iNNolutioN
agile innovative solutions

Scrum: A Manager's Guide
Agile Development Practices East 2010
November 16, 2010
Orlando, Florida
by Kenny Rubin

Includes slides from: Mountain Goat Software

www.innolution.com

1

✺ Background of Kenny Rubin

Author

オブジェクト指向開発の
プロジェクト管理

Succeeding with Objects

Scrum: A Manager's Guide

Trainer/Coach

Trained more than 15,000 people in Agile/Scrum, SW dev and PM

Provide Agile/ Scrum coaching to developers and executives

CST CERTIFIED ScrumTrainer

Experience

Former Managing Director

ScrumAlliance
transforming the world of work

My first Scrum project was in 2000 for bioinformatics

GENOMICA

IBM **Executive**

SMALLTALK

ParcPlace

2

⚡ Agenda

Scrum Framework

Concepts

Roles

Benefits

Industry Results

Pitfalls / CSFs

⚡

Scrum Framework

❋ Scrum Team

5

❋ Scrum Framework

6
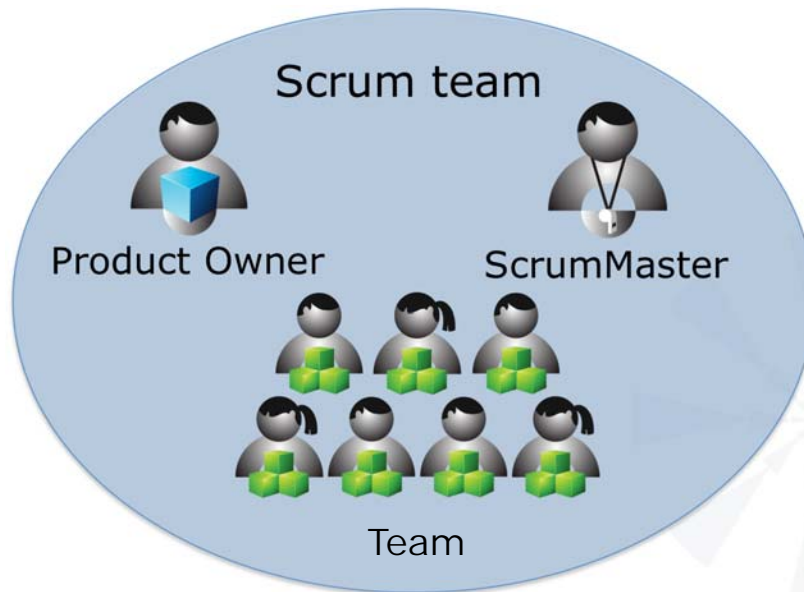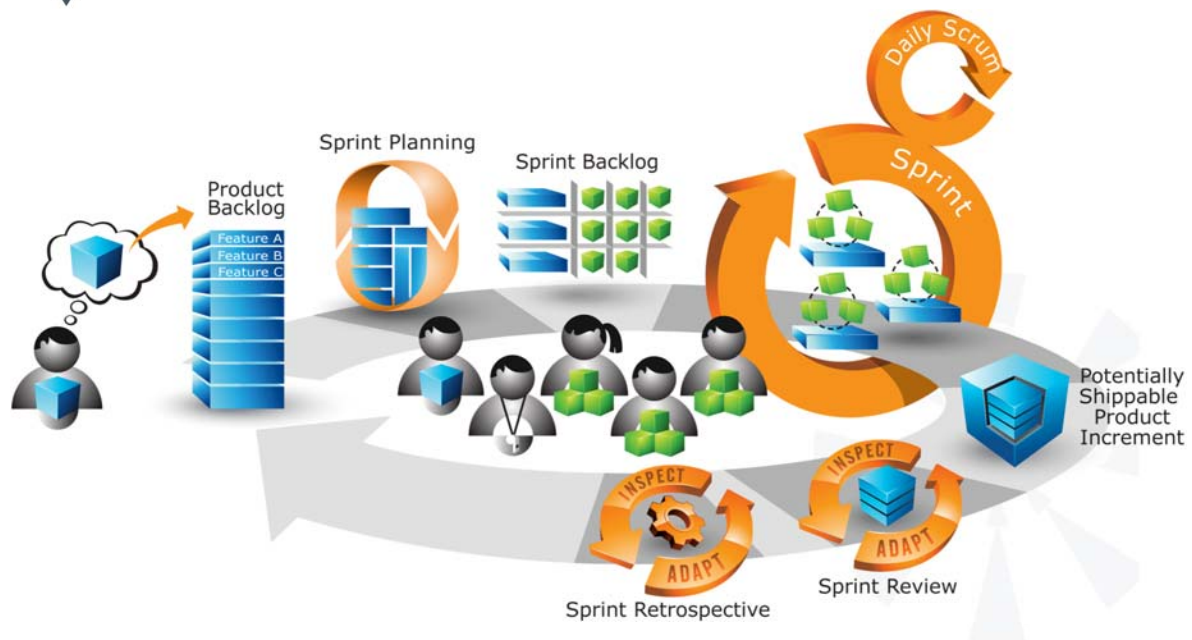
Prioritized Product Backlog

This is Prioritized Product Backlog

As a frequent flyer, I want to A	3
As a frequent flyer, I want to B	5
As a frequent flyer, I want to C	5
As a frequent flyer, I want to D	2
As a frequent flyer, I want to E	2

- Owned by Product Owner

- Captures product requirements (features to be developed or changes to be made)

- Prioritized by value, cost, knowledge and risk

- Each feature has an associated relative effort estimate provided by the Team

Sprint Planning

Sprint planning is **first** part of **eve**ry sprint

Sprint 1 | Sprint 2

Sprint Planning

Product Backlog

Feature A
Feature B
Feature C

Sprint Backlog

- Team members and Product Owner agree on Product Backlog Items to build

- Team members define tasks for each Product Backlog Item & organize them into a Sprint Backlog

Represents what to do
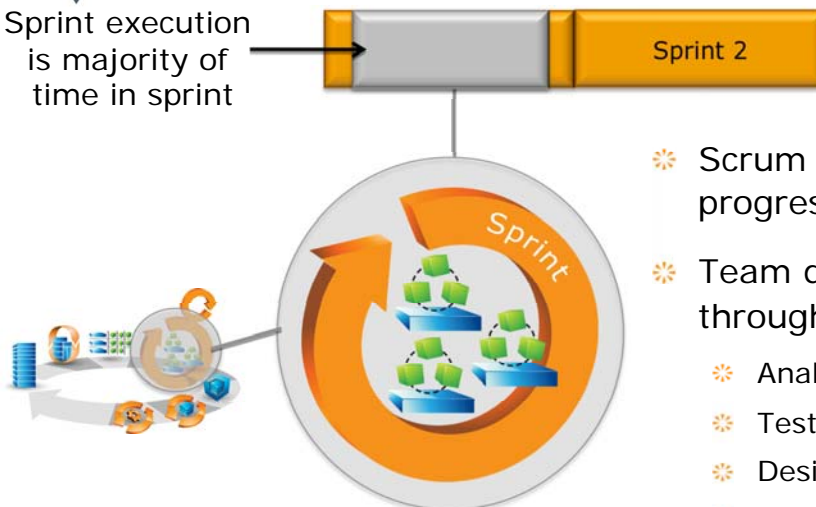
Represents tasks for how to do it

❄ Sprint Backlog

This is Sprint Backlog

- ❄ Owned by the Team
- ❄ Captures Sprint implementation strategy
- ❄ Consists of tasks
- ❄ Supports tracking of remaining effort

Code the UI	8
Write test fixture	6
Code middle tier	12
Write tests	5
Automate tests	4

❄ Sprint Overview

Sprint execution is majority of time in sprint
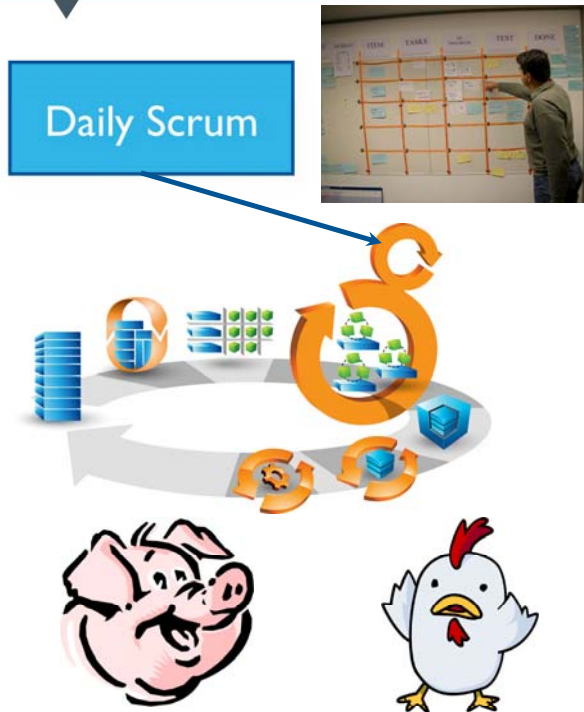

Sprint 2

Sprint – 2 to 4 weeks minus times for planning & review activities

- ❄ Scrum projects make progress in a series of Sprints
- ❄ Team develops each feature through opportunistic use of:
  - ❄ Analysis
  - ❄ Testing
  - ❄ Design
  - ❄ Code
  - ❄ and so on
- ❄ Product is potentially shippable after every Sprint

❋ What is the Daily Scrum

- ❋ Parameters
  - ❋ Daily
  - ❋ 15-minutes
  - ❋ Stand-up
  - ❋ Not for problem solving
- ❋ Three questions:
  - ❋ **What did you do yesterday?**
  - ❋ **What will you do today?**
  - ❋ **What obstacles are in your way?**
- ❋ Chickens and pigs are invited
  - ❋ Only pigs can talk
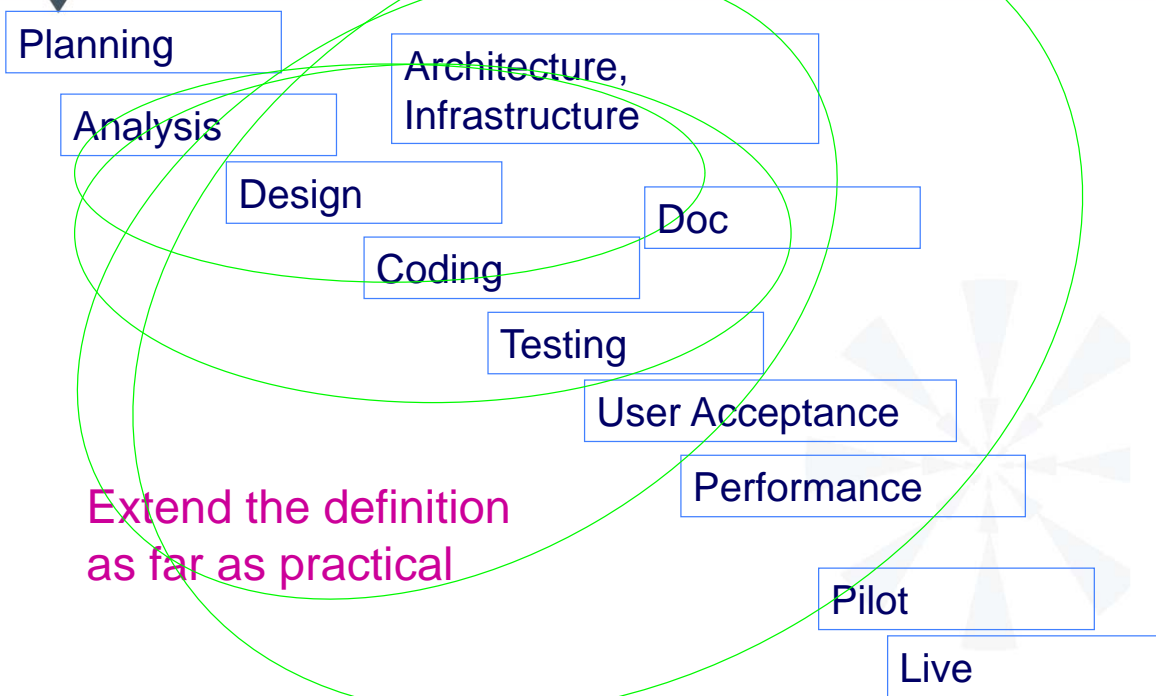- ❋ Help avoid other unnecessary meetings

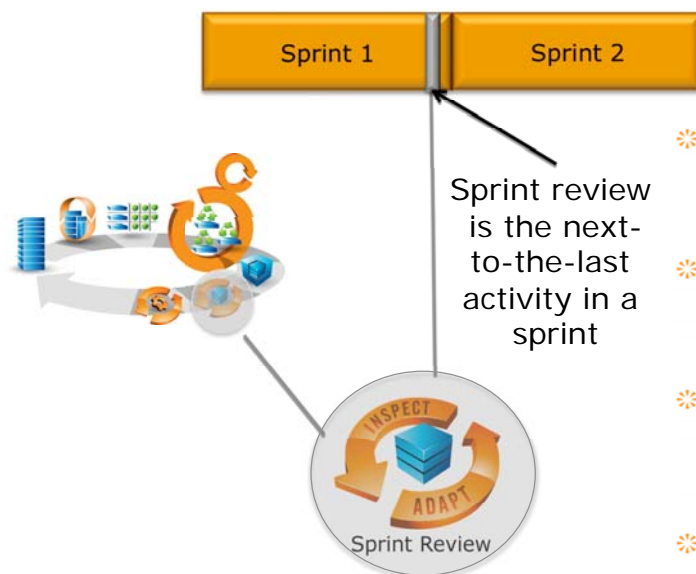# ❋ Potentially Shippable Product Increment

- ❋ Must produce a potentially shippable product increment each Sprint

- ❋ State of confidence

- ❋ Are you done? Except possibly for some release-specific work

Potentially shippable ≠ shippable

Illustration of "Done"

Planning

Analysis

Architecture, Infrastructure

Design

Doc

Coding

Testing

User Acceptance

Performance

Extend the definition as far as practical

Pilot

Live

The Sprint Review

Sprint 1 | Sprint 2

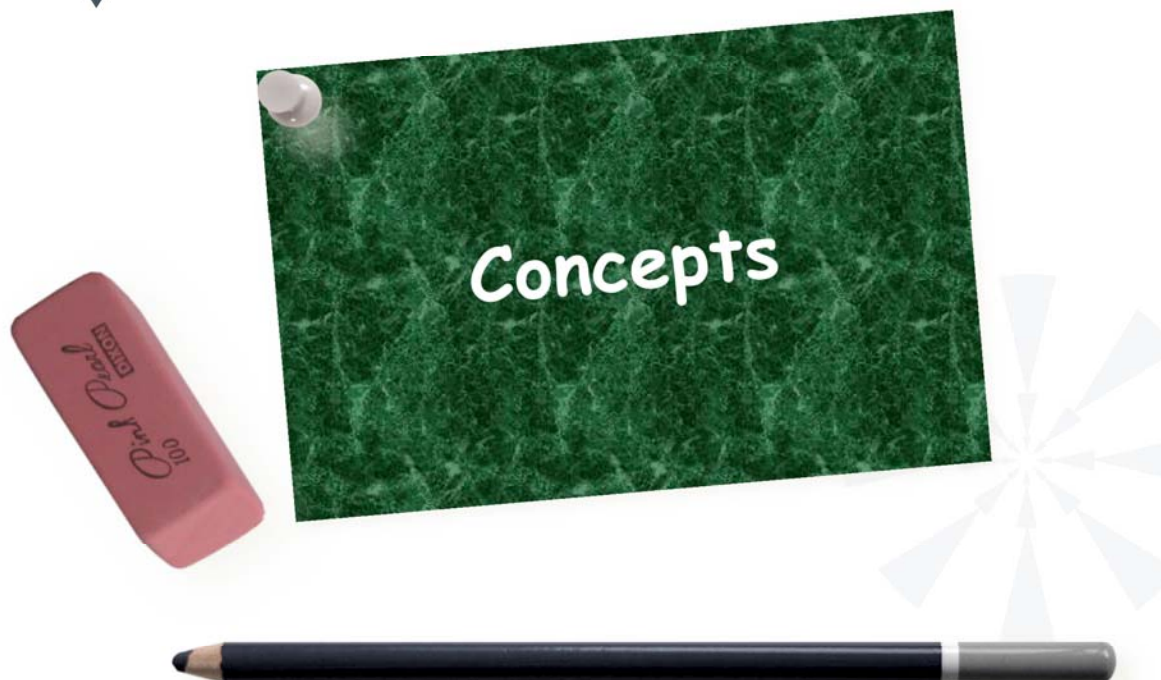Sprint review is the next-to-the-last activity in a sprint

INSPECT
ADAPT
Sprint Review

- Scrum Team presents to stakeholders what it accomplished during the Sprint

- Typically takes the form of a demo of new features or underlying architecture

- Informal
  - 2-hour prep time rule
  - No slides

- Whole team participates

- Invite the world

✳ Sprint Retrospective

| Sprint 1 | Sprint 2 |

Sprint retrospective is
last activity in sprint

Sprint Retrospective
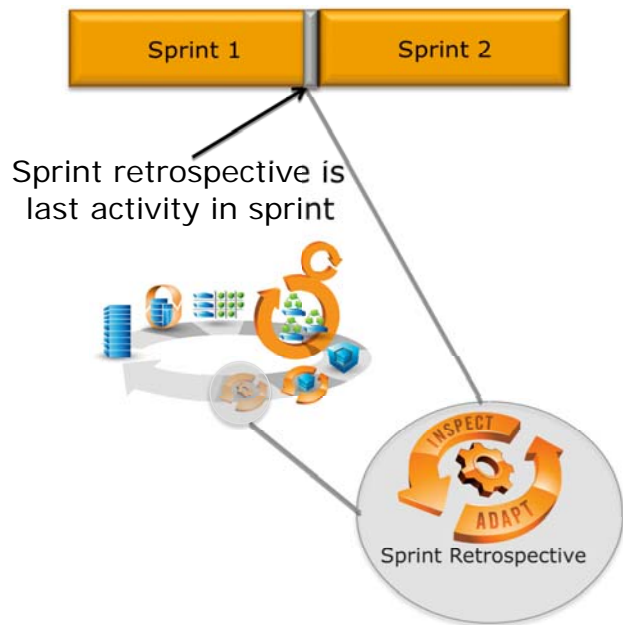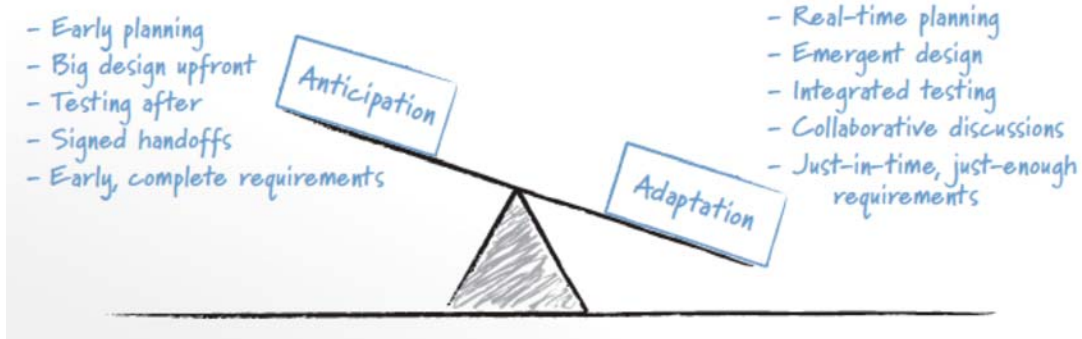
- ✳ Inspect and adapt for the process
- ✳ Periodically take a look at what is and is not working
- ✳ Done after every sprint
- ✳ Whole team participates:
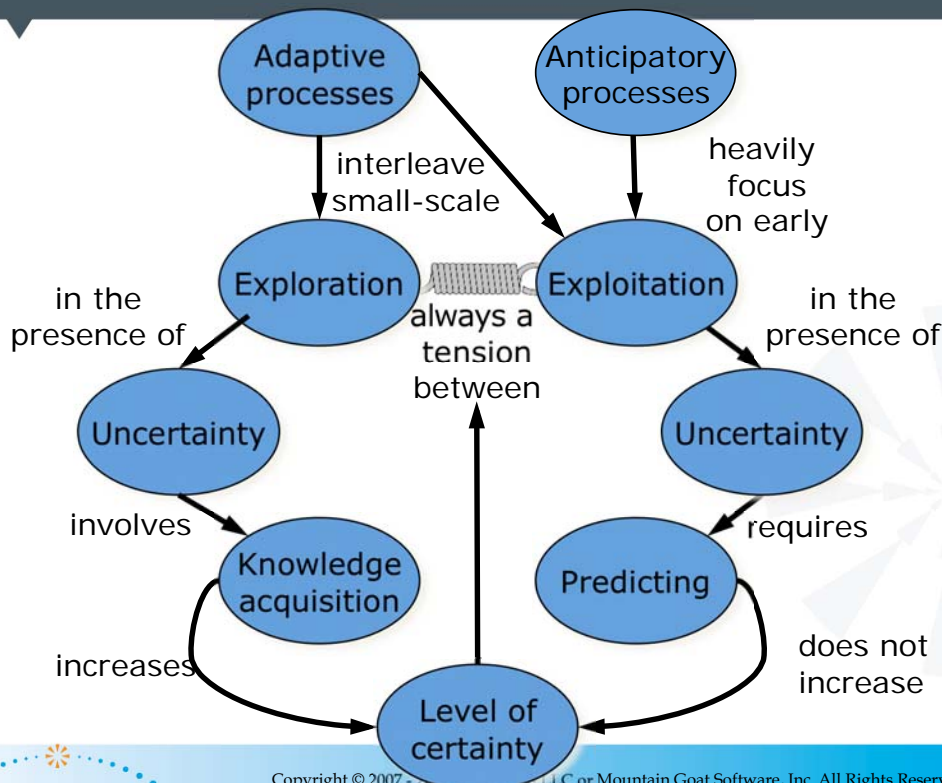 - ✳ Team
  - ✳ ScrumMaster
  - ✳ Product Owner

Concepts

Need to Balance Anticipation with Adaptation

- Early planning
- Big design upfront
- Testing after
- Signed handoffs
- Early, complete requirements

Anticipation

Adaptation

- Real-time planning
- Emergent design
- Integrated testing
- Collaborative discussions
- Just-in-time, just-enough requirements

Exploration versus Exploitation

Adaptive processes

Anticipatory processes

interleave small-scale

heavily focus on early

Exploration

always a tension between

Exploitation

in the presence of

in the presence of

Uncertainty

Uncertainty

involves

requires

Knowledge acquisition

Predicting

increases

does not increase

Level of certainty

Technical Debt –
Desired Release Date
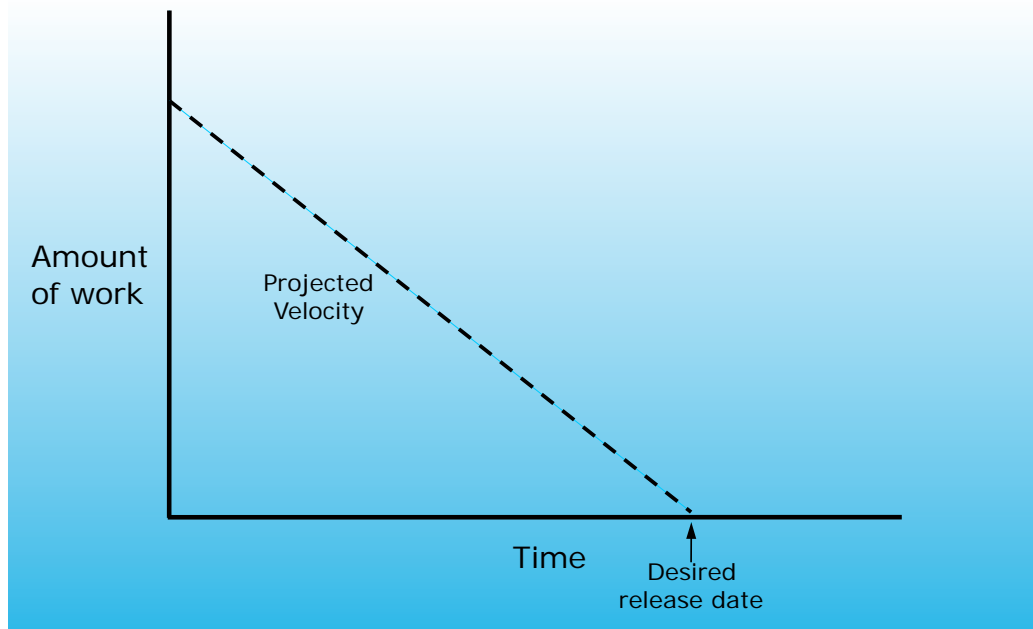
Amount
of work

Projected
Velocity

Time

Desired
release date

Technical Debt –
Likely Release Date
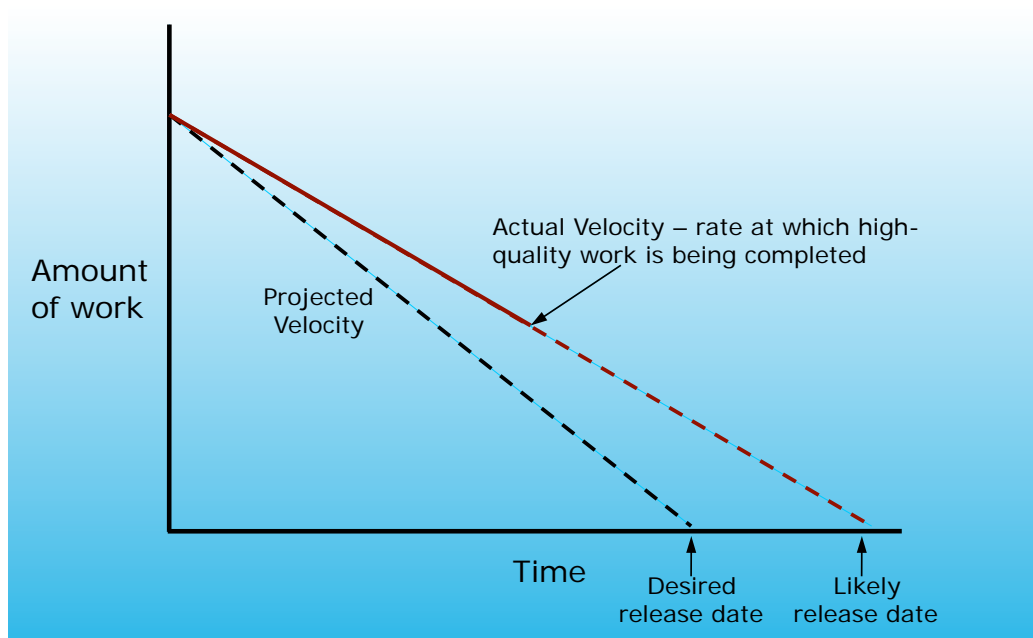
Amount
of work

Projected
Velocity

Actual Velocity – rate at which high-
quality work is being completed

Time

Desired
release date

Likely
release date

Compromised Release Date Leading to Technical Debt

Amount of work

Projected Velocity

"Accelerated Velocity" – velocity needed to finish all of the features by the desired date

This area represents the accrued technical debt

Time

Desired release date

Likely release date
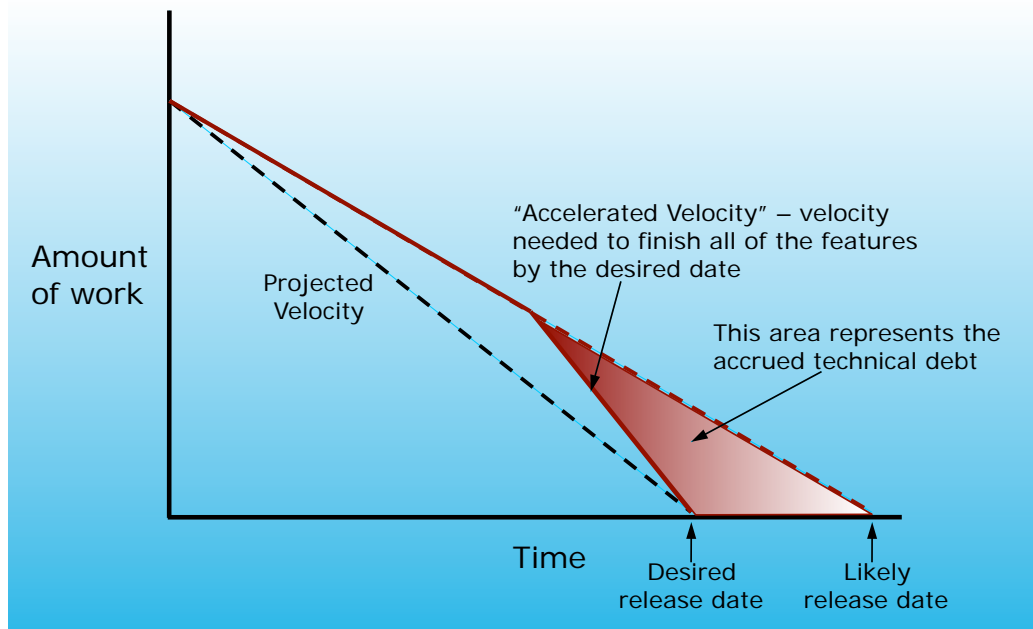
More Compromises; More Technical Debt

Actual Velocity of 2nd Release is much slower due to 1st Release technical debt

"Accelerated Velocity"

This area represents additional accrued technical debt

Amount of work

Velocity graph of 1st Release

Time

1st Release

2nd Release
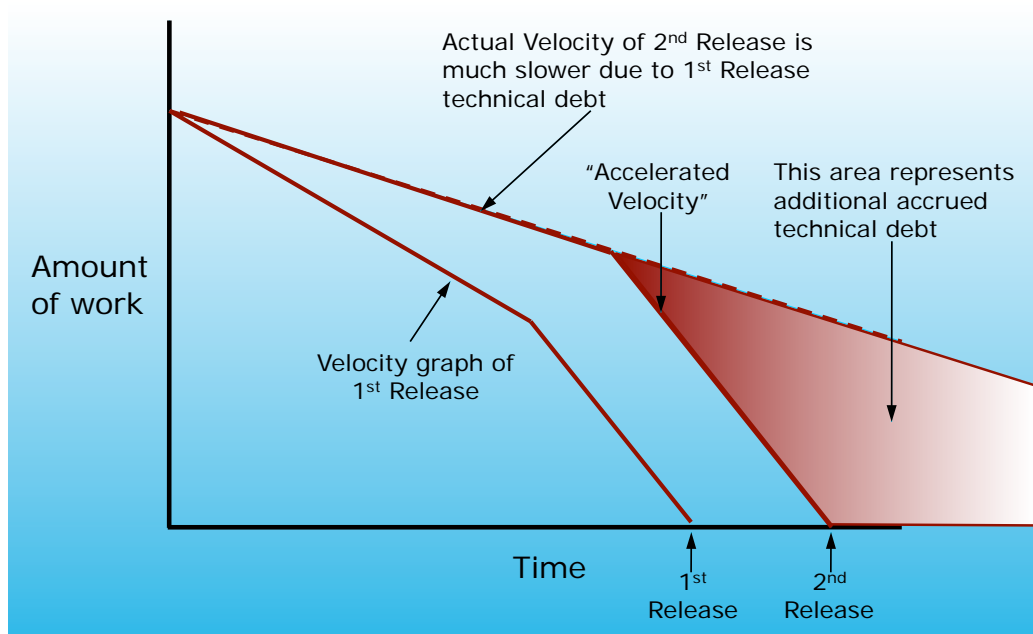
Technical Debt – Make It Visible

* Like financial debt, not all technical debt is bad
* All business people have visibility into financial debt on their balance sheet
* Most business people have no clue what kind of technical debt is in their systems

Assets		Liabilities	
Cash	$600,000	**Current Liabilities**	
Acct Receivable	$450,000	Notes Payable	$100,000
		Accounts Payable	$75,000
		ST Technical Debt	**$90,000**
Tools & Equipment	$250,000	**Long Term Liabilities**	
		Notes Payable	$300,000
		LT Technical Debt	**$650,000**
...	...	...	...

Technical Debt – Make Economically Viable Choices
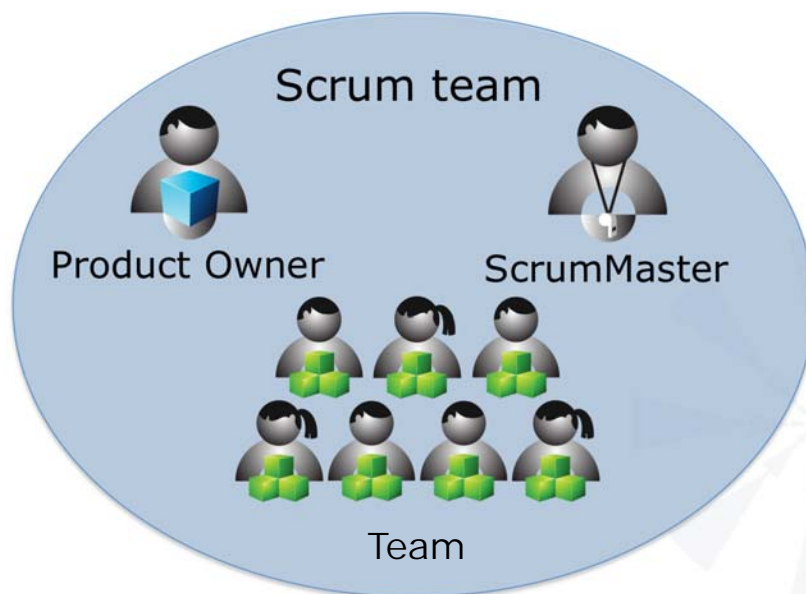
* Can't meet delivery date with all features

* Option 1 – delay shipping the product by three months to finish the work

    * Calculated cost of delay = $450k (lost sales)

* Option 2 – take on technical debt

    * Estimated additional cost of $60k to repay debt with interest after the product ships
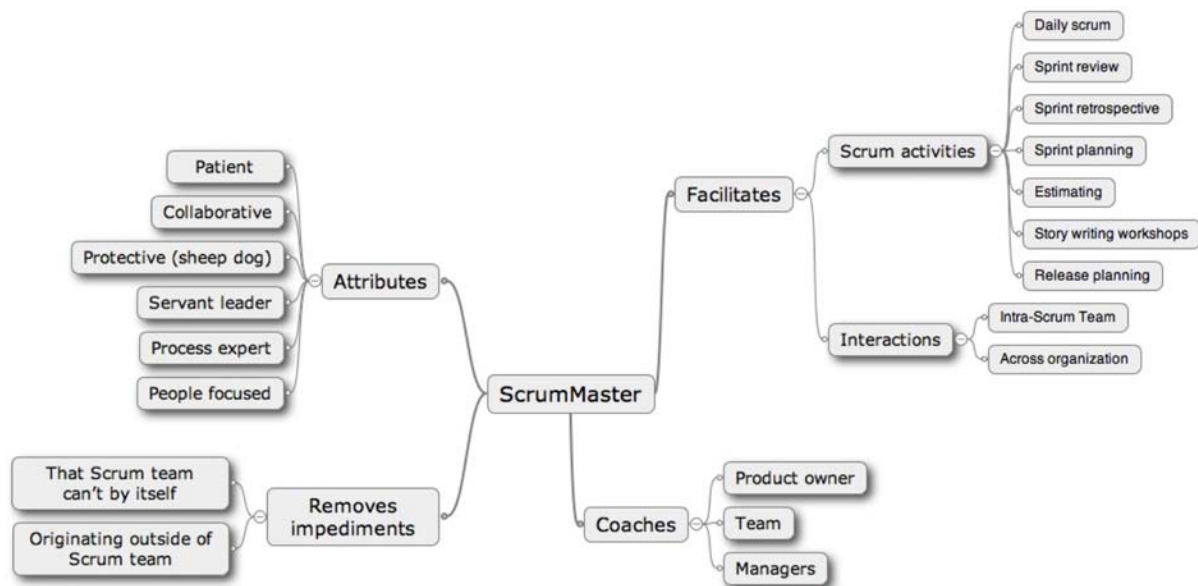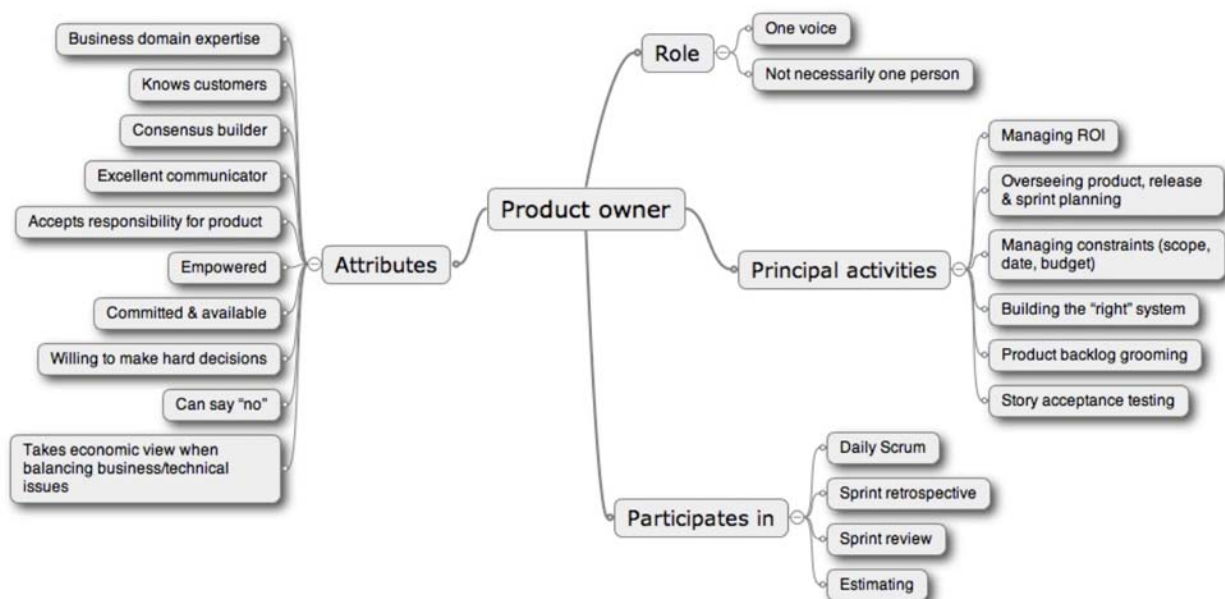   * Probably some additional costs since other projects might have a delay
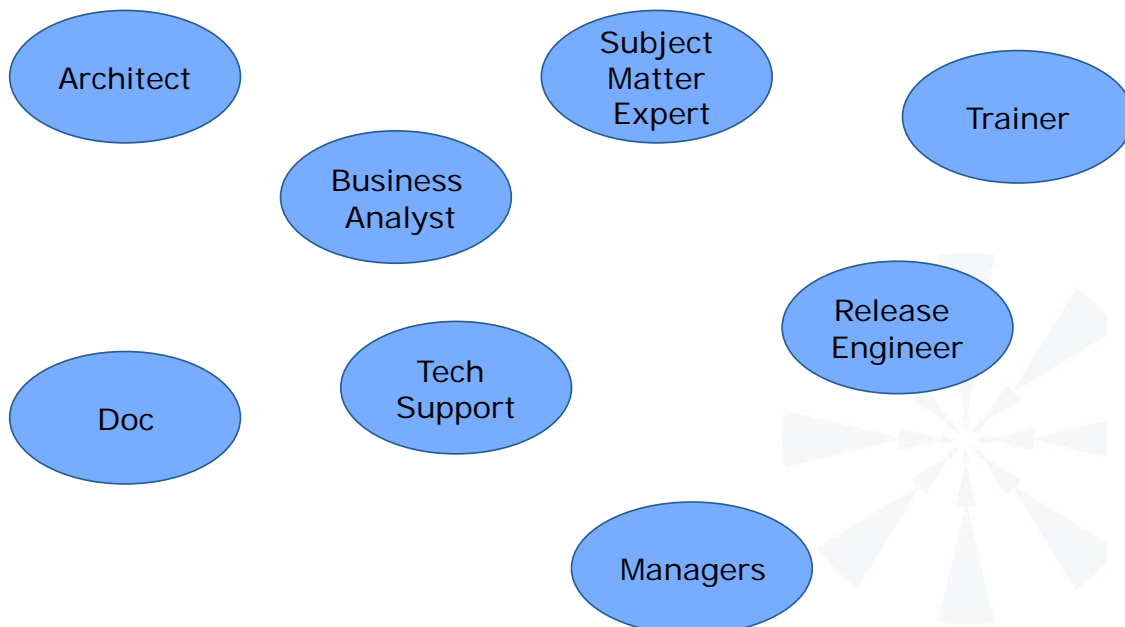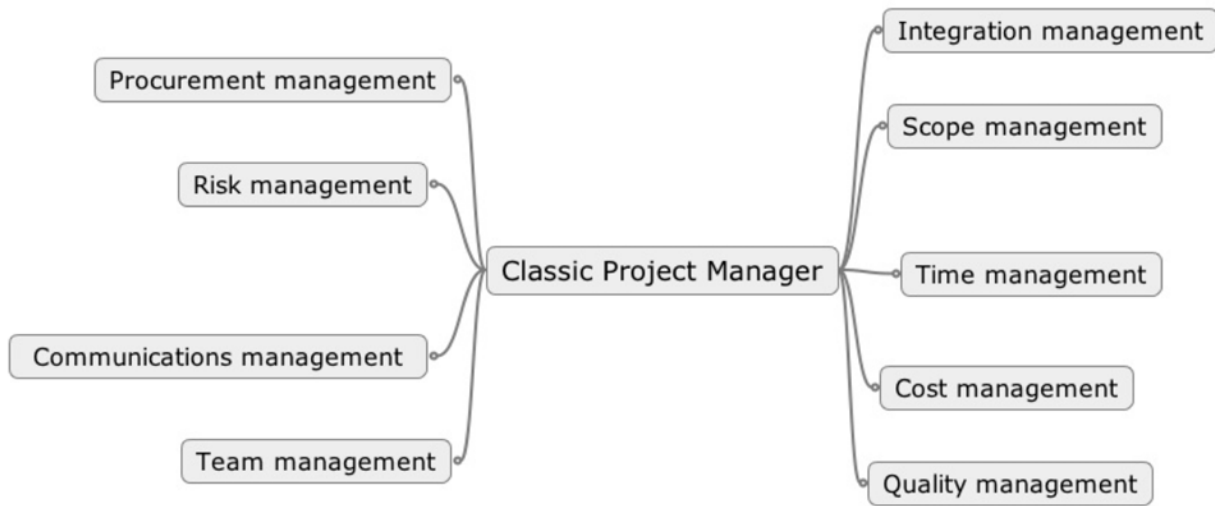
Roles

Scrum Team

Scrum team

Product Owner   ScrumMaster

Team

ScrumMaster

- **ScrumMaster**
  - **Attributes**
    - Patient
    - Collaborative
    - Protective (sheep dog)
    - Servant leader
    - Process expert
    - People focused
  - **Removes impediments**
    - That Scrum team can't by itself
    - Originating outside of Scrum team
  - **Facilitates**
    - **Scrum activities**
      - Daily scrum
      - Sprint review
      - Sprint retrospective
      - Sprint planning
      - Estimating
      - Story writing workshops
      - Release planning
    - **Interactions**
      - Intra-Scrum Team
      - Across organization
  - **Coaches**
    - Product owner
    - Team
    - Managers

Product Owner

- **Product owner**
  - **Attributes**
    - Business domain expertise
    - Knows customers
    - Consensus builder
    - Excellent communicator
    - Accepts responsibility for product
    - Empowered
    - Committed & available
    - Willing to make hard decisions
    - Can say "no"
    - Takes economic view when balancing business/technical issues
  - **Role**
    - One voice
    - Not necessarily one person
  - **Principal activities**
    - Managing ROI
    - Overseeing product, release & sprint planning
    - Managing constraints (scope, date, budget)
    - Building the "right" system
    - Product backlog grooming
    - Story acceptance testing
  - **Participates in**
    - Daily Scrum
    - Sprint retrospective
    - Sprint review
    - Estimating

❋ Team

Characteristic	Description
Self organizing	Team decides how best to organize to meet the Sprint goals
Self managing	Every member of the Team is responsible for "managing" the Team
Cross-functionally Sufficient	No "roles," but all necessary skills (e.g., QA, Programming, UI Design, etc.) necessary to go from Product Backlog to potentially shippable product increment
Right sized	Seven plus or minus two team members
Empowered	Authority to do whatever is needed to meet commitment
Committed	Committed to delivering Sprint features
Focused	Members should be full-time or at least not spread "too" thin
Immutable	During a Sprint, Team structure does not change; membership can change between Sprints

❋ What About These Roles?

Architect

Subject Matter Expert
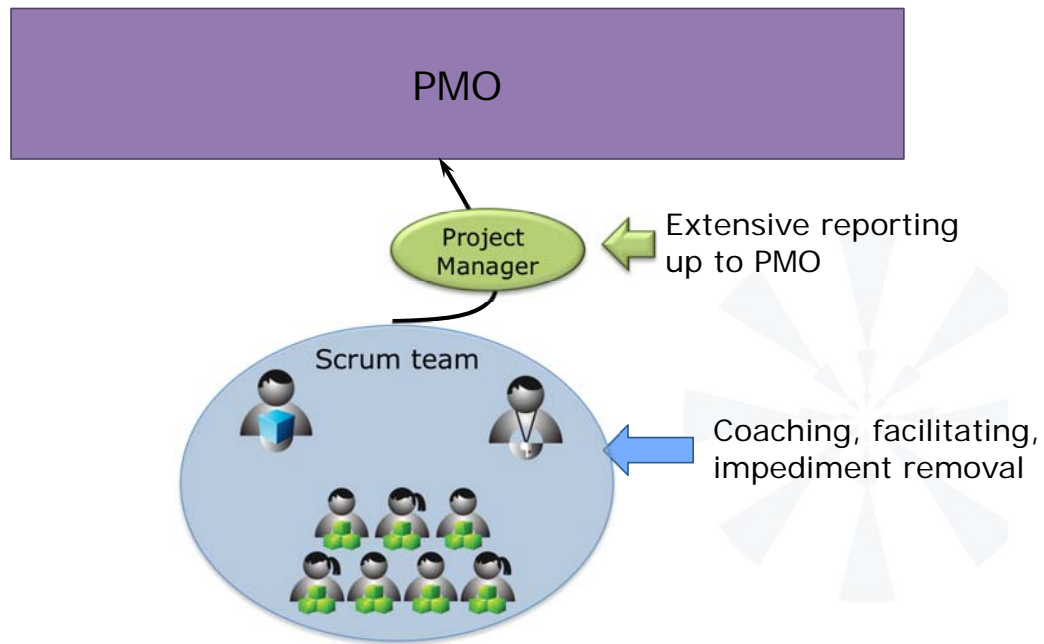
Trainer

Business Analyst

Doc

Tech Support

Release Engineer

Managers

Classic Role of Project Manager

Frequent Mapping of Responsibilities

Area	PO	SM	Team	Other Mgr
Integration	✔			?
Scope	Macro-level		Sprint-level	
Time	Macro-level		Sprint-level	
Cost	✔		Story/task estimating	
Quality	✔	✔	✔	✔
Team			✔	Formation
Communication	✔	✔	✔	✔
Risk	✔	✔	✔	✔
Procurement	✔			✔

Project Manager as Reporter

PMO

Project Manager

Extensive reporting up to PMO

Scrum team

Coaching, facilitating, impediment removal

Project Manager as Impediment Remover

Project-owning company in NYC

Project Manager

NYC-based impediment removal & sub-contract management

Sub-contractor/partner in India

Scrum Master

80% of team members in India; ScrumMasters located in India to be close to team
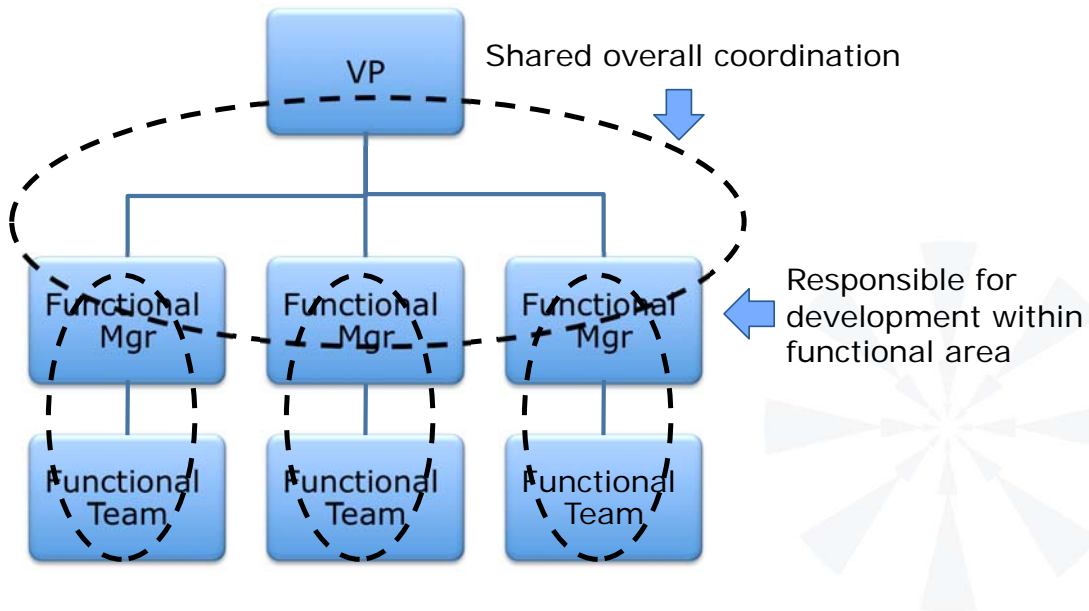
Project Manager as Logistics Coordinator
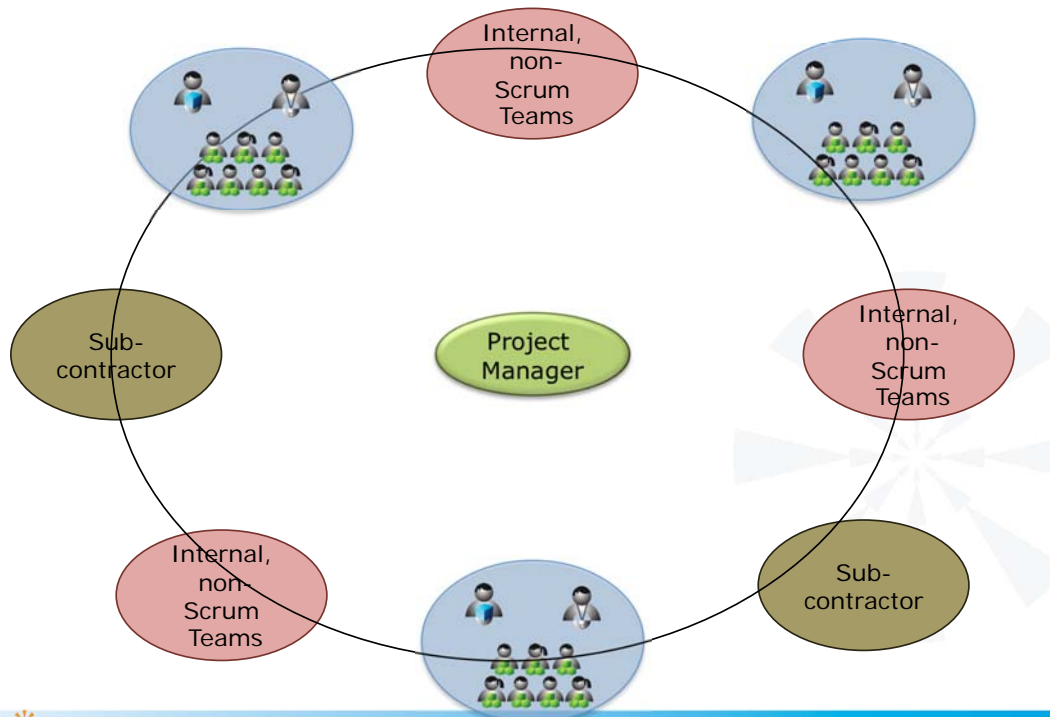
Functional Managers

✳ Manage people with similar, specialized skills

　✳ Manage professional development and career progression

　✳ Do personnel performance reviews

✳ Manages activities in a functional area

　✳ E.g., development, QA, UX, Art, etc.

✳ Aliases:

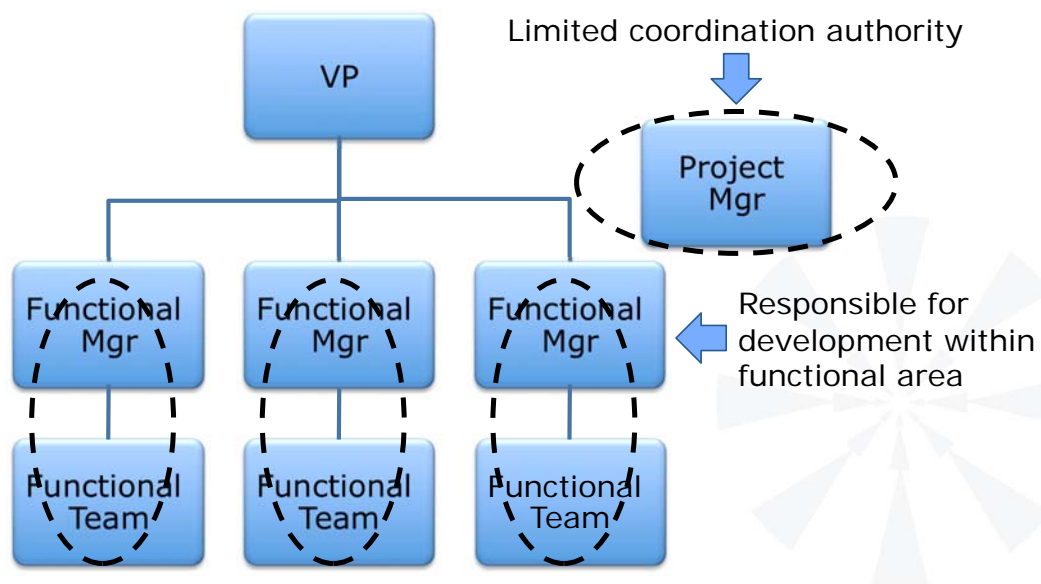　✳ Line manager, resource manager, and others

Functional Team

Shared overall coordination

Responsible for development within functional area

Functional Matrix Team

Limited coordination authority

Responsible for development within functional area

Balanced Matrix Team

VP

Oversees project

Project Mgr

Functional Mgr Functional Mgr Functional Mgr

Functional Team Functional Team Functional Team

Shared responsibility, authority and decision making

Project Matrix Team

VP

Primary responsibility & authority

Project Mgr

Functional Mgr Functional Mgr Functional Mgr

Assign personnel
Provide functional expertise

Project Team

Project Team

In charge

VP

Project Mgr

Functional Mgr

Functional Mgr

Functional Mgr

Assign personnel
No formal involvement

Project Team

Scrum Teams & Communities of Practice

Scrum Teams

Functional Managers
Community Leaders

Programmer
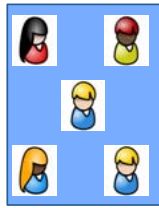Community of Practice

UI Designer
Community of Practice

Tester
Community of Practice

DBA
Community of Practice
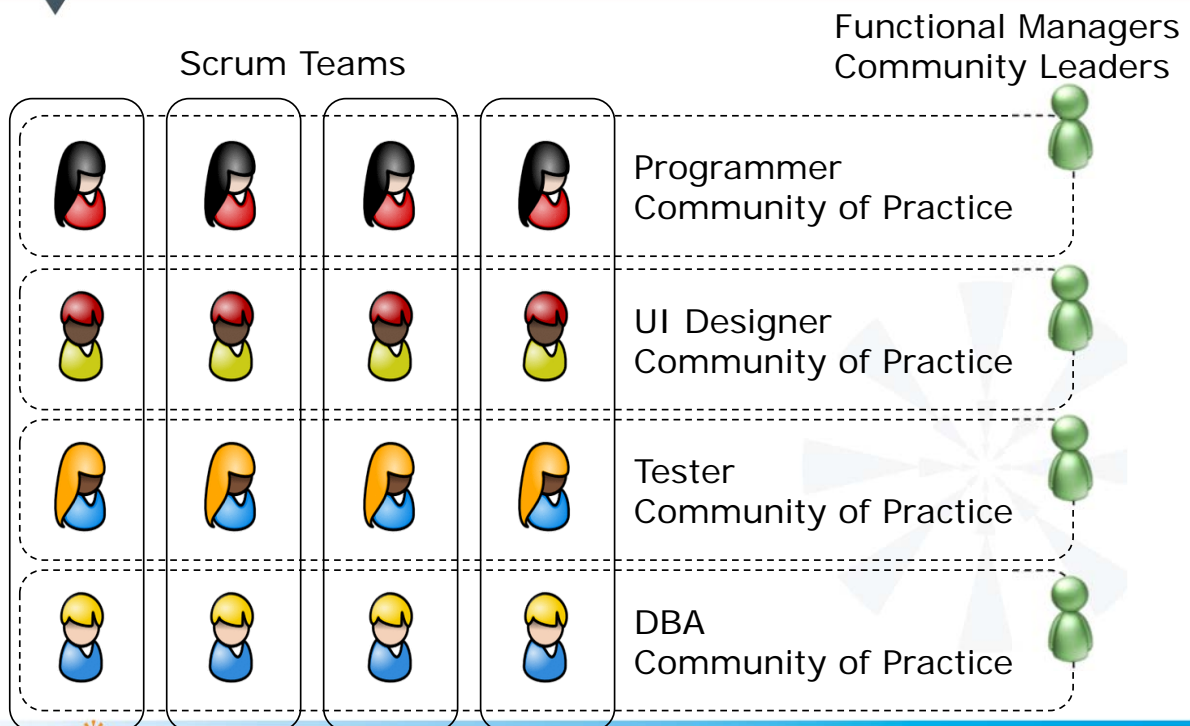
Does Self Management Mean No Managerial Control?

Managers define projects (sandboxes)

Teams self manage within their sandbox
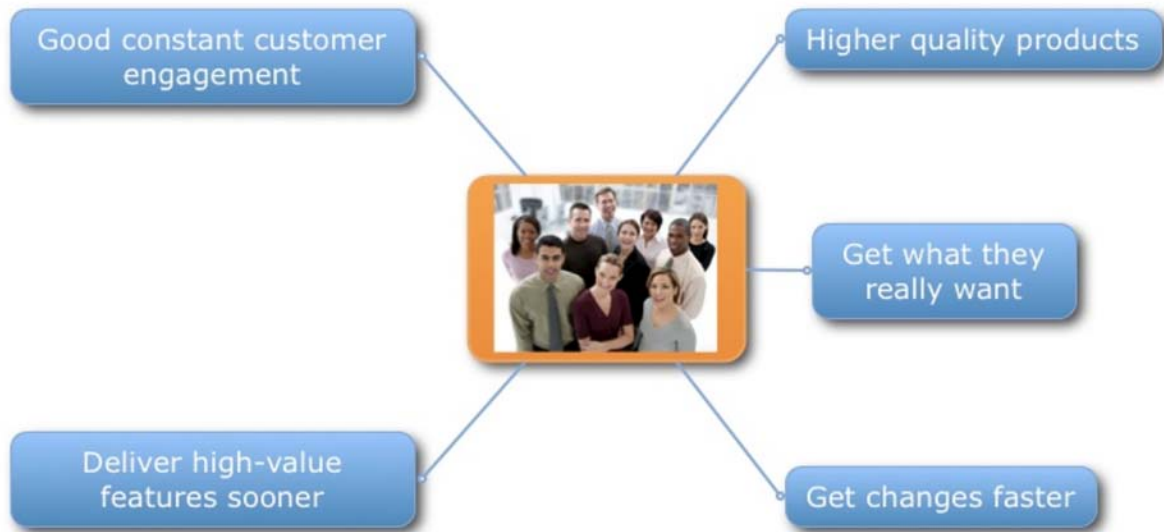
Managers define teams (who gets to play in the sandboxes)
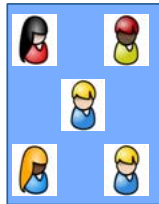
Senior/Executive Manager

- Understand scope of change and timeframe required

- Provide directional leadership

- Facilitate impediment removal

- Promote transparency

- Make the hard choices

- Membership and leadership of transition team

- Are available

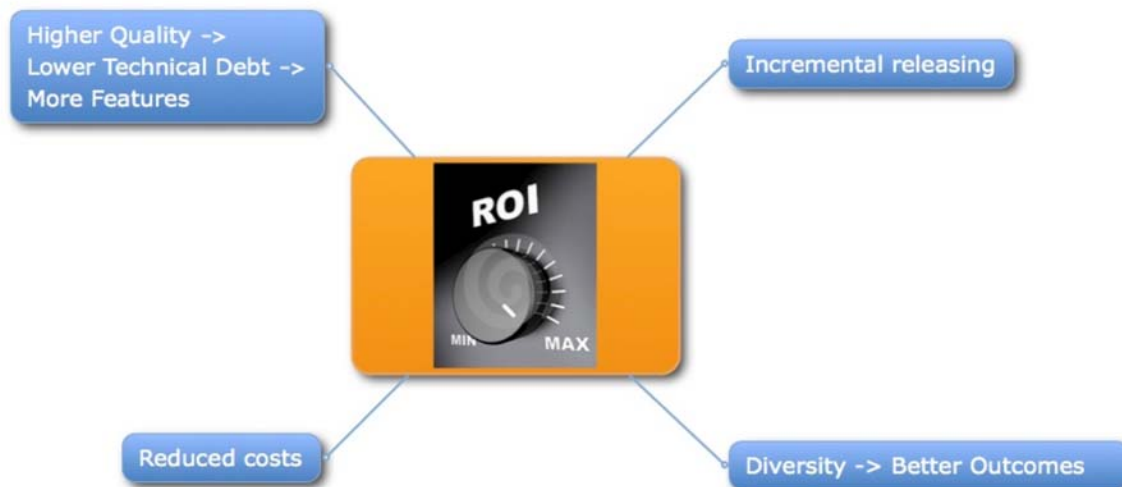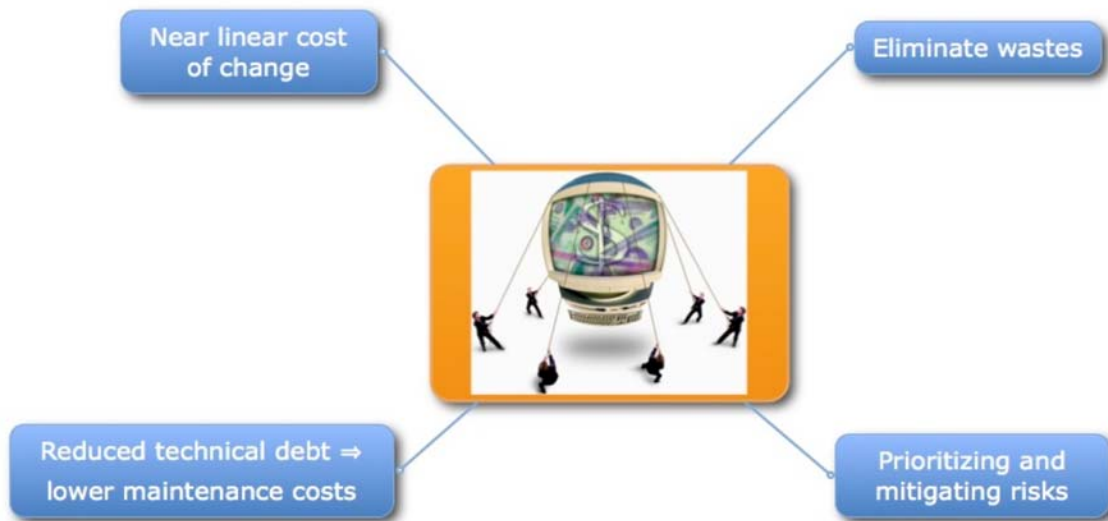Benefits

Benefits

Delighted Customers

Good constant customer engagement

Higher quality products

Get what they really want

Deliver high-value features sooner

Get changes faster

Improved ROI

Higher Quality ->
Lower Technical Debt ->
More Features

Incremental releasing

Reduced costs

Diversity -> Better Outcomes

Reduced Costs

Near linear cost of change

Eliminate wastes

Reduced technical debt ⇒ lower maintenance costs

Prioritizing and mitigating risks

49

Fast Results

All-at-once development

Minimum marketable features

Incremental innovation

50

Confidence to Thrive in a Complex World

- Visibility
- Simply development process
- Meaningful decision points
- Manage changing priorities
- Frequent and useful feedback
- Increased predictability
- Fixed-date & Fixed-scope planning

More Joy!

- Working at a sustainable pace
- Excitement & accomplishment
- Improved team member relationships
- Empowerment
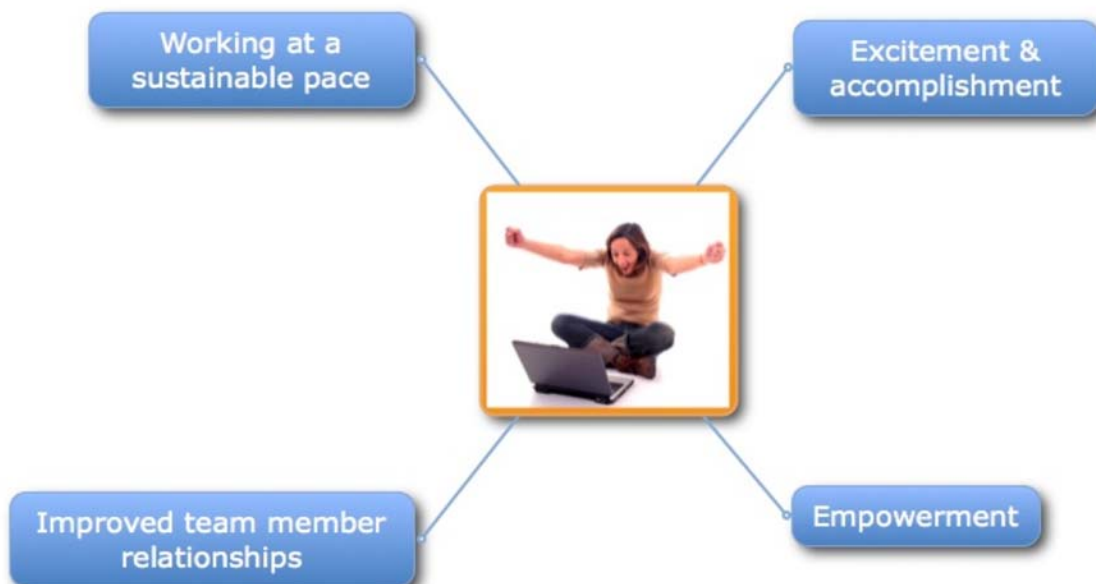
REASONS FOR ADOPTING AGILE

The main reasons cited by respondents as to why they decided to adopt Agile practices hinged on overall process flexibility, productivity, and quality. Other aspects like reducing cost were cited less often.
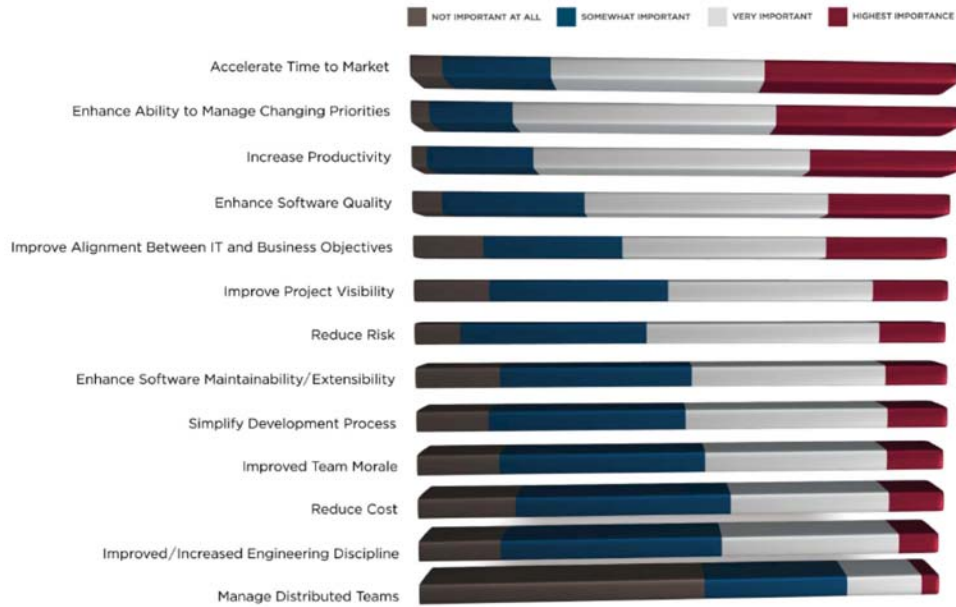
	NOT IMPORTANT AT ALL	SOMEWHAT IMPORTANT	VERY IMPORTANT	HIGHEST IMPORTANCE

- Accelerate Time to Market
- Enhance Ability to Manage Changing Priorities
- Increase Productivity
- Enhance Software Quality
- Improve Alignment Between IT and Business Objectives
- Improve Project Visibility
- Reduce Risk
- Enhance Software Maintainability/Extensibility
- Simplify Development Process
- Improved Team Morale
- Reduce Cost
- Improved/Increased Engineering Discipline
- Manage Distributed Teams

Version One. (2009). *The state of agile development*: *Fourth Annual Survey*. Alpharetta, GA: Author.

Industry Results

⁂ Comparative Agility Industry Data

| | -2 Std | | -1 Std | ■ | Mean | | +1 Std | | +2 Std |
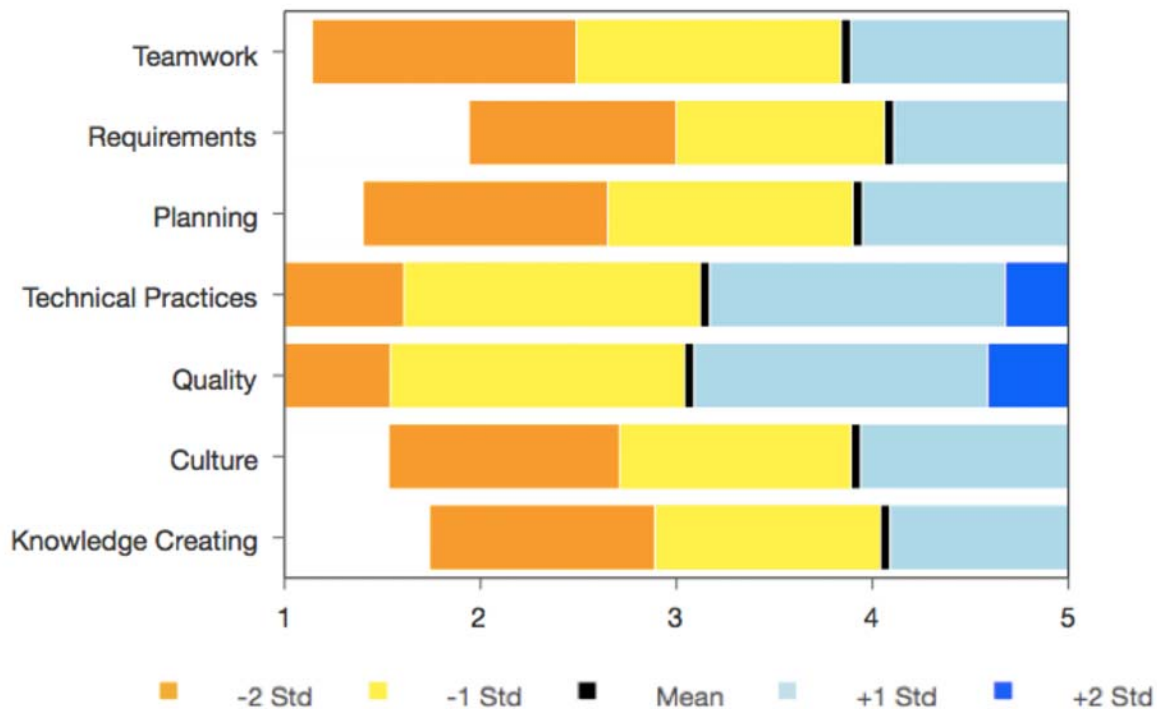
⁂ Industry Study –
Agile Success Rate

What percentage of Agile projects have been successful from the perspective of your organization?
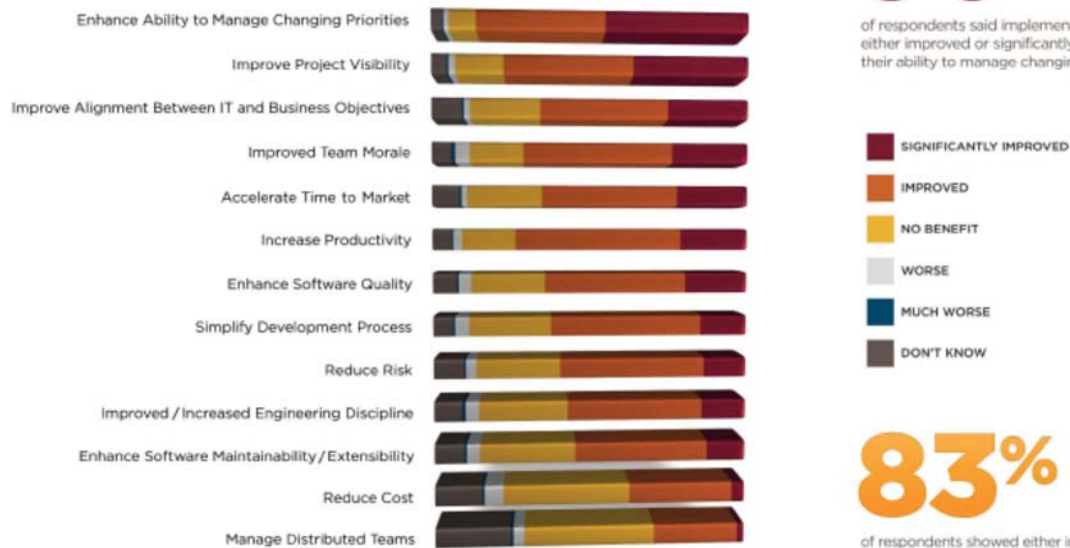*(choose closest answer)*

0% Projects: 5%
10% Projects: 4%
25% Projects: 4%
50% Projects: 11%
75% Projects: 21%
90-100% Projects: 55%
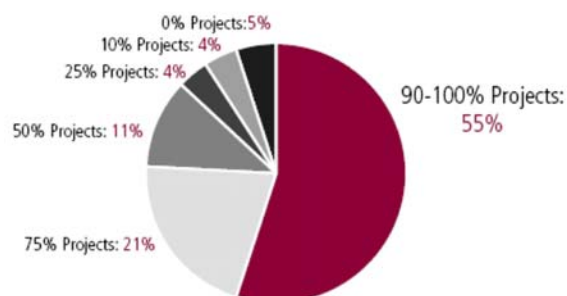
Successful Projects	Respondents
0%	4.8%
10%	3.7%
25%	3.8%
50%	11.5%
75%	21.2%
90% - I cant say 100% but pretty close.	37.6%
100% - I cant remember a project using Agile methods that wasn't considered successful.	17.4%

Version One. (2008). *The state of agile development*: *Third Annual Survey*. Alpharetta, GA: Author.

56

☀ Industry Study – Agile Benefits

BENEFITS OBTAINED FROM IMPLEMENTING AGILE

Respondents cited increased flexibility and productivity as the strongest gains won by a shift to Agile.

Enhance Ability to Manage Changing Priorities
Improve Project Visibility
Improve Alignment Between IT and Business Objectives
Improved Team Morale
Accelerate Time to Market
Increase Productivity
Enhance Software Quality
Simplify Development Process
Reduce Risk
Improved / Increased Engineering Discipline
Enhance Software Maintainability / Extensibility
Reduce Cost
Manage Distributed Teams

90%
of respondents said implementing Agile either improved or significantly improved their ability to manage changing priorities

- ■ SIGNIFICANTLY IMPROVED
- ■ IMPROVED
- ■ NO BENEFIT
- ■ WORSE
- ■ MUCH WORSE
- ■ DON'T KNOW

83%
of respondents showed either improvement or significant improvement in project visibility

Version One. (2009). *The state of agile development*: *Fourth Annual Survey*. Alpharetta, GA: Author.

☀ Industry Study – Agile Benefits

☀ Survey of 400 international respondents

Cost
Alignment
Quality
Time to Market

0% 10% 20% 30% 40% 50%

Improvement
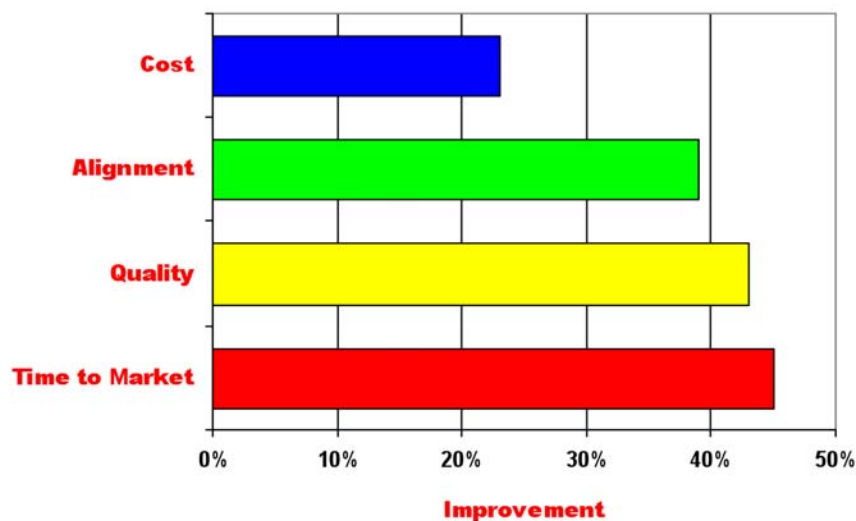
Barnett, L. (2006). And the agile survey says. *Agile Journal, 1*(1).

Industry Study – Agile Benefits

☀ Survey of 642 international respondents

☀ 62% were from firms with less than 1,000 people
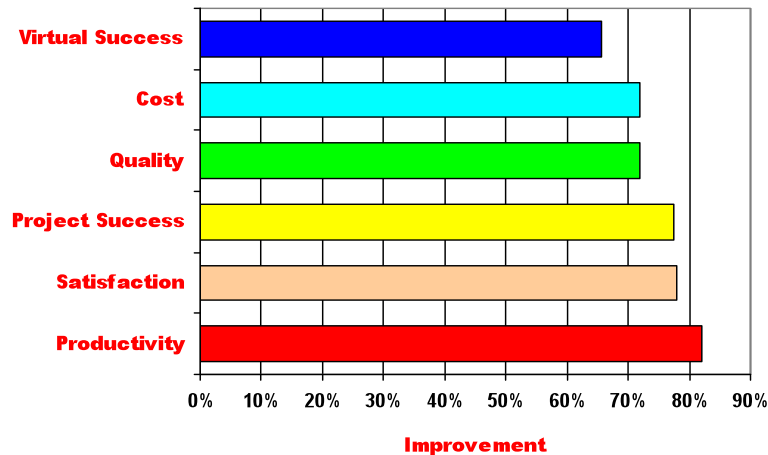
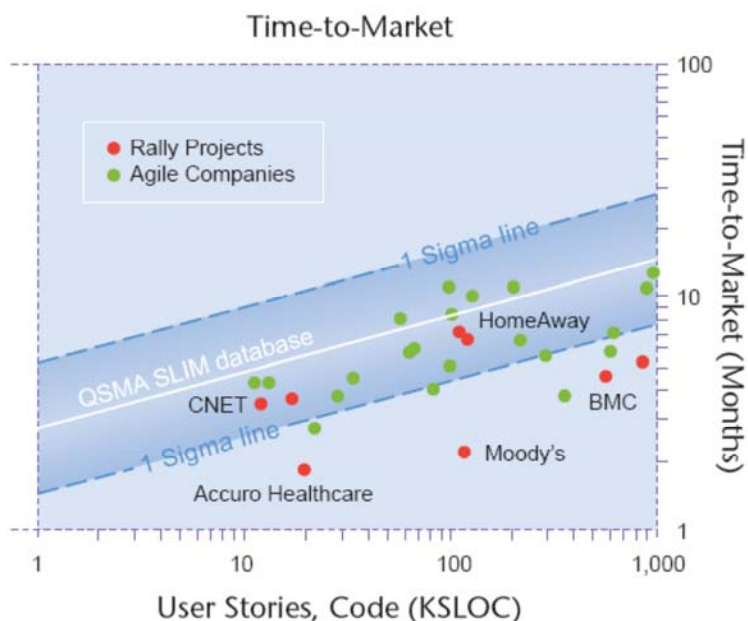Ambler, S. W. (2008). *Agile adoption survey*. Retrieved October 17, 2008, from http://www.ambysoft.com/downloads/surveys/AgileAdoption2008.ppt

Industry Study – Speed to Market

☀ Agile projects are 37% faster to market

☀ Some agile projects are 50% faster
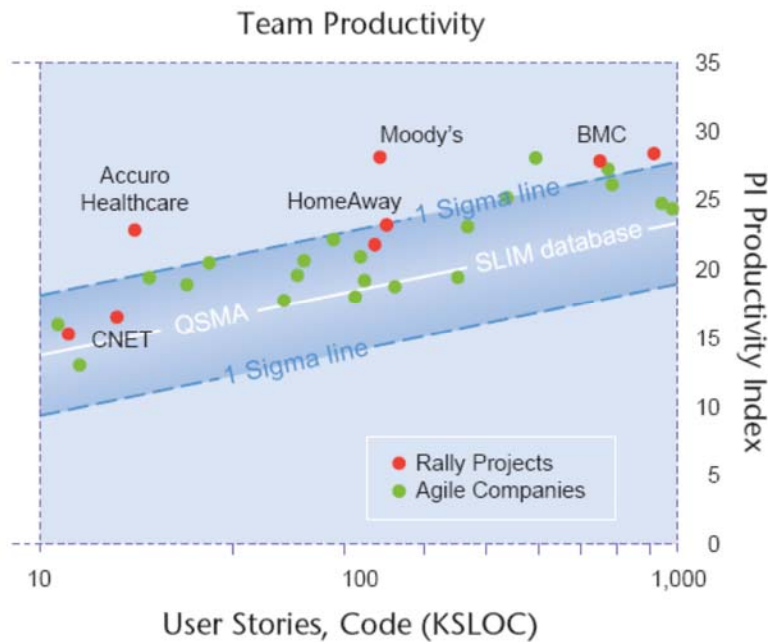
☀ Compared 26 Agile projects to 7,500 non-Agile projects

The Agile Impact Report: Proven Performance Metrics from the Agile Enterprise, QSMA Report, 2008.

Industry Study – Productivity Increase

Team Productivity

* Average 16% increase in productivity
* Some had a 25% increase

The Agile Impact Report: Proven Performance Metrics from the Agile Enterprise, QSMA Report, 2008.

Biotech Company Results

	Waterfall	Scrum
Use Case pages	3,000	
User Stories		1,400
Calendar months	9	12
Person months	540	54
Lines of Java code	58,000	51,000
Lines of Java code per person-month	120	840

Salesforce.com –
2006 Delivery Results
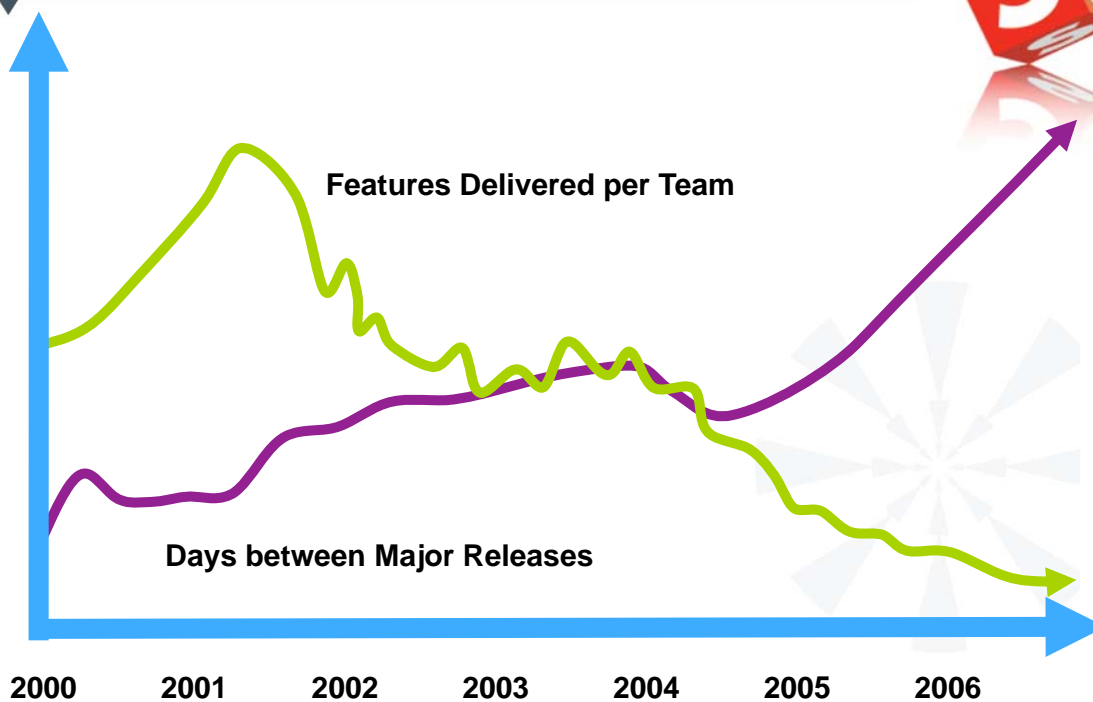
Features Delivered per Team

Days between Major Releases

2000 2001 2002 2003 2004 2005 2006

63

Salesforce.com –
2007 Transformation Results
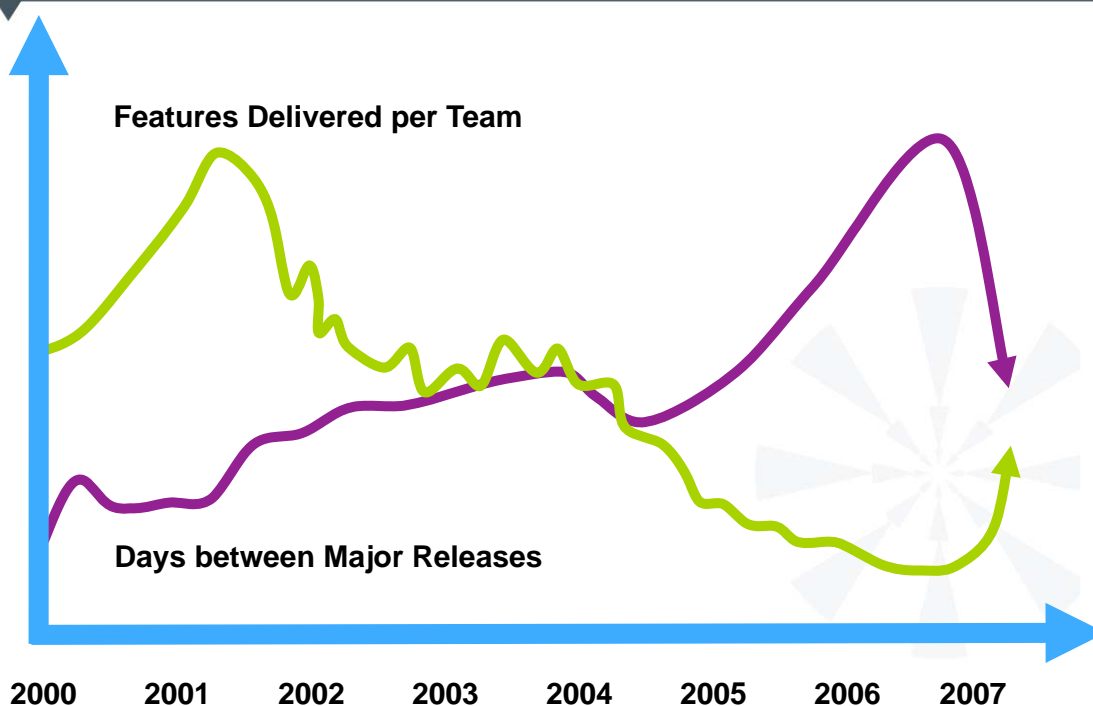
Features Delivered per Team

Days between Major Releases

2000 2001 2002 2003 2004 2005 2006 2007

64

Pitfalls / CSF

☀ Believing Dilbert!

© Scott Adams, Inc./Dist. by UFS, Inc.
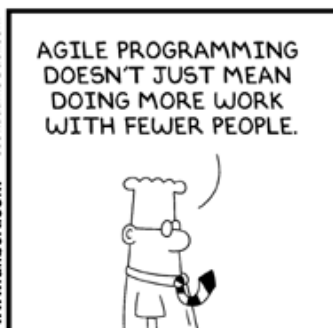
Be Aware

- Some developers & managers will not like it

- It may get harder in months 3 to 6

- Requires intense customer (product owner) involvement to "build the right system"

- Requires intense developer involvement to "build the system right"

- May require change in compensation policy

- Management change from command to leadership

- Adding more people will not likely be an answer

Additional Issues with Agile

- Agile is simple, but it can cause a lot of changes

- Need to be willing to work in time boxes (might not feel natural)

- Engineering, design and testing practices need to become more Agile

- Interactions and communication need to become clearer, more personal and more transparent

- Honesty, trust and a sense of commitment need to be developed

- Agile "exposes" issues quickly and effectively, but it doesn't "solve" them

- When the going gets tough it is easy to slip back into the old way of doing things.

- Courage is essential

Will You Change or Will You Change Scrum?

- Following Scrum out of the book isn't enough; a change in thought process is required and is hard

- Be ready for a major change that must be managed carefully. It can take years in larger organizations

- Don't modify Scrum; let Scrum point out the dysfunctionalities that need to change

- Most of the effort isn't in implementing Scrum but rather removing the dysfunctionalities it points out

- Change will occur person by person

Agilephobia

noun.
strong fear or dislike of agile, usually due to the uncertainty of change

- I'm afraid I'll have nothing to do
- I'm afraid I'll lose my job
- I'm afraid people will see how little I actually do
- I'm afraid I won't be able to keep up
- I'm afraid I won't be able to learn the new software
- I'm afraid this will mean hard work
- I'm afraid I'll be fired if the decisions we make don't work out
- I'm afraid of conflict and trying to reach consensus
- It's just so much easier and safer when someone else tells me exactly what to do

☀ Overcoming Agilephobias

- ☀ Being aware of what people fear is half the challenge

- ☀ Acknowledge these agilephobias and over time work to have solutions for them

- ☀ Over time people will lose their fear of losing their job

  - ☀ But will likely still need revised job descriptions in an agile enterprise

☀ Industry Study – Greatest Concerns About Agile

GREATEST CONCERNS ABOUT ADOPTING AGILE

Corporate concerns about the adoption of Agile methodologies were most commonly centered around its implementation. Respondents said their companies were concerned about the lack of upfront planning (46%) or the lack of documentation (34%).

- Lack of upfront planning
- Lack of documentation
- Loss of management control
- Lack of predictability
- Management opposed to change
- Lack of engineering discipline
- Development team opposed to change
- Inability to scale
- Regulatory compliance
- Reduced software quality
- Quality of engineering talent
- No Concerns
- Other

Version One. (2009). *The state of agile development*: *Fourth Annual Survey*. Alpharetta, GA: Author.

BARRIERS TO FURTHER ADOPTION OF AGILE

Management opposition and lack of upfront planning were the most commonly cited barriers to further adoption of Agile within companies.
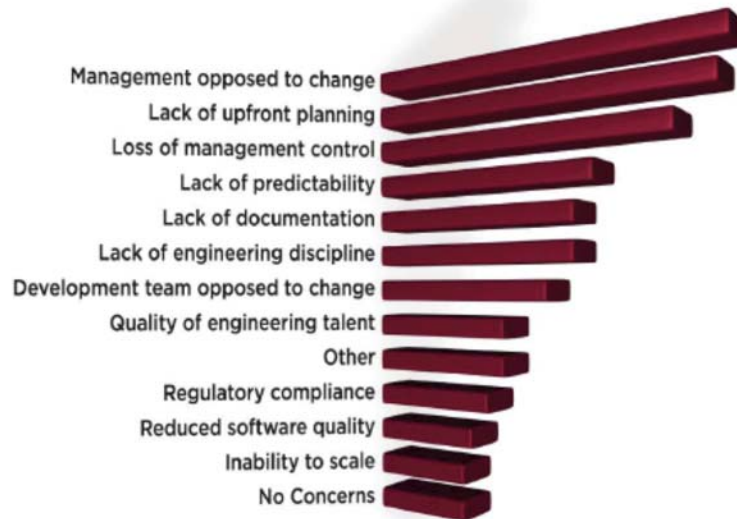
- Management opposed to change
- Lack of upfront planning
- Loss of management control
- Lack of predictability
- Lack of documentation
- Lack of engineering discipline
- Development team opposed to change
- Quality of engineering talent
- Other
- Regulatory compliance
- Reduced software quality
- Inability to scale
- No Concerns

Version One. (2009). *The state of agile development*: *Fourth Annual Survey*. Alpharetta, GA: Author.

Contact Info for Kenny Rubin

innolution
agile innovative solutions

Email:	krubin@innolution.com
Website:	www.innolution.com
Phone:	(303) 827-3333
LinkedIn:	www.linkedin.com/in/kennethrubin
Twitter:	www.twitter.com/krubinagile
Scrum: A Manager's Guide Book Chapters	tinyurl.com/yj4m72c
Comparative Agility Website	www.comparativeagility.com