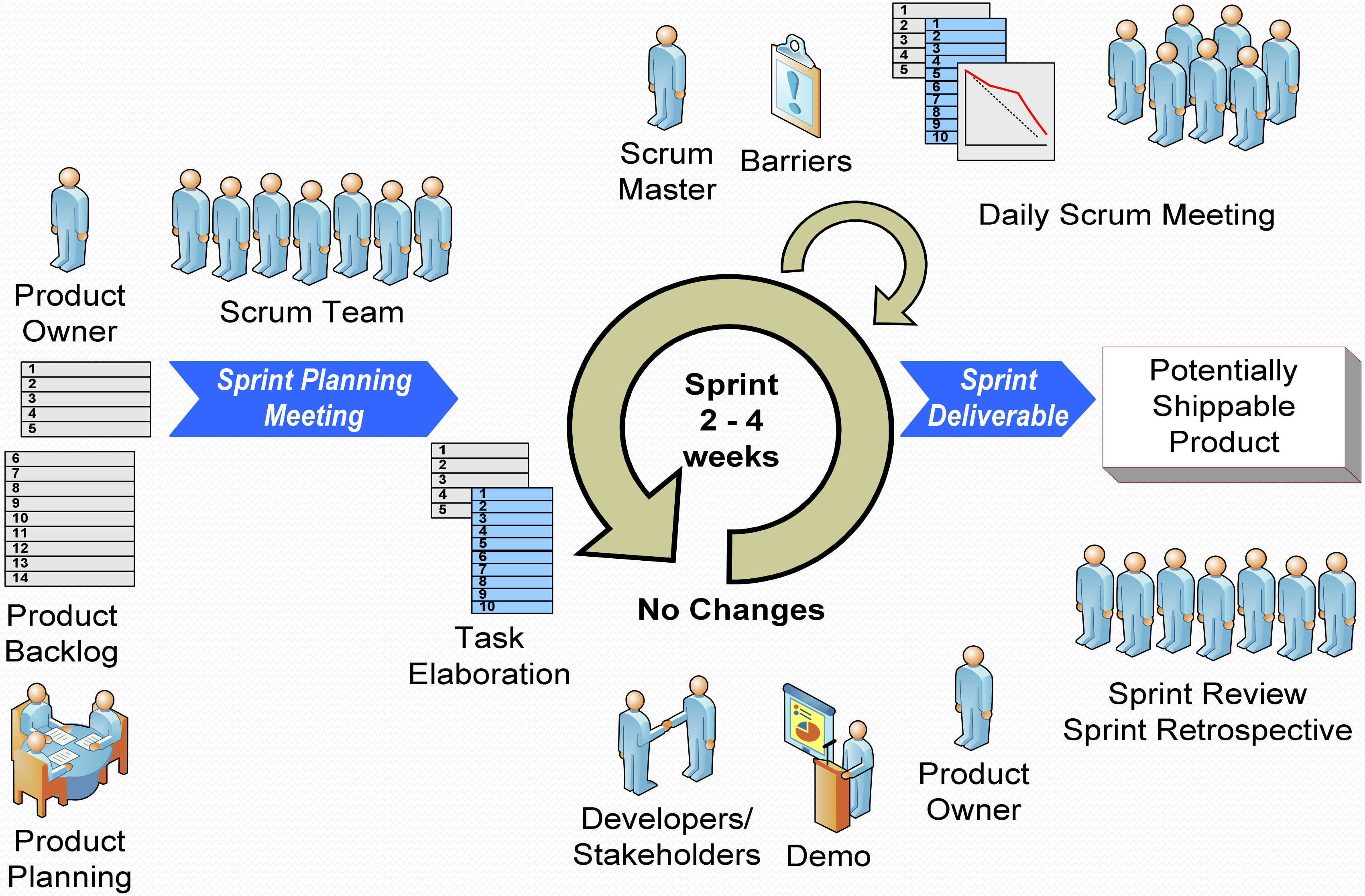
Practice – Burn Down Chart

- Shows the cumulative work remaining in a Sprint, day-by-day
- The Burn down Chart is used as a tool to guide the development team to successful completion of a Sprint on time with working code that is potentially shippable as a product
- The development team gains a better understanding of work to be done as time progresses and may find that they need to add new tasks to the Sprint Backlog to complete the Product Backlog items selected.
- Defects may be identified and logged as additional tasks. While these are viewed primarily as unfinished work on committed tasks, it may be necessary to keep track of them separately. ##

Iteration burndown chart



Typical Scrum Process



Scrum Master

Barriers

Daily Scrum Meeting

Product Owner

Scrum Team

Sprint Planning Meeting

Sprint 2 - 4 weeks

Sprint Deliverable

Potentially Shippable Product

Product Backlog

Task Elaboration

No Changes

Sprint Review
Sprint Retrospective

Product Planning

Developers/ Stakeholders

Demo

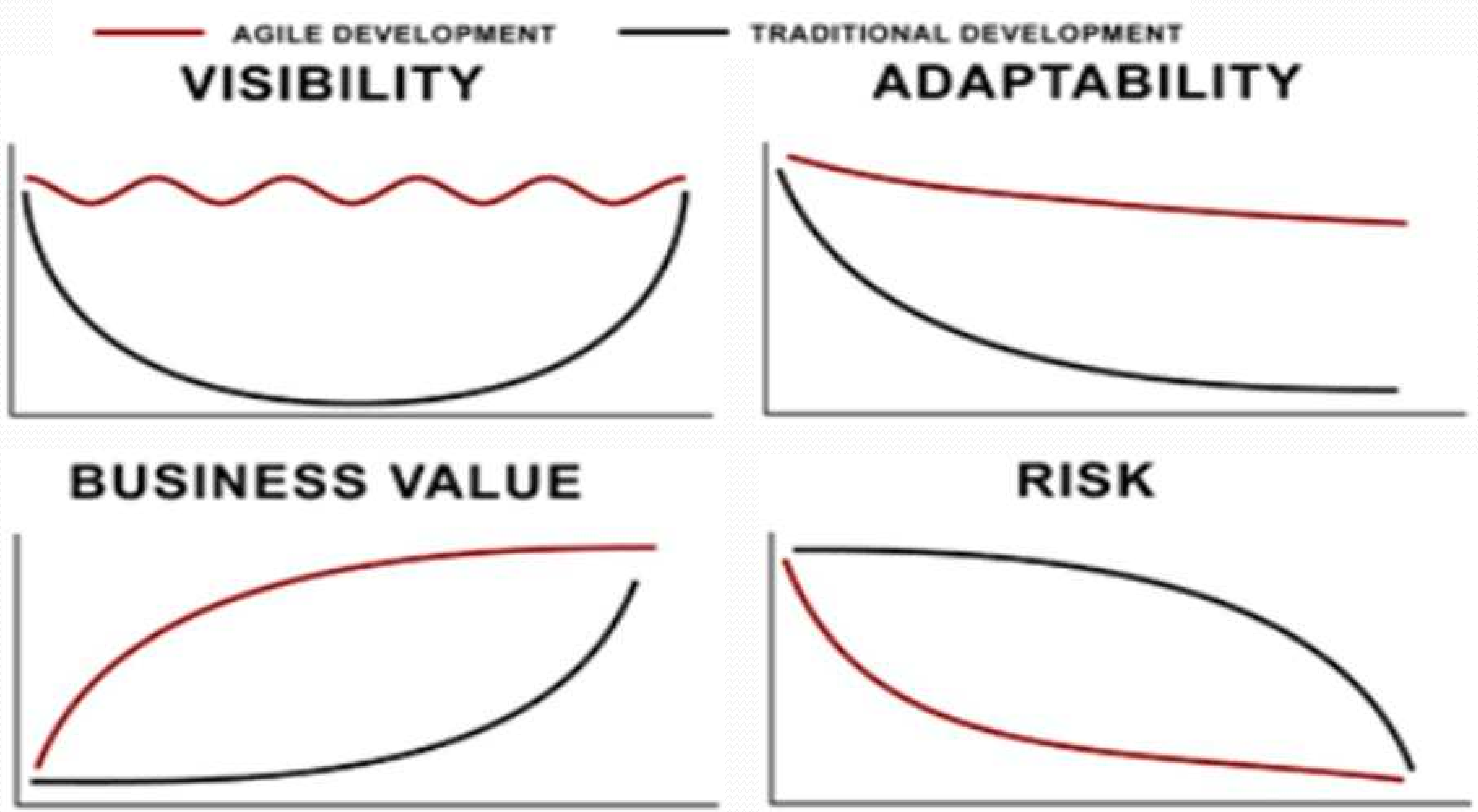
Product Owner

Scrum Principles : Summary

- Customer satisfaction by rapid, continuous delivery of useful software
- Working software is delivered frequently (weeks rather than months)
- Working software is the principal measure of progress
- Even late changes in requirements are welcomed
- Close, daily cooperation between business people and developers
- Face-to-face conversation is the best form of communication (co-location)
- The best architectures, requirements and design comes from self organized team ##

Scrum VS Traditional Development

Agile Development Value Proposition



Scrum Benefits & Constraints

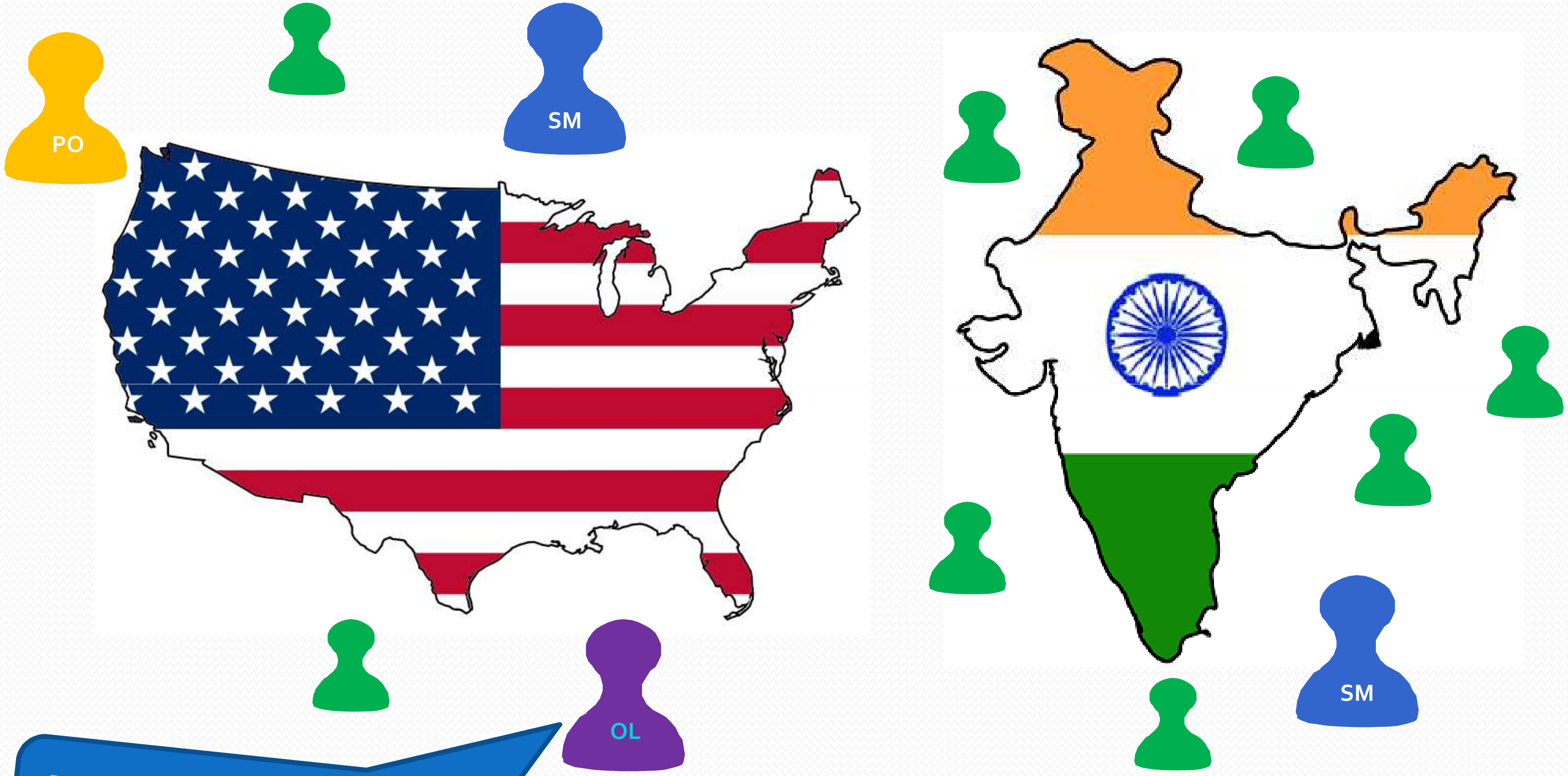
Benefits –

- More predictable deliveries
- Early returns on investments, Working Software delivered and in use sooner
- Risk mitigation provided by shorter delivery cycles
- Quick response to changes in customer needs
- Realistic assessment of progress
- Relatively high productivity and quality

Constraints –

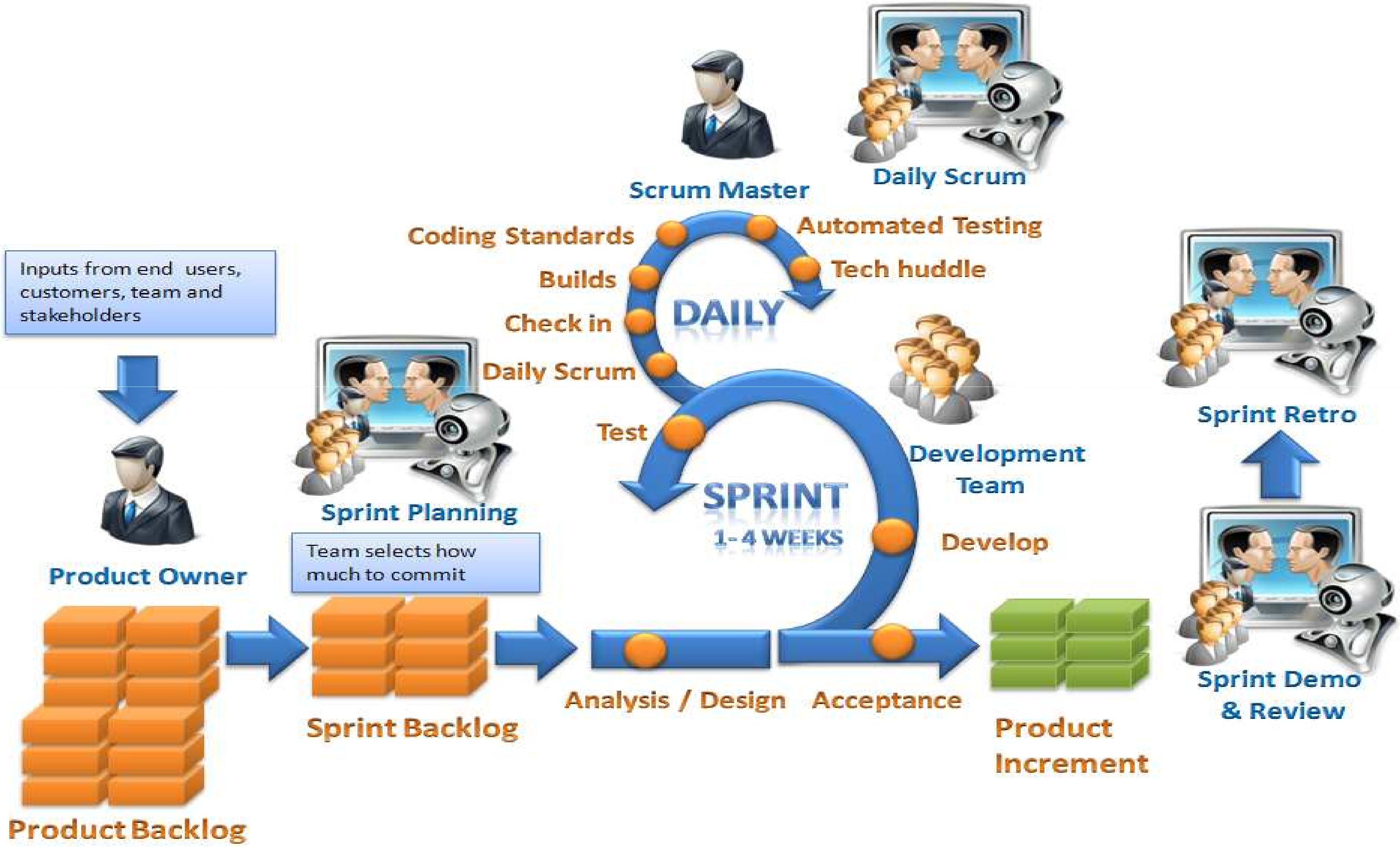
- Agile promotes face to face communication and expects people to be co-located
- Many People required at onsite Which is a costly affair
- Daily productivity is limited to 8 hrs.
- Knowledge restricted to single geography

Distributed Scrum

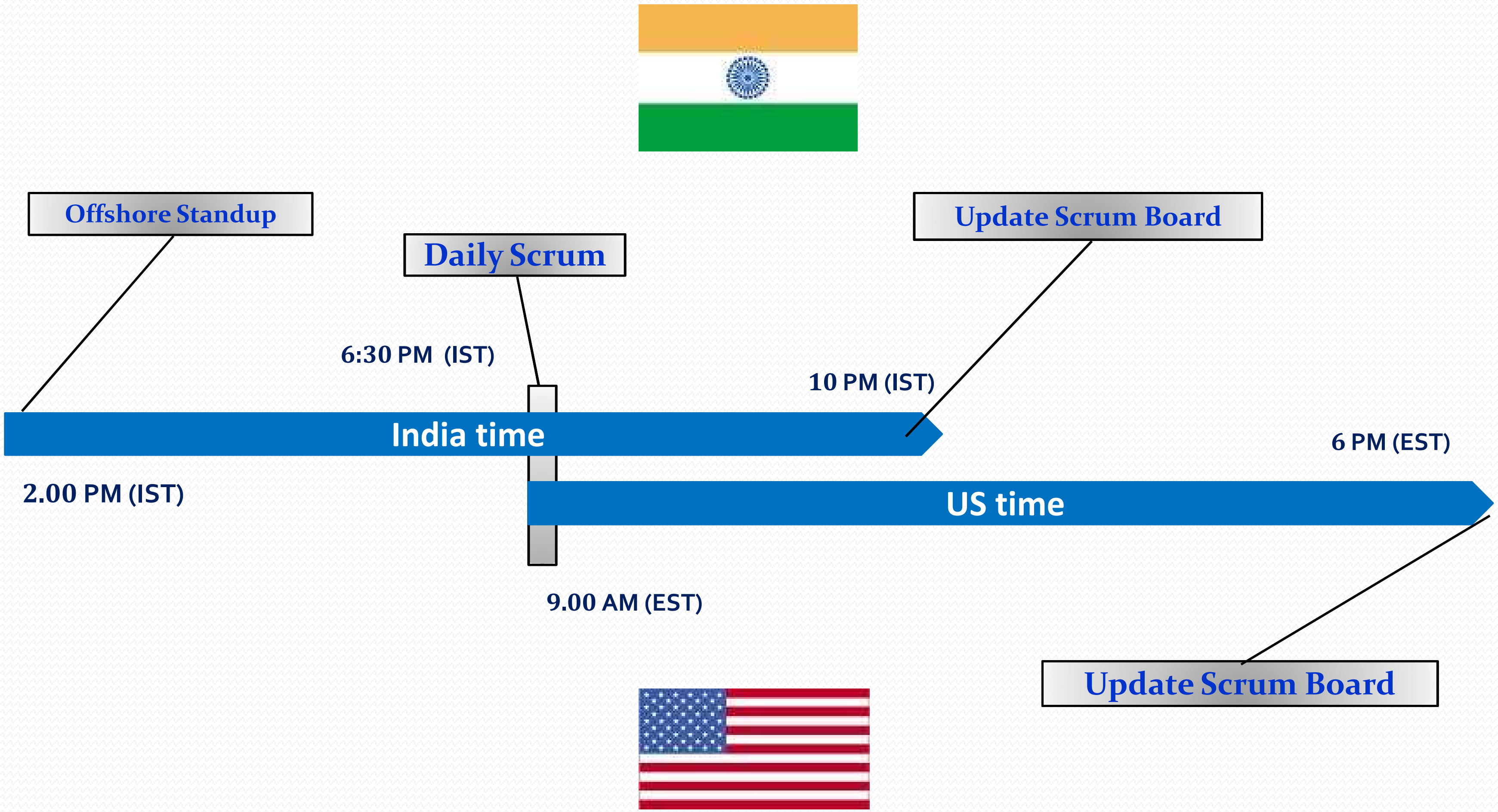


Onsite Liaison (OL) – Point of Contact between Offshore Devs, Onsite Devs, Scrum Master and the Product Owners

Distributed SCRUM Process



A Day in Distributed Scrum



Offshore Standup

Daily Scrum

Update Scrum Board

6:30 PM (IST)

10 PM (IST)

India time

6 PM (EST)

2.00 PM (IST)

US time

9.00 AM (EST)

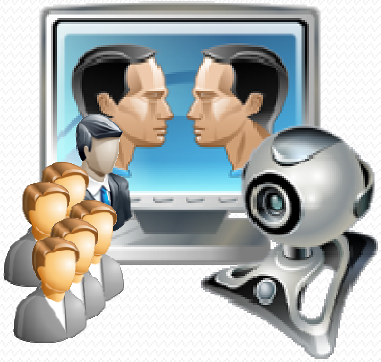
Update Scrum Board

Communication in Distributed Scrum

Pre - SPM

🇮🇳

- Review Product Backlog
- Post questions to PO in advance
- Commitment plan for Sprint planning.
- Defects Prevention analysis for previous Sprint
- Analysis of Estimation variance for previous Sprint



SPM

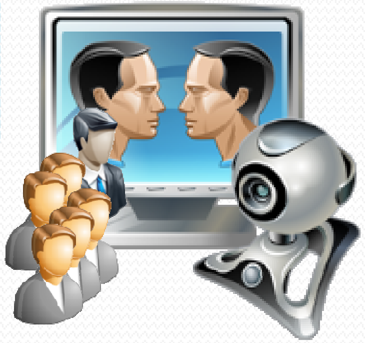
🇮🇳 🇺🇸

- Estimate Sprint backlog
- Task out the stories
- Commit

Story Discussion

🇮🇳

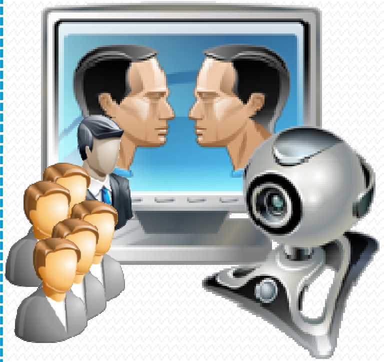
- Discuss each story on the Sprint task board
- Knowledge sharing within the team
- Action plan for the Sprint



Daily Scrum

🇮🇳 🇺🇸

- Each developer updates:
  - What was done since last meeting ?
  - Are there any impediments ?
  - What is he planning to do till next meeting?
- Scrum Master ensures that the impediments are sorted out.



Demo &Retro

🇮🇳 🇺🇸

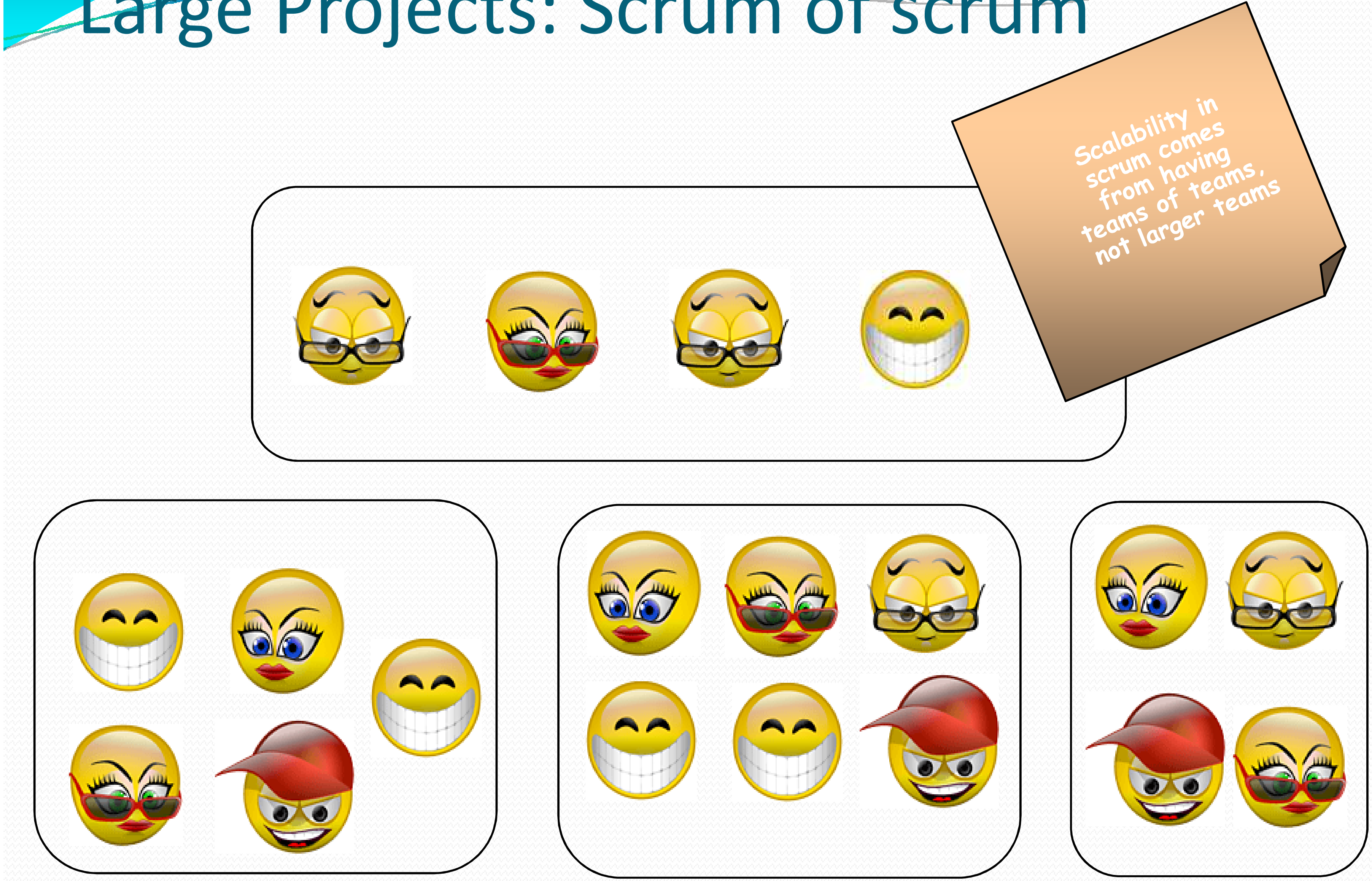
- Demonstrates features accomplished in each story to PO /stake holders
- Retrospective to find out:
- What went well?
- What needs improvement ?
- Checks previous retro to see if we improved on 'needs improvement' items

Information Sharing in a Collocated Team

Benefits in Distributed SCRUM

- OGILE: Cost benefits of going offshore and going Agile
- Distributed SCRUM (OGILE) offers same velocity as localized SCRUM (AGILE)
- OGILE offers advantages of strong Offshore development model
- Knowledge distribution across Geographies
- Fully distributed SCRUM has more value than a localized SCRUM – “Dr. *Jeff Sutherland*”

Large Projects: Scrum of scrum



Scalability in scrum comes from having teams of teams, not larger teams

Running the scrum of scrums

Attendees:

- Each team sends an individual contributor
- If 4 or fewer teams – it is OK to send the scrummaster as well
- Rotate based on whose skills are needed most

Frequency:

- Some say daily
- MWF or TuTh often used
- These are problemsolving meetings
- Not timeboxed to 15 min

Agenda:

- Everybody answers 4 questions
- Attendees discuss the product backlog for the scrum of scrums

The four questions

1 What has your team done since we last met?

2 What will your team do before we meet next?

3 What's in your teams way?

4 What are your about to put into another teams way?

Rule:

No names during this discussion !

End of Session