

# Introduction to Agile Software Development Practices

Dev Barua, CSM, CSPO, PMP  
Sr. Development Manager



Tech solutions. Human support.





# Scope of the presentation

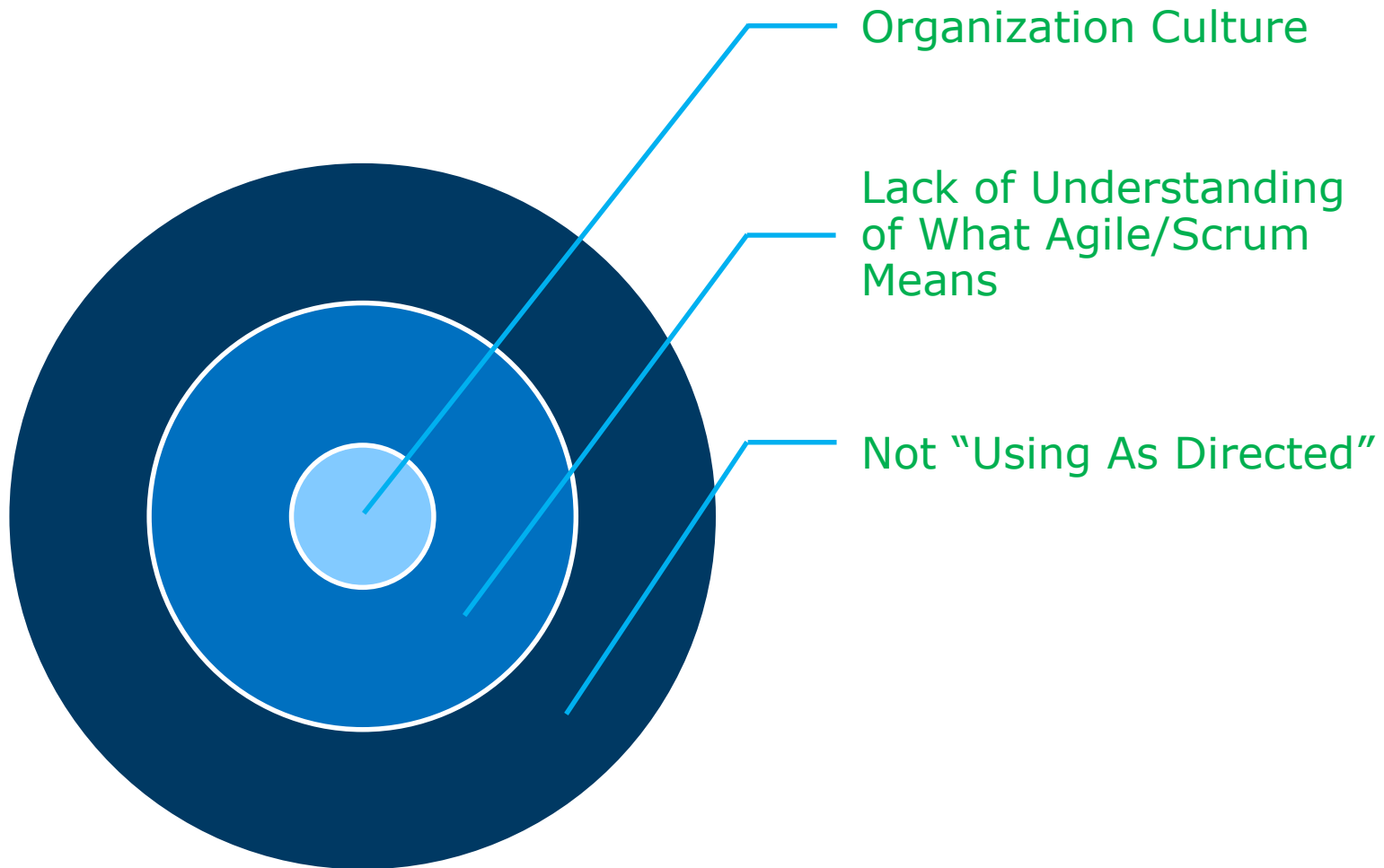
---

- Why organizations fail to adopt the agile development process
- Introduction to Agile Development
- Types of agile methods and an introduction to scrum
- How the agile-scrum process has been adopted at HelpSystems

# Top 3 Reasons Companies Struggle with Agile and Scrum



# Top Three Reasons



# Introduction to Agile



# What is Agile ?

---

## Agile is about enabling business results

- An approach to product development
- Adaptive – there is no “Specific or The Agile Method”
- To “be agile” put the values and principles into practice
- Processes such as Scrum and eXtreme Programming (XP) are considered “Agile” because they adhere to Agile’s Values & Principles



# Agile Methods

---



## Agile Methods

Extreme Programming (XP)

Scrum

Lean Software Development

Kanban

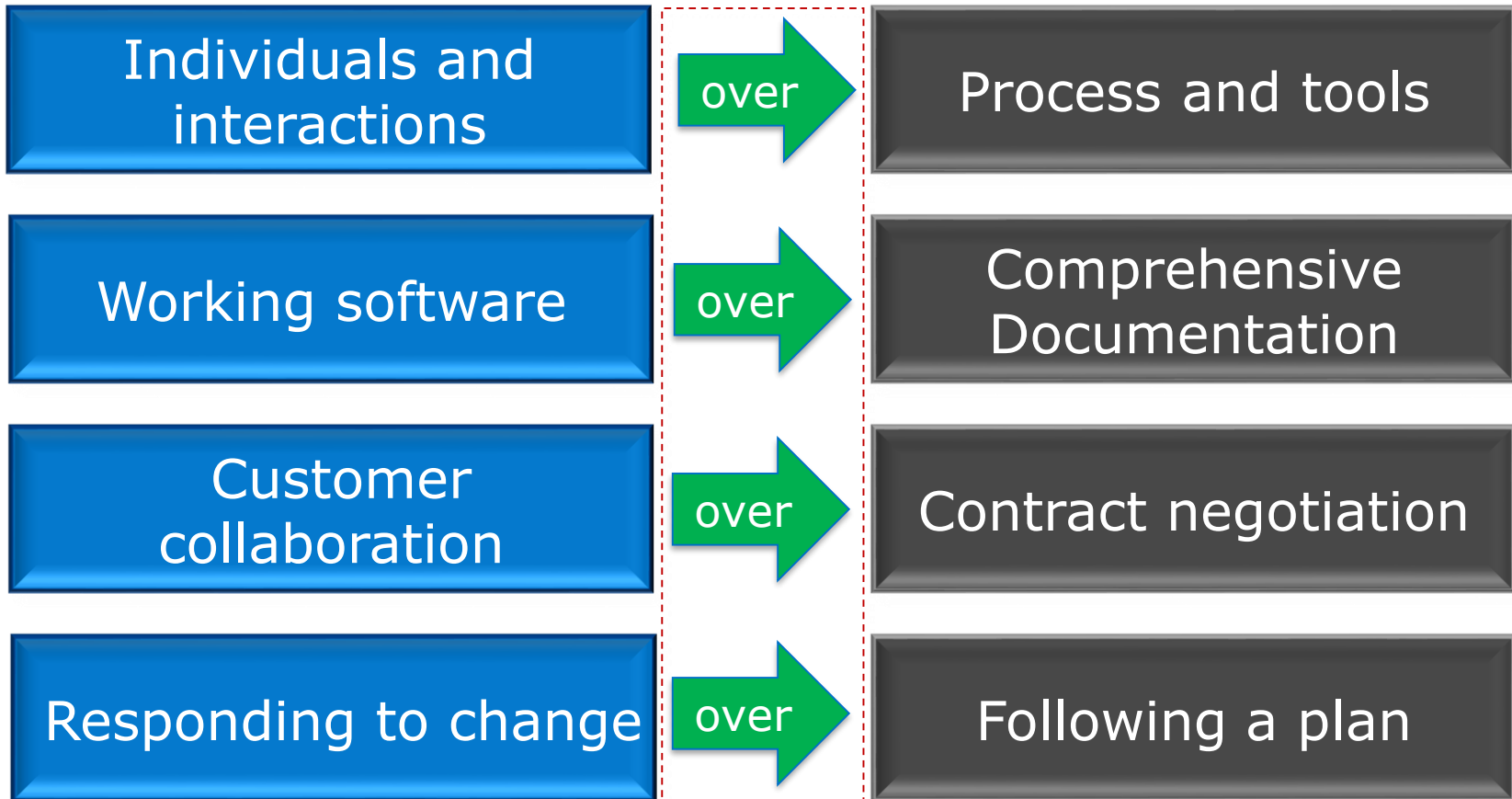
DSDM

Crystal Methodology

FDD



# The Agile Manifesto—a statement of values



Source: [www.agilemanifesto.org](http://www.agilemanifesto.org)



# Introduction to Scrum



# Introduction to Scrum

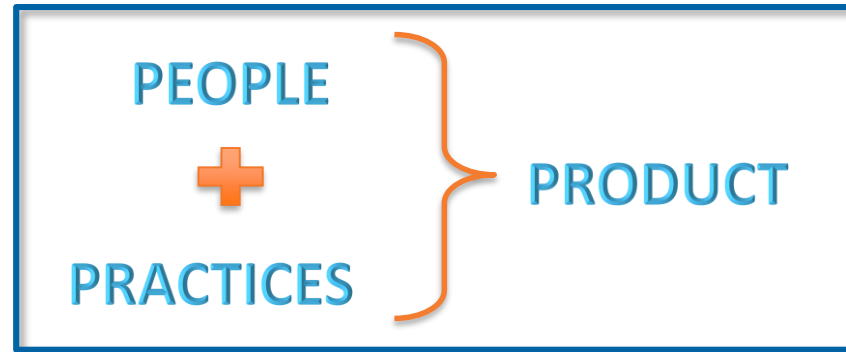
## It is...

- A simple **framework** that can be understood and implemented in short time
- An approach to **managing complexity**
- A **collaborative** effort that enables an engaged **ongoing dialog**
- **Most popular** Agile Method used today
- Has **industry supported** standards (roles, tools, certification, etc.)
- A culture change for the **entire organization**

## It is not...

- A **methodology**
- A license to do **NO documentation** (some documents still need to be created)
- A silver bullet for all that **ails software development**
- A framework which provides **detailed plans** for every contingency

# Mindset of SCRUM?



In the figure above, these three things depicts a simple equation:

***People use Practices to develop Product***



# Scrum Theory

Three pillars uphold every implementation of empirical process control: transparency, inspection, and adaptation.



## Transparency

Significant aspects of the Scrum process must be visible to those responsible for the outcome, i.e. the scrum team



## Inspection

Scrum users must frequently inspect Scrum artifacts and progress toward a Sprint Goal to detect undesirable variances.



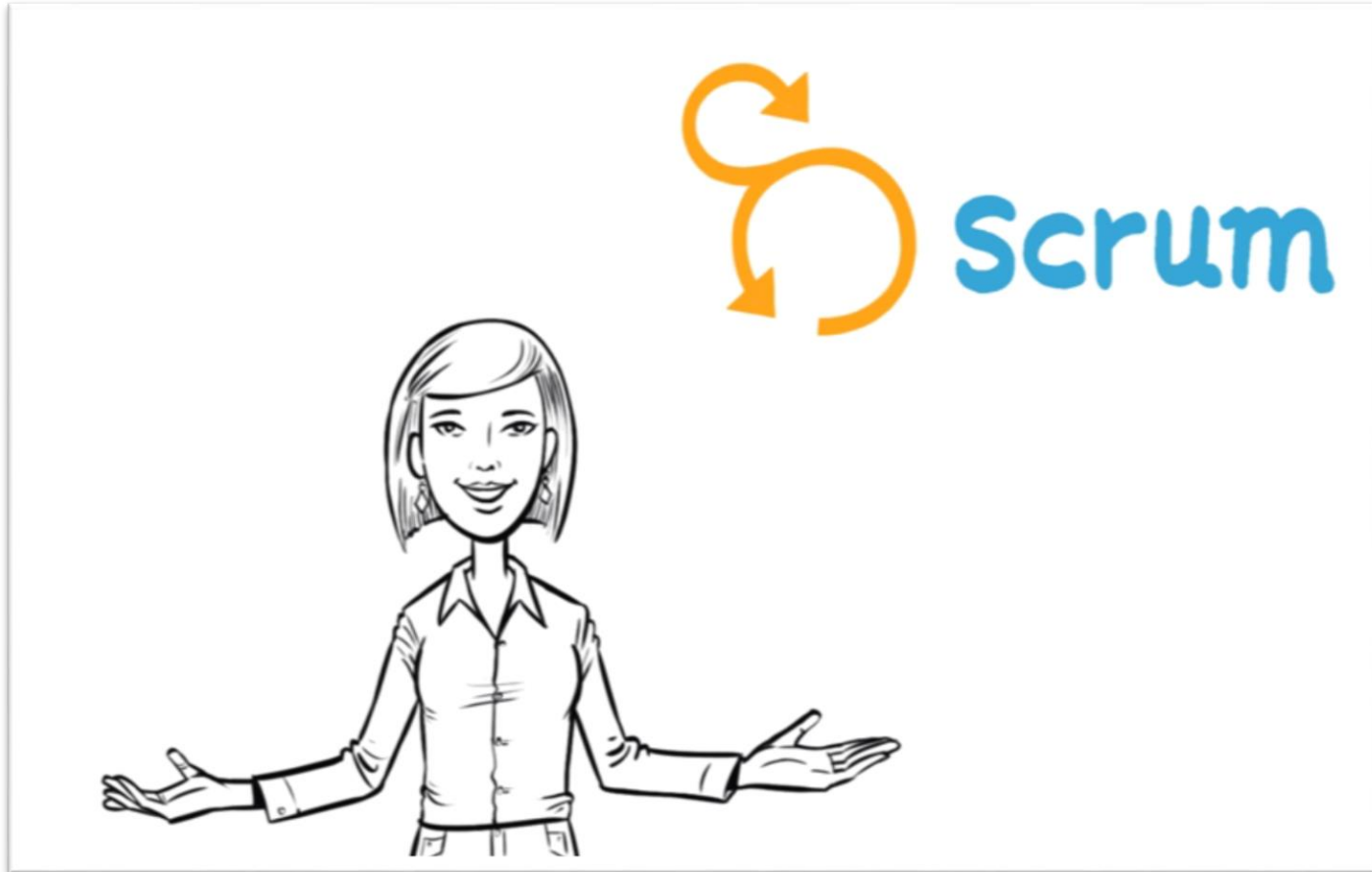
## Adaptation

If the Scrum determines that one or more aspects of a process deviate outside acceptable limits, the process must be adjusted.

Source: Scrum Guide by Ken Schwaber and Jeff Sutherland

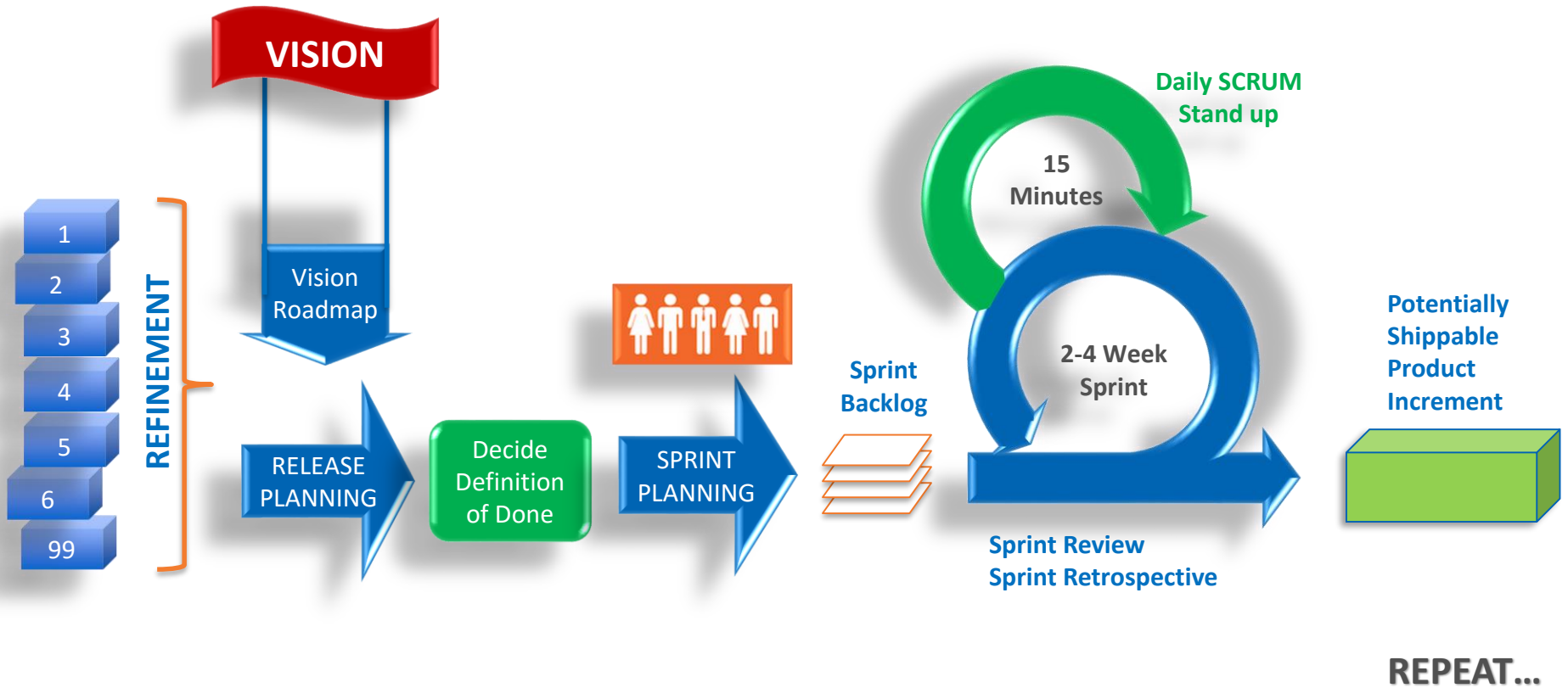


# Scrum Process





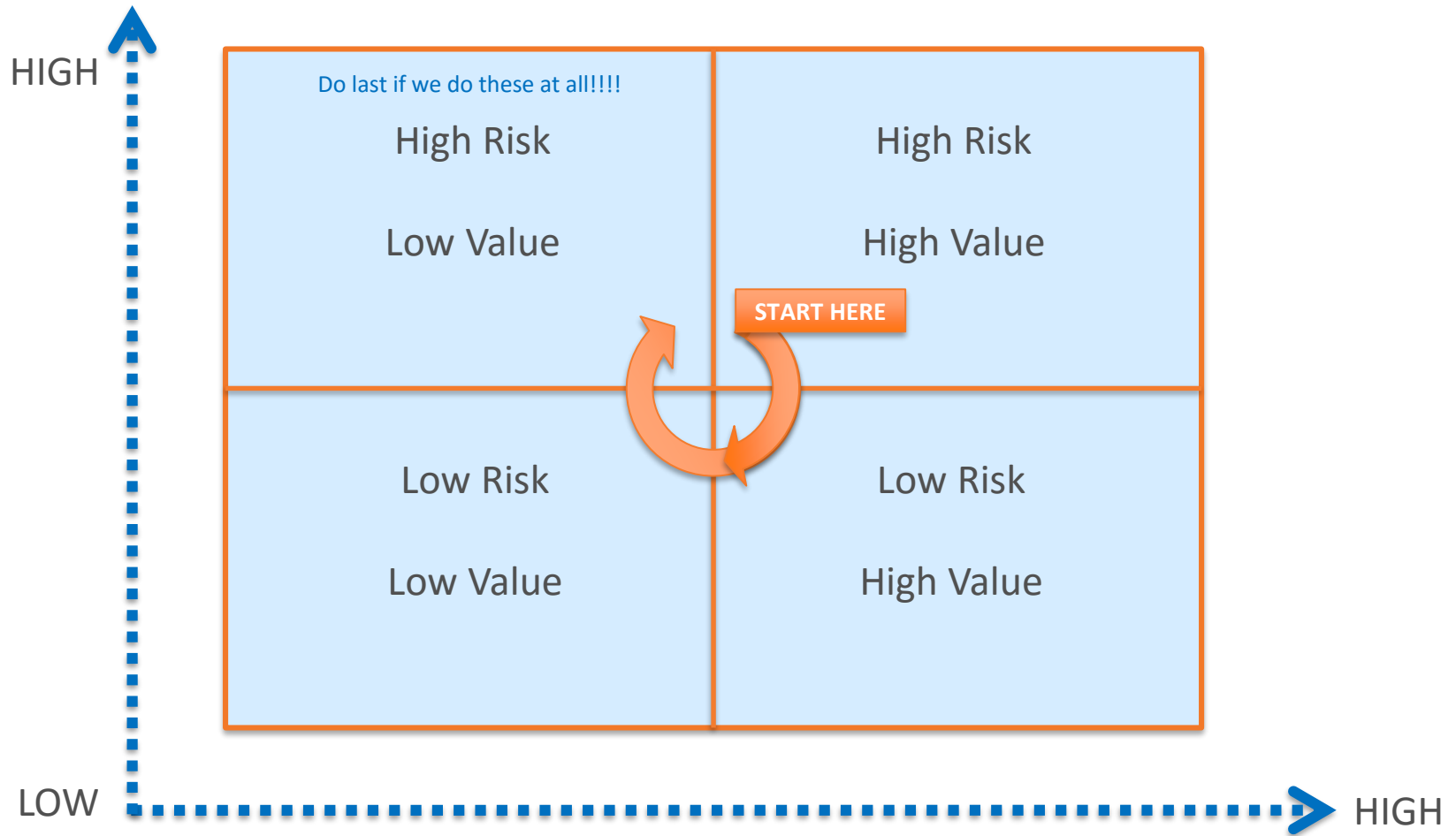
# SCRUM Development Process







# Product Backlog - Order/Re-order methods







# Elements of SCRUM - 3<sup>3</sup>

## PEOPLE- 3

- Scrum Team
- Product owner
- Scrum Master

## Ceremonies - 3

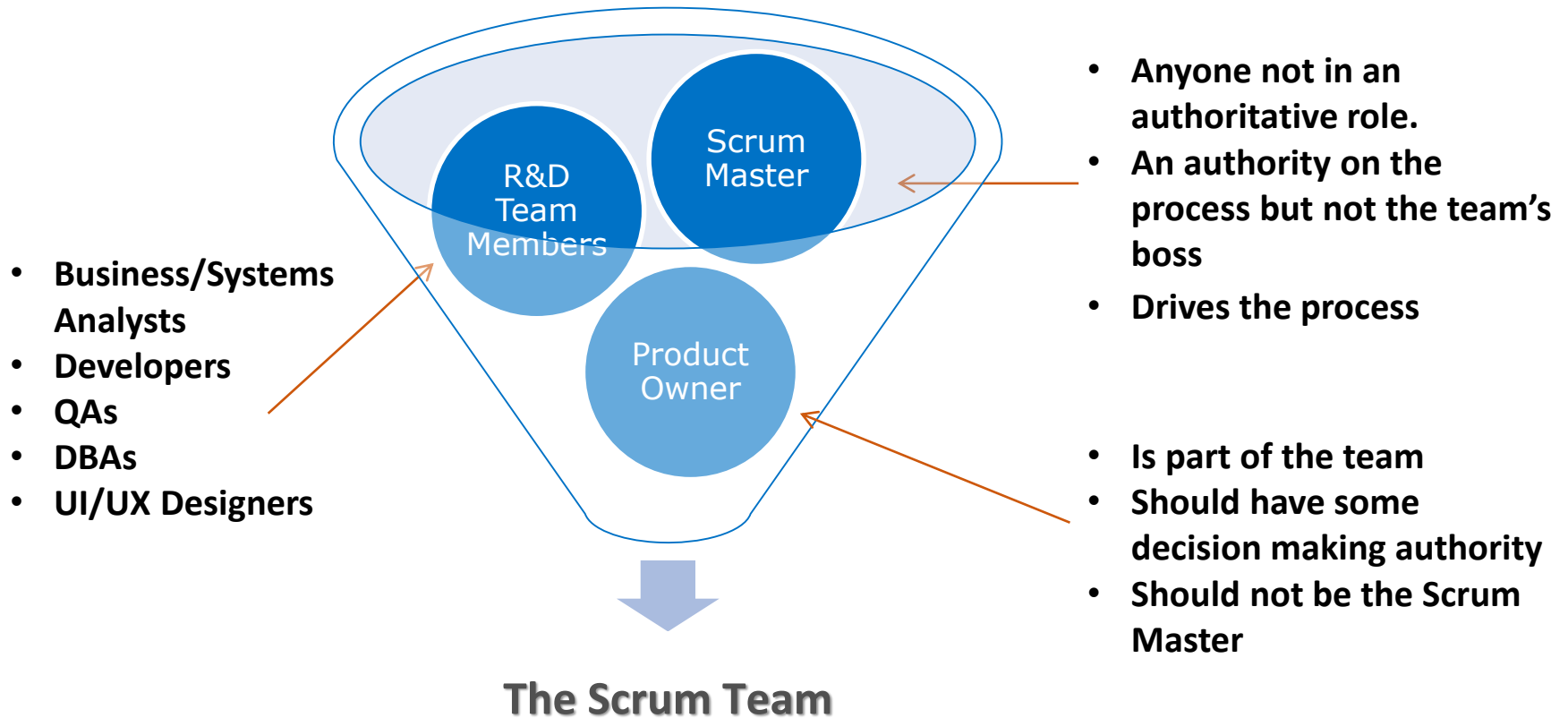
- Sprint planning
- Daily stand-up
- Sprint Review

## Artifacts - 3

- Product backlog
- Sprint backlog
- Burn down charts



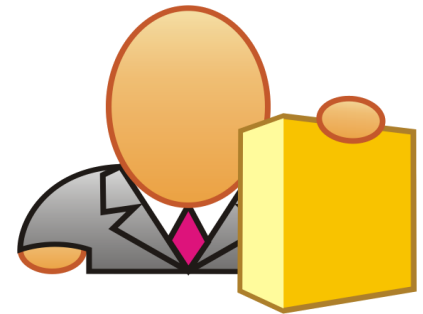
# Scrum Team – who are part of it?





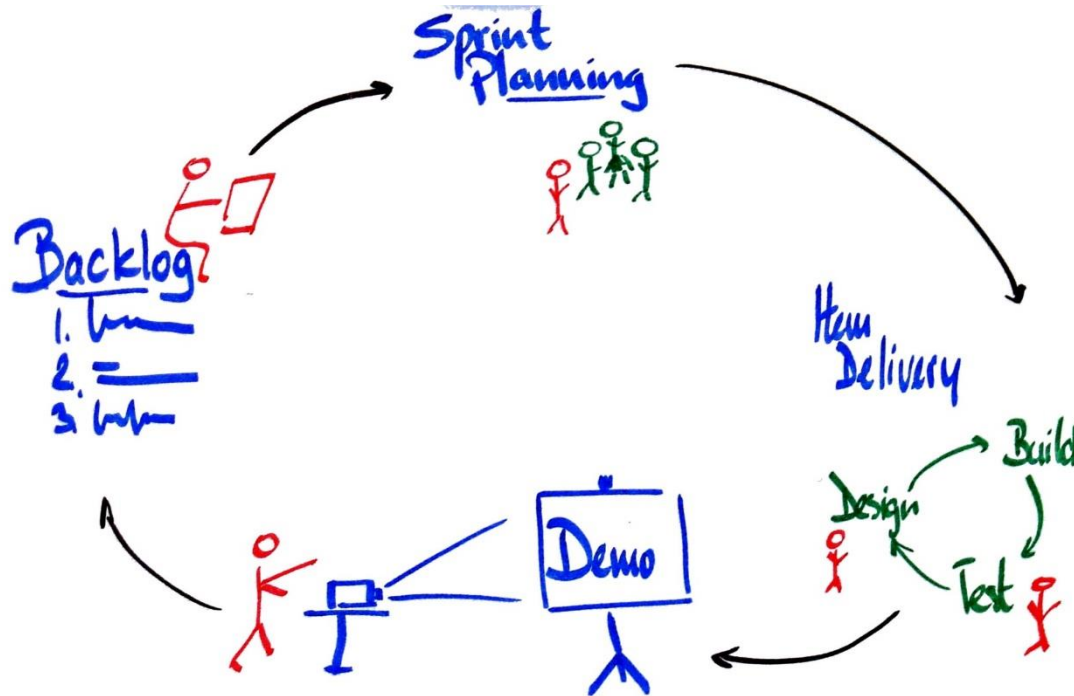
# Product owner

- Team Member who is accountable for the Business for the value of the Team's Work Results
- Be the primary interface with the Stakeholders, and can be a Subject Matter Expert (SME)
- Have a vision for the Product, its Releases, and the Sprints
- Responsible for maximizing the value of the product; define and prioritize the features of the product according to market value
  - Can change features and priority every X days
- Maintain and groom a prioritized Backlog of Product Backlog Items on regular basis
- Ensuring that the Product Backlog is visible, transparent, and clear to all, and shows what the Scrum Team will work on next;
- Determine what product the team will build, and drive the team as a sustainable pace
- Guides product development
  - Adjust features and priority every Sprint, as needed
- Seek guidance from the development team
- Has final say on the work product; can accept or reject work results (Sprint)





# Product Owner in Every Step of SCRUM



- Be the primary interface with the Stakeholders, and can be a **Subject Matter Expert (SME)**
- Acting as **Product Owner** during Sprint Planning, Sprint Demo and Backlog Grooming
- Acting as **Team Member** during Sprint Cycle.



# The Scrum Master

- Represents management to the project
- Responsible for ensuring Scrum is understood and enacted
- Scrum Masters do this by ensuring that the Scrum Team adheres to Scrum theory, practices, and rules
- Removes impediments
- Ensure that the team is fully functional and productive
- Enable close cooperation across all roles and functions
  - Manage Scrum Team's relationship with the Product Owner
- Shield the team from external interferences



***A Scrum Master Can be from any number of team roles: Project Manager, Team Lead, Dev Lead, QA, etc.***



# Elements of SCRUM

## Roles - 3

- Product owner
- Scrum Master
- Team

## Ceremonies - 3

- Sprint planning
- Daily stand-up
- Sprint Review

## Artifacts - 3

- Product backlog
- Sprint backlog
- Burn down charts



# Agile Ceremonies

---

- **Sprint (or Iteration) Planning:** Held at the beginning of each sprint for the team so commit Product Backlog Items to the Sprint Backlog
- **Daily Stand up:** a 15 minute time-boxed event for the Team to inspect, adapt and transparently synchronize on the Sprint Goal
- **Sprint Review (through Demo):** Feedback mechanism for stakeholders to see working product increments and for the Product Owner to inspect and adapt on the Product
  - **Sprint Retrospective:** an inspect and adapt mechanism for the Development Team regarding their Process
- **Release Planning:** a session that comes from the eXtreme Programming framework
  - A Release is typically made up of multiple Sprints or Iterations



# Elements of SCRUM

## Roles - 3

- Product owner
- Scrum Master
- Team

## Ceremonies - 3

- Sprint planning
- Daily stand-up
- Sprint Review

## Artifacts - 3

- Product backlog
- Sprint backlog
- Burn down charts





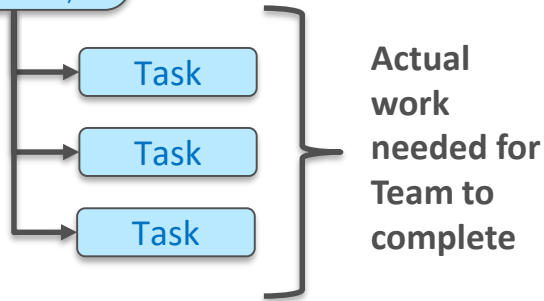
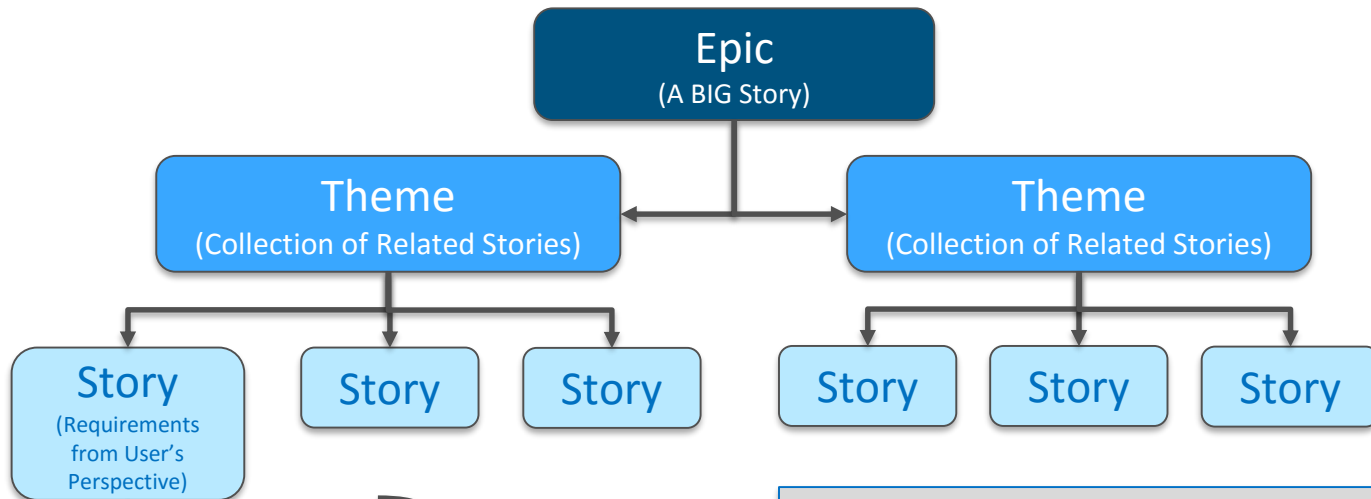
# Product Backlog

---

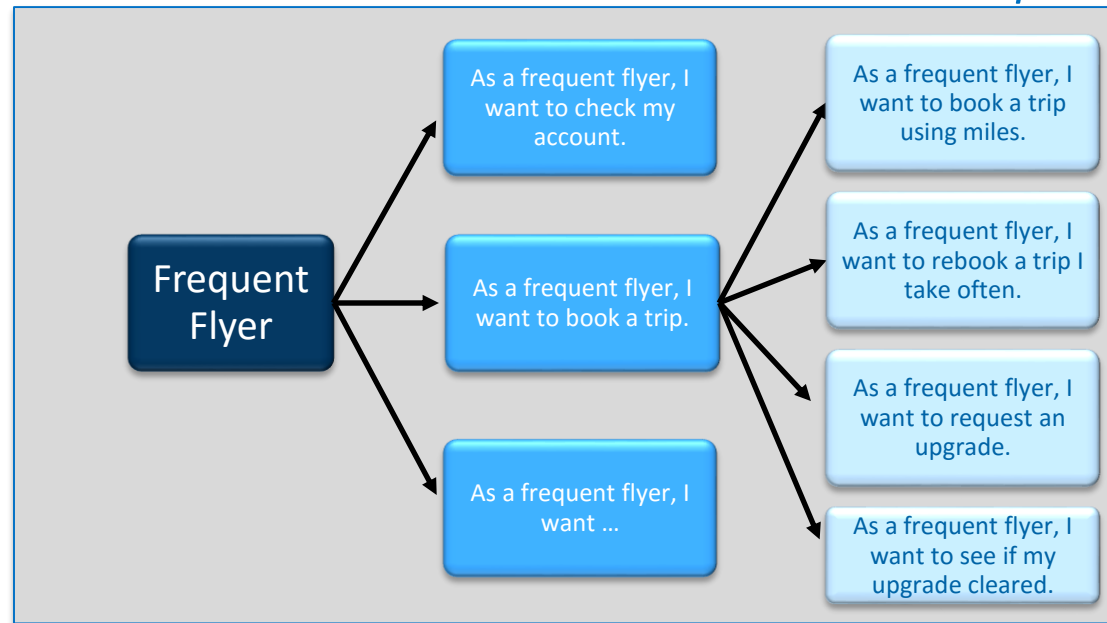
- **SINGLE SOURCE OF REQUIREMENTS and TRUTH**
- An ordered list of everything that potentially be in the Product
  - All features, functions, requirements, enhancements, and fixes
- Is never complete; constantly changes to identify product needs
- Product Owner owns the product backlog
- The Product Owner is responsible for the Product Backlog, including its content, availability, and ordering
  - Frequently re-ordered/re-prioritized
- Reprioritized at the start of each Sprint



# How to breakdown backlog items



*Example:*





# How to write user stories

- User Stories typically follow a simple template:

*As a <type of user>, I want to <goal>, so that <reason>*

Example 1: *As a HR Admin, I want to see all of the time off requests in a queue, so that I can approve those in the order that I have received them.*

- “If we work on the back-end of the software development, how can I write a user story?”

Example 2: *As the document imaging system, I want to receive all data as valid, well-formed XML so that I don't have to worry about syntax checking.*

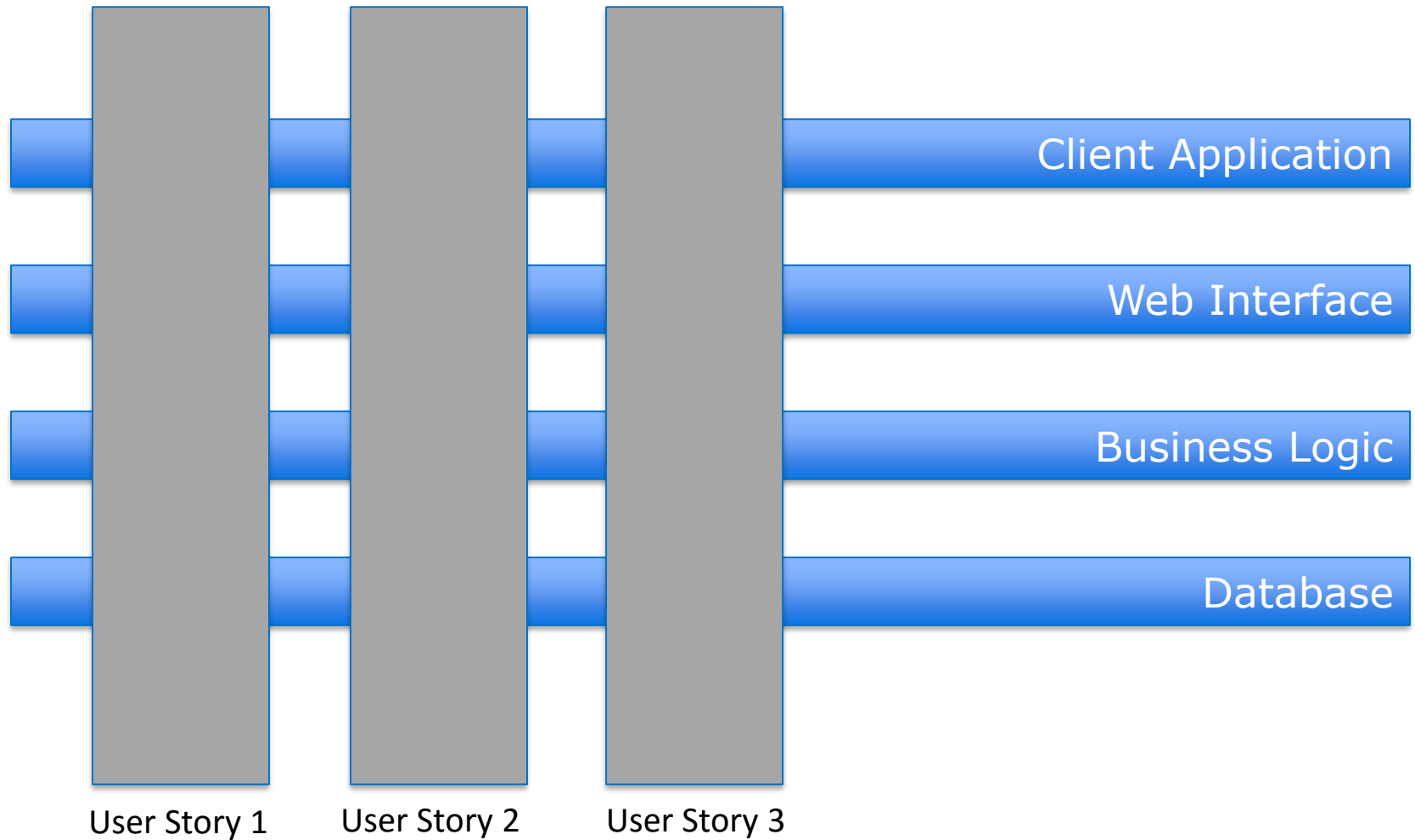
- If the syntax doesn't fit any of the above mentioned format, write the backlog with syntax, used in Feature Driven Development

*<action> <result><object>*

Example 3: *Display mail icon on the queue*  
*Add delete button on each queue*



# Work Breakdown Method – Vertical Slicing





# Criteria for good user stories: INVEST

## *Independent*

User stories should be independent of each other during their execution. Dependencies lead to problems for estimating and prioritizing.

## *Negotiable*

Stories are not contracts; they leave or imply some flexibility. Stories are the negotiations units in Scrum and are agreed to in planning and are delivered.

## *Valuable*

Stories are, by definition, units of value that are requested by the users and customers.

## *Estimable*

Team should be able to agree to the user stories for the correct estimates. If the team cannot estimate the stories, then the stories are ambiguous

## *Sized Appropriately*

Stories should be small enough to be completed in one sprint (Iteration). It is better to have a single focus per user story.

## *Testable*

If the requirements are not testable, requirements are vague. Likewise, each story needs to be verifiable, so that the team can determine when it is done.

Source: 3Back.com

**“Done” or “Not Done”, no “partially finished” or “done except”**



# Estimates of Work through User Stories

---

- For estimation, Story Points for user stories are widely used; T-Shirt size is also used by some scrum teams (Small, Medium, Large, X-Large), calendar days, and weeks
- Story points indicate the size and complexity of the story relative to other stories.
  - A story with two story points is expected to take twice as long as a story with one point
- Story Point determined by the team
  - No averages!
  - Make sure everyone is involved, team consensus
- Select a pointing system, for example
  - Fibonacci sequence: 1,2,3,5,8,13, 21+
  - Utilize a process such ‘planning poker’
- Set smallest story to a “1” and largest story to your highest number to establish markers



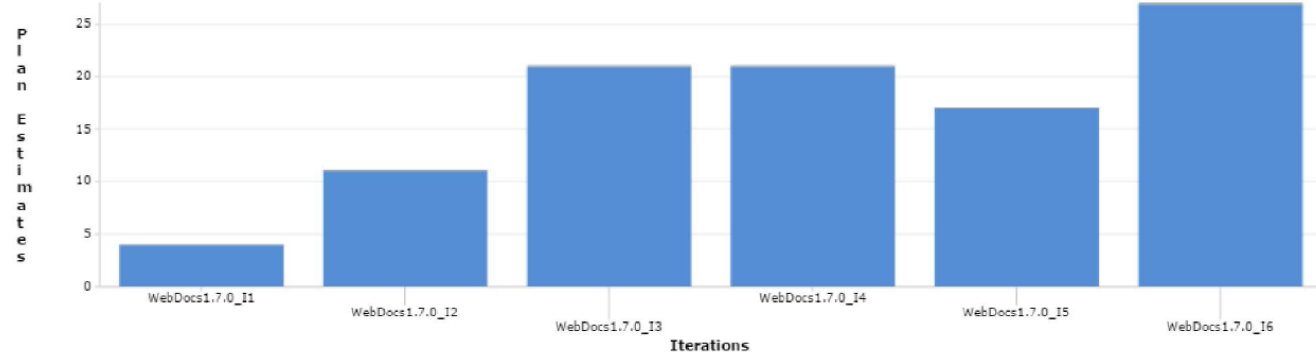
# Definition - Velocity

- Velocity is how much product backlog effort a team can handle in one sprint. This can be estimated by reviewing previous sprints, assuming the team composition and sprint duration are kept constant. It can also be established on a sprint-by-sprint basis, using commitment-based planning.
- Once established, velocity can be used to plan projects and forecast release and product completion dates.

**Ideal Velocity: 15.4 Points**

**Actual Velocity: 27 Points**

Average velocity (last 20 iterations): 16.83 Points





# Sprint Backlog

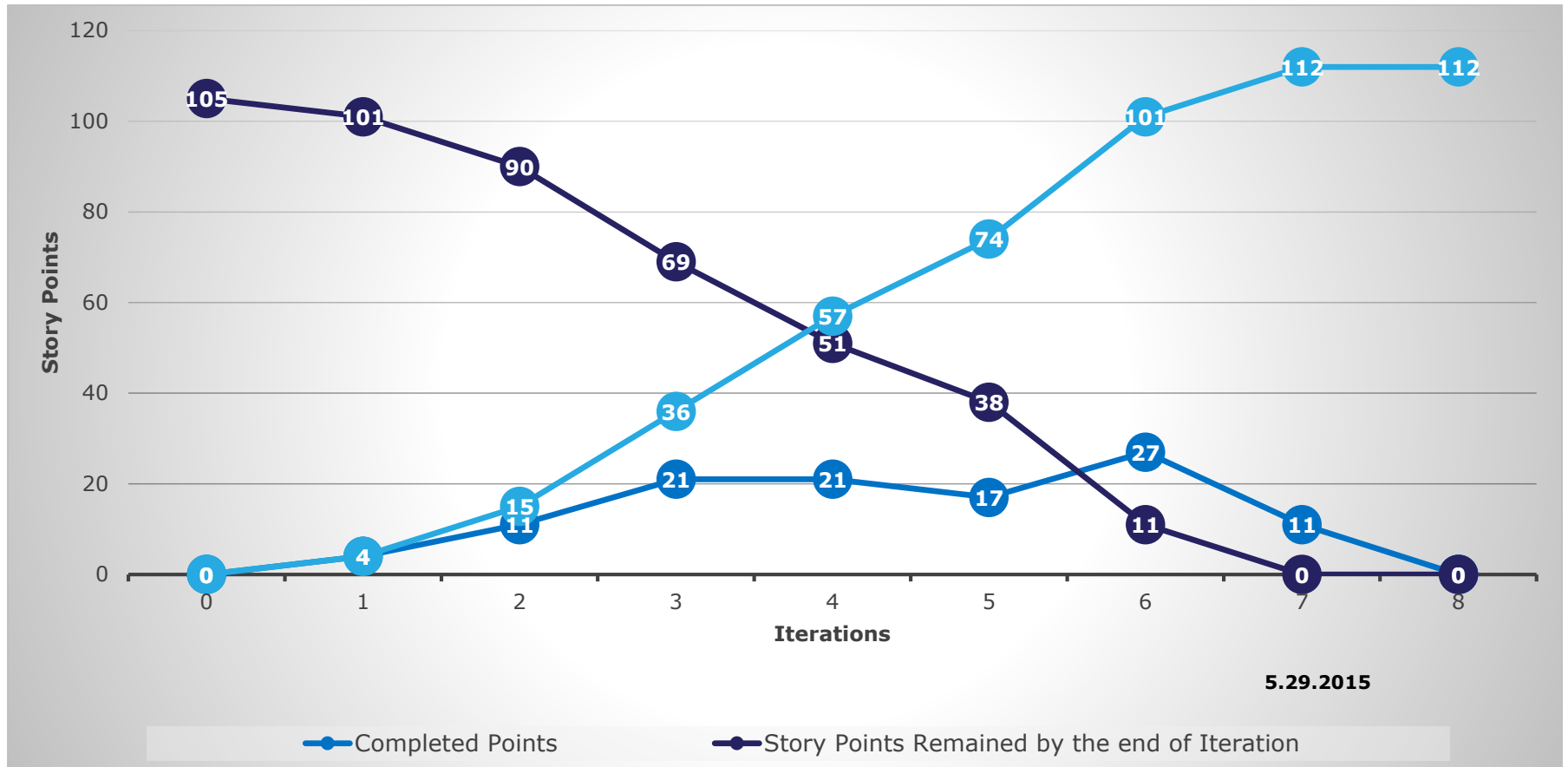
---

- Collect the agreed upon Sprint tasks in a Sprint Backlog
- Similar structure to the Product Backlog
- Team members sign up for tasks, they aren't assigned by the Scrum Master
- Estimated work is entered in the Sprint Backlog
- Any team member can add, delete or change tasks on the Sprint Backlog during the Sprint
- Tasks for the Sprint emerge during the planning session and during the Sprint
- Tasks (content, estimates, sign-up) can change during the Sprint





# An Sprint Burn down chart

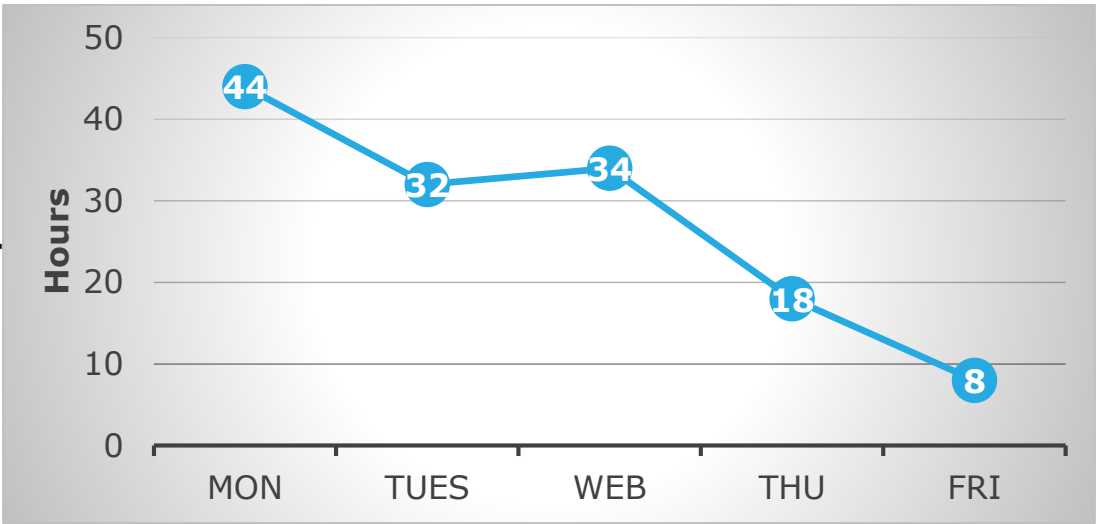




# Burn down chart continues...

Tasks	Mon	Tues	Web	Thu	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	7	
Test the middle tier	8	16	16	11	8
Write online help	12				
<b>TOTAL HOURS</b>	<b>44</b>	<b>32</b>	<b>34</b>	<b>18</b>	<b>8</b>

burndown char



# Agile-Scrum process at HelpSystems



# Product Releases – Feature vs. Maintenance

---

- R&D Group of HelpSystems drives two kinds of product releases
  - Feature Release (With new feature and/or existing feature enhancements)
  - Maintenance Release (Product defect fixes)



# Product Feature Release Process

1. Release Planning
2. User Story planning and estimation
3. Release kickoff
4. Agile-Scrum Process
  - a) 2-weeks long Development and Testing Iterations
  - b) Daily Stand up
  - c) Iteration Wrap ups
  - d) Product Demo during Iteration Wrap ups
  - e) Iteration Retrospectives
5. Product Beta Release
6. Regression Testing





# Product Maintenance Release Process

---

1. Release Planning
2. Detailed estimations on defects in the release scope
3. Agile-Scrum Process
  - a) Daily Stand up
4. Product Beta Release
5. Regression Testing





# Few last thoughts

---

- Agile transformation does not happen overnight
- Need to have
  - Goal for the transformation
  - Organization leadership buy in
  - Established minimal set of best practices
  - Management of expectation



# References and Resources:

- Scrum
  - [www.mountangoatsoftware.com](http://www.mountangoatsoftware.com)
  - [www.controlchaos.com](http://www.controlchaos.com)
  - [www.scrumalliance.org](http://www.scrumalliance.org)
  - [www.3back.com](http://www.3back.com)
  - [www.collaborive](http://www.collaborive.com)
  - Agile Software Development with Scrum
    - Ken Schwaber and Mike Beedle
  - (GVU) Agile Project Management with Scrum
    - Ken Schwaber and Mike Beedle
  - General information
    - [www.agilealliance.com](http://www.agilealliance.com)
- [www.agilemodeling.com](http://www.agilemodeling.com)
- [www.ambysoft.com](http://www.ambysoft.com)
- [www.agiledata.org](http://www.agiledata.org)
- [www.enterpriseunifiedprocess.com](http://www.enterpriseunifiedprocess.com)
- [www.implementingscrum.com](http://www.implementingscrum.com)





Thank You