# Enterprise Agile

Aashish Vaidya
August, 2016
Twitter: @aashishvaidya  Linkedin: www.linkedin.com/pub/aashish-vaidya/2/13/168/

# 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

*That is, while there is value in the items on the right, we value the items on the left more.*

| | | |
|---|---|---|
| Kent Beck | James Grenning | Robert C. Martin |
| Mike Beedle | Jim Highsmith | Steve Mellor |
| Arie van Bennekum | Andrew Hunt | Ken Schwaber |
| Alistair Cockburn | Ron Jeffries | Jeff Sutherland |
| Ward Cunningham | Jon Kern | Dave Thomas |
| Martin Fowler | Brian Marick | |

# Principles behind the Agile Manifesto

- **Our highest priority is to satisfy the customer** through early and continuous delivery of valuable software.

- **Welcome changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage.

- **Deliver working software frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

- **Business people and developers must work together daily** throughout the project.

- **Build projects around motivated individuals**.  Give them the environment and support they need, and trust them to get the job done.

- The most efficient and effective method of conveying information to and within a development  team **is face-to-face conversation**.

- **Working software is the primary measure of progress.**

- Agile processes promote sustainable development. The sponsors, developers, and users should be **able to maintain a constant pace indefinitely**.

- Continuous attention to **technical excellence and good design enhances agility**.

- **Simplicity**--the art of maximizing the amount of work not done--is essential.

- The best architectures, requirements, and designs emerge from **self-organizing teams**.

- At regular intervals, the **team reflects on how to become more effective**, then tunes and adjusts its behavior accordingly.
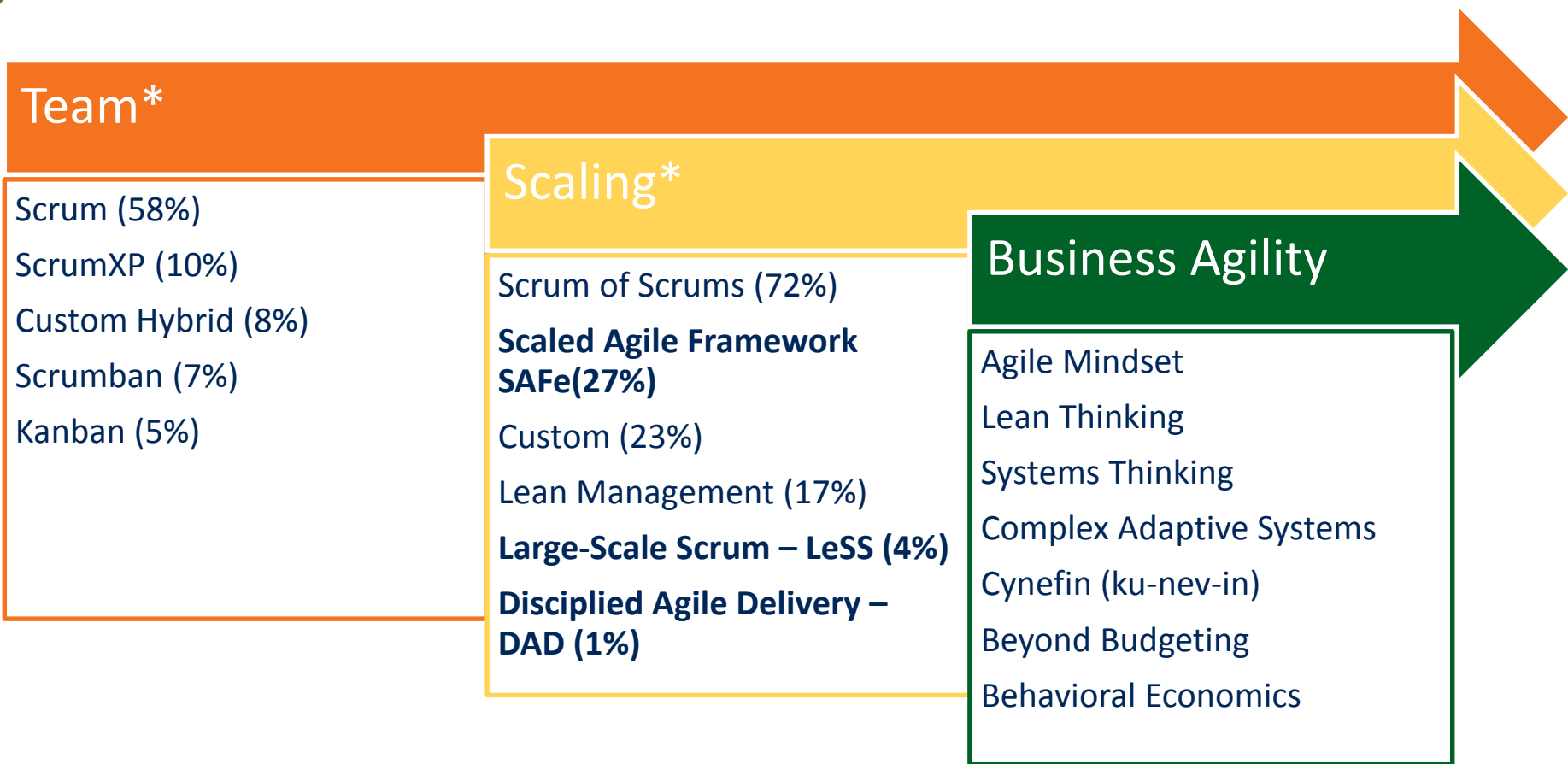
# Agile Evolution

Team → Scaling → Business Agility

# Enterprise Scaling has Two Aspects

Scaling — Across large # of teams

Scaling — Across divisions

# Agile Evolution

**Team***
- Scrum (58%)
- ScrumXP (10%)
- Custom Hybrid (8%)
- Scrumban (7%)
- Kanban (5%)

**Scaling***
- Scrum of Scrums (72%)
- **Scaled Agile Framework SAFe(27%)**
- Custom (23%)
- Lean Management (17%)
- **Large-Scale Scrum – LeSS (4%)**
- **Disciplined Agile Delivery – DAD (1%)**

**Business Agility**
- Agile Mindset
- Lean Thinking
- Systems Thinking
- Complex Adaptive Systems
- Cynefin (ku-nev-in)
- Beyond Budgeting
- Behavioral Economics

*Note: Source VersionOne -10th Annual State of Agile Report

# Scrum

# Scrum Practices Prescribed by the Scrum Alliance

| 3 Roles | 4 Artifacts | 5 Activities |
|---------|-------------|--------------|
| • Product Owner<br>• Scrum Master<br>• The Team | • Product Backlog<br>• Sprint Backlog<br>• Product Increment<br>• Definition of Done | • Backlog Refinement<br>• Sprint Planning<br>• Daily Scrum<br>• Sprint Reviews (Demos)<br>• Sprint Retrospectives |

# SCALED AGILE FRAMEWORK (SAFe)

# Quick Hits on SAFe

- Four-tier structure: Portfolio, Valuestream, Program, and Team

- Appropriate transition model if moving from traditional program-project management to Agile

- Teams based on ScrumXP

- Program/Project scale 50-150

- Dean Leffingwell – creator, promoter of the framework

- Website: scaledagileframework.com

SAFe® 4.0 for Lean Software and Systems Engineering
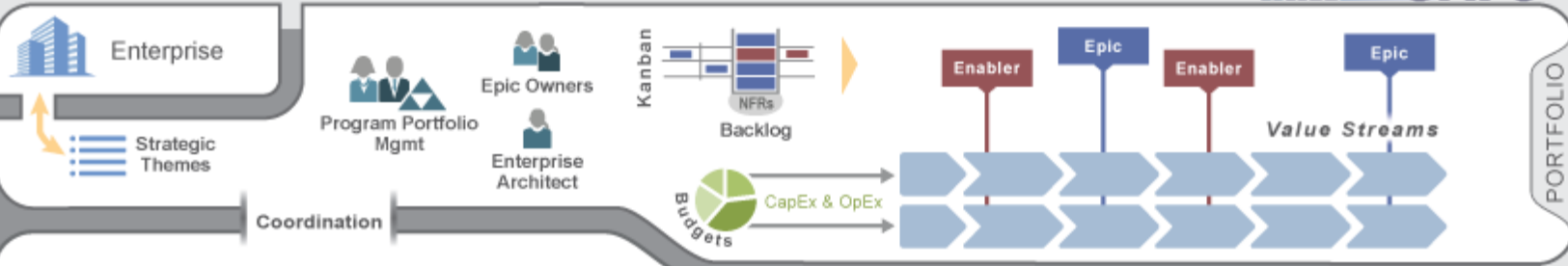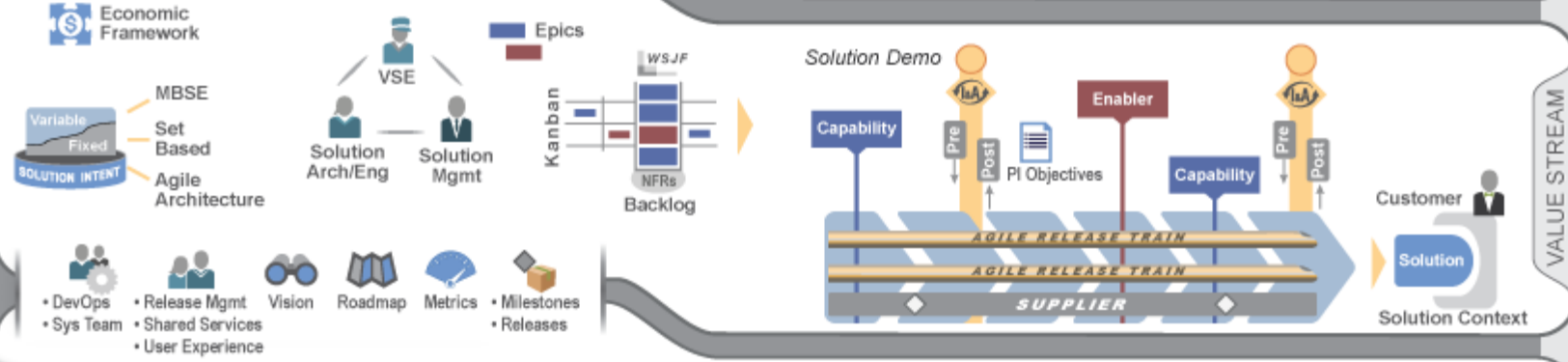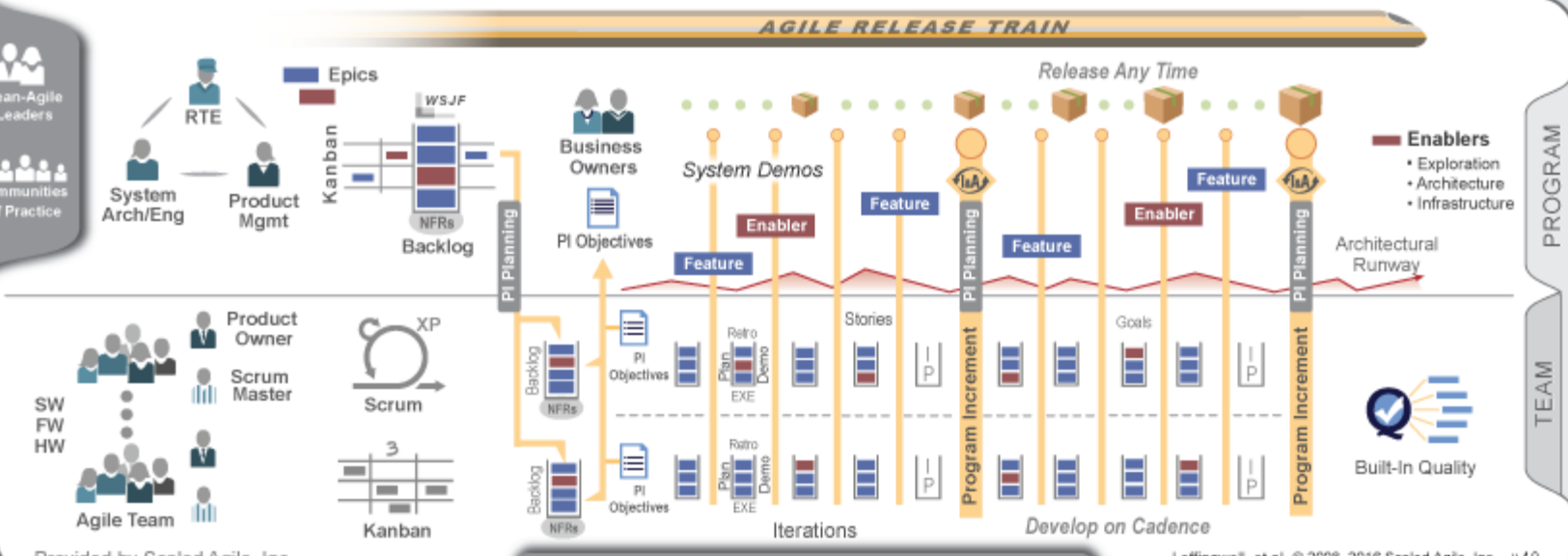
# LARGE SCALE SCRUM (LeSS)

# Quick Hits on LeSS

- Product-centric view
- LeSS: 2-8 teams (10-70 people)
- LeSS Huge: 8+ teams (70+ people)
- Each LeSS "cluster" has single product (or project) focus
- Uses the basic concept of Scrum of Scrum and extends it for other activities
- Craig Larman & Bas Vodde – creators and promoters of the framework
- Website: https://less.works
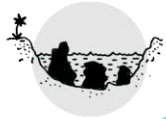
# Large Scale Scrum
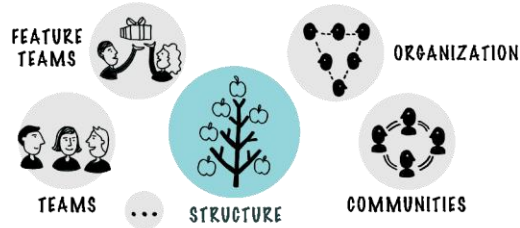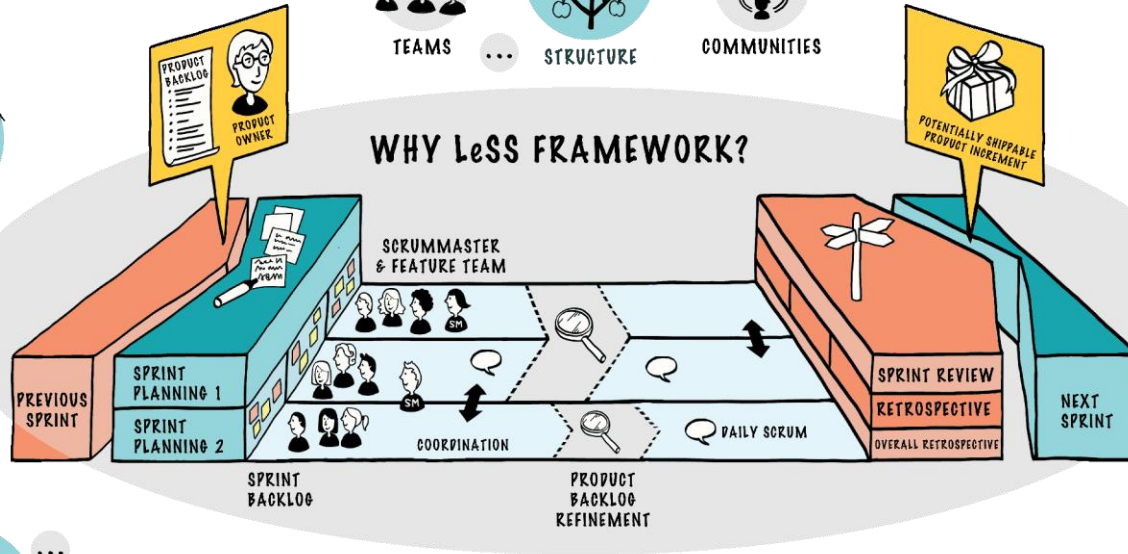
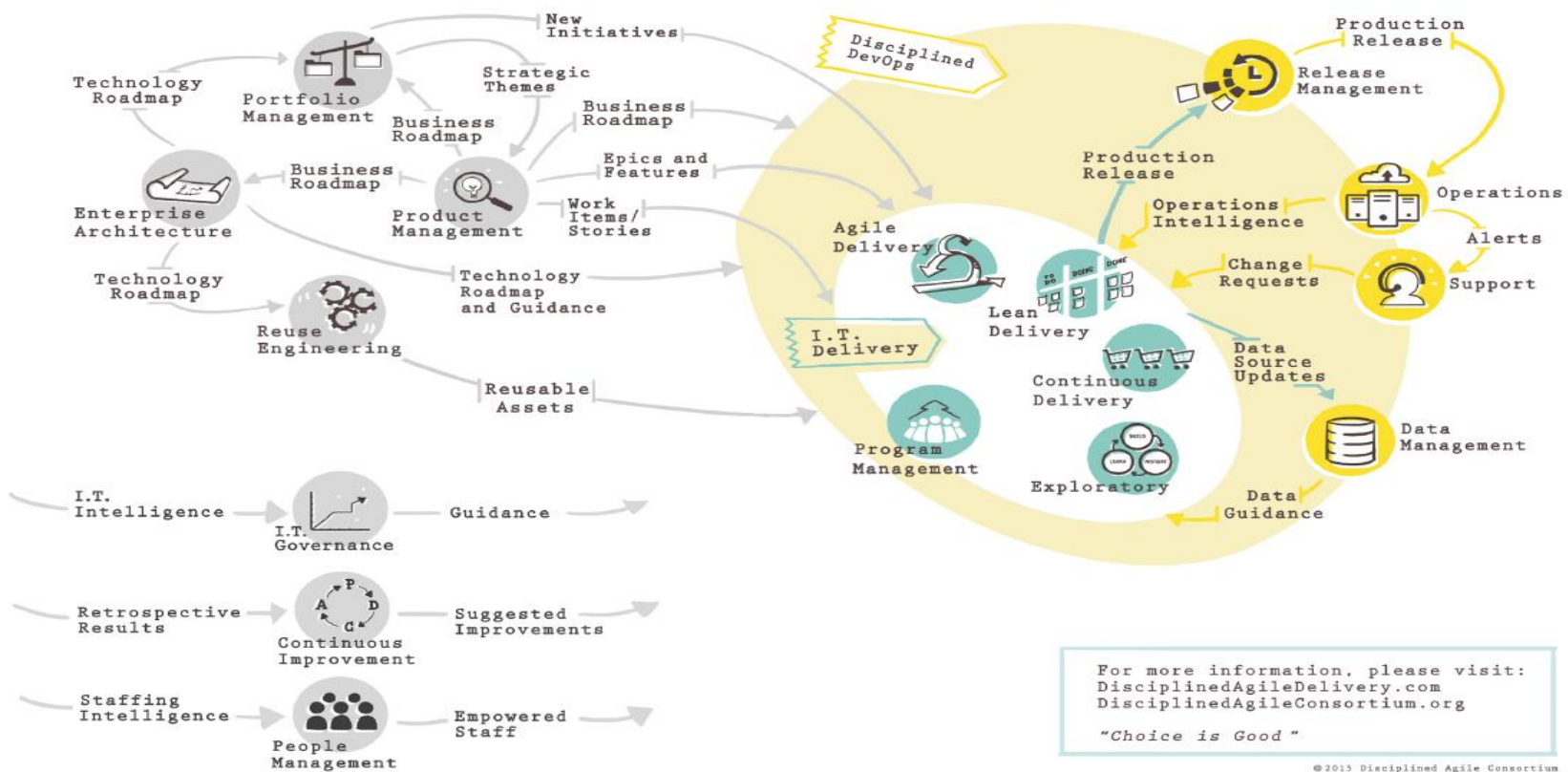# DISCIPLINED AGILE 2.0
## (FORMERLY KNOWN AS DAD)

# Quick Hits on DA2.0

- More program perspective
- Basic construct for a program/project: Inception, Construction, Transition
- Employs 4 different lifecycles
- Transitional model
- Scott Ambler – creator and promoter of the framework
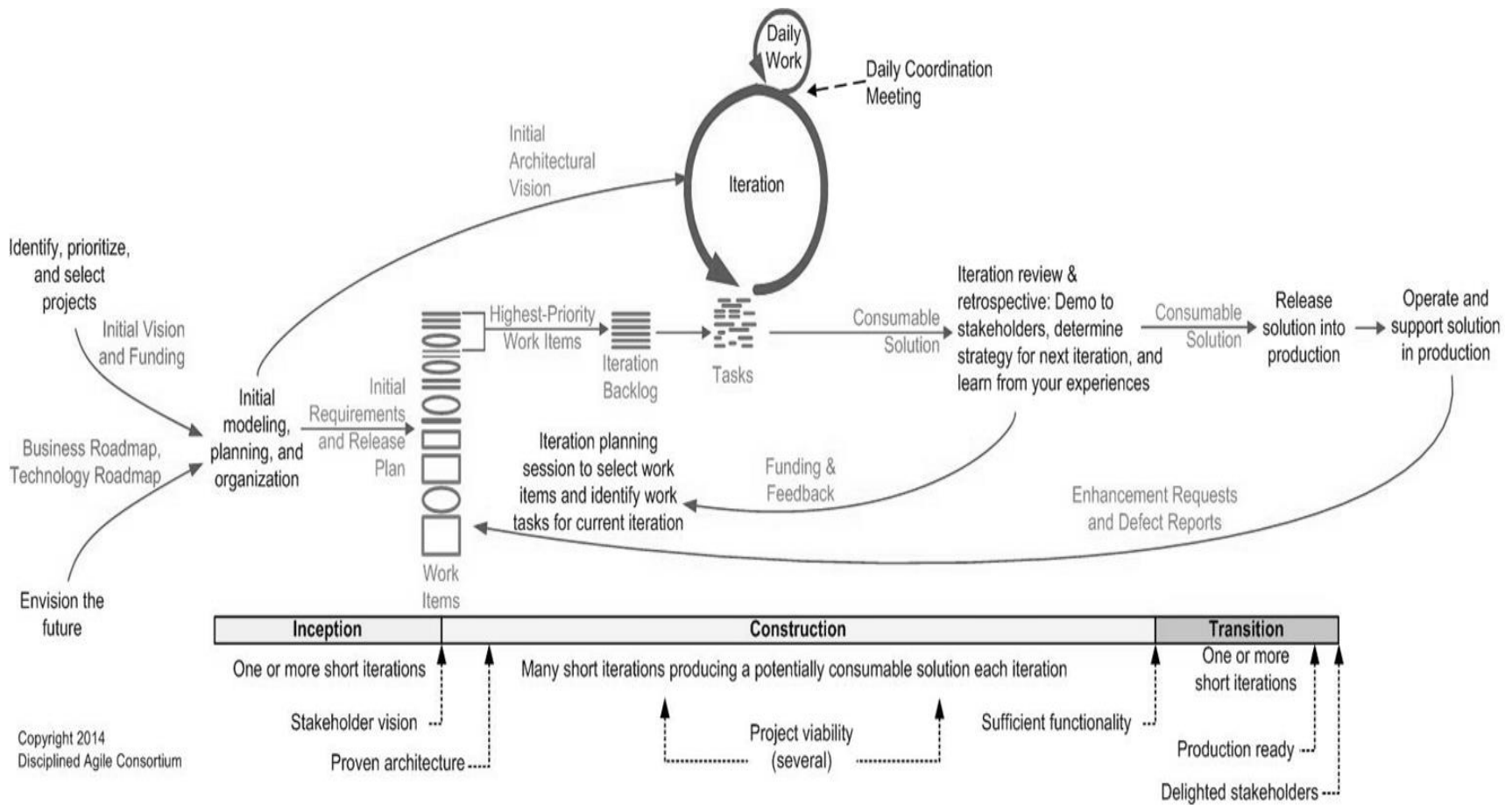- Website: http://www.disciplinedagiledelivery.com/

# Disciplined Agile 2.0

# Disciplined Agile Delivery (DAD)

- Agile/Basic Lifecycle (Scrum-like)

# Disciplined Agile Delivery (DAD)
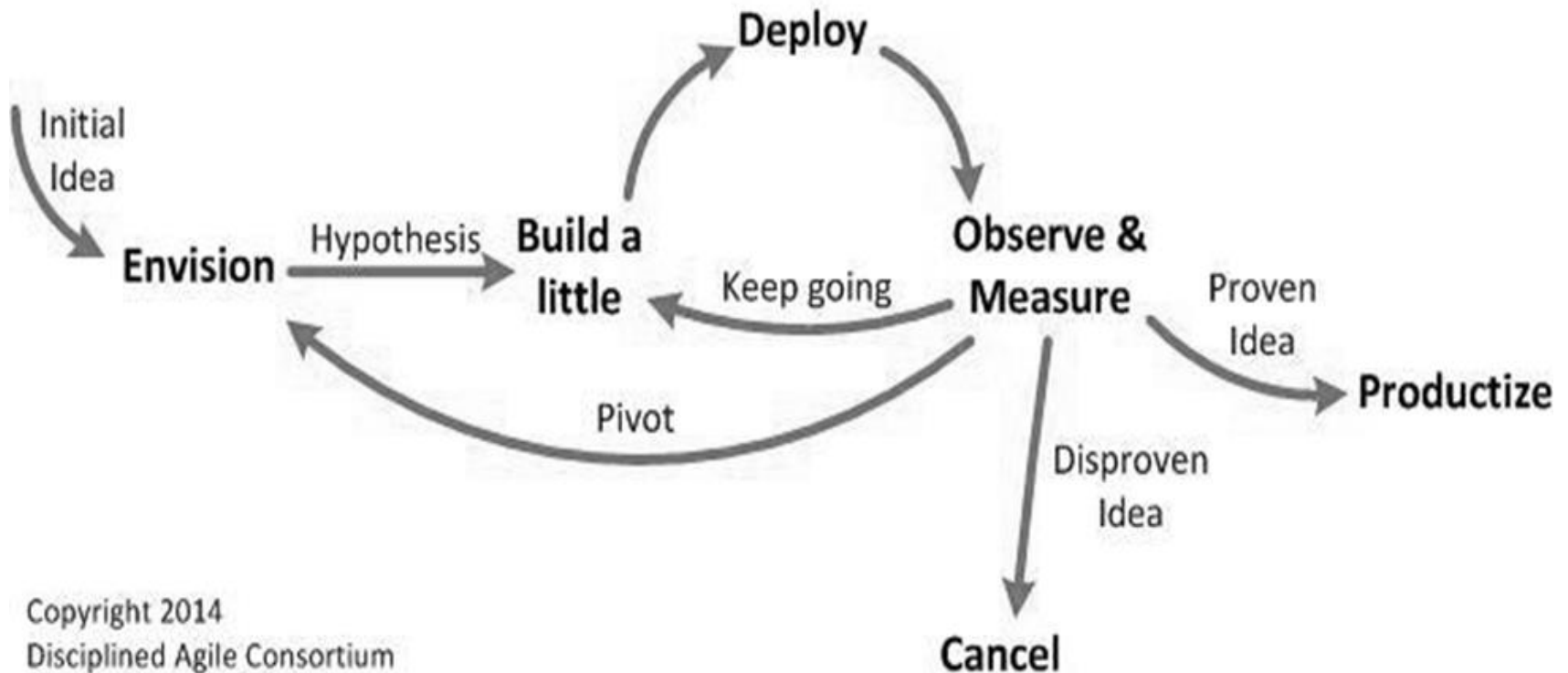
- Advanced Lean Lifecycle (Kanban-like)

# Disciplined Agile Delivery (DAD)

• Continuous Delivery Lifecycle (purer-Kanban)

# Disciplined Agile Delivery (DAD)

• Exploratory Lifecycle (Lean-Startup)
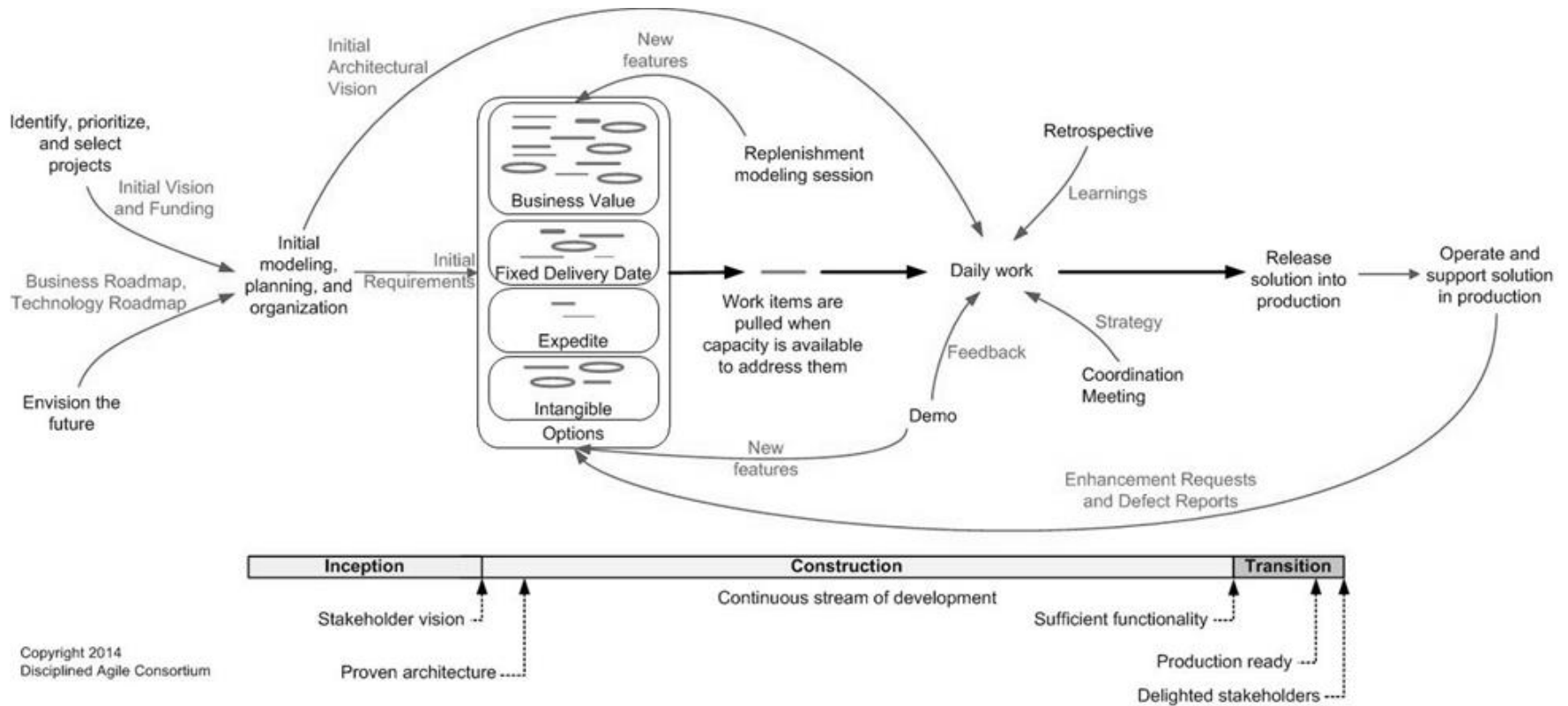


Copyright 2014
Disciplined Agile Consortium

# COMPARE AND CONTRAST

# Team Makeup Compare and Contrast

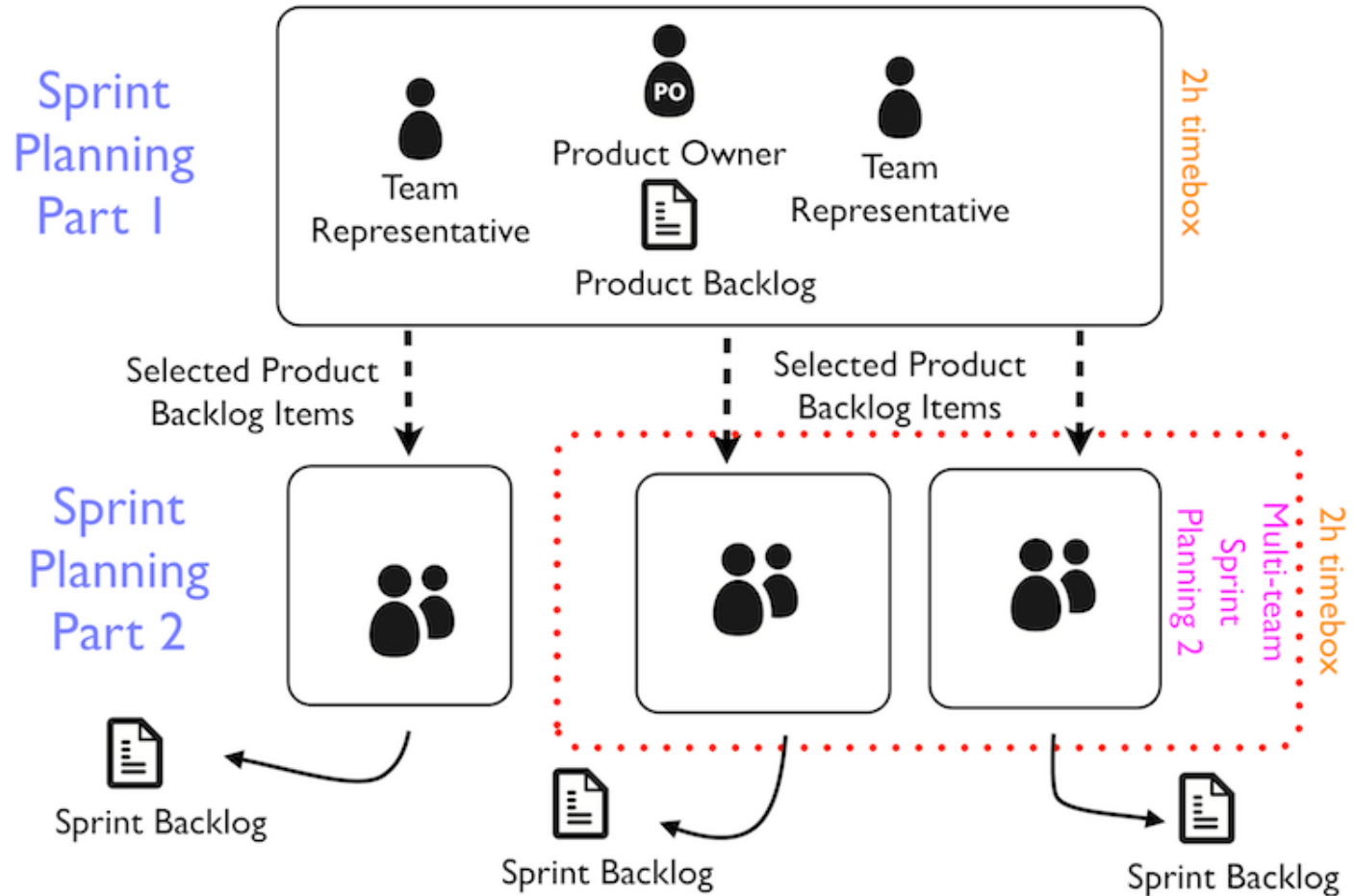| | SCRUM | SAFe | LeSS | DA 2.0 |
|---|---|---|---|---|
| Roles | PO<br><br>SM<br><br>Team Members | PO (shared up to 2 teams)<br>SM (shared up to 2 teams)<br>Team Members | LeSS: PO (shared 2-8 teams)<br>LeSS Huge: PO + Area POs<br><br>SM (1-3 teams)<br><br>Team Members | Primary Roles: PO, Team Lead, Team Member, Architecture Owner, Stakeholder<br>Secondary Roles : Specialists (brought in temporary basis) |
| Team Size | 7±2 | 7±2 | 7±2 | Small: 2 to 15<br>Medium: 25-30<br>Large: 35+ (Project or Product Delivery team) |
| Team Type | Feature (cross-functional) | Component or Feature | Feature (cross-functional) | Component, Feature, or Functional |

# Team Makeup Compare and Contrast

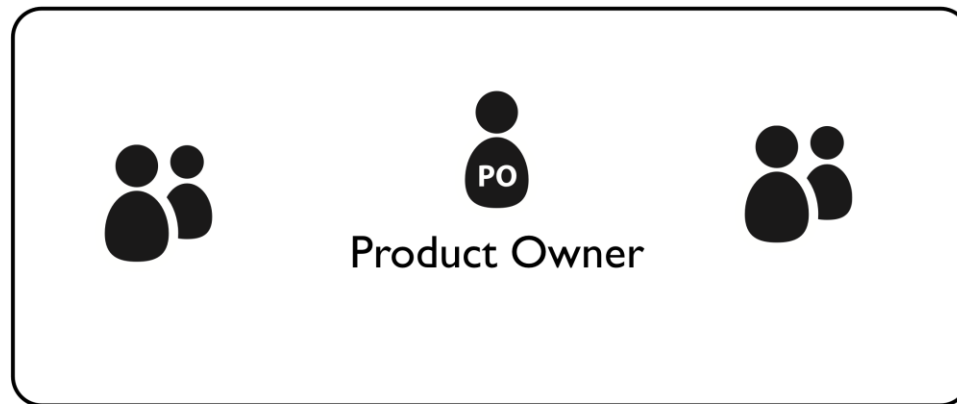|  | SCRUM | SAFe | LeSS | DA 2.0 |
|---|---|---|---|---|
| Team Method | Scrum | ScrumXP or Kanban | Scrum | Depends on Delivery Lifecycle chose |
| Project Scaling | Add teams | Add teams | Add teams | Grow teams |
| Project Size | 5-varies | 50-150 (5-15 teams) | LeSS: 10-70 LeSS Huge: 50+ |  |

# Team Level Practices Compare and Contrast

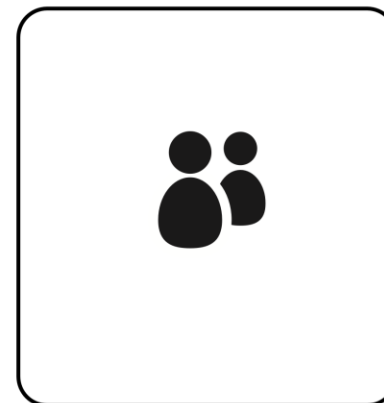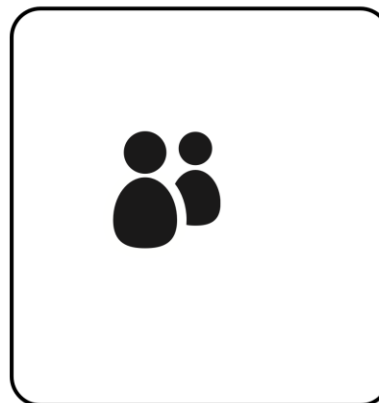| | SCRUM | SAFe | LeSS | DA 2.0 |
|---|---|---|---|---|
| Mid-Horizon Planning | Not Specified | Once every 3 months, whole team, whole project | Continual Planning | Grow Teams |
| Iteration Planning | Whole Team | Whole Team | See Diagram | Depends on Lifecycle chosen |
| Retrospectives | Whole Team | Whole Team | See Diagram | Depends on Lifecycle chosen |
| Sprint Review | Whole Team, Stakeholders | System wide demos, All Teams, Stakeholders | See Diagram | Depends on Lifecycle chosen |
| Backlog Refinement | Whole Team | Whole Team | See Diagram | Depends on Lifecycle chosen |

# LeSS Sprint Planning

# Reviews & Retrospective
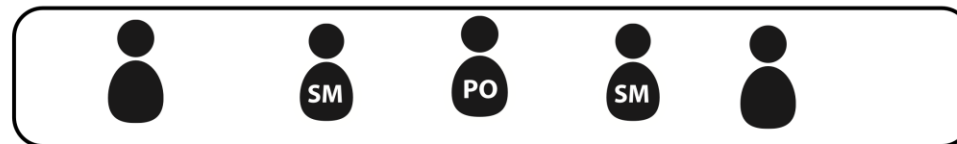


Sprint Review

Product Owner

2h timebox

Retrospective

1.5h timebox

Overall Retrospective

# Backlog Refinement

# Team Level Practices Compare and Contrast

| | SCRUM | SAFe | LeSS | DA 2.0 |
|---|---|---|---|---|
| Iteration Cadence | Any | Synchronized for all Teams | Synchronized for all teams | Depends on Lifecycle chosen |
| Iteration Duration | 1-4 weeks | 2 weeks x 6 + 1 week IP iteration | 2 weeks (but could be 1-4 weeks) | Depends on Lifecycle chosen |
| Coordination and Alignment | Scrum Of Scrums | Release Planning Meeting, Program Team | • "Just talk"<br>• Through code (CI)<br>• Daily Scrum – observers, scouts<br>• CoPs<br>• Scrum of Scrums<br>• Open Space | Depends on Lifecycle chosen |
| Product Increment | Every Sprint | Every Release (or earlier as desired) | Every Sprint | Depends on Lifecycle chosen |
| Technical Practices | Not Specified | • Agile Architecture (intentional and emergent)<br>• Continuous Integration<br>• Test-First<br>• Code Refactoring<br>• Pair Work<br>• Collective Code Ownership | Emergent design XP practices | Not Specified |

# CAMBIA SCALING PRACTICES

# High Level View of Scaled Agile Approach



Business Case

Business Vision

Program Funding

Program Inception

Quarterly Planning

Quarterly Planning

Quarterly Planning

Quarterly Planning

Periodic Deployments

Program Realization
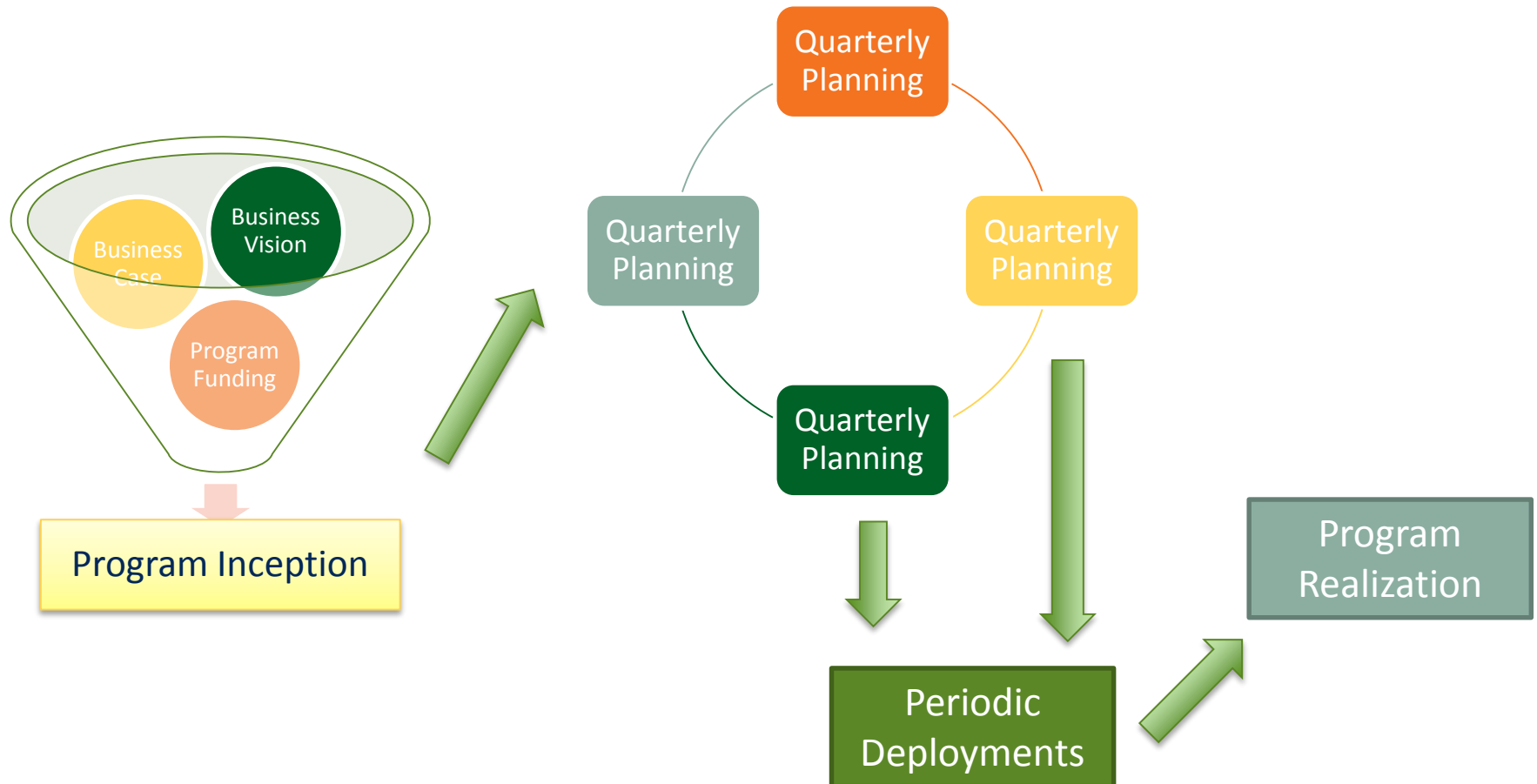
# Synchronized Quarterly and Sprint Process

Project Scrum of Scrums

Sprint 2

Sprint 1

Sprint N

Hardening Sprint

Quarterly Pre-Planning

Quarterly Planning

All Teams on same Quarterly and Sprint Planning Cadence

Coordinated Program Releases
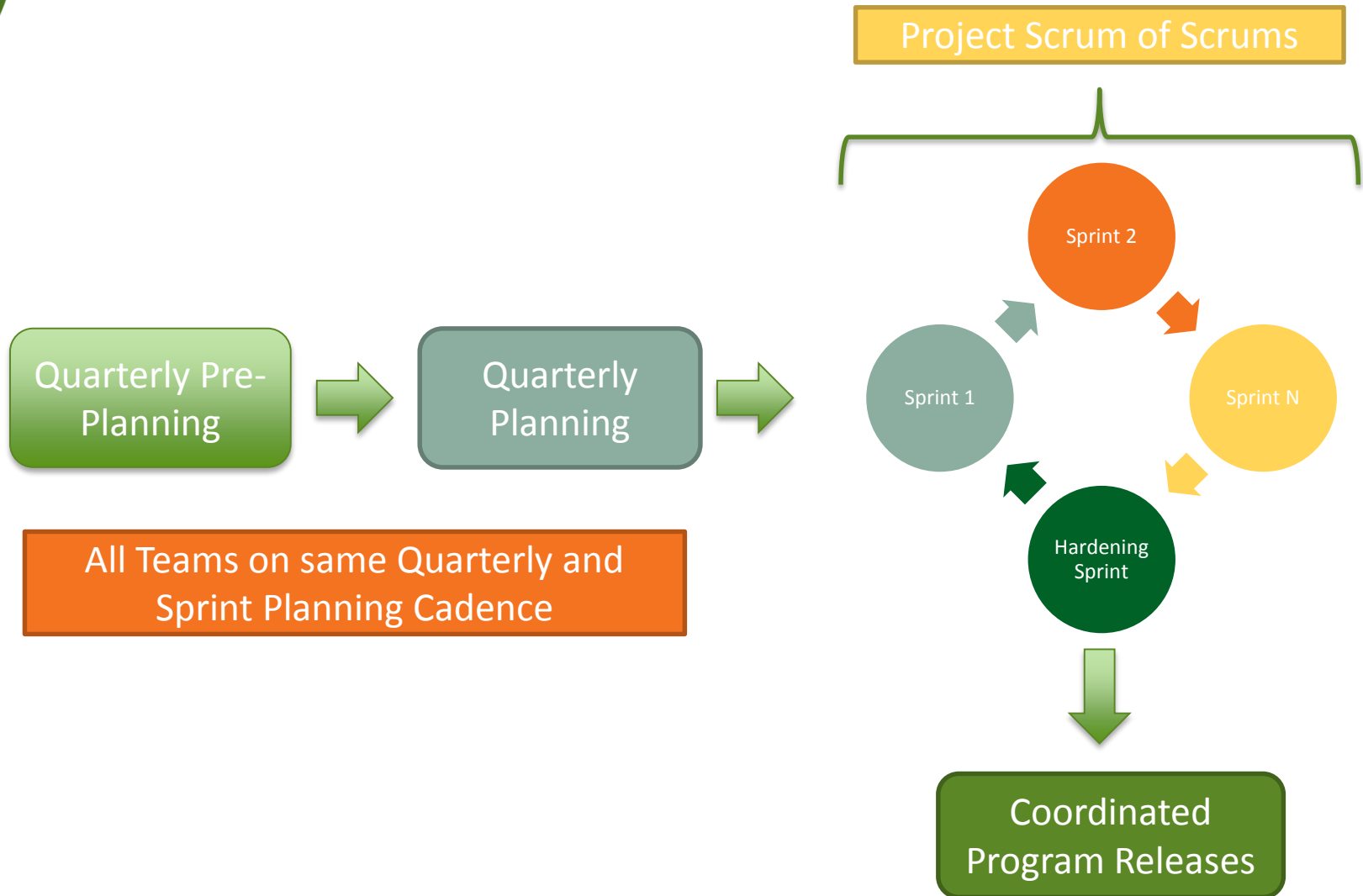
# Quarterly Planning Meeting

# Enterprise-Wide Synchronized Sprint Schedule

# Rolling 90-180 Days Grooming

**Sprint 1**
- Features and Epics groomed for **180 days (Sprints 2-9)**
- Stories groomed **90 days (Sprint 2-5)**

**Sprint 2**
- Features and Epics groomed for **180 days (Sprints 3-10)**
- Stories groomed **90 days (Sprint 3-6)**

**Sprint 3**
- Features and Epics groomed for **180 days (Sprints 4-11)**
- Stories groomed **90 days (Sprint 4-7)**

**Sprint 4....**
- Rinse and Repeat

# Scrum of Scrum Meetings

Project Scrum of Scrums

Team A

Team B

Team C

Team D
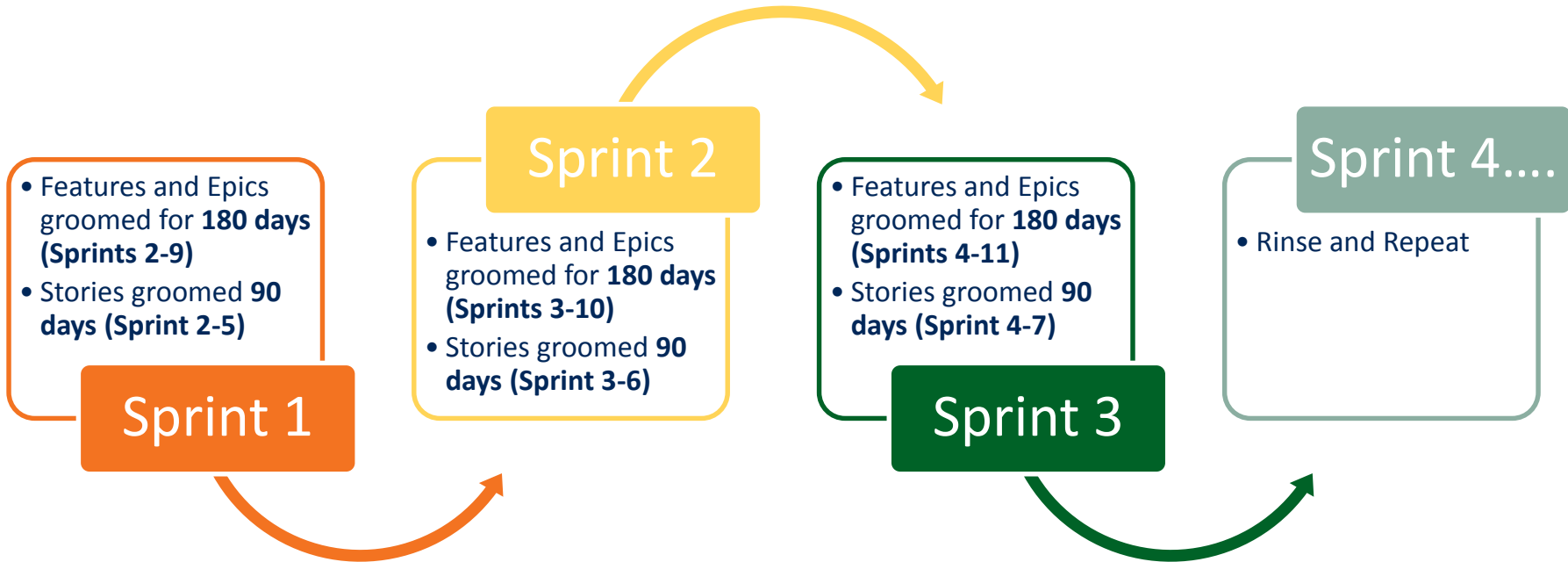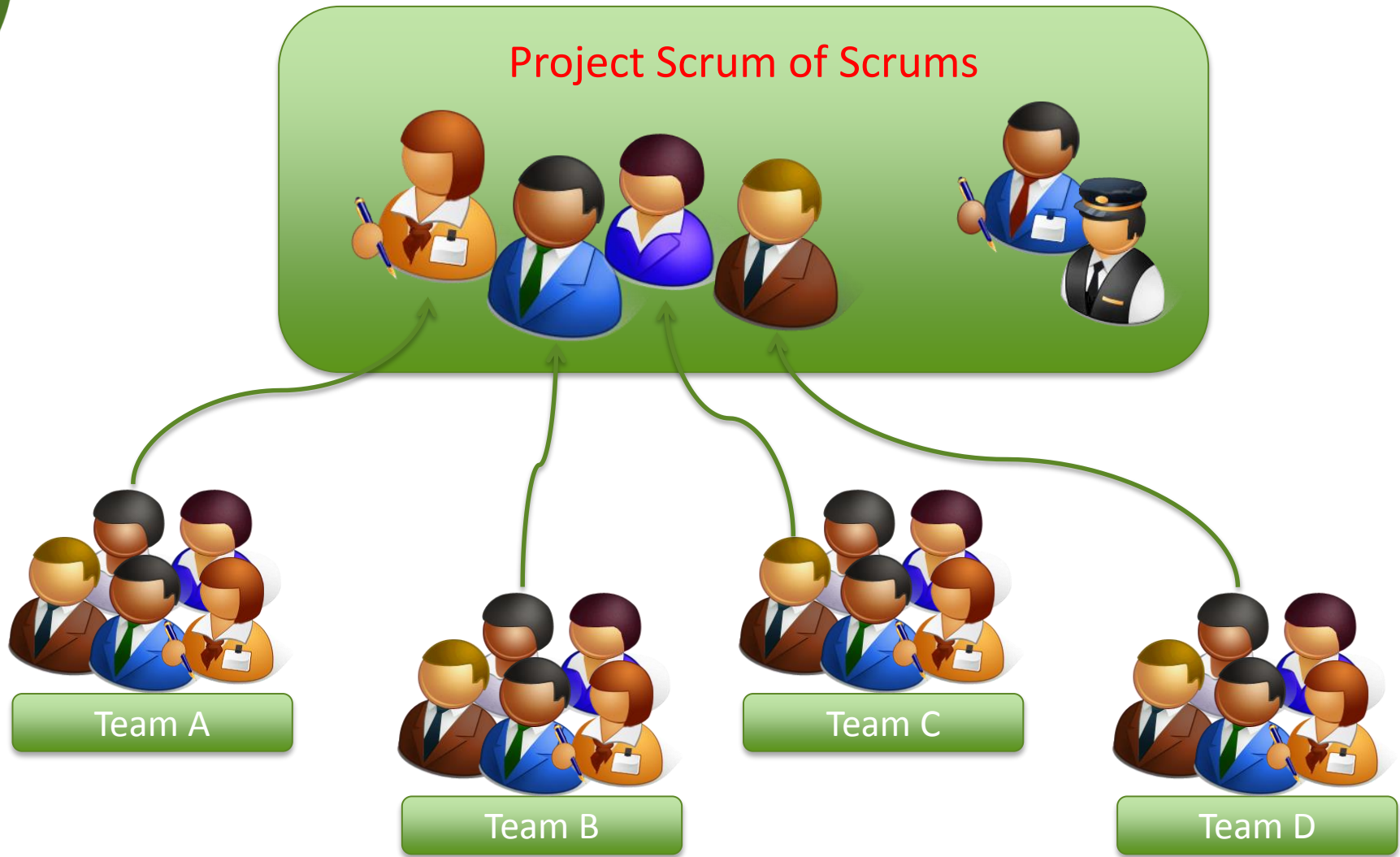
# Communities of Practice



= Skillset

Agile Teams

ScrumMaster BPE

Development BPE

SQ Engineer BPE
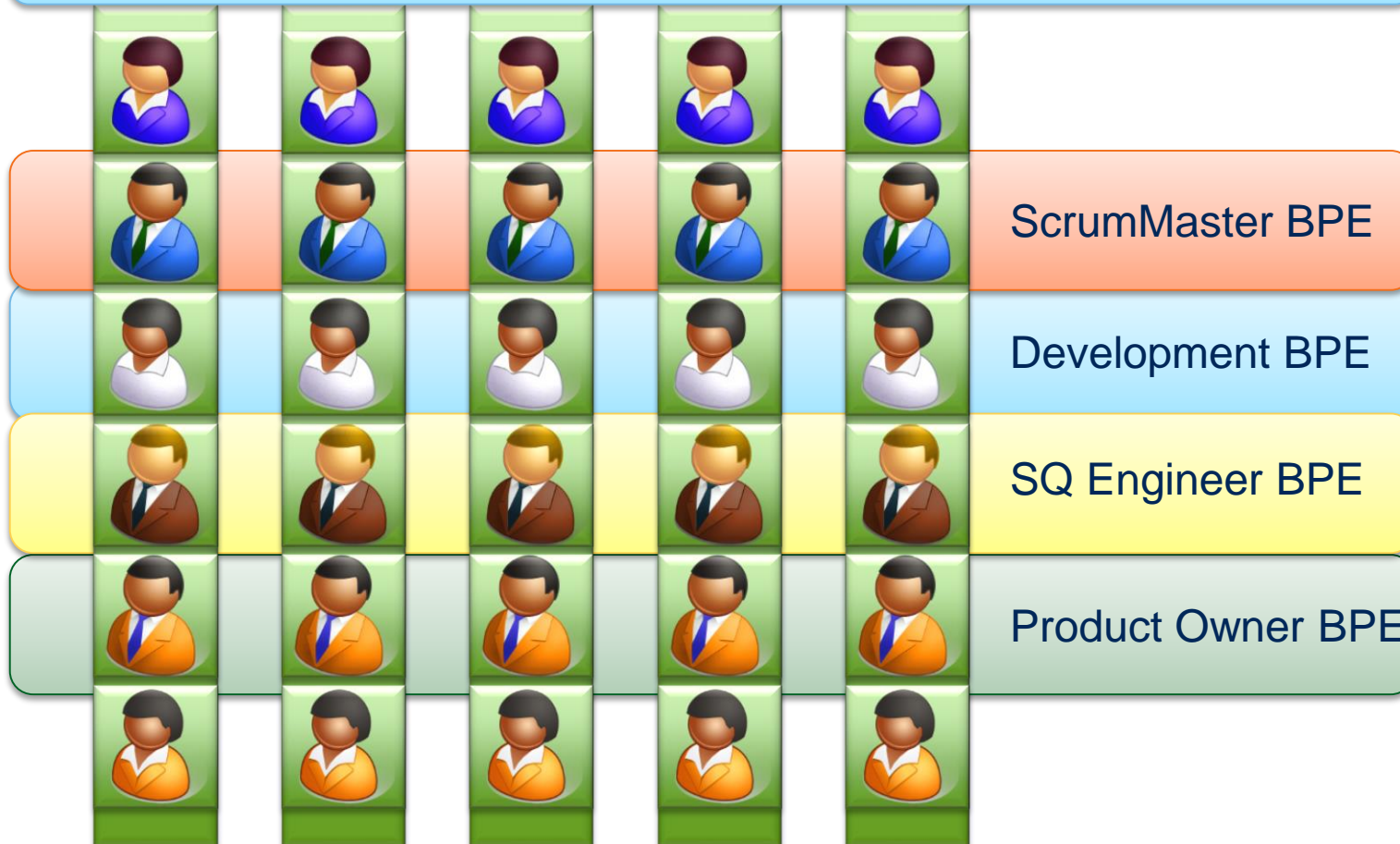
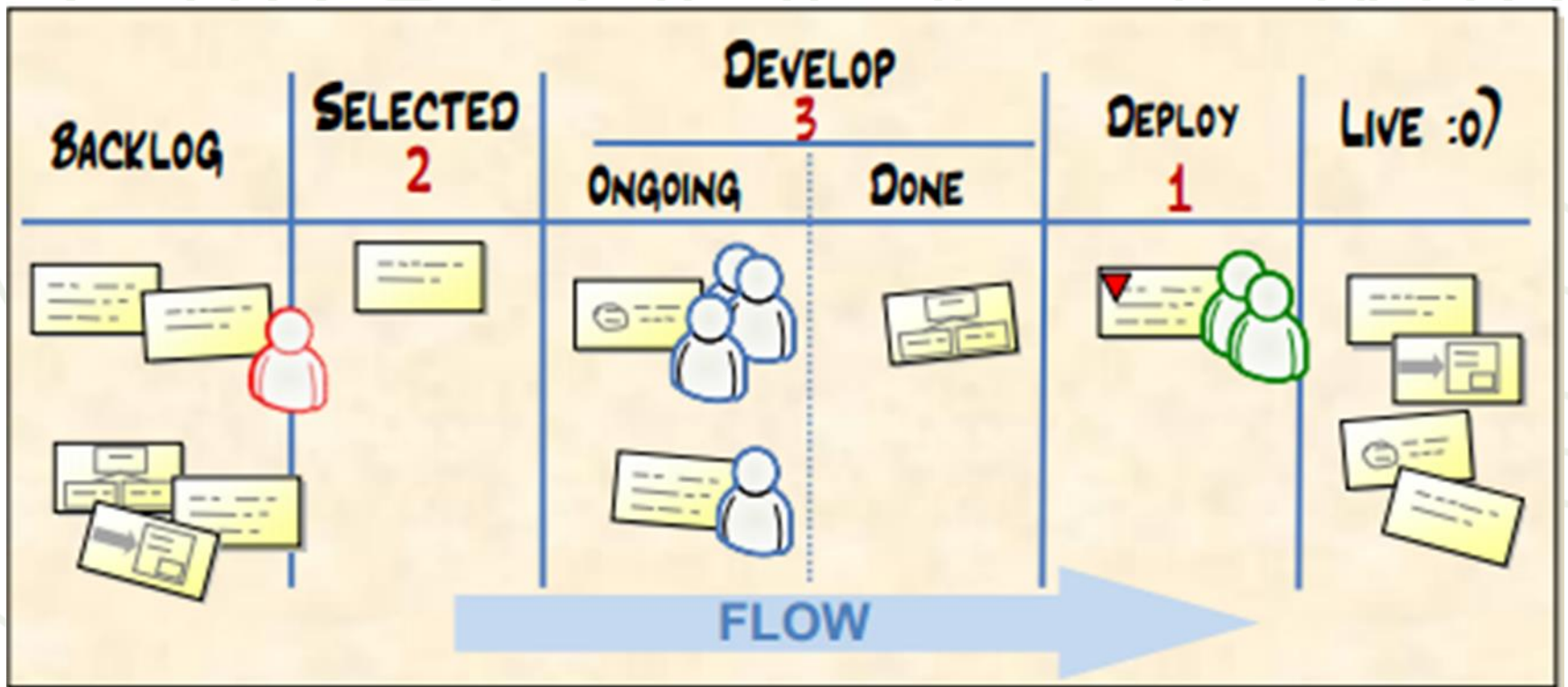Product Owner BPE

# APPENDIX A
# AGILE METHODS

# Scrum Team Makeup

- Skillset: cross-functional team to complete all their work with minimal dependencies
- Size: 5-9 team members + ScrumMaster + Product Owner

# KANBAN

# Kanban is Less Prescriptive Than Scrum

## Practices

- Visualization of Workflow
- Impose Limit on Work-in-Process

## Other Considerations

- Measure the Cycle Time
- Make Process Policies Explicit
- Use Models to Recognize Improvement Opportunities

# Kanban Team Makeup

- Skillset: cross-functional desired.  But, could be highly specialized as well.

- Size: None specified.  About 10 is what we generally see

# Presenter Info

- Aashish Vaidya
- Twitter: @aashishvaidya
- Linkedin: www.linkedin.com/pub/aashish-vaidya/2/13/168/

# Further References

- Scrum Alliance Coaching Retreat 2016: Goldilocks Agile At Scale

- PNSQC 2014 Paper: Does DAD Know Best, Is it Better to do LeSS or Just be SAFe? Adapting Scaling Agile Practices into the Enterprise

- PNSQC 2012 Paper: On the Way to Meeting a Mandate: Transititoning to Large Scale Agile