



*“Caution: Scaling Ineffective Scrum Can Lead to More Chaos”*

Michael S. McCalla  
Agile Project and Portfolio Summit  
Harrisburg University of Science and Technology

# About Me

---

- Agile Coach and Owner of Achieving Agility
  - We provide agile and lean coaching, training, and transformation services
- 15 Years in Software Product Management/Development in the Financial Domain
  - Grew up in a number of vendor start-ups
  - Played many different roles along the way
- Agile Practitioner for 8 years
- Agile fits me like a glove and I am very passionate about it
- Spent last 4 years coaching organizations adopt agile principles and practices
  - Coached on the Portfolio, Program, and Team Level
  - Led Transformation Efforts
  - Helped Teams Build Great Quality Products

# Agenda

---

- Rate Your Scrum Team(s)
- Teams within the Scaling Frameworks
- Why Agile?
- Effective vs. Ineffective Scrum
- Scaling Ineffective Scrum Case Study
- Essentials for Effective Scrum
- Revise Scrum Team(s) Score

# Rate your Scrum Team's Effectiveness

“If your organization has scrum teams, how would you rate their effectiveness on a scale 1 to 5?”

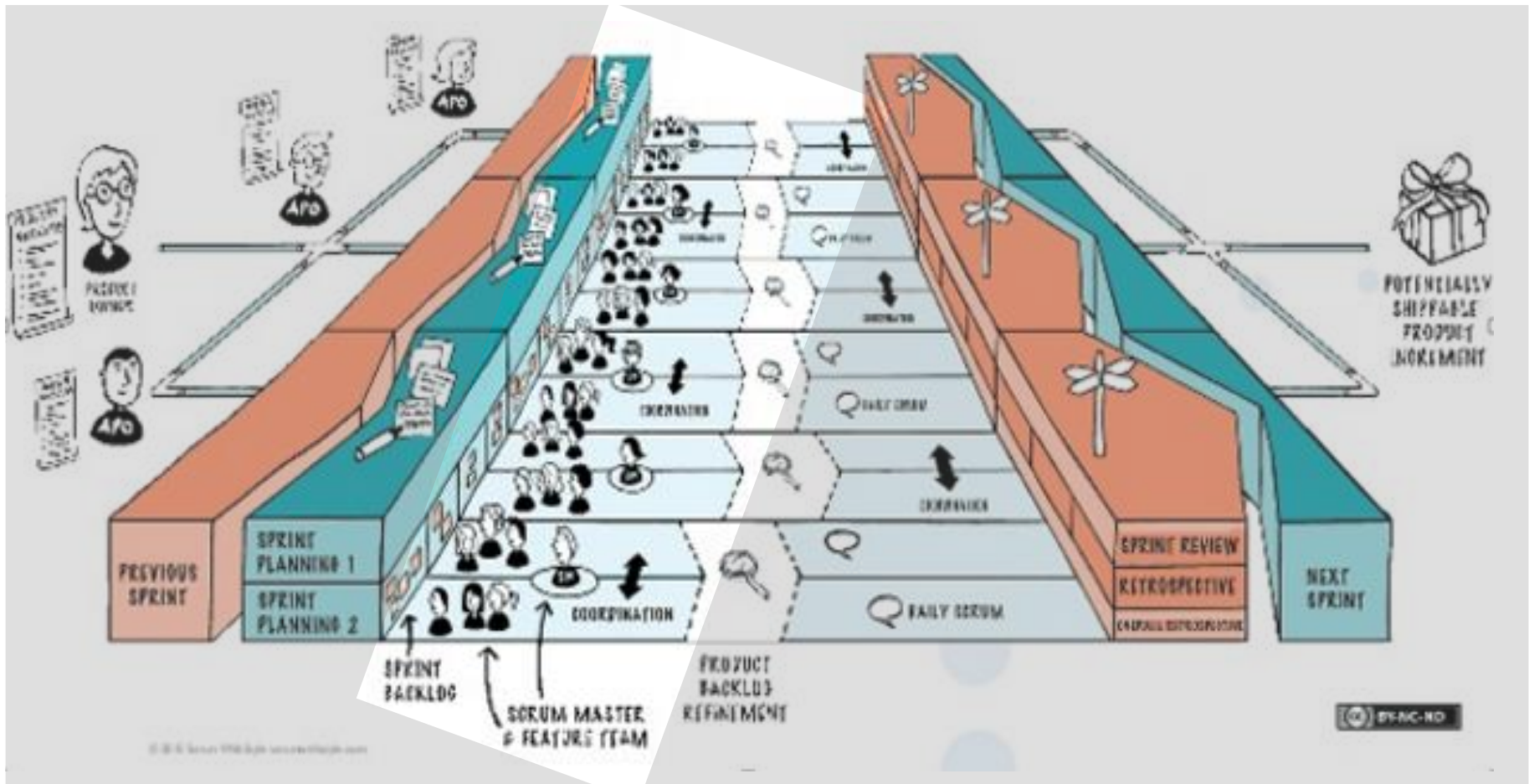
Scrum Report Card	
Initial Score (1-5):	
Revised Score Categories (1-5)	
Cross Functional Teams	
High Focus on Quality	
Product Owner Selection	
Thin Vertical Slices	
Swarming	
Effective Scrum Masters	
Average:	

# Scaling Agile Frameworks

---

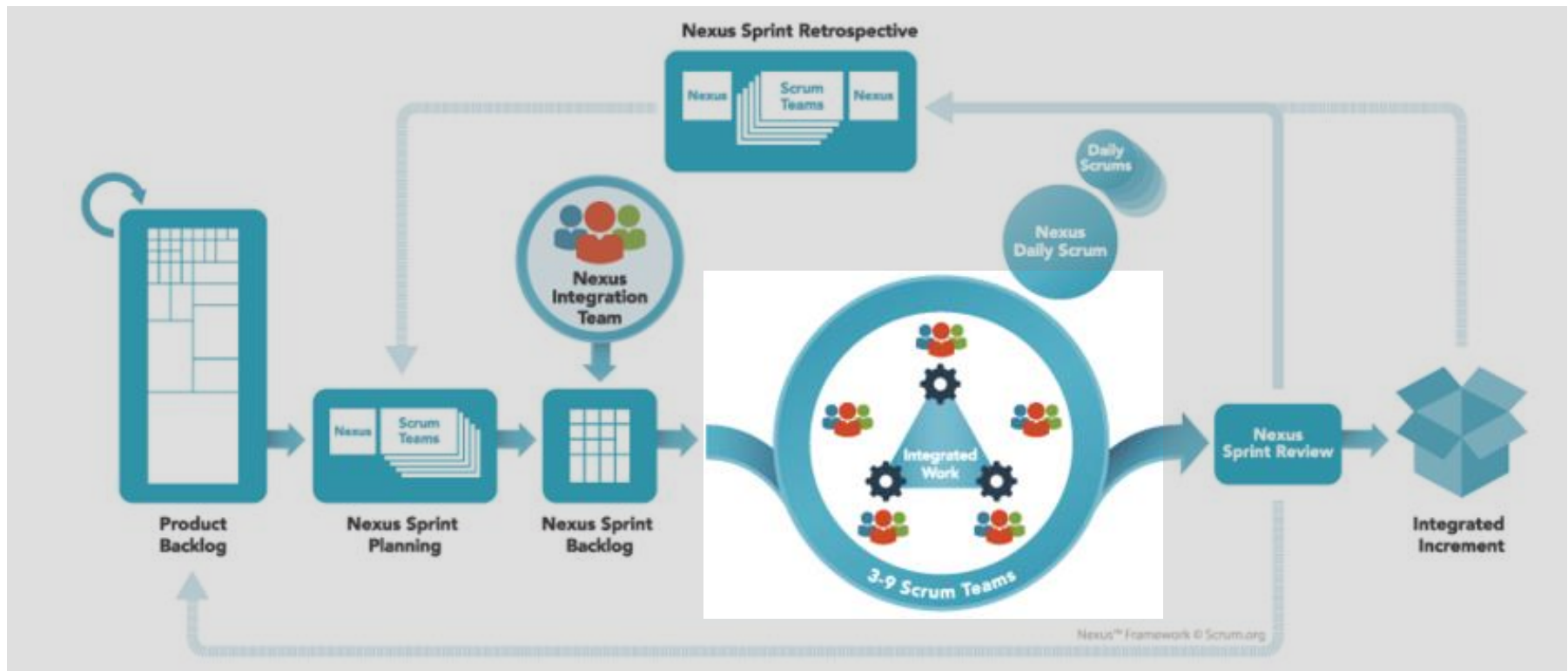
*“A brief look at the role of the **team** within the frameworks”*

*“Feature teams balance specialization and flexibility”*

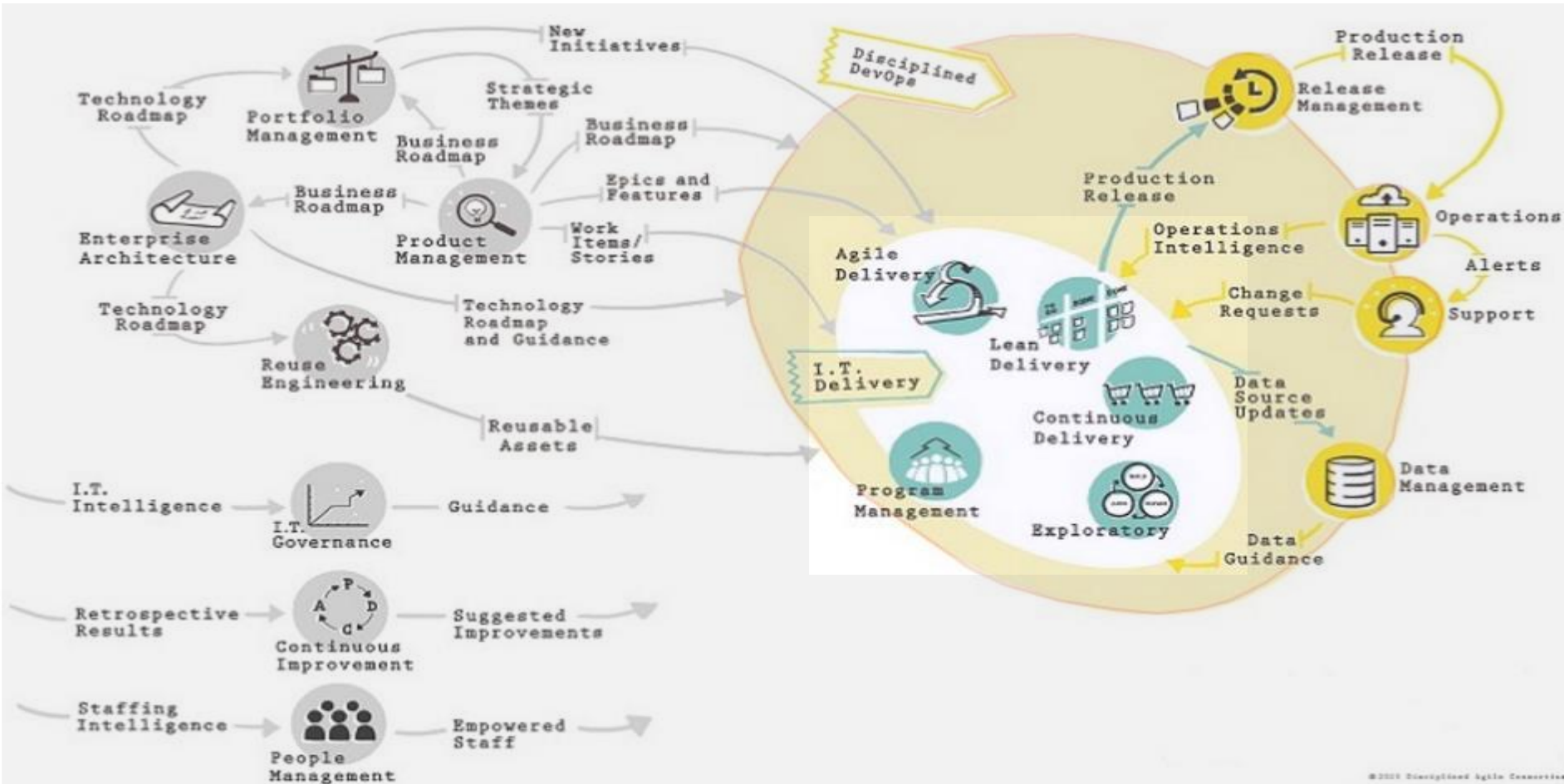


# NEXUS™ FRAMEWORK

*“The Scrum Teams are responsible for developing increments of potentially releasable software, as prescribed in Scrum”*



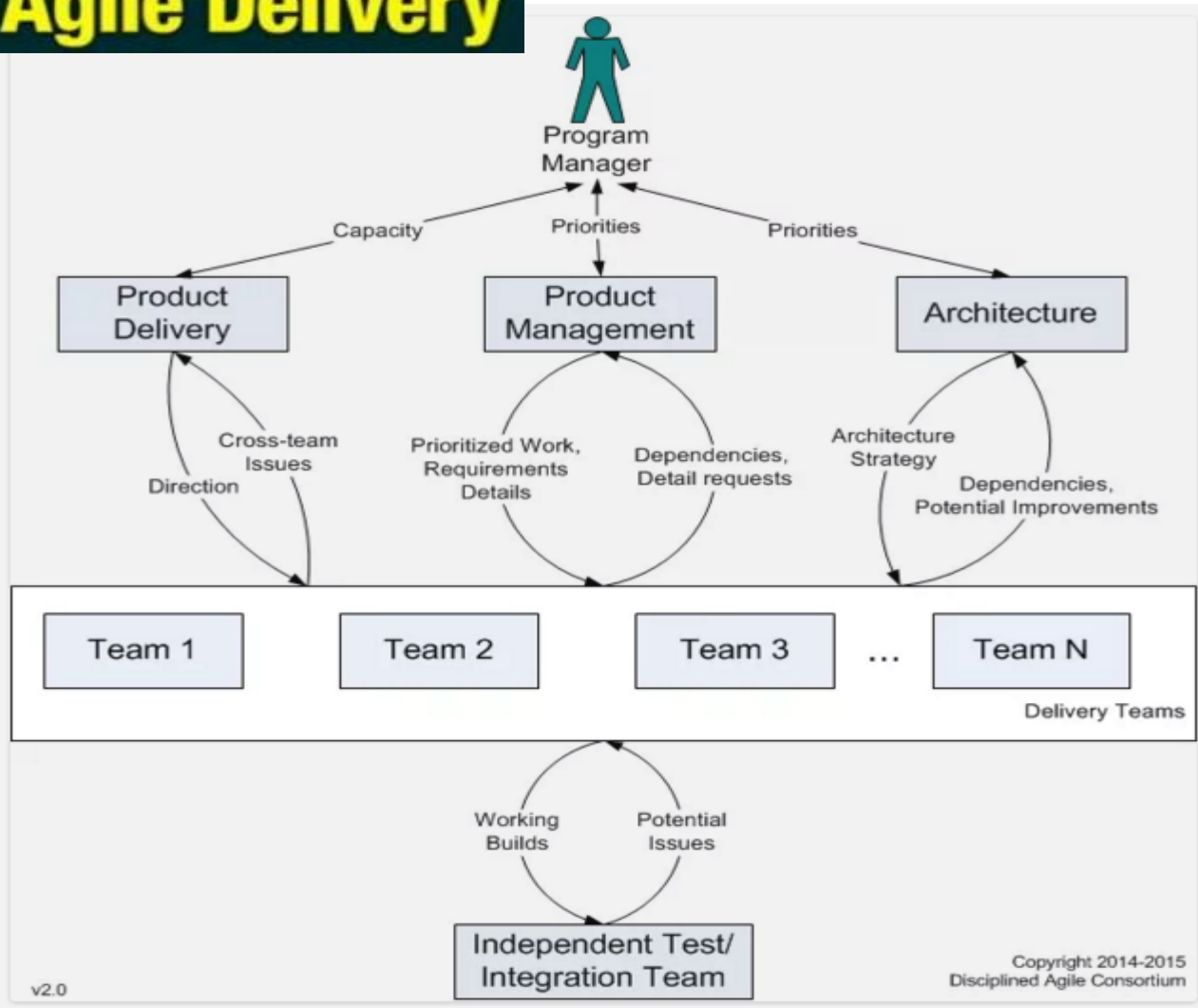
# Disciplined Agile Delivery



Achieving Agility

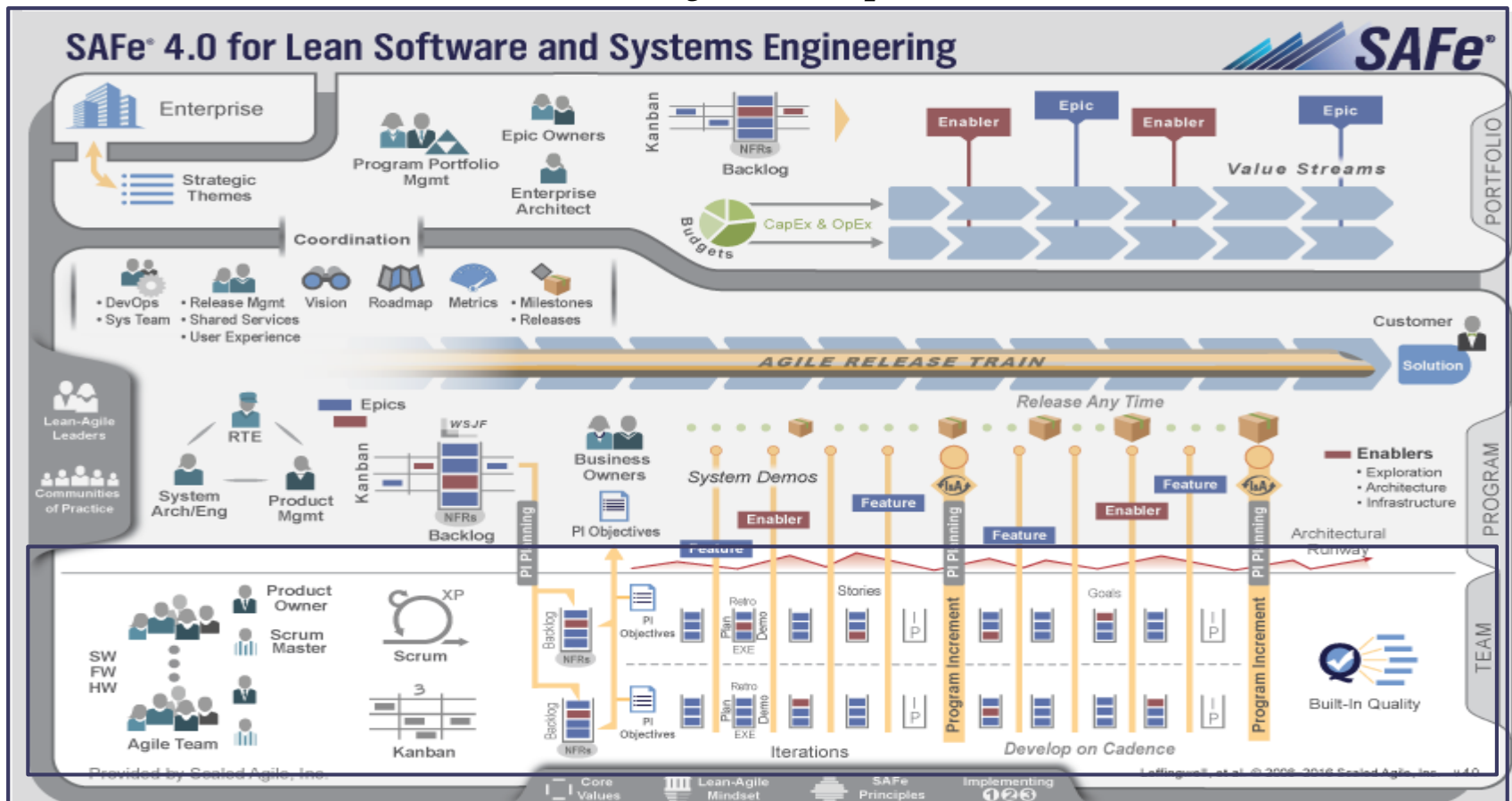


# Disciplined Agile Delivery



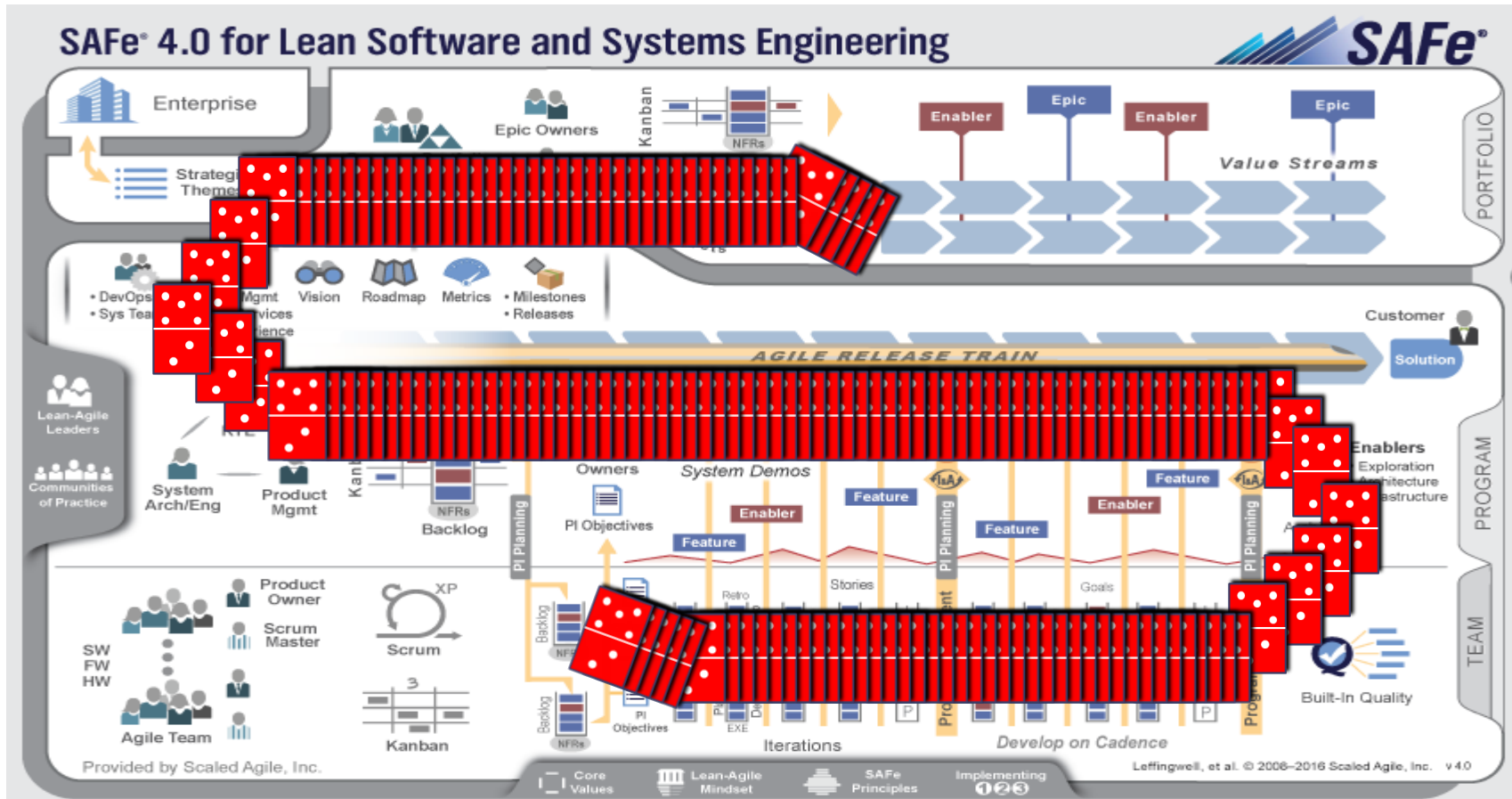
“Agile deemphasizes specialized roles and considers all team members equal – everyone pitches in to deliver a working solution regardless of their job description”

*“Without Effective Agile Teams, comprised of empowered and motivated individuals organizations cannot scale Agile to achieve larger business benefits of Enterprise Lean-Agile development”*



# The Ripple Effect of Ineffective Scrum

*“Ineffective Scrum impacts the whole organization”*





# Defining Effective/Ineffective Scrum

Effective Scrum
Cross Functional/Generalizing Specialist
High Focus on Quality
Tech Debt Periodically Paid Down
A True Product Owner
Work Units: Thin Vertical Slices
Swarm on Stories
Trained Scrum Master
Continuous Flow

Ineffective Scrum
Component Teams/Specialist
High Focus on Meeting Deadlines
Tech Debt Steadily Accrued
Product Owner Proxy
Work Units: Horizontal Slices
Waterfall The Sprint
Project Manager with the Scrum Master Label
Big Bang Delivery

# Case Study: Scaling Ineffective Scrum

---

## Non-Cross Functional Teams

- Maintained Matrix Environment
- Teams were Aligned By Component

## No Training

- Outdated Training Document Distributed
- Teams were Stood Up within One Quarter

## Lack of Organizational Alignment

- No Priority: “Everything will get done”
- IT Driven Transformation

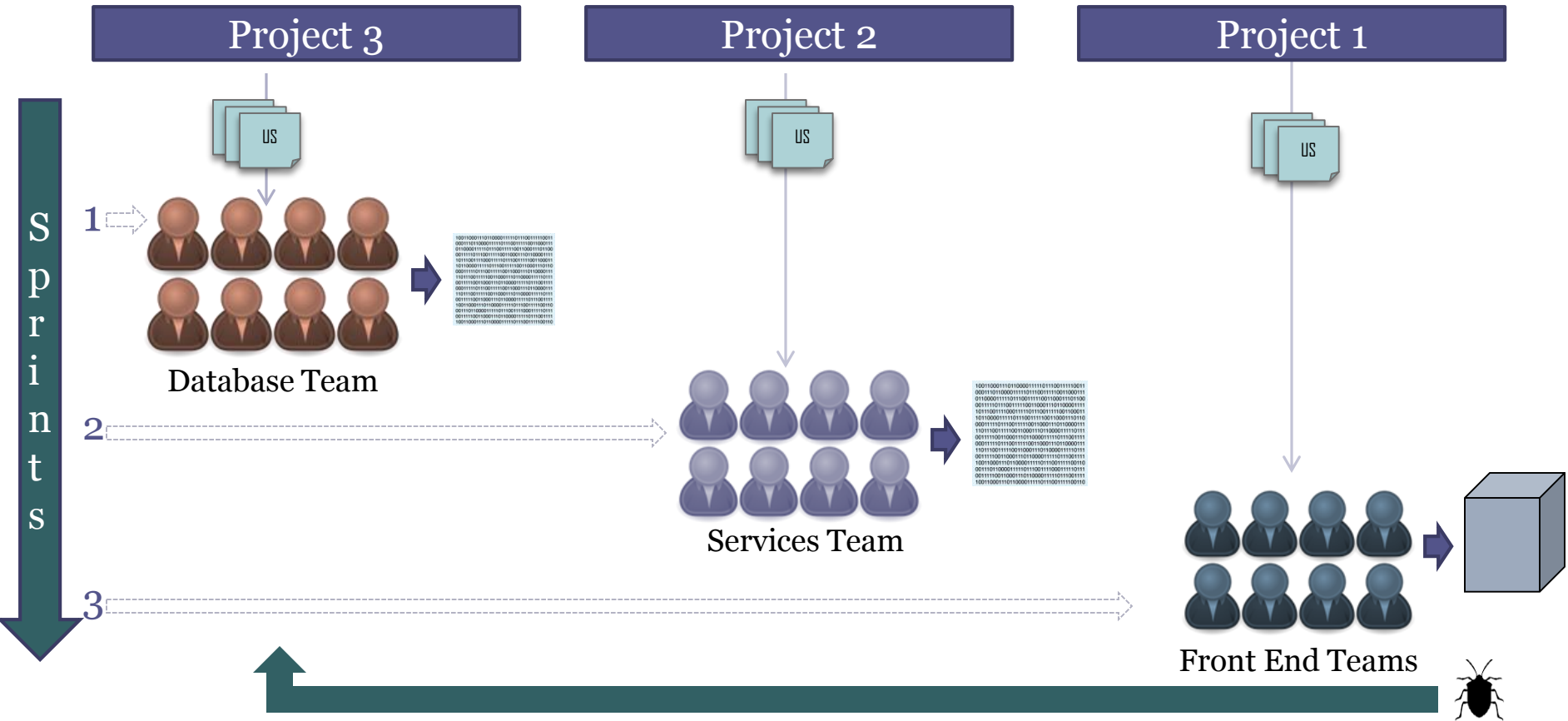
## No Project WIP Limits

- PMO Managed Pipeline in Traditional Fashion
- Utilization Focused

## No ALM Tool

- Excel Backlogs in SharePoint
- No Visibility

# Scaling Ineffective Scrum: Flow of Work



# Case Study: Scaling Ineffective Scrum

---

How did they know that their implementation was failing?

- Time to Market decreased 30%
- Defects Increased
- Project Budgets were Exponentially Exceeded
- Employee Survey Revealed Employee Satisfaction Decreased
- Most Importantly.....

*Business Owners and Customers Expressed Dissatisfaction*



# Scrum Essentials

---

*“Lets look at the Prerequisites and Practices  
that make Scrum Teams Great”*

# Cross Functional Teams

## Cross Functional Definition:

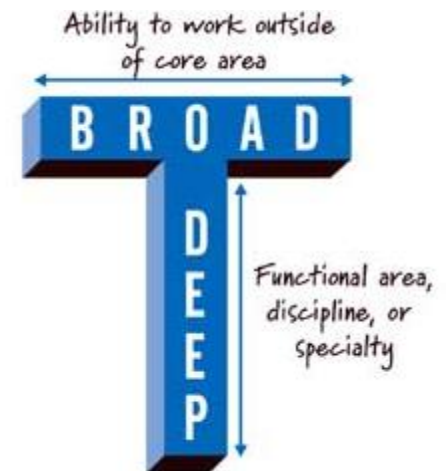
- A group of software professionals that can deliver customer value at the end of an iteration.

## Characteristics:

- Focus is Throughput: Useable Features
- Customer Centric
- Shared Team Responsibilities/Accountability

## Benefits:

- Eliminates Coordination/Resourcing Needs
- Drives More Effective Communication
- Breeds a Healthy Environment of Trust and Learning
- Team Members become T Shaped

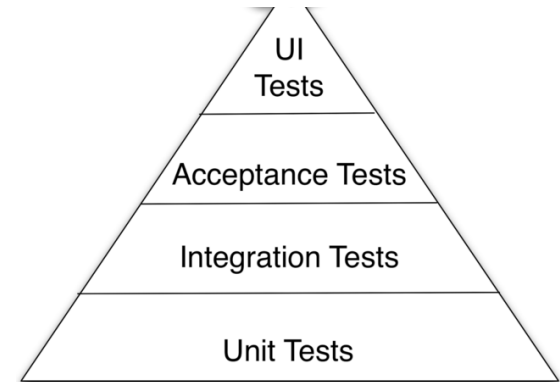


*“Rate Your Team(s) in this Category”*

# Quality: A First Class Citizen

---

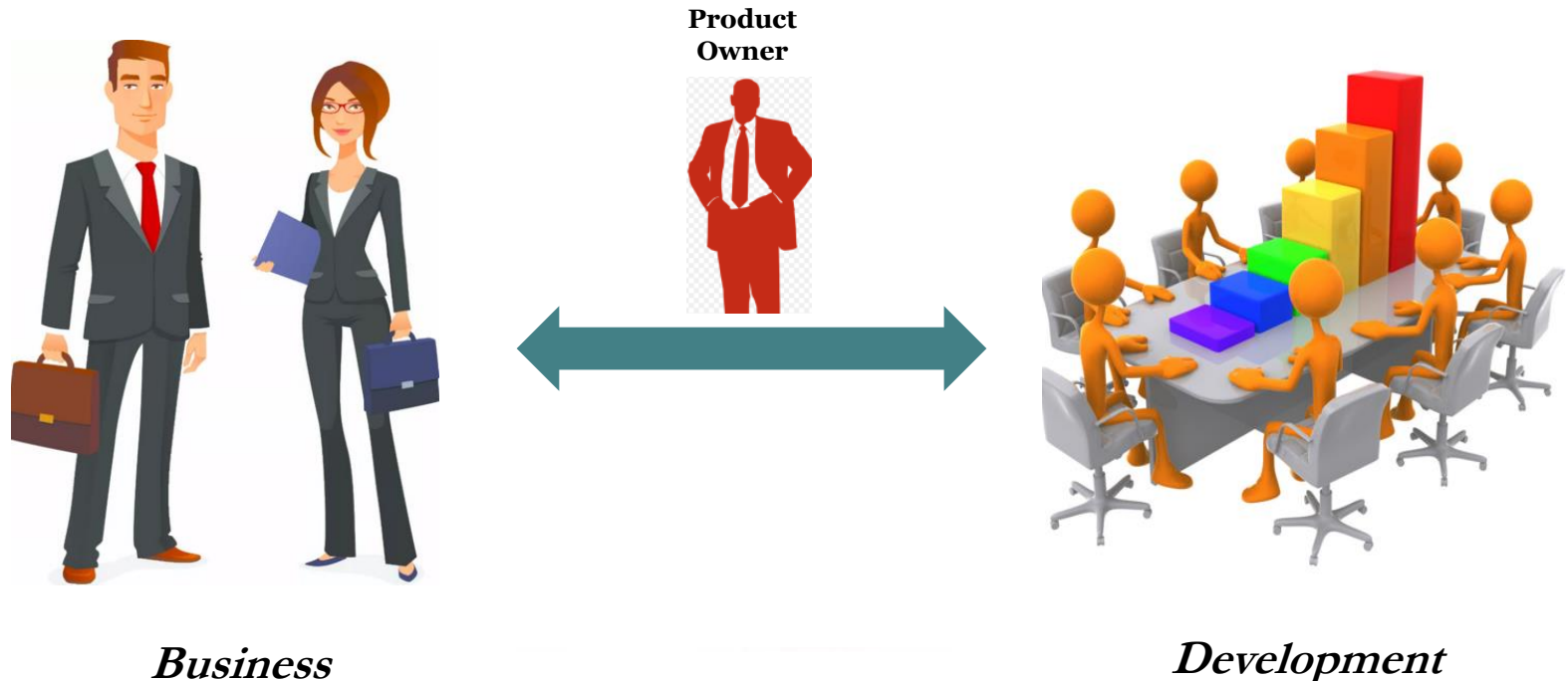
- Quality is Non-Negotiable in Agile
- Quality is an investment
- Reliability of the product
  - Application Outages
  - Hard Errors
  - Scalable
  - Easily Maintainable
- Technical Debt is periodically paid down
  - Managed on the backlog with everything else
  - Team educates the Product Owner about possible ramifications
- Testing is inherently built into the team's daily processes
  - Adopt Test Driven Development (TDD)
  - Testing is performed at different levels (unit, acceptance, integration, regression)
  - Invest in test automation and continuous integration



*“Rate Your Team(s) in this Category”*

# Product Owner: Select the Best Candidate

*How Ken and Jeff Envisioned the Role*



# Product Owner: Select the Best Candidate

---

- Becoming a Product Owner is not a demotion
  - Should be one of the most powerful positions in an organization
- Empowered to make decisions
- Respected amongst his peers and trusted
- Not a BA who writes stories
  
- Does this look like the job description of a BA?
  - Manages stakeholder relationships, communication & expectations
  - Represents the business and is responsible for value delivery
  - Accountable for the vision, scope, and scale of the product
  - Clarifies the customer need to the team
  - Responsible for “what” we build

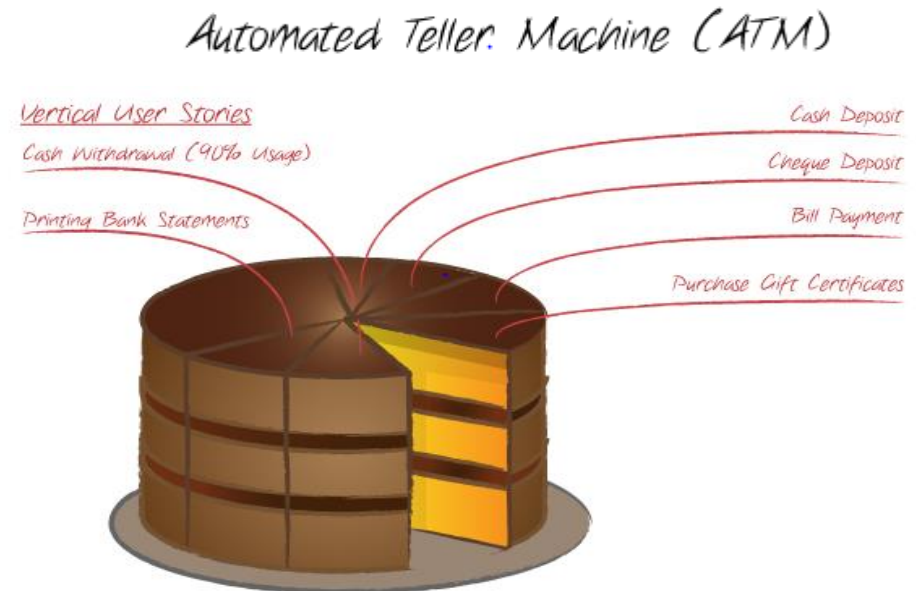


*“Rate Your Team(s) in this Category”*

# Thin Vertical Slices

- Work in Agile projects is organized by **Units of Value**, rather than by **Architectural Layer**. This forces early integration between all layers, even before the full set of requirements are known
- User Stories/Features are **Thin Vertical Slices** across the Tech Stack delivered incrementally
- Requirements are gathered from the users perspective

*“Rate Your Team(s) in this Category”*



# Swarming

*Swarming is a when a team of people collectively work on an one story to get it finished before moving on to another.*

## Team Approach:

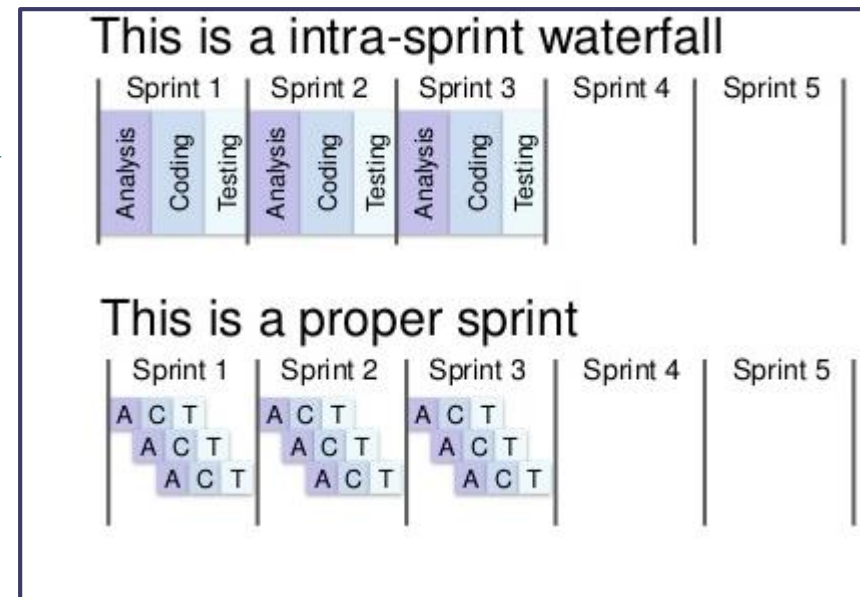
- “There is no such thing as testers and developers on a team. Everybody has to be ready to contribute in any way to get things to done”

## WIP Limits:

- Drives Swarming and Continuous Flow

## Benefits:

- Quicker Feedback from PO
- Forces Integration Earlier and Often
- Mitigates Risk
- Cross Training
- More Focus on Quality



# Swarming Demonstrated

D Dev Task    T Test Task

## Non-Swarming Team

	TO DO	IN PROGRESS	DONE
PBI 'A'		D D D D D D D D D D	D
PBI 'B'	T T T	D D D D D D D D	
PBI 'C'	T T	D D D D D D D D D D	D D
PBI 'D'	T T T	D D D D D D D D D D	D

Total Sprint Stories: 10  
 Stories Started: 10  
 Stories Finished: 2 (20%)

## Swarming Team

	TO DO	IN PROGRESS	DONE
PBI 'A'			D D T T D D D T
PBI 'B'	D T	T T D D	
PBI 'C'	D D T D D T	T D	
PBI 'D'	D D D T D D D T		

Total Sprint Stories: 10  
 Stories Started: 8  
 Stories Finished: 8 (80%)

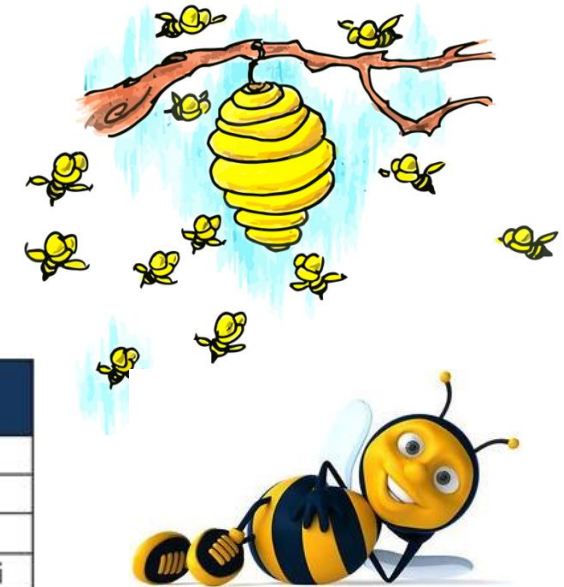




# Swarming Approach

## Traditional Approach

Task	Hours	Person Responsible	Day Worked On
Task 1	8	Renee	Monday
Task 2	4	Renee	Tuesday
Task 3	4	Renee	Tuesday
Task 4	2	Todd	Monday
Task 5*	4	Todd	Wednesday



## Developing Team

Task	Person Responsible-With Expertise	Person With Capabilities
Task 1	Renee	Renee, Todd, Mick
Task 2	Renee	Renee, Mick, Greg
Task 3	Renee	Renee, Todd, Greg, Marie
Task 4	Todd	Todd, Marie, Renee, Mick, Heidi
Task 5*	Todd	Todd, Marie, Renee, Mick

## Swarming

Task	Person Responsible	Person With Capabilities
Task 1	Renee (Marie to watch and learn)	Renee, Todd, Mick
Task 2	Mick	Renee, Mick, Greg
Task 3	Greg	Renee, Todd, Greg, Marie
Task 4	Todd (Heidi to watch and learn)	Todd, Marie, Renee, Mick, Heidi
Task 5*	Todd (Heidi to watch and learn)	Todd, Marie, Renee, Mick

*“Rate Your Team(s)  
in this Category”*

# Effective Scrum Masters

---

*Scrum Masters are Servant Leaders and:*

- Change Agents: “What is holding us back?”
- Agile Champion: “Does this practice adhere to the principles and values?”
- Facilitators: “How can I put you in a position to succeed?”
- Process Improver: “I think we can do this better”
- A Mediator: “How can we resolve this conflict?”
- Enabler: “How do we get it done?”
- Team Protector: “Our office hours are from 11-12, you can come back with those questions then”
- Motivator: “We can do this”



*“Rate Your Team(s) in this Category”*

# Words of Wisdom

---

- Start Small: Standup one or two teams working on a product and master scrum.
  - Involve the business early on and select the right PO
  - Inspect and Adapt along the way
  - Organizational constraints need to be escalated and dealt with
- Train: Provide the education they need to succeed to everyone involved.
- Coach: Bring in a experienced Scrum Master or Agile Coach
- Expand Incrementally
  - Disseminate the original team members to other teams to be champions
- Introduce one agile program across multiple teams
  - Inspect and Adapt
- Scale Up!
- Don't stand for traditional waterfall practices with Agile labels!