Agile Estimation & Planning

Manjit Singh
Agile Coach

20 Years of software development, management & delivery experience

14 Years of agile experience; started using XP in 2000 @ IBM

Agile Coach & Trainer

Manjit

Founder & Organizer DC Agile User Group

Consulted, trained, or coached with people and teams from IBM, SRA, UMUC, NSF, DOJ, DOL, NSF

MS Computer Science @ SUNY Albany

Agenda

We Will Cover

- Scrum Process Overview
- Scrum Terminology
- User stories
- Relative Estimation
- “Scoping” (and Sizing)
- Planning

We Will **NOT** Cover

- Agile Mindset
- Agile Values & Principles
- Scrum In-Depth
- Scrum Process
- Scrum Ceremonies
- User Story Writing
- Acceptance Criteria
- Estimating User Stories

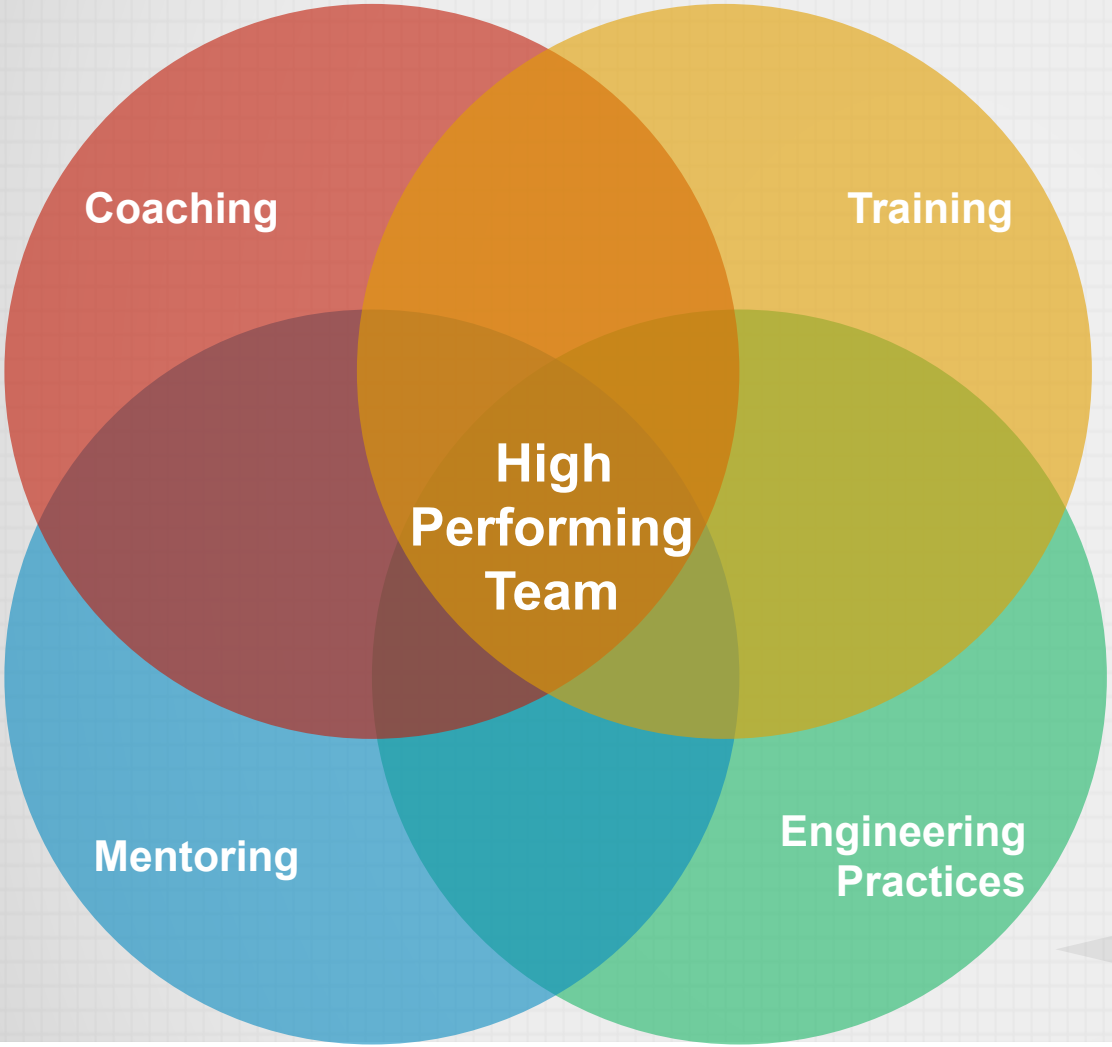
About Us

Accelerating Agility

We are a team of product craftsmen specializing in the use of Agile and Lean methodology to help companies create high performing teams and deliver high value products that maximize competitive advantage.

Agile Consulting

Accelerating Agility

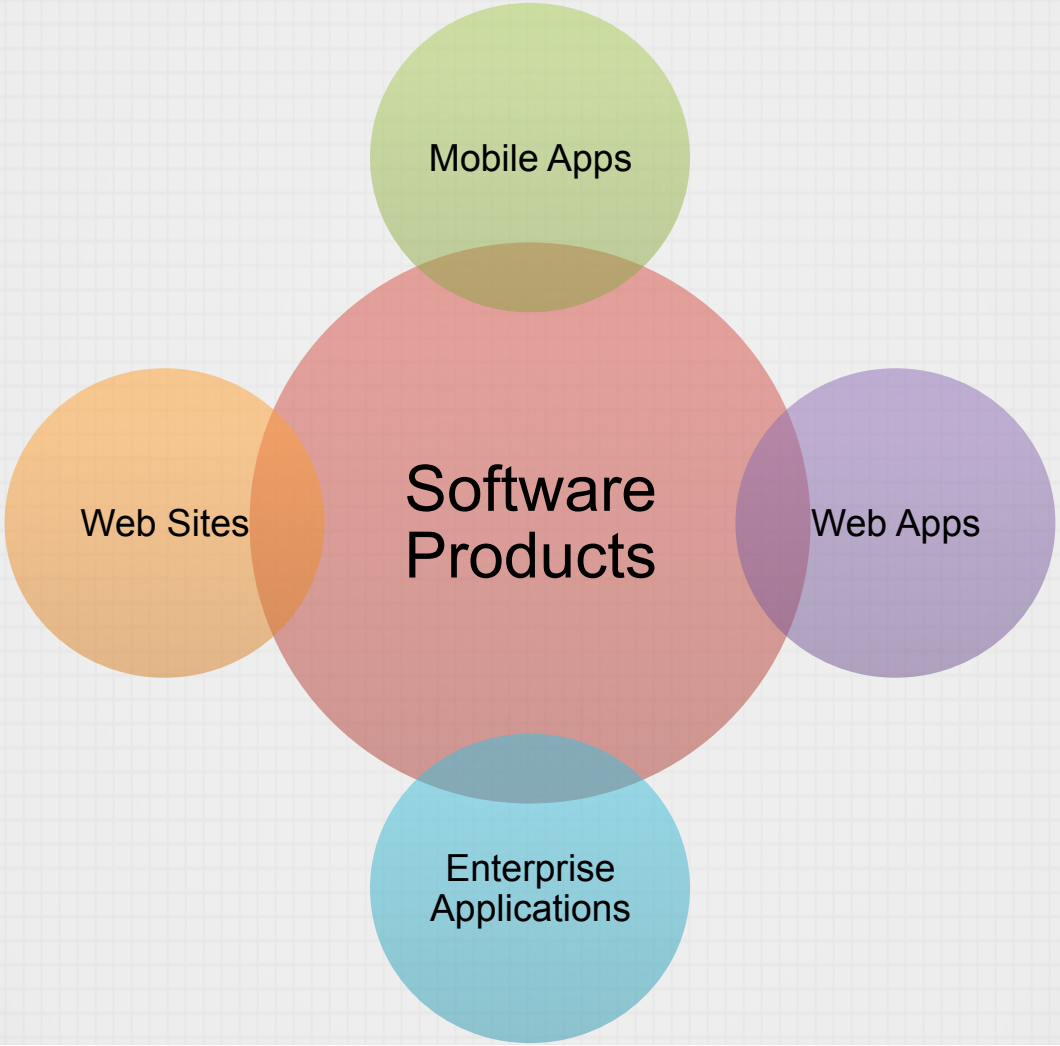


Agile Transformation

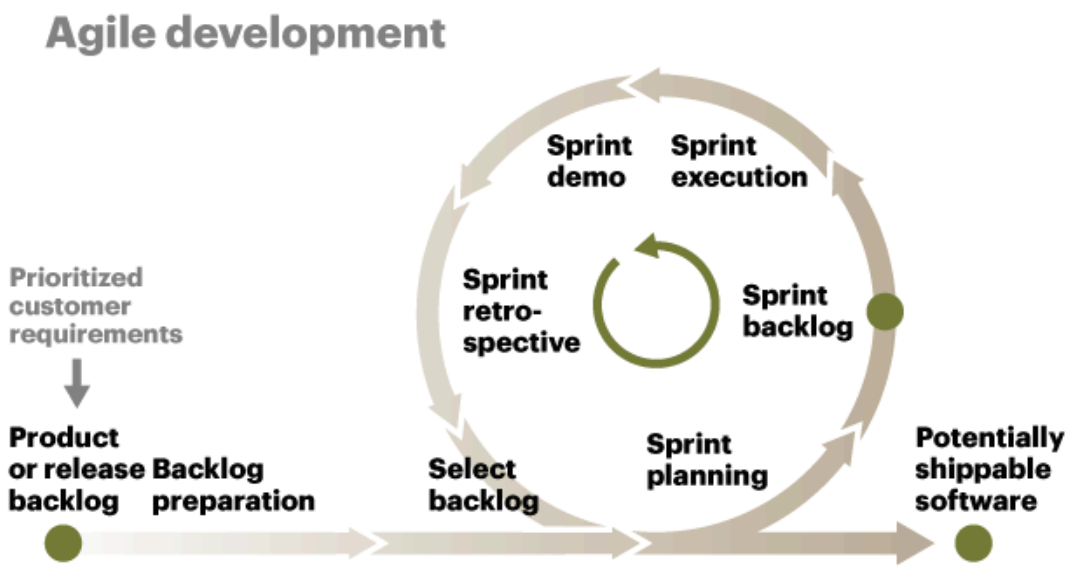
- Team and One-on-One Agile Coaching
- Customized Agile Methodology Training
- Product Owner/Scrum Master/Team Member Mentoring
- Practical, hands-on learning of Software Engineering Practices (TDD, ATDD, Continuous Integration, Pair Programming, etc.)

Agile Product Development

Accelerating Agility



Waterfall vs Agile

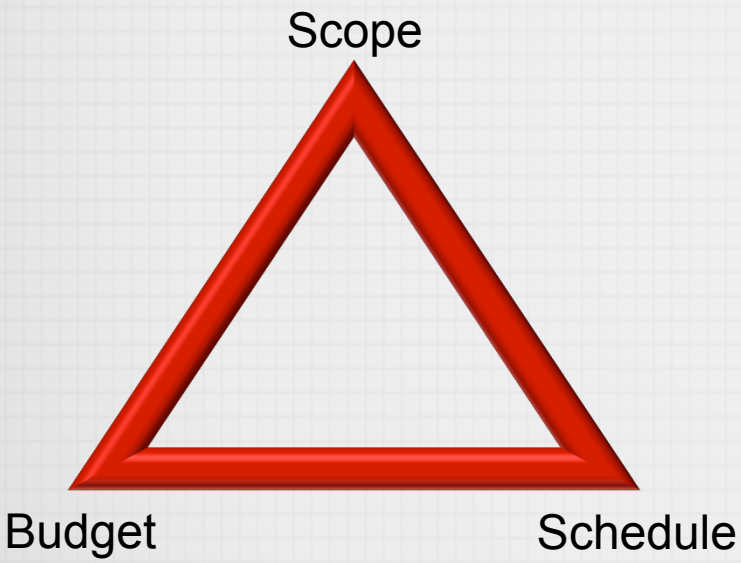


Planned and stage-gated development	⟷	Incremental and iterative development
Well-defined requirements	⟷	Customer-oriented backlog
Structured decision mandates	⟷	Empowered teams, end-to-end responsibility
Structured and planned delivery	⟷	Frequent delivery and feedback
Validation of complete product	⟷	Early validation through stakeholder participation
Focus on leveraging expertise	⟷	Focus on learning

Comparing Waterfall & Agile

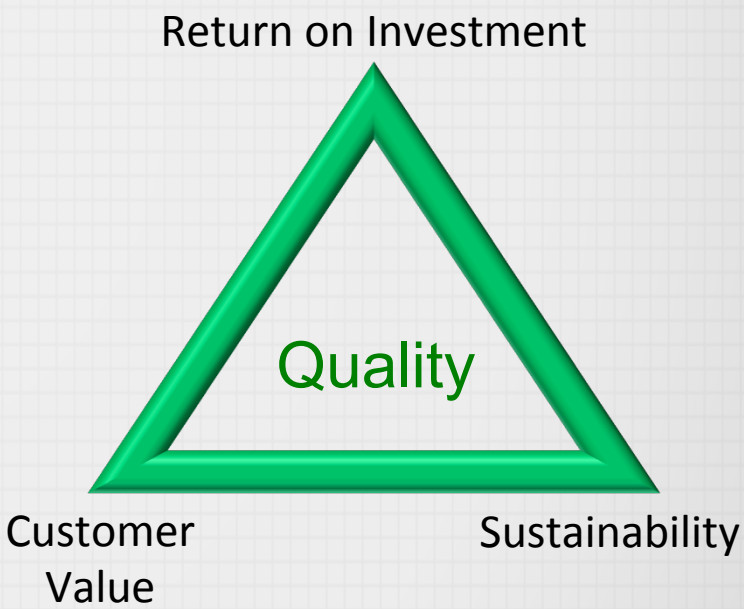
Old Approach

Create a project plan, then measure success by **adherence to plan.**

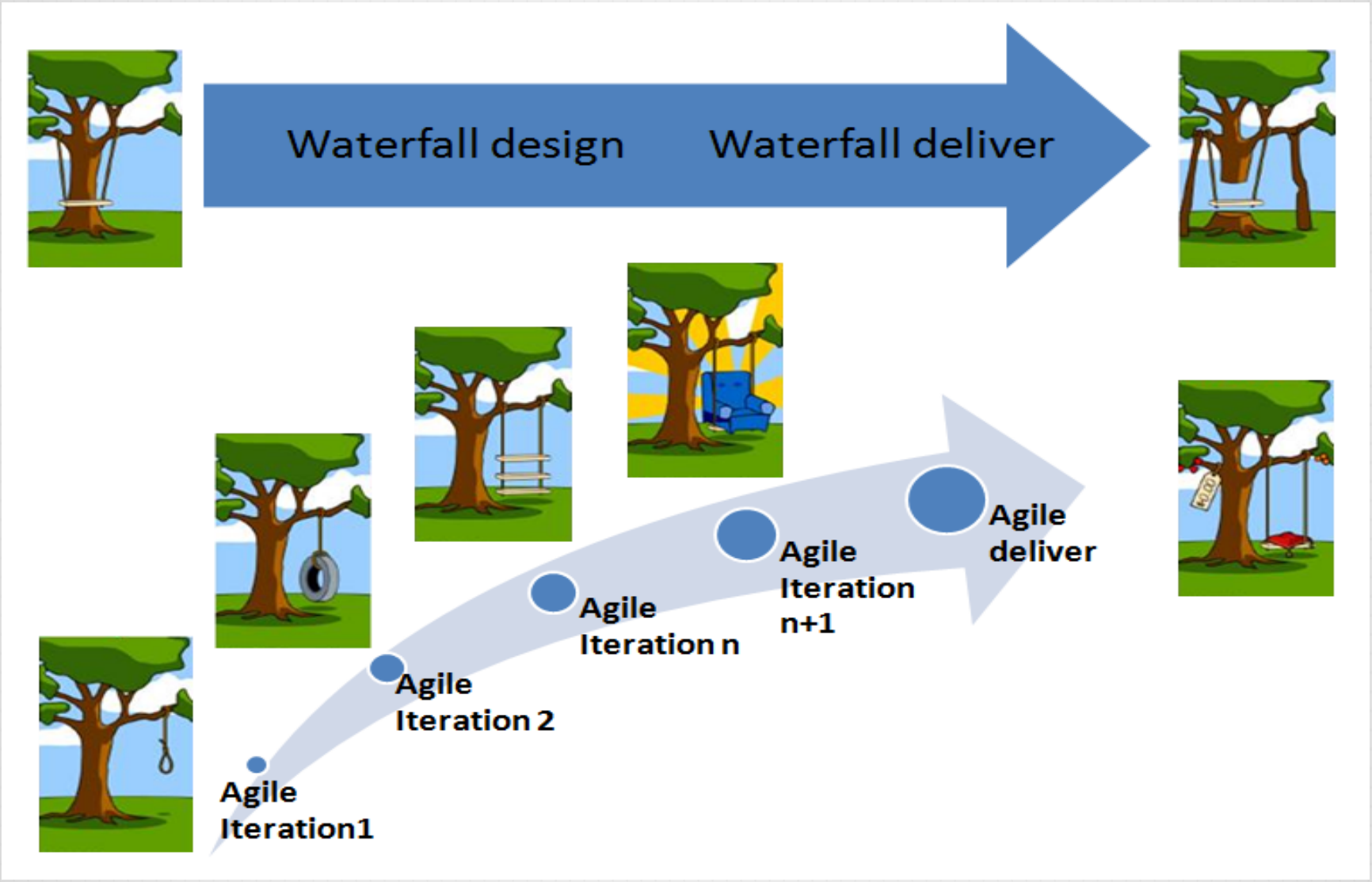


New Approach

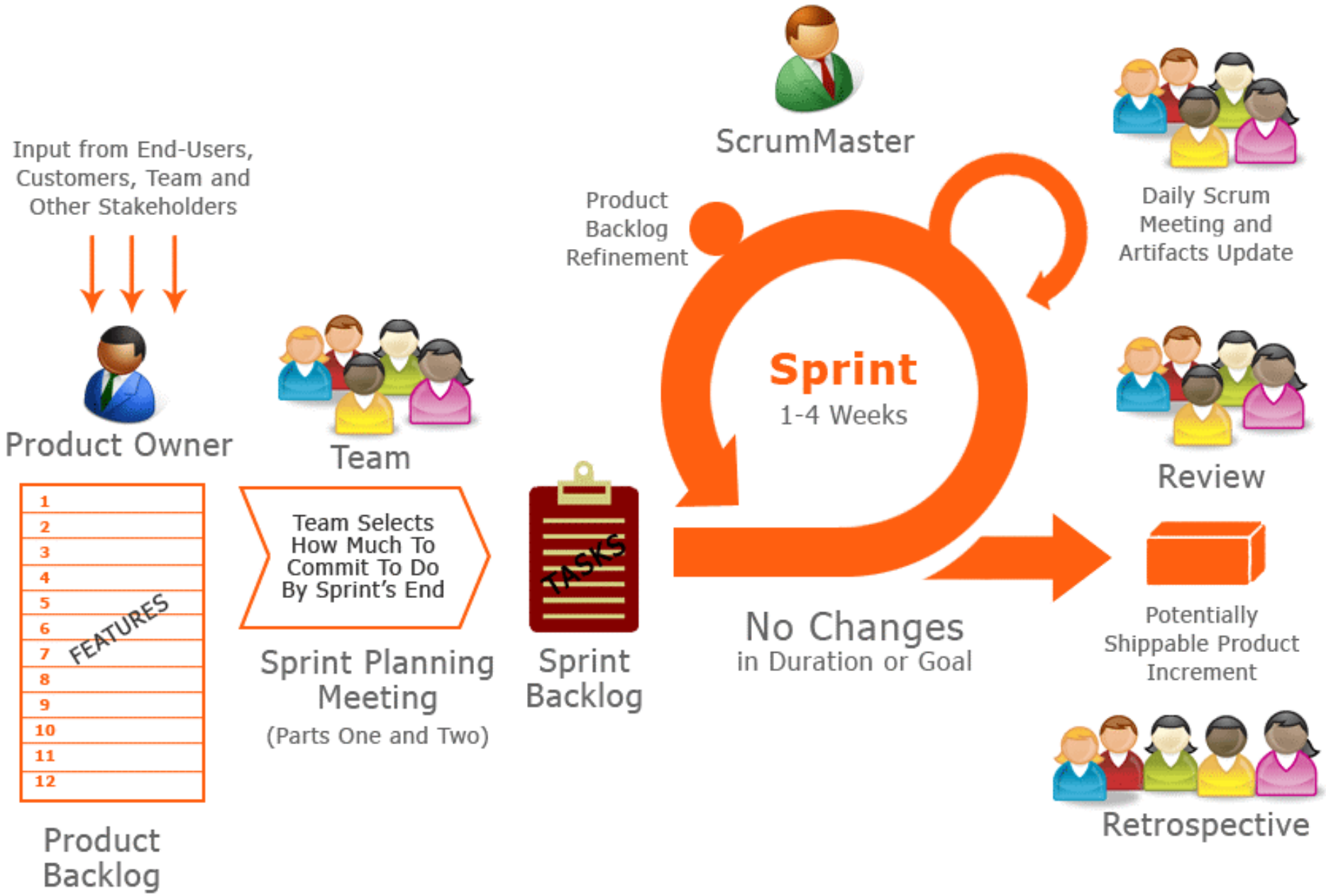
Maximize value and quality within specified project constraints.



The Solution



Scrum Process



Scrum Terminology

Epic

- A large, high level user story.
- Cannot be completed in a single Iteration (Sprint).

User Story

- A short, simple description of a feature, written from the perspective of a user.

Theme

- A set of related User Stories.

User Story Format:

As a <*user type/role*>,

I <*want/need/can*> **so that** <*benefit/value*>.

Epic and User Story Examples

Epic

Support during checkout.

User Stories

As a buyer, I want to pay for my order using a credit card, so that I can checkout quickly.

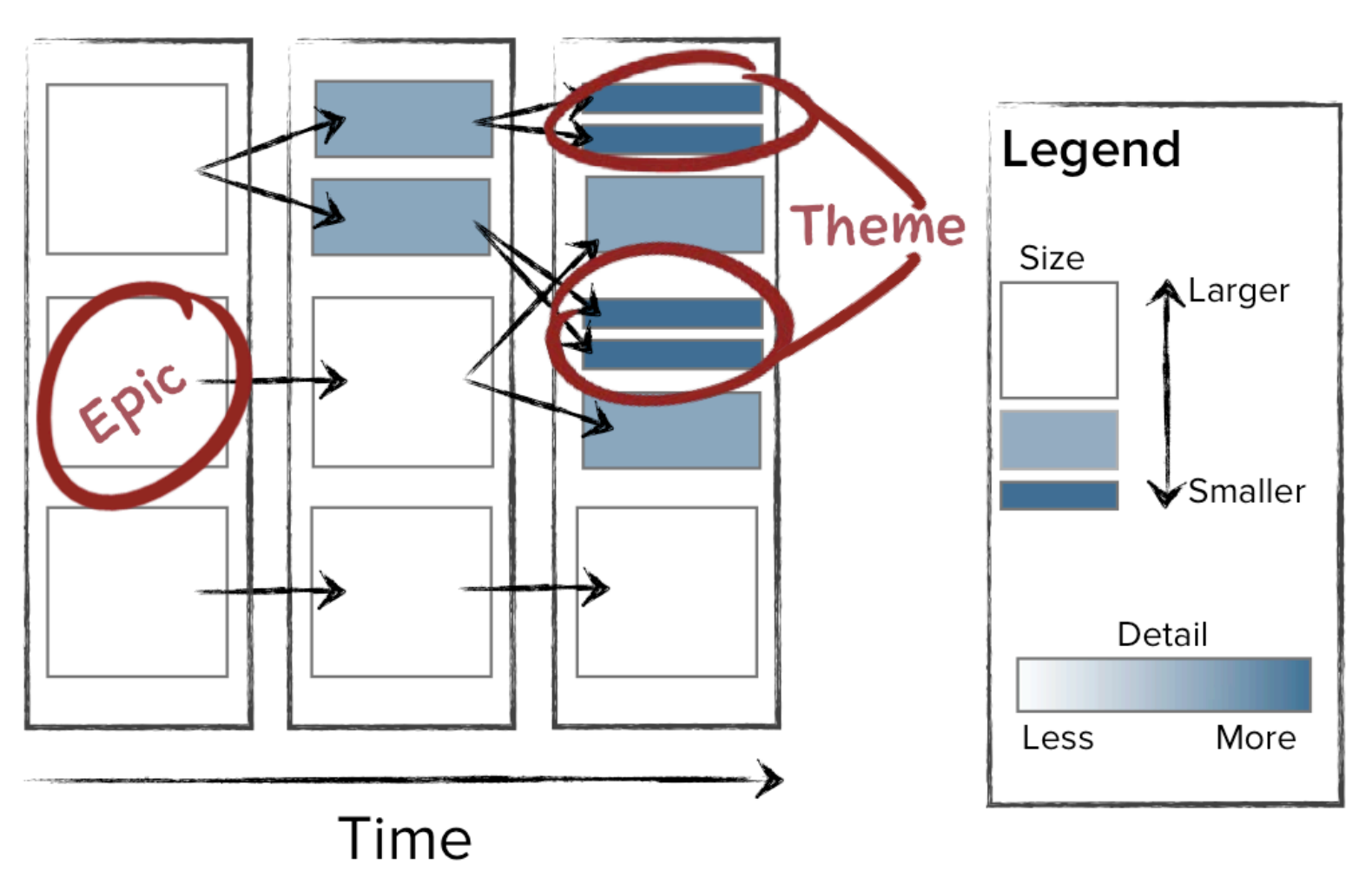
As a buyer, I want to save my credit online payment card info in my account, so that I don't have to reenter the card info each time I checkout.

As a buyer, I want to pay for my order using my AMEX Rewards Points, so that I can redeem my points.

As a buyer, I want to pay for my order using my AMEX card, so that I can get Rewards Points.

As a buyer, I want to pay for my order using Bitcoin, so that I can use my digital currency.

Refining the Product Backlog



Agile Estimation

AGILE SOFTWARE DEVELOPMENT

Relative Estimation

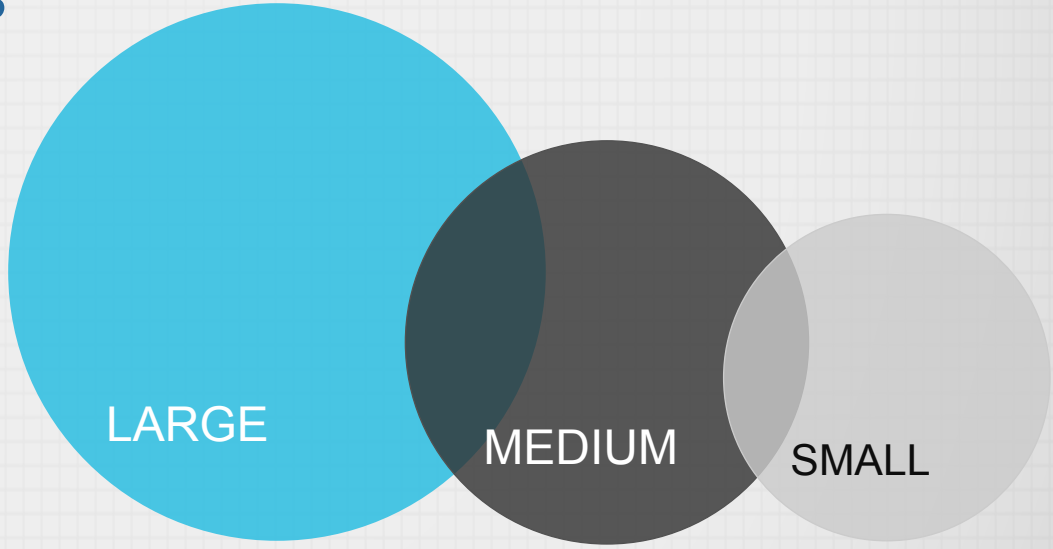
1. Based on principle that comparing is much quicker and more accurate than deconstructing
   - compare the relative effort of completing a new User Story (requirement) to the relative effort of a previously estimated (and perhaps even finished "DONE") User Story

2. Applied at a product backlog level rather than at a sprint backlog level.

Agile Estimation Basics

Estimates Size (Not Duration)

- Relative level of effort for reach User Story
- Represents complexity of work, not time to complete
- Uses relative units



Measures of Size

Traditional vs Agile Estimation

- Lines of Code
- Function Points

- Story Points
- Ideal Days

Agile Estimation Units

Two Popular Units of Estimation

- Story Points
  - Purely relative measure
  - Averages out across many stories
  - Don't "decay" over time
- Ideal Developer Days
  - About 4-5 hours, on average
  - Direct linkage to absolute time

Story Point Sequences

Story Point Estimation

Scales with increasing gaps between elements:

- Fibonacci Numbers: 1, 2, 3, 5, 8, 13
- T-Shirt Sizes: XS, S, M, L, XL, XXL
- Doubles: ½, 1, 2, 4, 8, 16

Avoid **false precision** to avoid false expectations.

When Estimating…

- Team decides scale
- Discuss and agree on sizing criteria
- Find the reference point

Story Point Estimation

Influenced by:

1. Complexity
2. Functionality
3. Current knowledge/uncertainty
4. Level of Effort (LoE) (volume of work)
5. Risk

Level of Effort Considerations

Story Point Estimation

1. Analysis
2. Design
3. Coding & Unit Testing
4. Data (database creation, revision, adding reference data, data clean up, data migration)
5. Functional/System Testing
6. User Acceptance Testing (UAT)
7. Documentation (design, online help, user guide, knowledge article, etc.)

Using Fibonacci Numbers

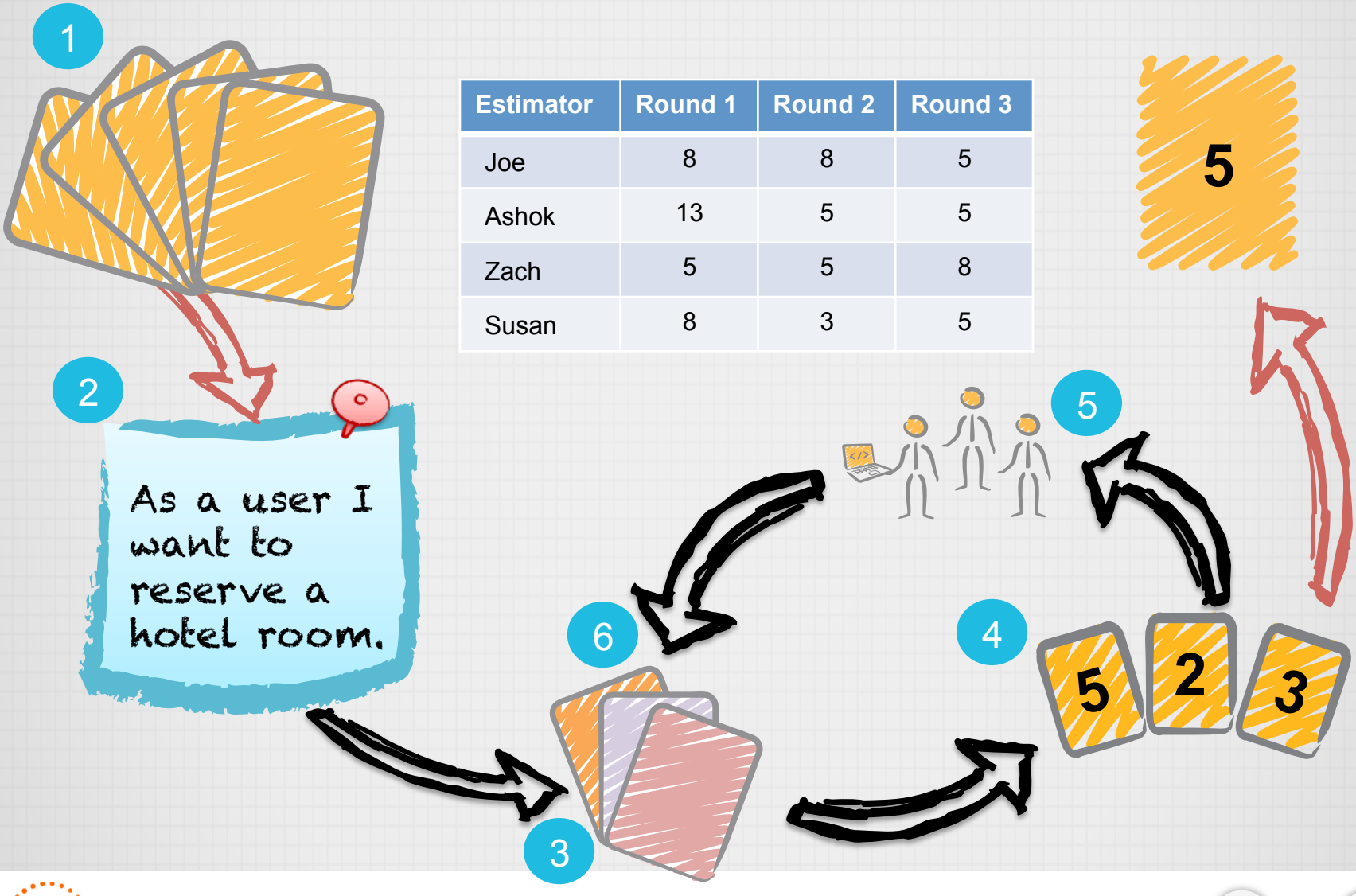
Story Point Estimation

- Make Reservation: 5
- Cancel Reservation: 3

Backlog Item	Story Points	Priority	Status
Allow a guest to make a reservation	5	P1	☑
As a guest, I want to cancel a reservation.	3	P1	☑
As a guest, I want to change the dates of a reservation.	5	P2	☑
Improve exception handling	8	P2	☒
TOTAL Sprint Velocity	**13**	**(Claim13 of 21)**	

Planning Poker Example

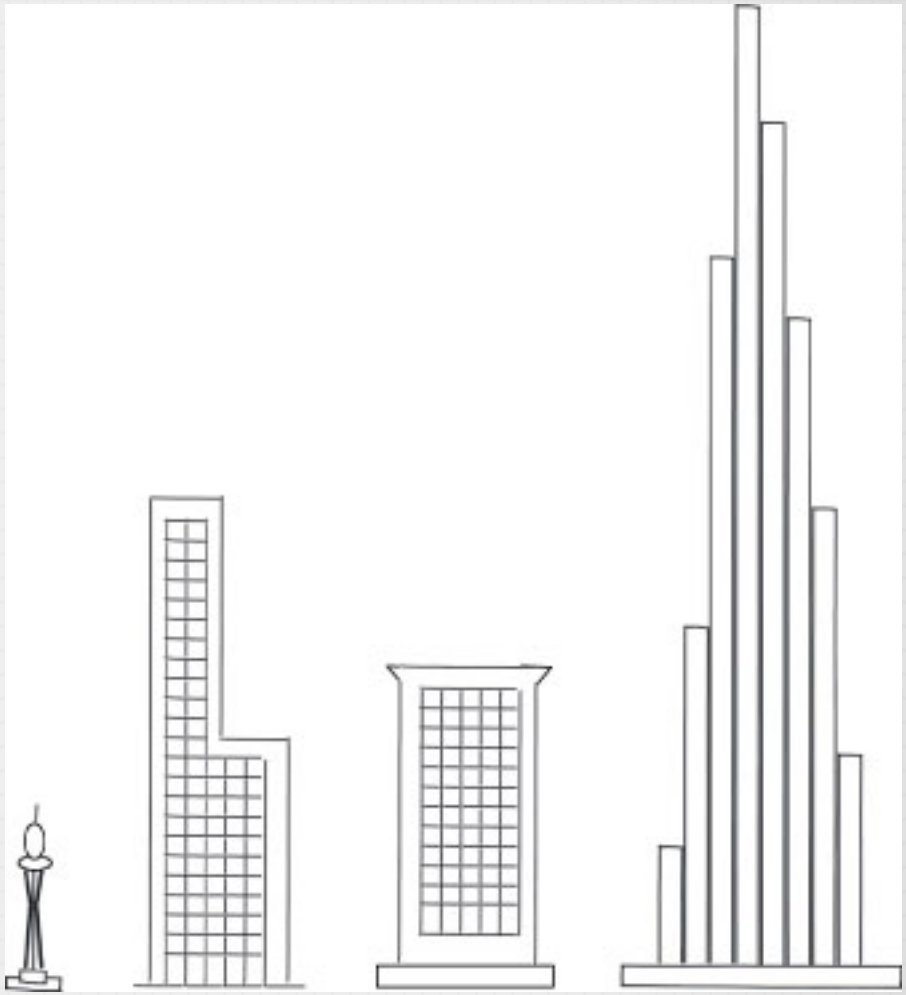
Estimator	Round 1	Round 2	Round 3
Joe	8	8	5
Ashok	13	5	5
Zach	5	5	8
Susan	8	3	5



Relative Estimation Analogy

How do we estimate how long it will take us to walk up all of these buildings using the stairs?

- How physically fit we are
- Obstacles in the stairwells?
- So, what do we do?



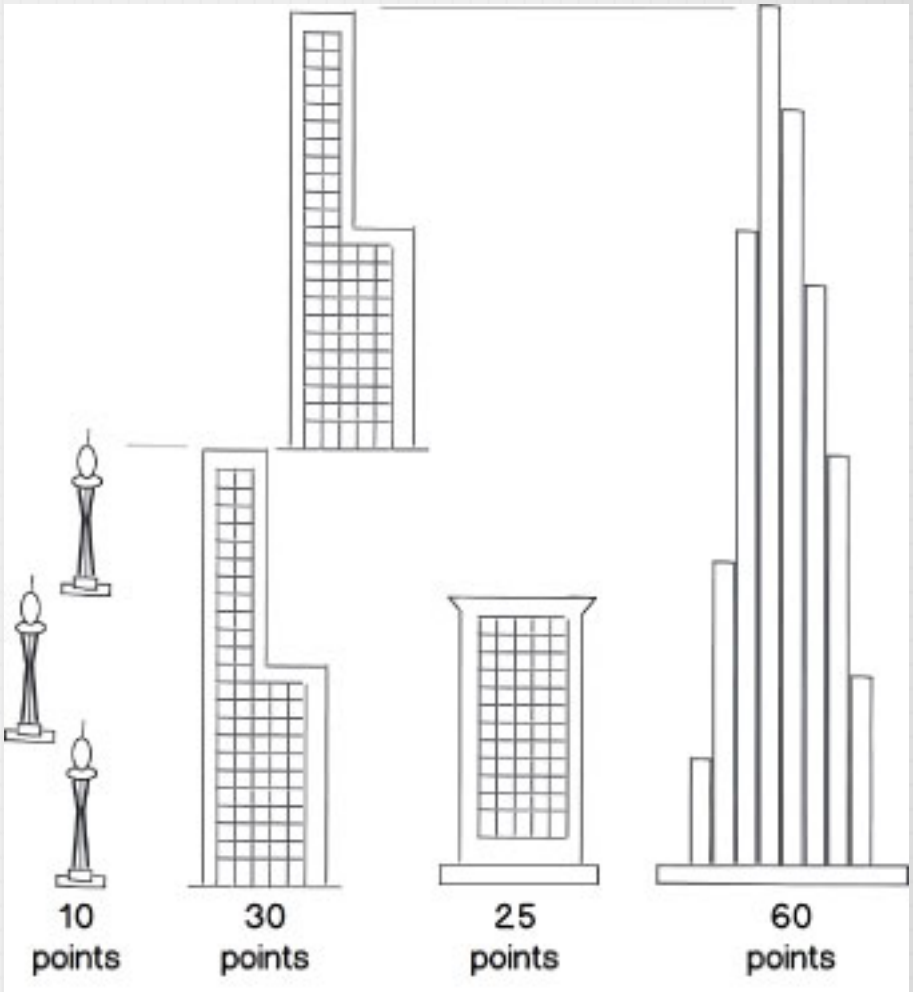
Courtesy: AxisAgile (http://www.axisagile.com.au/blog/estimation/relative-estimation-communication/)

Relative Estimation Analogy

First, we classify our buildings in relative terms by making some quick comparisons.

NOTE:

- **The points are simply relative markers to help us compare.**
- **The numbers do not relate to a specific unit of size or time.**

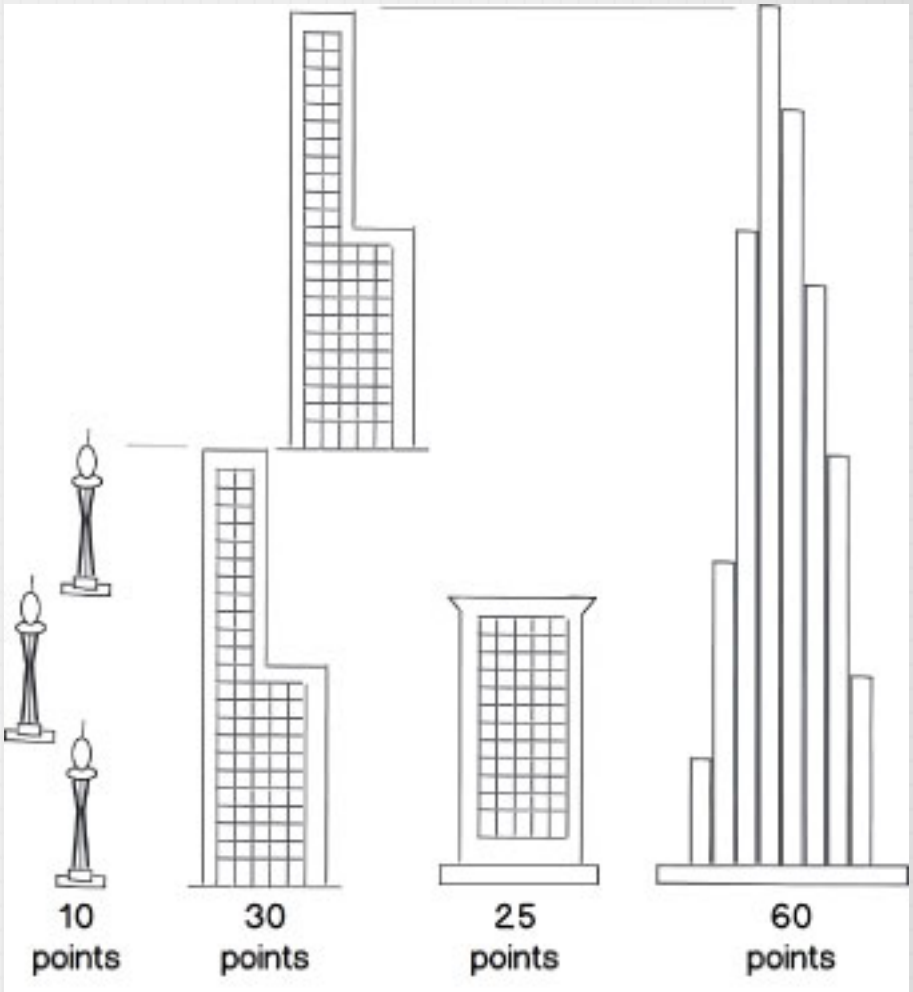


Courtesy: AxisAgile (http://www.axisagile.com.au/blog/estimation/relative-estimation-communication/)

Relative Estimation Analogy

What next?

1. Invest a little time to actually test our fitness and check the state of an indicative stairwell?
2. Time-box this experiment to 10 minutes.

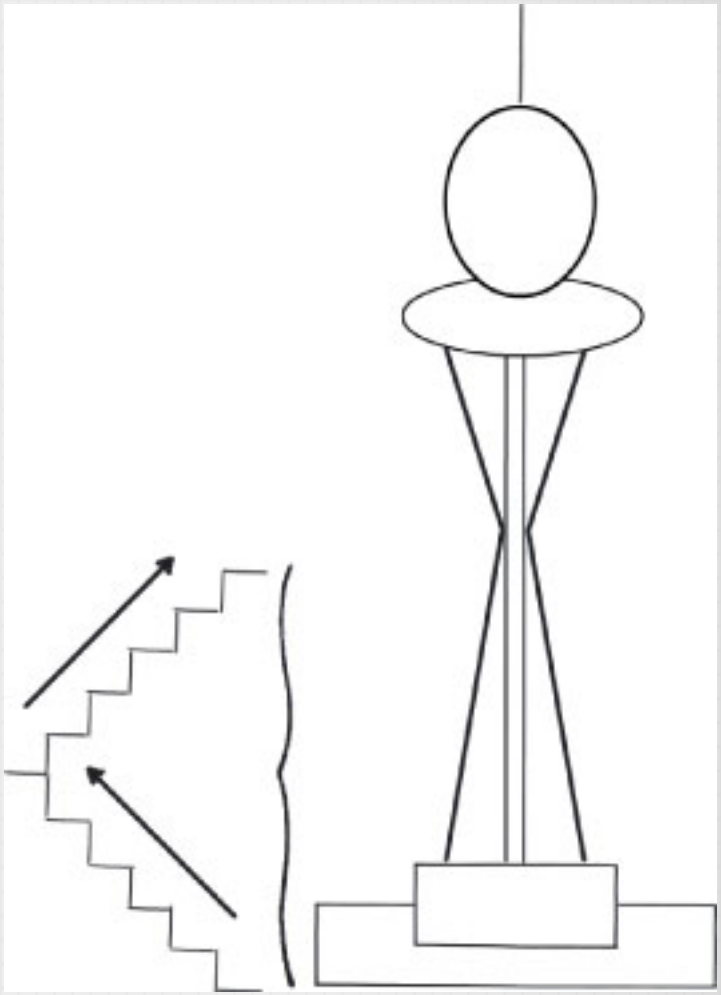


Courtesy: AxisAgile (http://www.axisagile.com.au/blog/estimation/relative-estimation-communication/)

Relative Estimation Analogy

After 10 minutes of actual stair climbing, we get halfway up the 10-point building.

- Velocity = 5 points/sprint



Courtesy: AxisAgile (http://www.axisagile.com.au/blog/estimation/relative-estimation-communication/)

Relative Estimation Analogy

But we need to know how long it will take us to reach the top of all four buildings!

- **Total Points:**

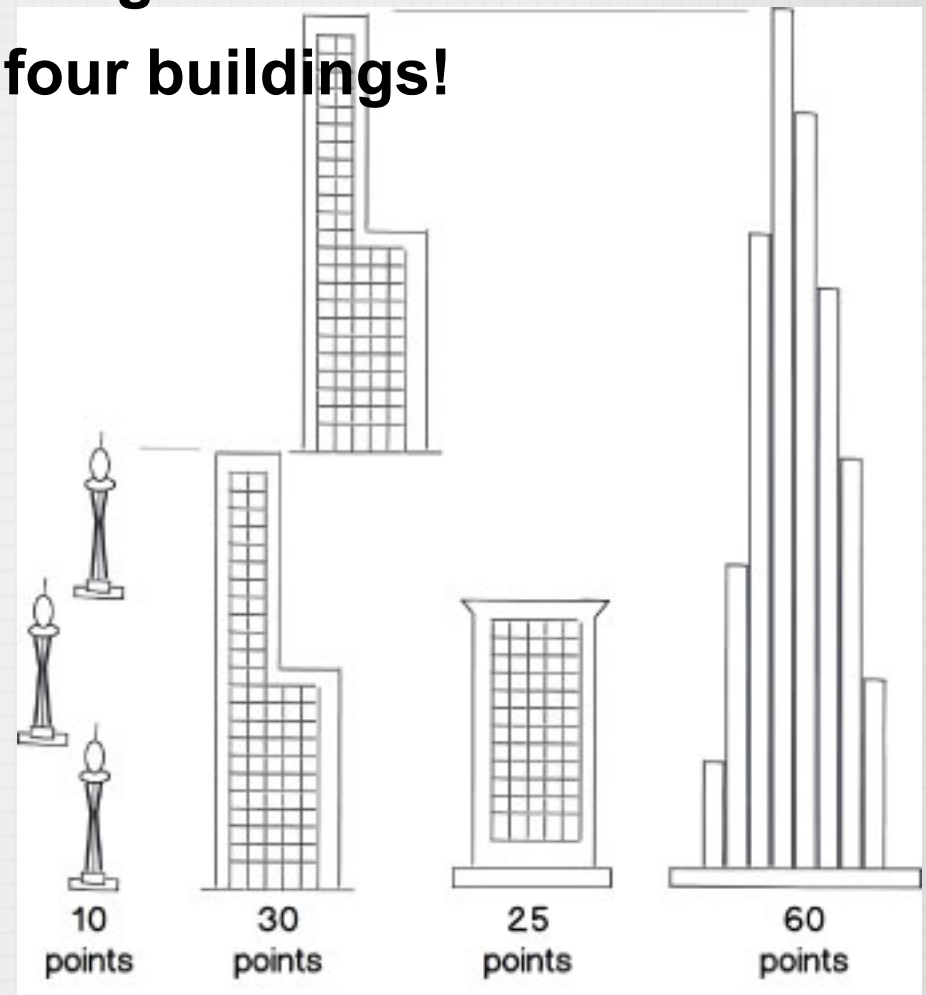
10 + 30 + 25 + 60 = 125 points

- **Total Sprints:** (total points/ velocity)

125 points/5 = 25 sprints

- **Total Duration:**

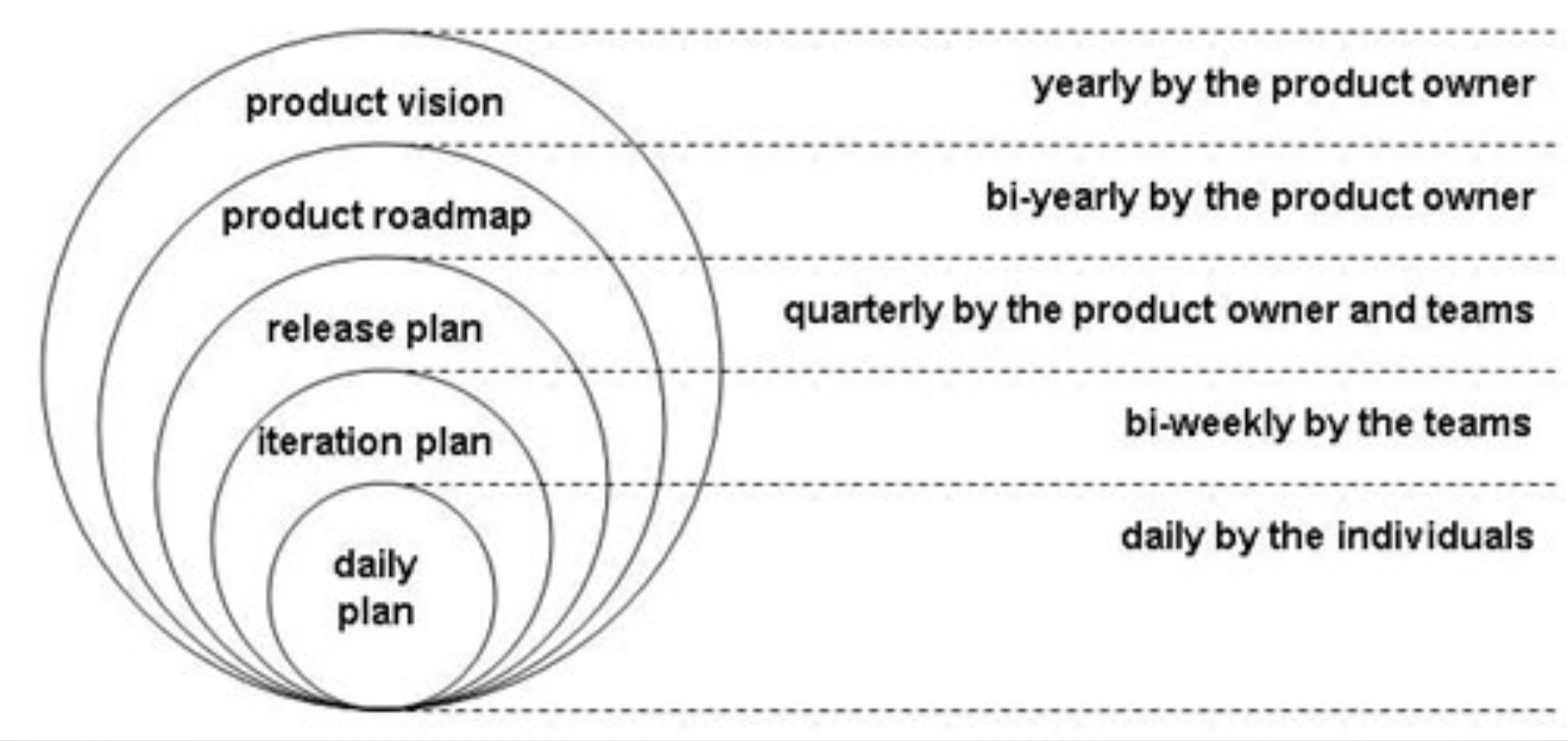
25 sprints x 10 mins = 250 mins



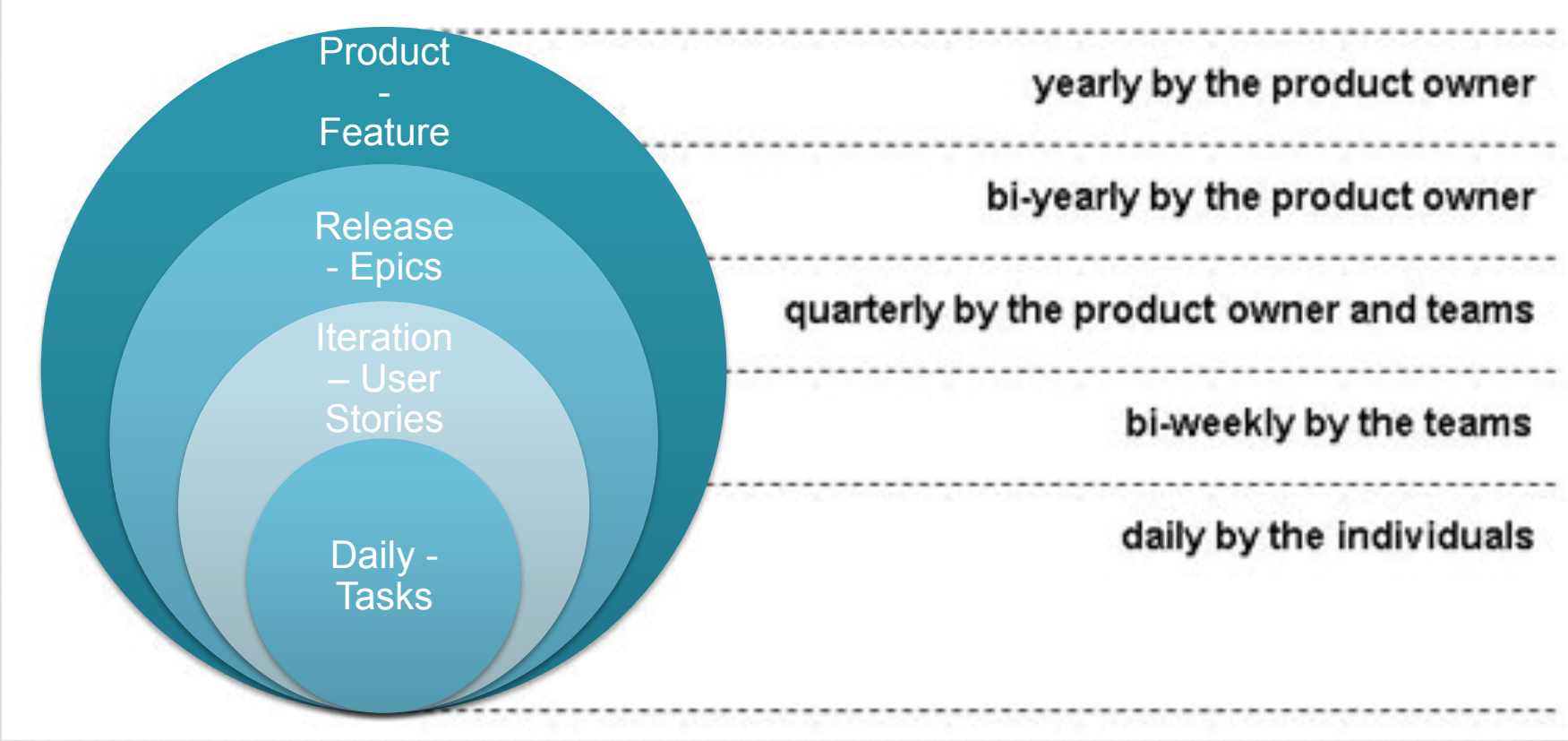
Courtesy: AxisAgile (http://www.axisagile.com.au/blog/estimation/relative-estimation-communication/)

Agile Planning

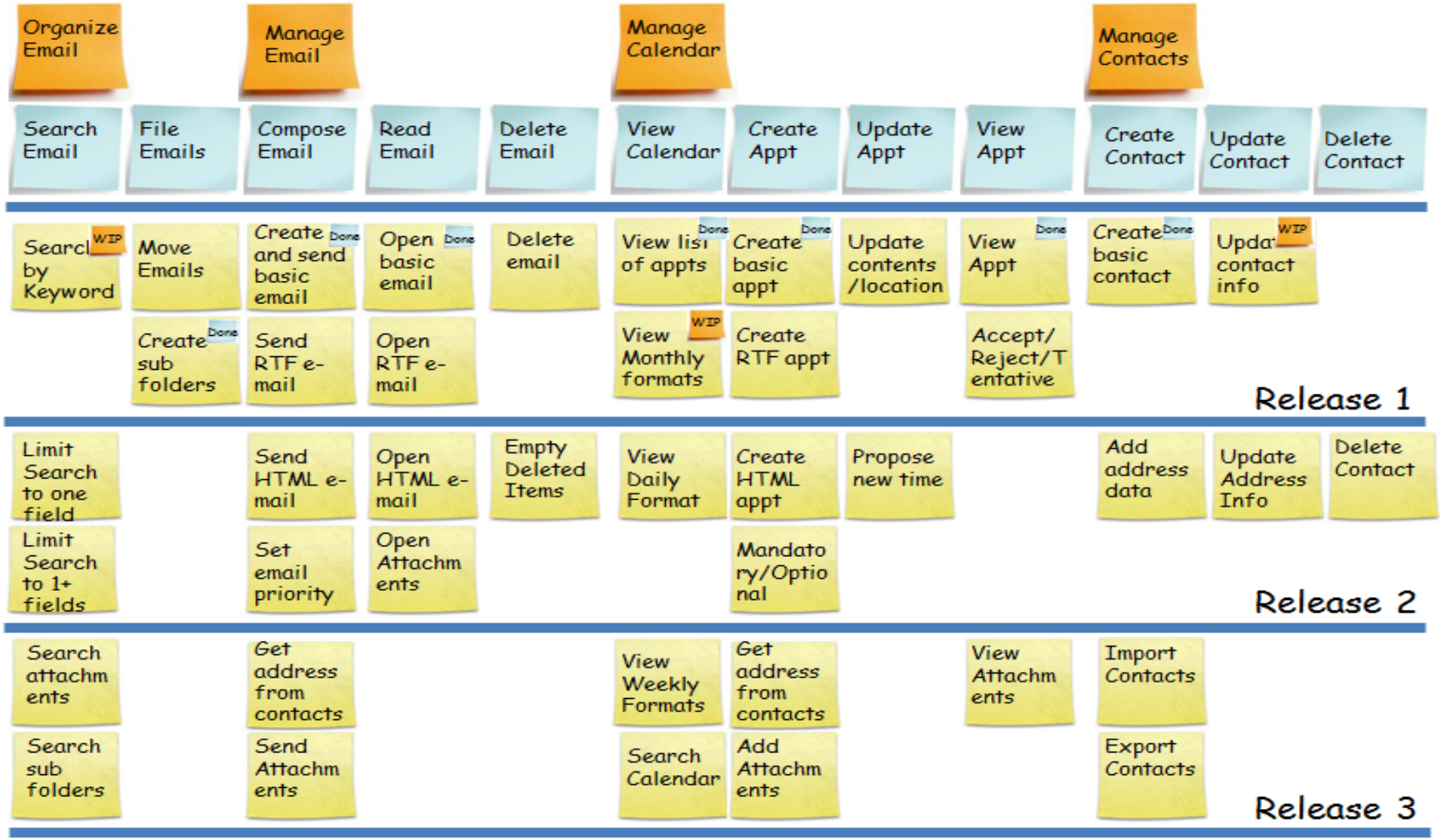
Levels of Agile Planning



Levels of Agile Planning



Release Planning using Story Map



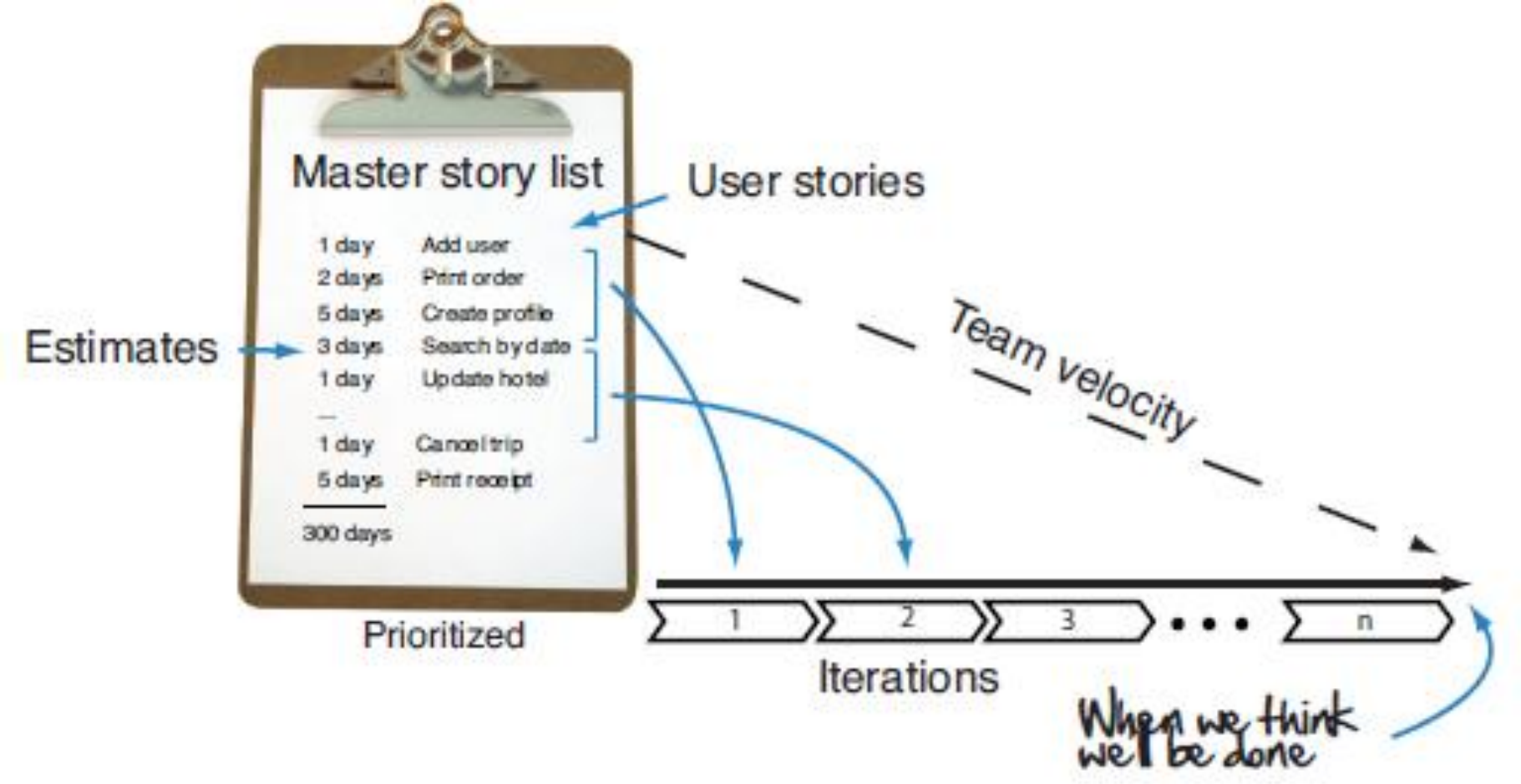
Courtesy: Winnipeg Agilist (http://winnipegagilist.blogspot.com/2012/03/how-to-create-user-story-map.html)

What Is Velocity?

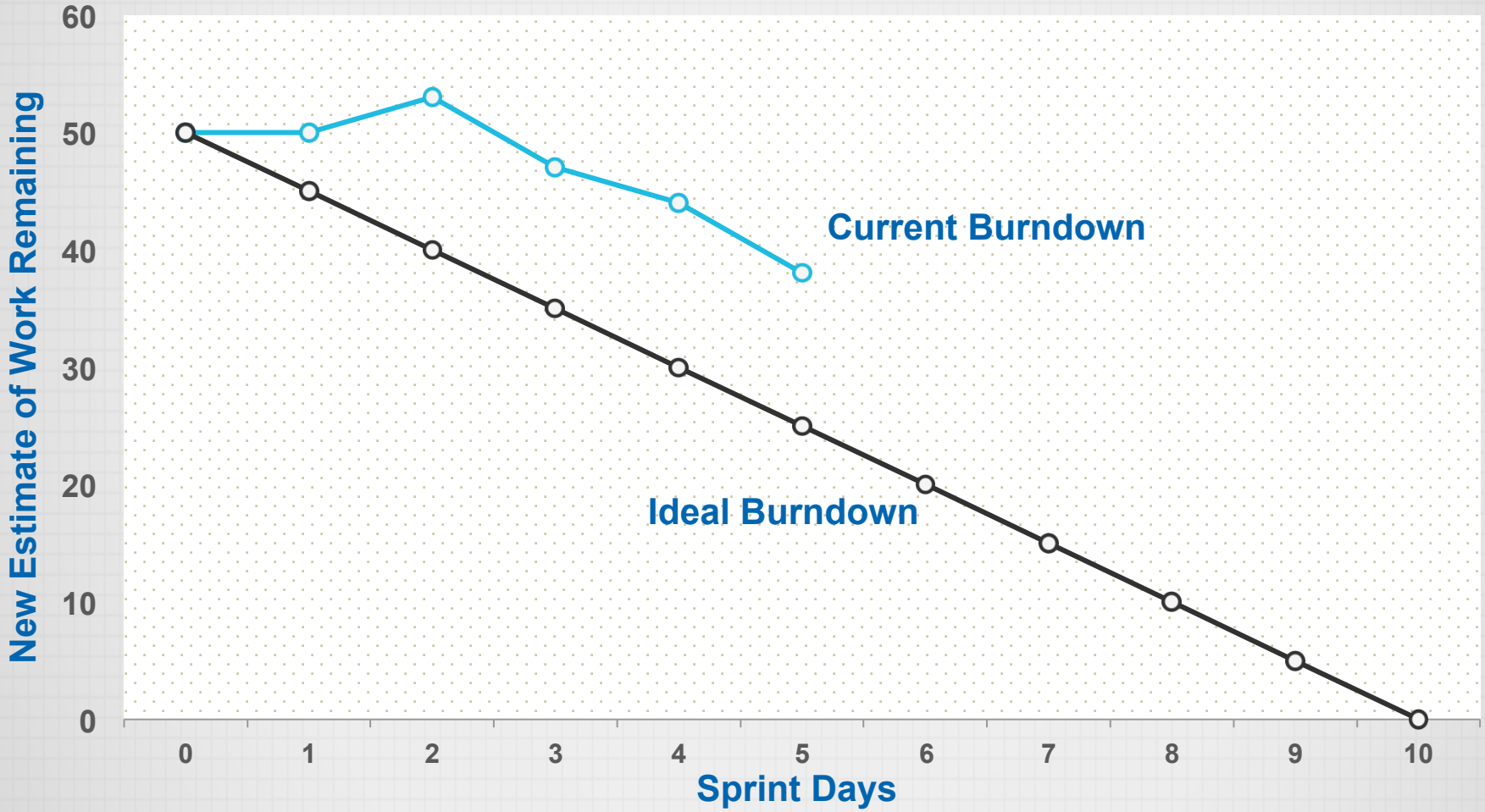
- Measures team capacity
- Takes into account:
  - Developer skill set
  - Domain knowledge
  - External interruptions
- Represents work actually completed

What is Velocity?

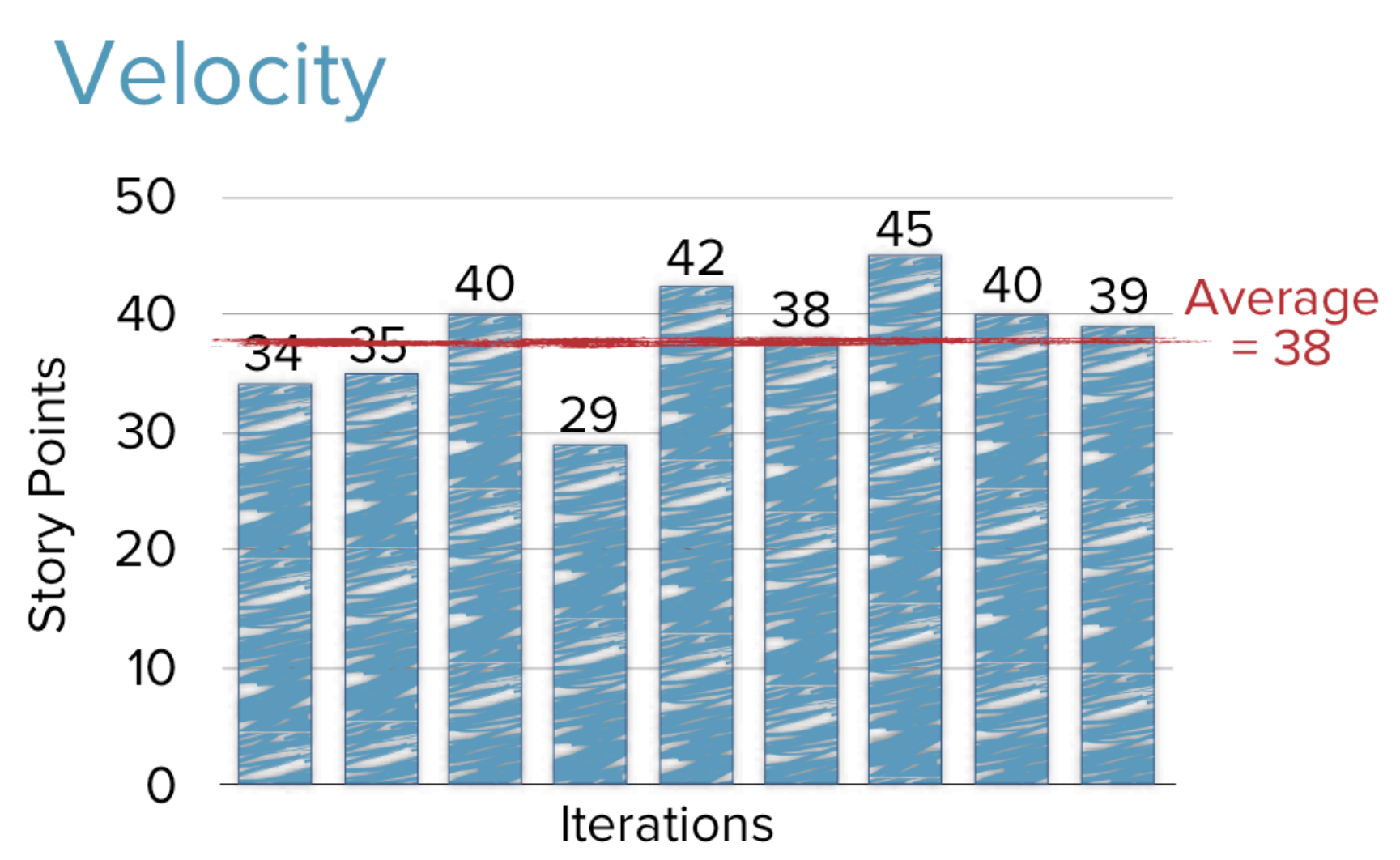
Velocity is the rate at which **scrum** development teams consistently deliver business value (i.e. convert product backlog items into completed functionality).



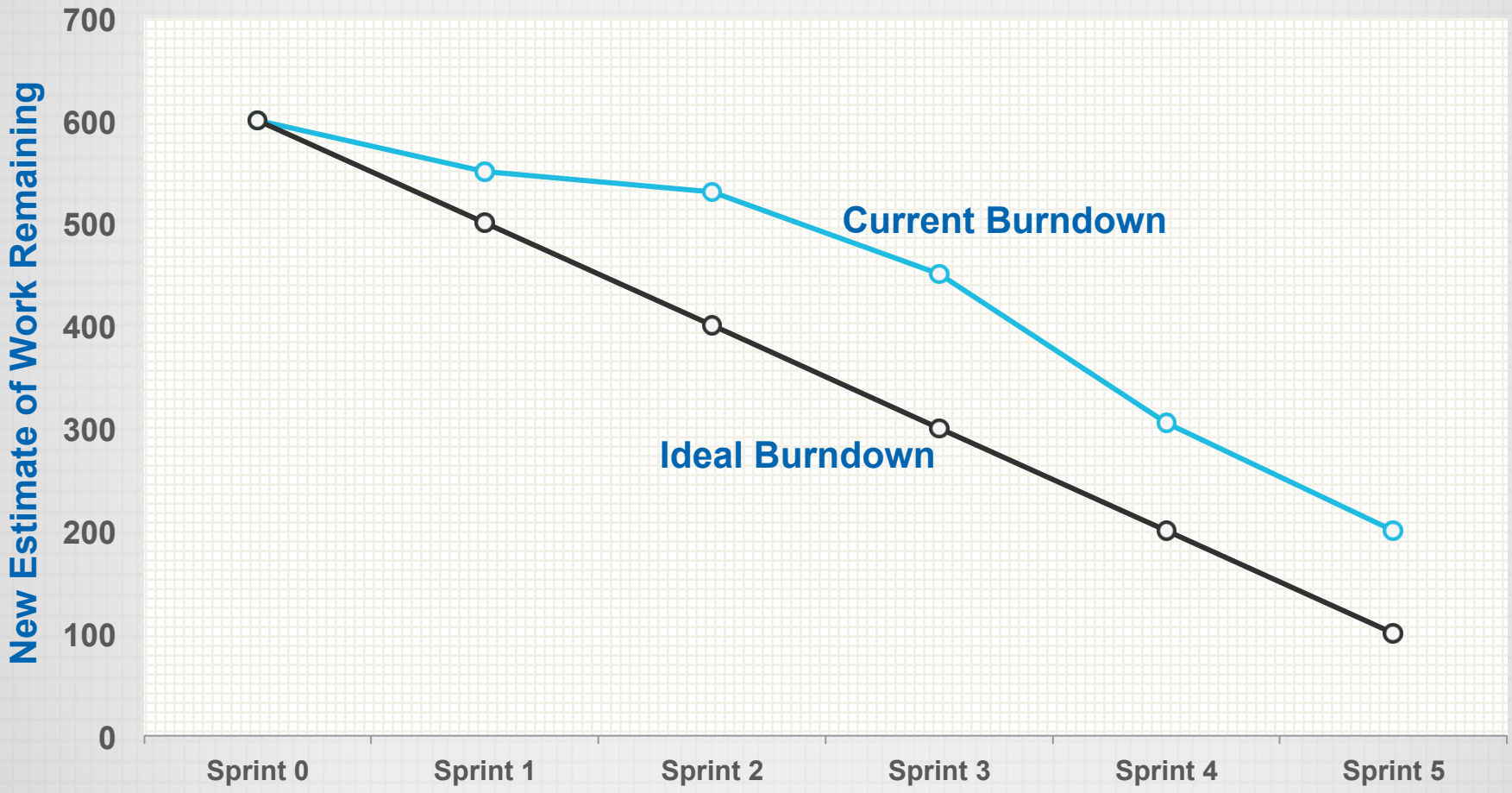
Sprint Burndown Example



Planning with Known Velocity



Release Burndown Example



Agile Estimating/Planning

1. **Date Driven** – (*Variable Scope*, **Fixed Time**)

   Product owner says "when" – lay out as many sprints as you can envision, working backwards to today

2. **Scope Driven** – (**Fixed Scope**, *Variable Time*)

   Guess how many sprints you will need to deliver the product as per the defined scope

3. **Budget Driven** – (*?? Scope*, **?? Time**)

   Estimate how many sprints (based on team size and sprint duration) until the money is gone

So How Do We Estimate Velocity?

Two Scenarios:

1. Team Already Assembled for Project
2. Don’t Have a Team Yet

Scenario 1: Team Already Assembled

Steps:

1. Estimate the required stories in story points the normal way.
2. Do Sprint Planning for two Sprints:
   a. Grab one story
   b. Split it into tasks
   c. Estimate the tasks in hours and ask the team if they can commit to finishing it
   d. Then repeat with additional user stories until the sprint is full

Scenario 1: Team Already Assembled

Steps (continued):

2. Do Sprint Planning for two Sprints:
   e. Add all the story points of all the stories in Sprint 1
   f. Add all the story points of all the stories in Sprint 2
   g. Use the two values as a range; say 18 to 24 points
   h. The two values now are your Best and Worst Velocity

      Forecast

   i. Now you can forecast the duration of the project i.e. between X – Y Sprints

Scenario 2: Don’t Have a Team Yet

You don't know who will do this project.

Steps:

1. Get a “representative team” together
2. Have the “representative team” do the steps in Scenario 1
3. Then you can **forecast**:

   "a project with three senior programmers, a senior tester, a junior tester, a good designer and a great database person can do this project in X - Y sprints."

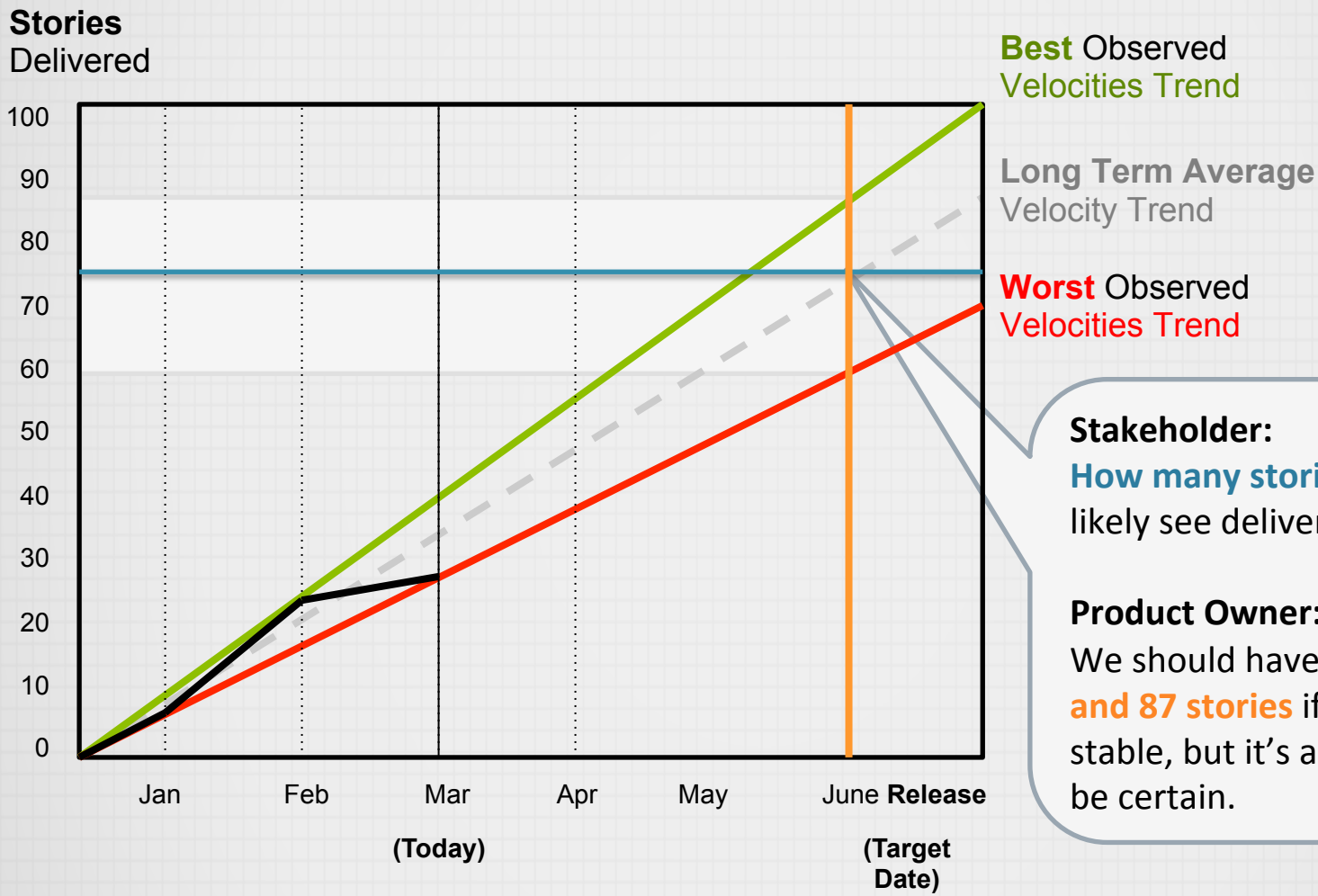
Budget Driven

- Sum the product backlog
- Estimate velocity as a range
- Divide the size of the product backlog by the velocity range

120÷20=

120÷15=

Date Driven – *Variable Scope*, Fixed Time



Stakeholder:
How many stories will we likely see delivered by **June?**

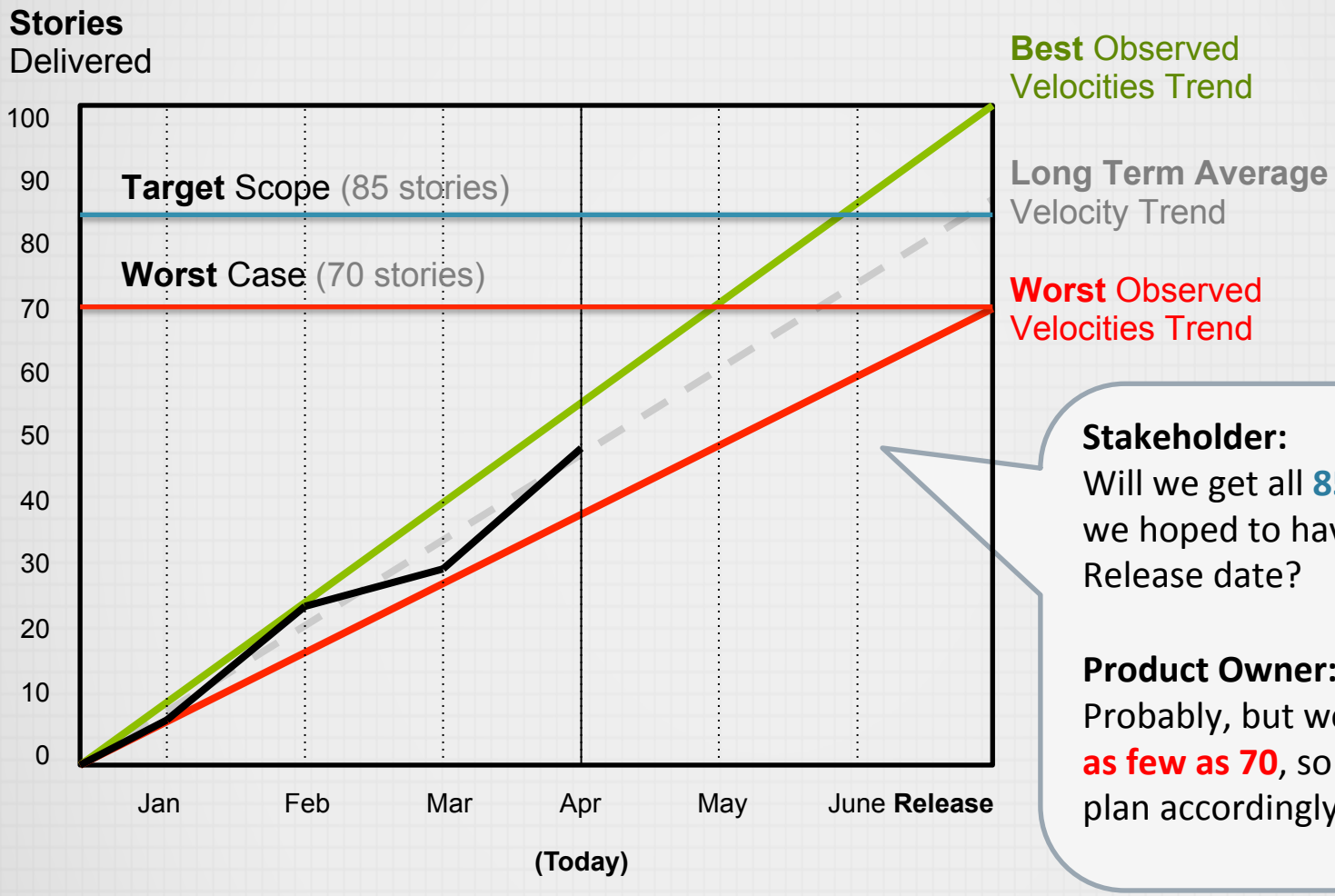
Product Owner:
We should have between **60 and 87 stories** if velocity stays stable, but it's a little early to be certain.

Date Driven – Fixed Time Projects

Project Schedule may look like this, based on estimated average (of best and worst case) velocity.

Sprint 0	Sprint 1	Sprint 2	Integration	Go Live
9/13 - 9/22	9/23 - 10/13	10/14 - 11/3	11/4 - 11/21	11/22
Everything in place to begin construction	*Code, test, and deliver first feature set, mimimum usable product*	*Code, test, and deliver secondary feature set*	*FINAL INTEGRATION AND SYSTEM TEST*	

Scope Driven – Fixed Scope, *Variable Time*



Stakeholder:
Will we get all **85 stories** that we hoped to have by the Release date?

Product Owner:
Probably, but we might have **as few as 70**, so we should plan accordingly.

Scope Driven – Fixed Scope, *Variable Time*

Sprint Number:		Sprint 1	Sprint 2	Sprint 3	Rollout 1
Sprint Dates:		7/1/2013 – 7/22/2013	7/23/2013 – 8/16/2013	8/17/2013– 9/10/2013	9/10/2013 – 9/30/2013
Sprint Theme:		*Base functionality*	*Buying*	*Selling*	
Stories	**Story Points**	Estimated Velocity - 34 Story Points	Estimated Velocity - 34 Story Points	Estimated Velocity - 34 Story Points	Pilot roll out
Simple Search	1	X			
Plan item up for bid	8	X			
Bid on an auction	4	X			
Register	5	X			
Flag problem postings	3	X			
Contact the seller	13	X			
Item alerts	13		X		
Online help	13		X		
Record seller feedback	8		X		
Review seller feedback	2		X		
Advanced search	8			X	
Email to a friend	3			X	
Customize my view	8			X	
Auction browser toolbar	13			X	
Retract a bid	2			X	

Is it Planning or Forecasting?

Photo Credit: Arielle Nadel
(https://flic.kr/p/7Acy2h)

Questions

Our Contact Details

Accelerating Agility

ADDRESS: 10311 Motor City Drive
Suite 750
Bethesda, MD 20817

WEBSITE: www.agilious.com

EMAIL: manjit@agilious.com

TELEPHONE: 240.244.3311

www.facebook.com/agilious

@agilious

THANK YOU