

Achieving Sustainable organisational agility with



Ahmed Sidky, Ph.D.

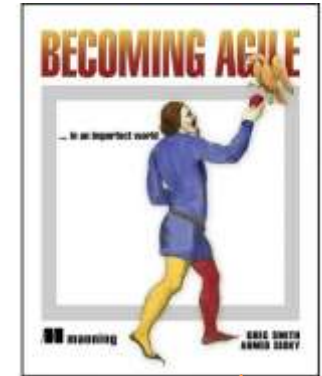
15 YEARS

years of experience in software development, management and delivery

Co-founded



Co-authored



on the PMI-ACP Steering committee



Virginia Tech
Ph.D. in Agile Transformation and Agility Assessment



Consulted, trained or coached with people and teams from ...



Agile2009 Conference
Program Chair

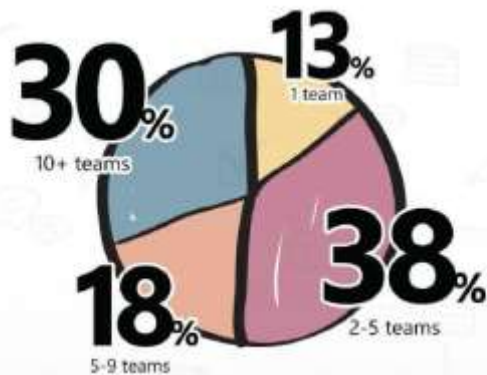




We can all agree, Agile Adoption on a Rise ...

HOW MANY TEAMS ADOPTED AGILE?

This year we've seen growth in the number of teams practicing agile at each organization surveyed. Nearly half of respondents worked at companies that had adopted agile practices across 5 or more teams (48%), up from 33% in 2011, and 30% said they had 10 or more agile teams.



Source: 7th Annual VersionOne State of Agile Development Survey

PERCENTAGE OF PROJECTS USING AGILE

31%
0-25%
of Projects

19%
26-50%
of Projects

13%
51-75%
of Projects

37%
76-100%
of Projects

Source: 7th Annual VersionOne State of Agile Development Survey

Agile Project Management Demand Trend



Source: IT Jobs Watch, March 2012



Meet Jack

Company: **Future Corp**

Size: **10,000 people**

Profession: **CIO**

Size of IT: **3000 People**

Goal: **Transform organization to Agile - ASAP**

Plan: ***Something like this***

1. Start training across IT – probably on Scrum
2. Picked a star, Stacy, in the IT organization and put her in charge of the transformation – in addition to her day job.
3. Two pilot projects were launched successfully (doing Scrum) !
4. Memo from the CIO that says we're moving to an agile/scrum process for all IT projects by the end of the year.
5. The plan is to launch five pilots/teams every quarter.
6. The CIO is meeting monthly with Stacy to track the number of projects who are adopting the agile process.
7. Stacy is procuring an agile tool to help teams be consistent in their agile process.



Meet Jack

Company: **Future Corp**

Size: **10,000 people**

Profession: **CIO**

Size of IT: **3000 People**

Goal: **Transform organization to Agile - ASAP**

Plan: ***Something like this***

1. Start training across IT
 2. Picked a star, Stacy, in the organization and put her in charge of the transformation – in addition to her day job.
Two pilot projects were launched successfully!
 3. Memo from the CIO that says we're going to an agile process for all IT projects by the end of the year.
 4. They plan to launch five pilots every quarter. The CIO is meeting monthly to track the number of projects who are adopting the agile process.
 5. They are procuring an agile tool to help teams be consistent in their agile adoption.
- Will it Work? Will we really be Agile?**
- For how long? How Sustainable will it be?**



Meet Jack

Company: **Future Corp**

Size: **10,000 people**

Profession: **CIO**

Size of IT: **3000 People**

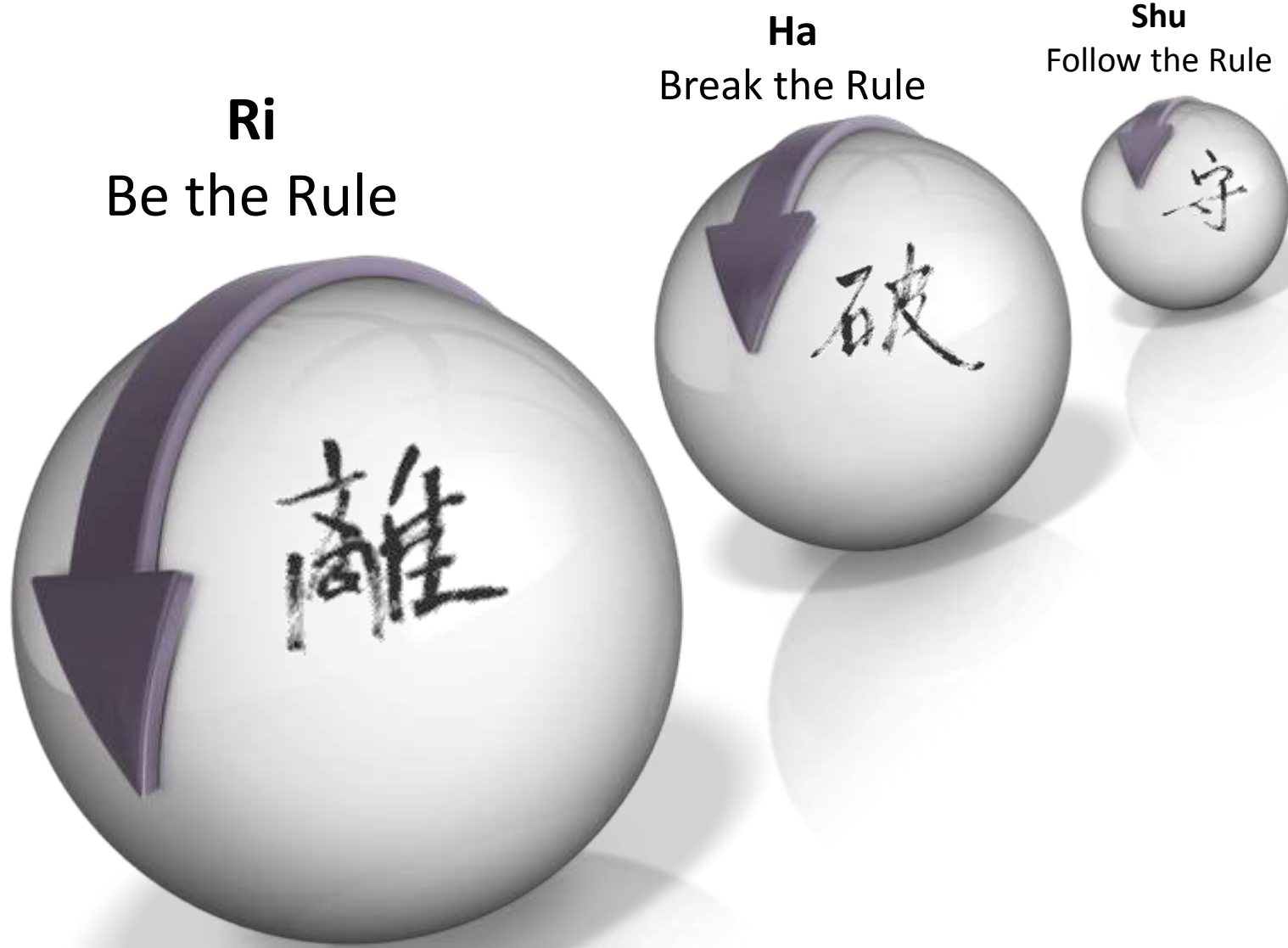
Goal: **Transform organization to Agile - ASAP**

Plan: ***Something like this***

1. Start training across IT
 2. Picked a star, Stacy, in the organization and put her in charge of the transformation – in addition to her day job.
Two pilot projects were launched successfully!
 3. Memo from the CIO that says we're going to an agile process for all IT functions by the end of the year.
 4. The plan is to launch five pilots every quarter. The CIO is meeting monthly to track the number of projects who are adopting the agile process.
 5. They are procuring an agile tool to help teams be consistent in their agile adoption.
- Will it Work? Will we really be Agile?**
- For how long? How Sustainable will it be?**

Agility and Stages of Learning

1. Shu – 2. Ha – 3. Ri



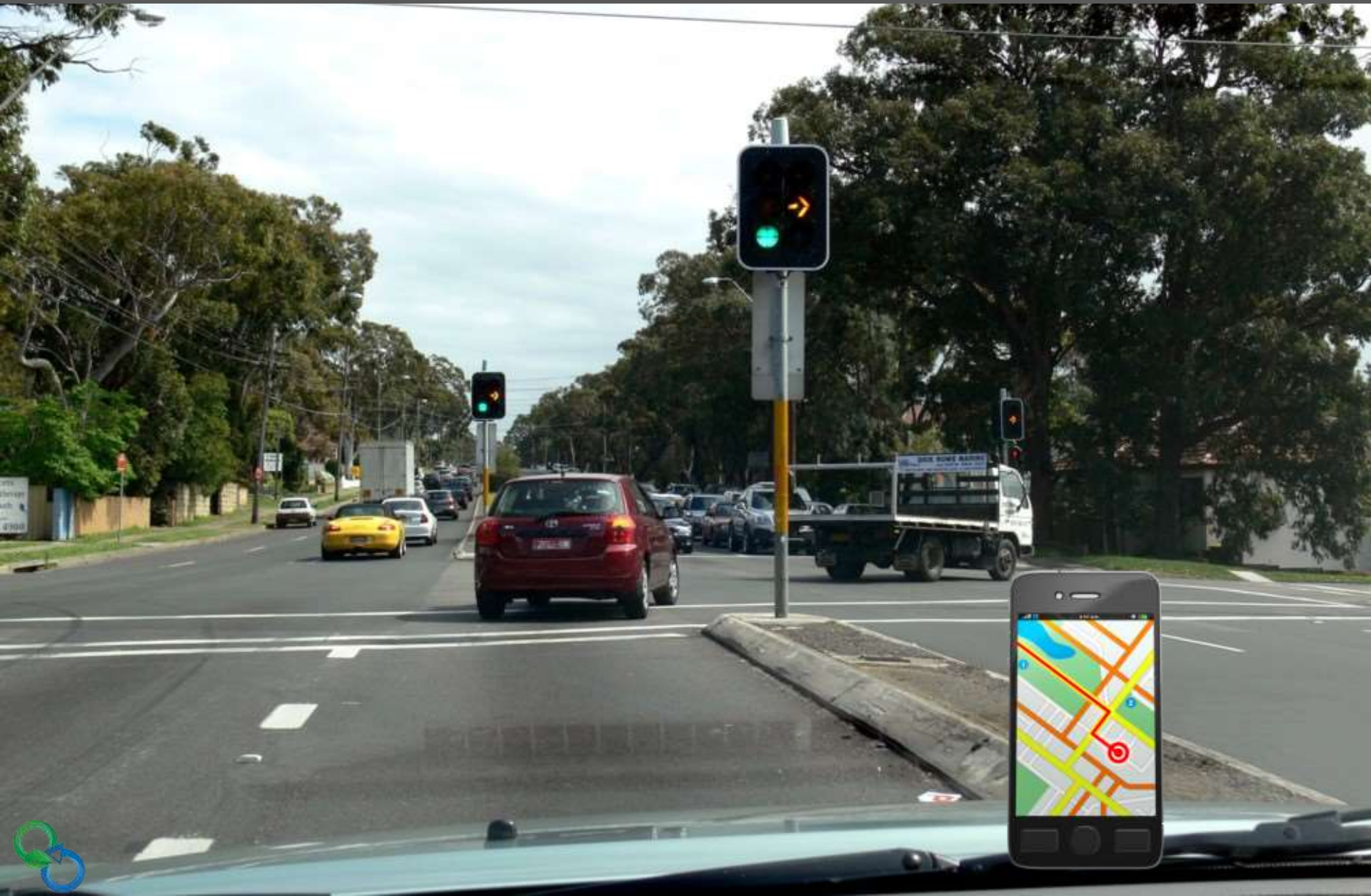
Agility and Stages of Learning

1. Shu (Following Precisely without Modification) – 2. Ha – 3. Ri



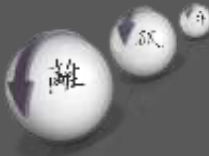
Agility and Stages of Learning

1. Shu – 2. Ha (Shifting Between Techniques) – 3. Ri



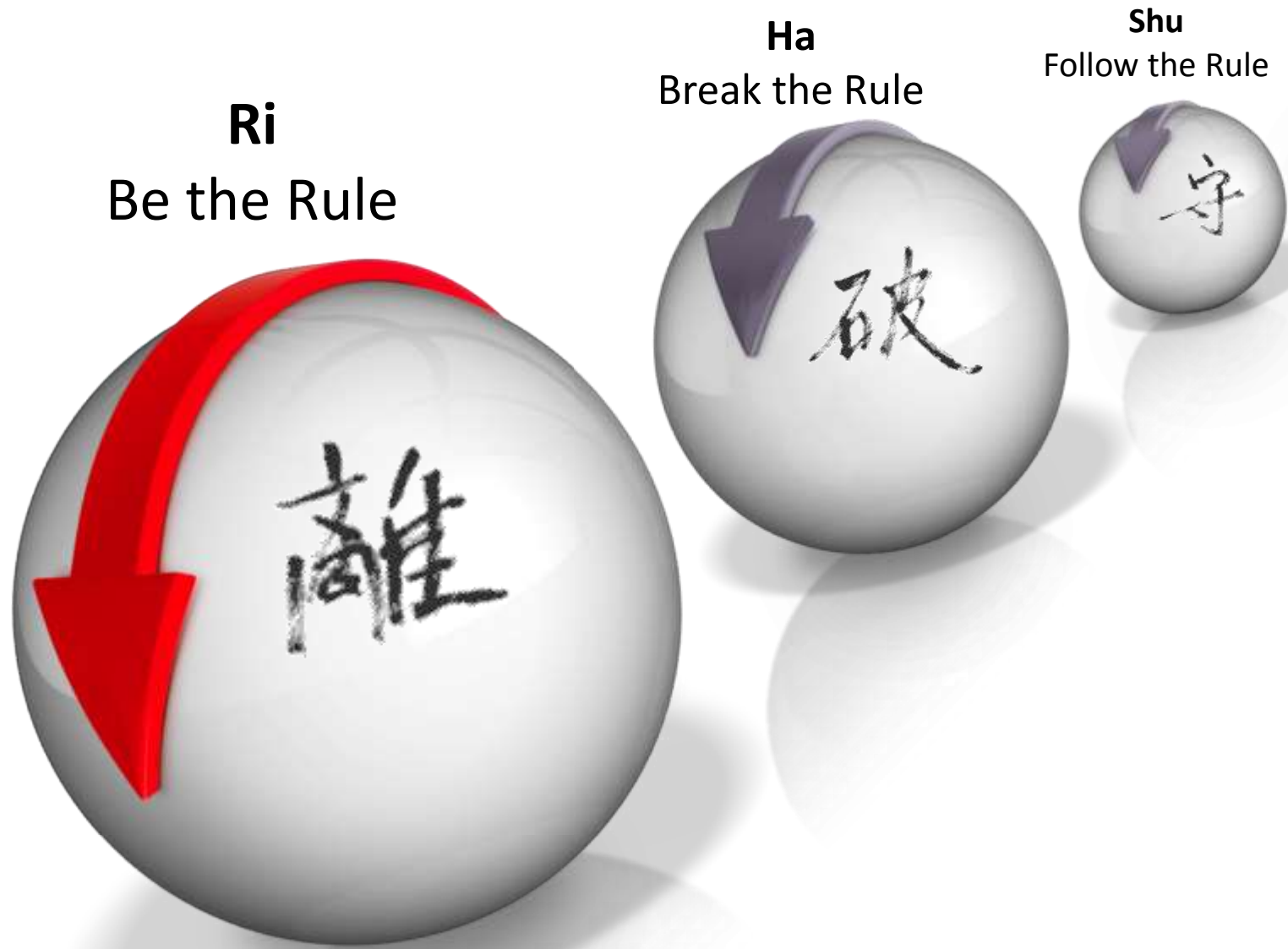
Agility and Stages of Learning

1. Shu – 2. Ha – 3. Ri (New techniques, even unconsciously)



Agility and Stages of Learning

1. Shu – 2. Ha – 3. Ri



What is Agile?

No Documentation

Process ?

No Architecture

Fad?

Framework ?

Chaos

Methodology ?

Cult?

No Planning

Approach ?

No Discipline

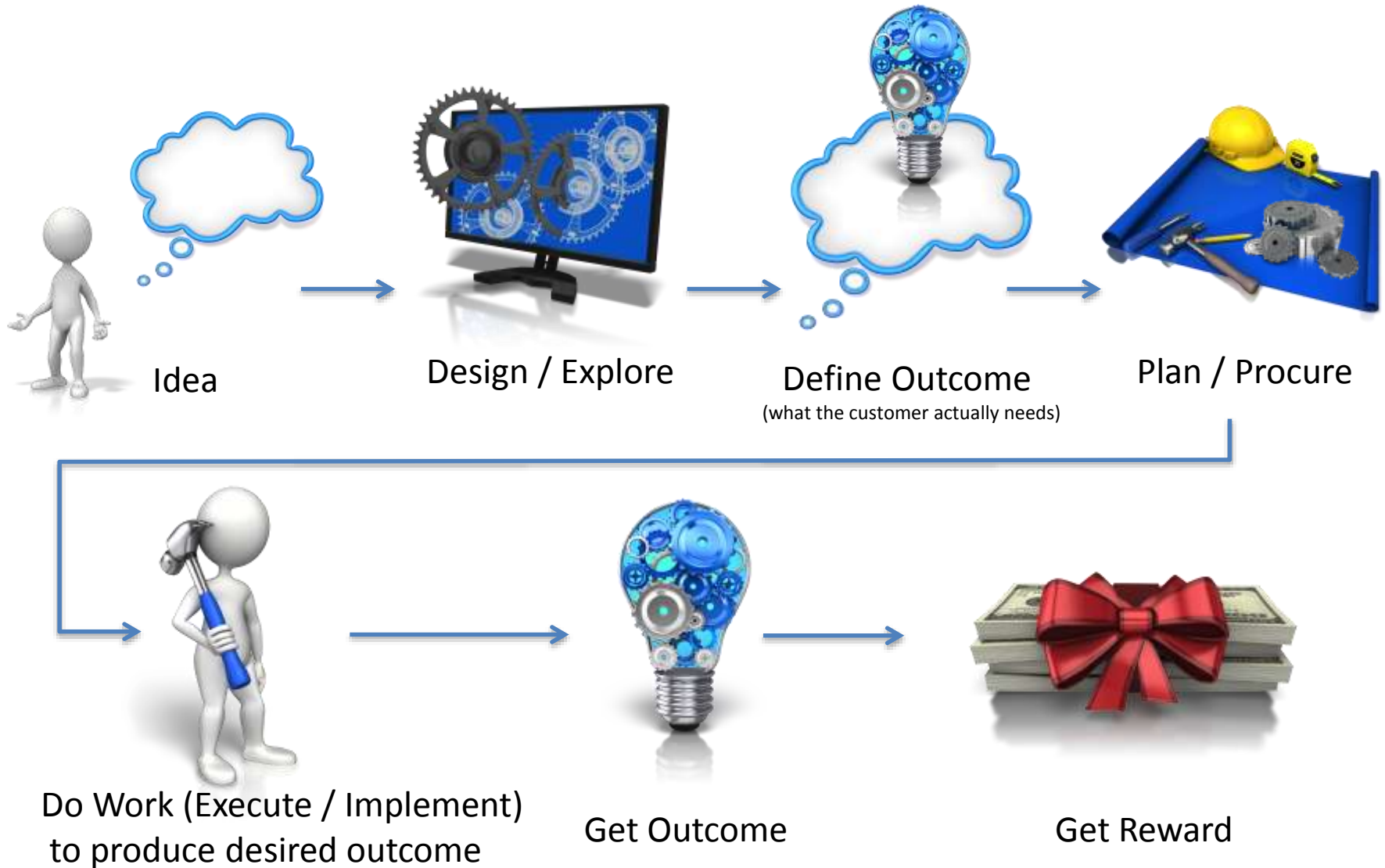


Agile is a Mindset ...

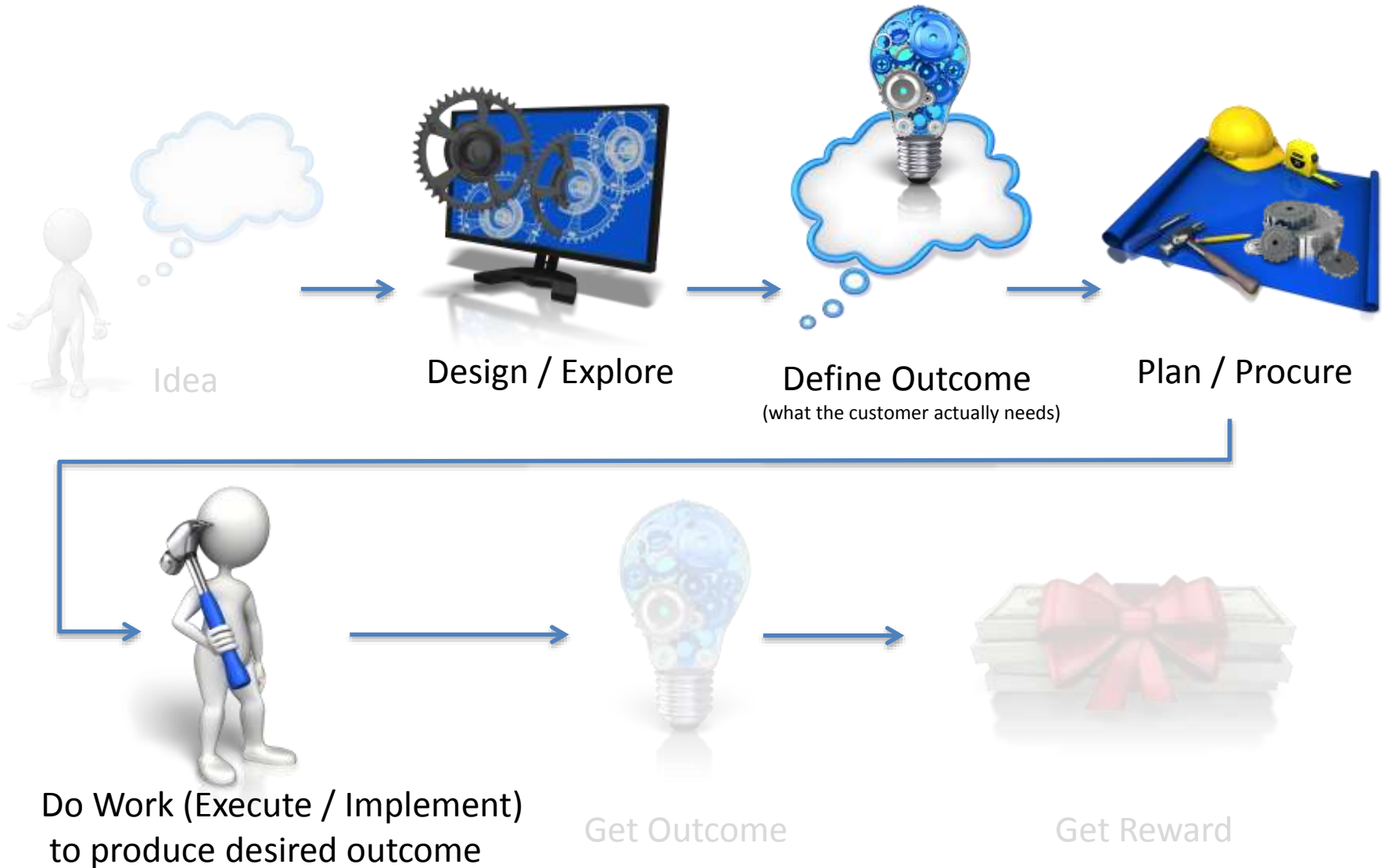
Your mindset is the established set of attitudes and habits you have about how to succeed at getting work done.



How do we get work done?

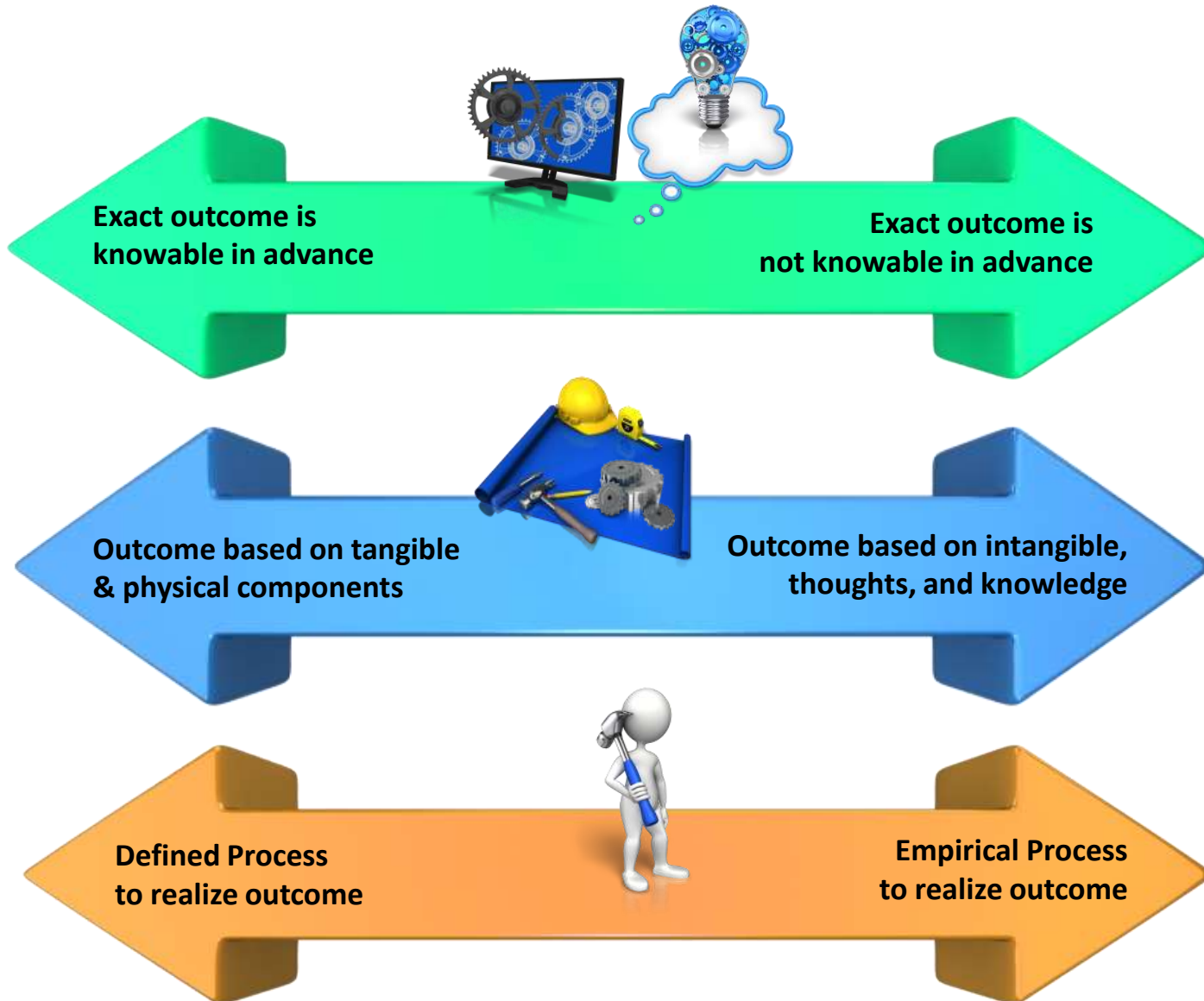


What is our Mindset towards ...



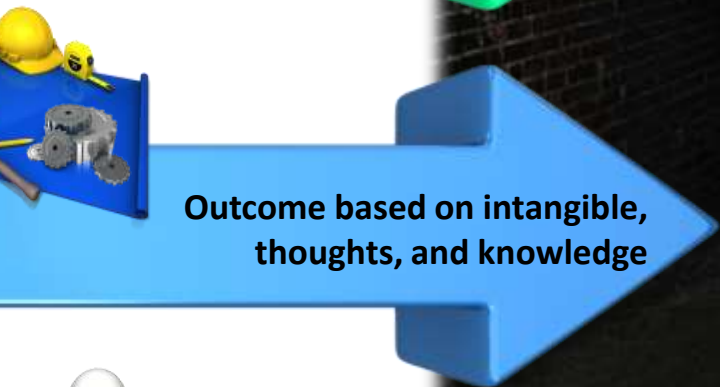
What is our Mindset towards ...

Assembly line mentality



Knowledge work Mentality

What is our Mindset towards ...



The Tunnel of Uncertainty

We have an idea where we are going, and we don't know exactly what the outcome will be, and we know there will be lots of changes and we really can't control all the players needed to produce the outcome according to plan.

What is our

Mindset

towards succeeding when there is

Uncertainty

What is our

established set of
attitudes and habits

towards succeeding when there is

Uncertainty

I believe that my **[Intelligence, Personality, Character]** is locked-down or fixed. My potential is determined at birth. It doesn't change.

Fixed Mindset



Desire to **avoid failure and look smart** in every situation and prove myself.

Avoids challenges and obstacles because risk of failure.

Stick to what they know and can do.

Failure is an impression of **lack of talent**, therefore quick to blame and be defensive.

Feedback and criticism is **personal** as it impacts self-image.

They **don't change** or improve so to this confirms that "they are as they are."

I believe that my **[Intelligence, Personality, Character]** can be continuously developed. My true potential is unknown and unknowable.

Growth Mindset



Desire **continuous learning**. Confront uncertainties.

Embracing challenges because will learn something new

Not afraid to fail – an opportunity to learn

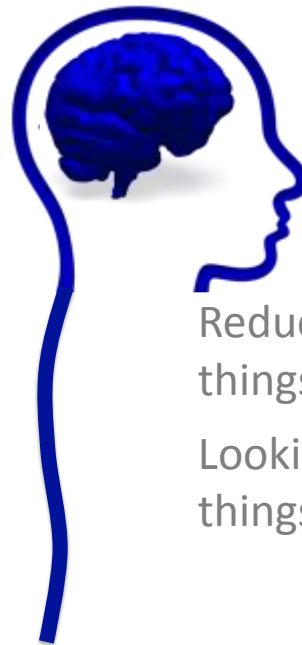
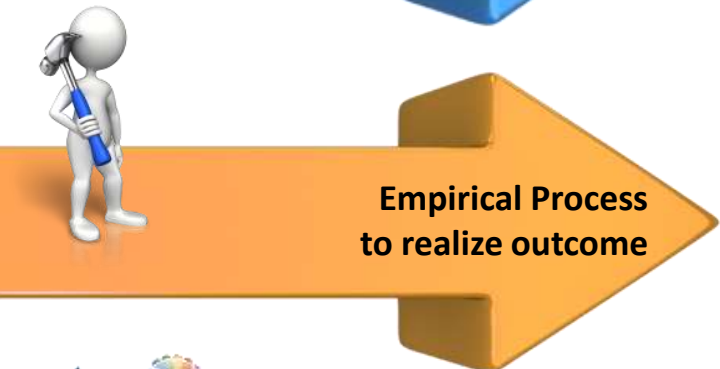
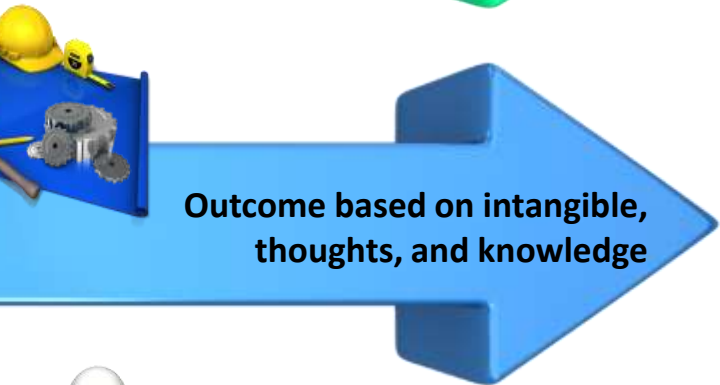
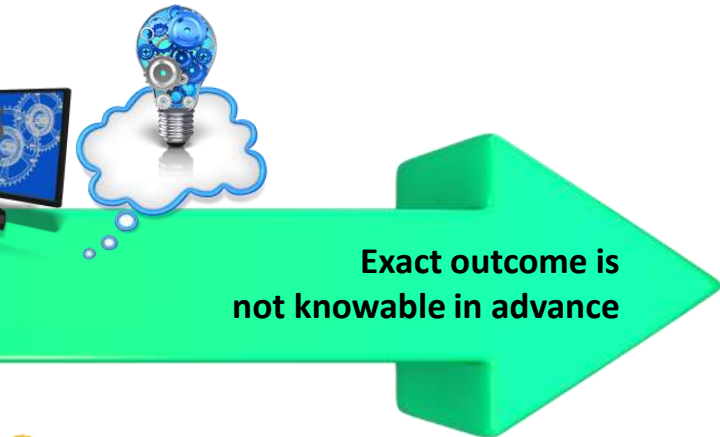
Put lots of effort to learn and master something new

Feedback and criticism is not about them but **about current capabilities**

Elicit feedback since it is a source of new information and learning

Context of Personal Life

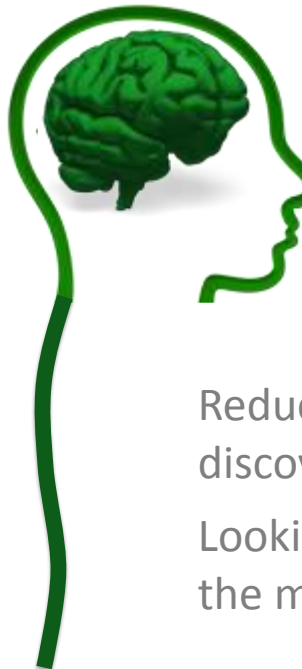
Which Mindset ?



Fixed Mindset
approach to
managing
uncertainty

Reducing uncertainty by “nailing things down.”

Looking to fix and confirm things.



Agile Mindset
approach to
managing
uncertainty

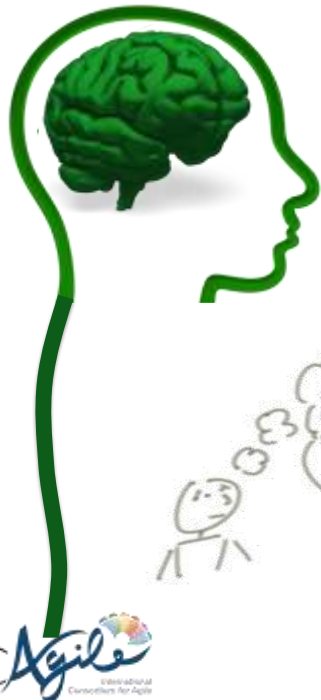
Reducing uncertainty by discovering and learning.

Looking to learn and discover in the most efficient way possible.



Fixed Mindset approach to using Iterations

Reducing uncertainty by “nailing things down.”

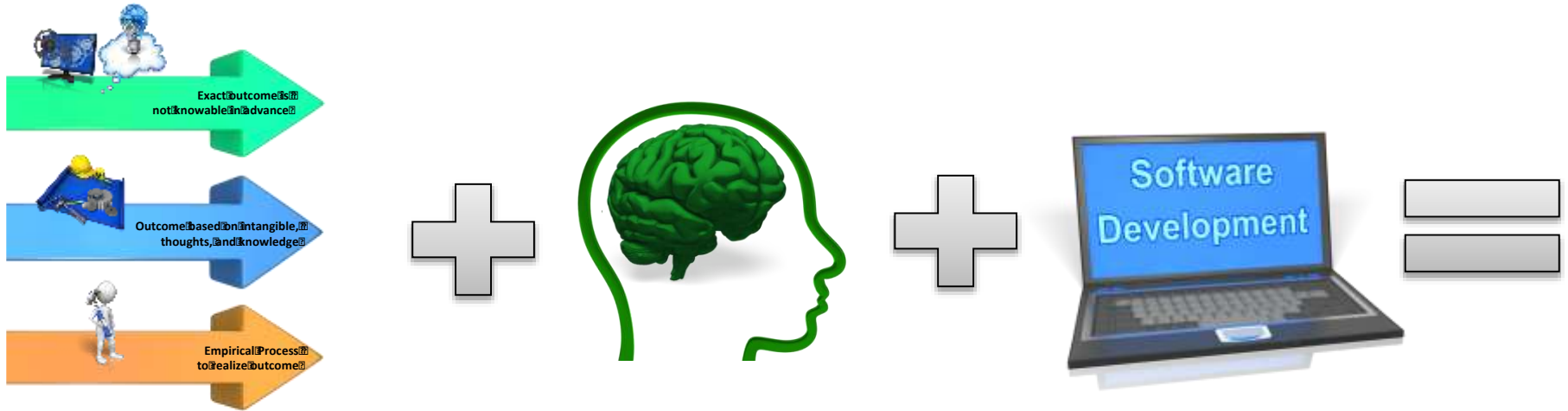


Agile Mindset approach to using iterations

Reducing uncertainty by discovering and learning.



How to manage **Uncertainty** using the **Agile Mindset** in the **Software domain**



THE 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



Agile is a mindset

[that in software world is]



Established through 4 values



Grounded by 12 principles, &



Manifested through many
many different practices

A mindset is the established set of **attitudes** held by someone

- Welcome Change
- Failing Early
- Build and Feedback loops
- Continuous Delivery
- Value-Driven Development
- Small value-add slices
- Learn through Discovery
- Continuous Improvement





Agile is a mindset

[that in software world is]



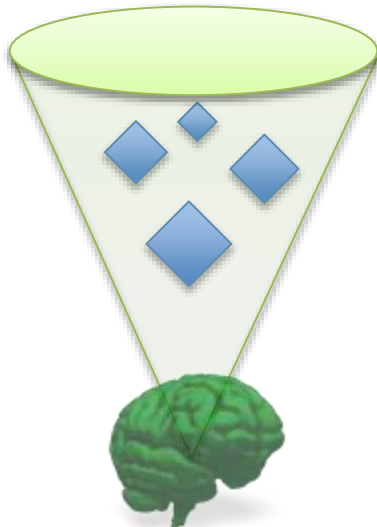
Established through 4 values



Grounded by 12 principles, &



Manifested through many
many different practices



A Value is an established ideal that the members of a given society regard as desirable

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan



Agile is a mindset

[that in software world is]



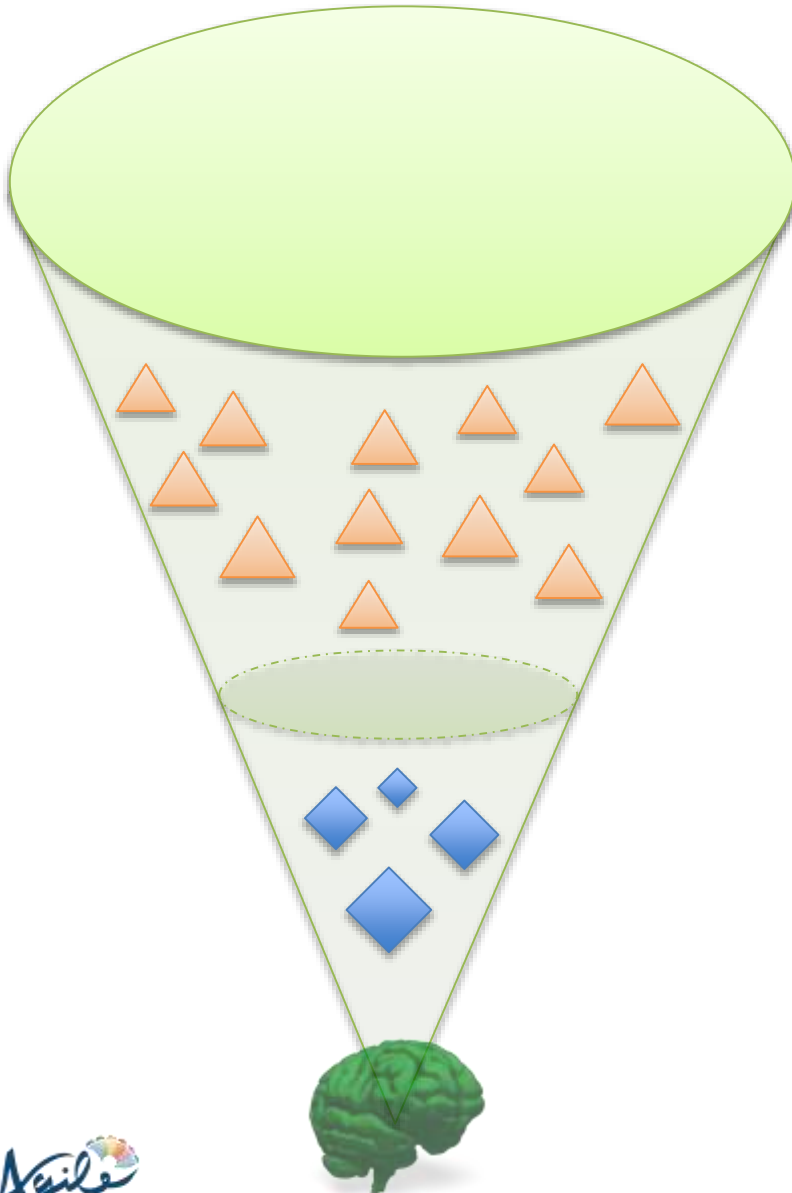
Established through 4 values



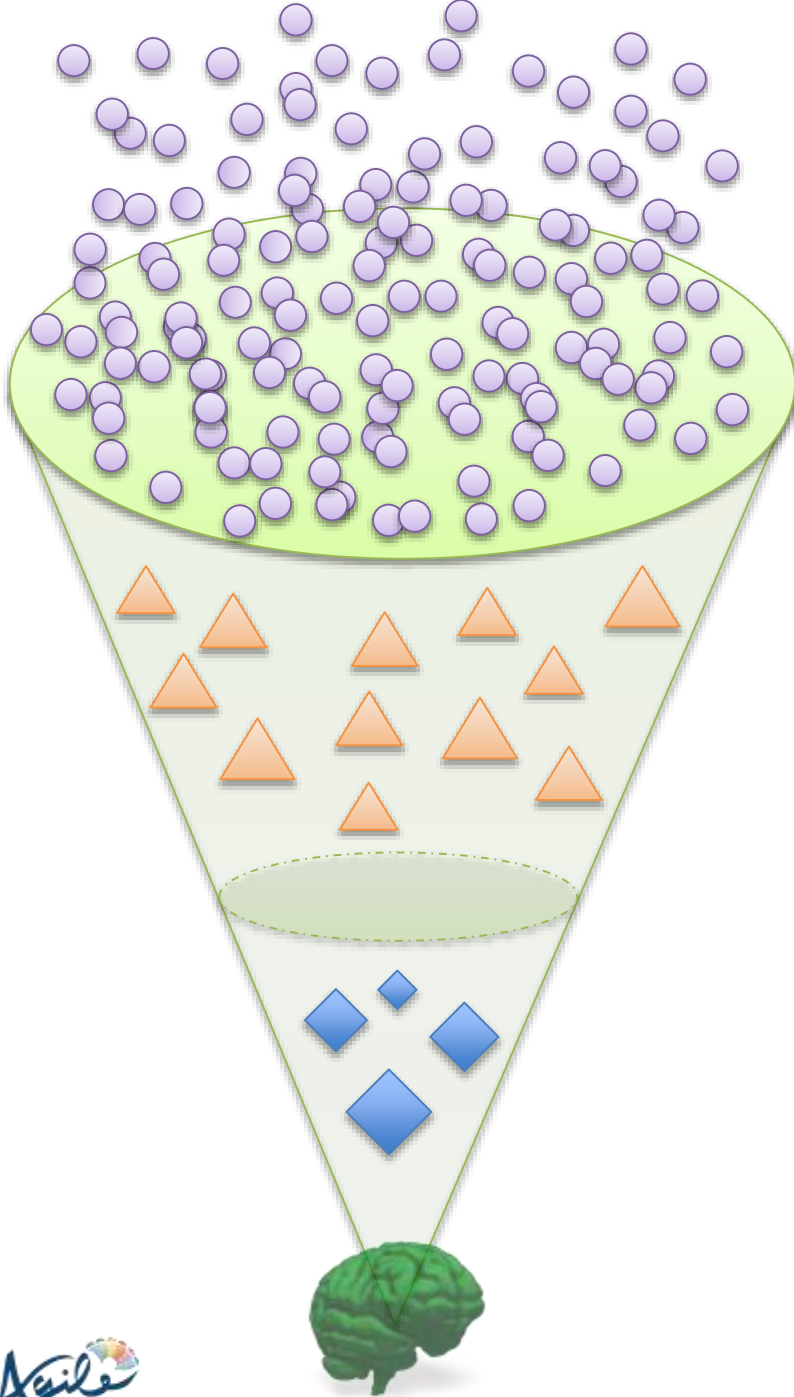
Grounded by 12 principles, &



Manifested through many
many different practices



1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity--the art of maximizing the amount of work not done--is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



Agile is a mindset

[that in software world is]



Established through 4 values



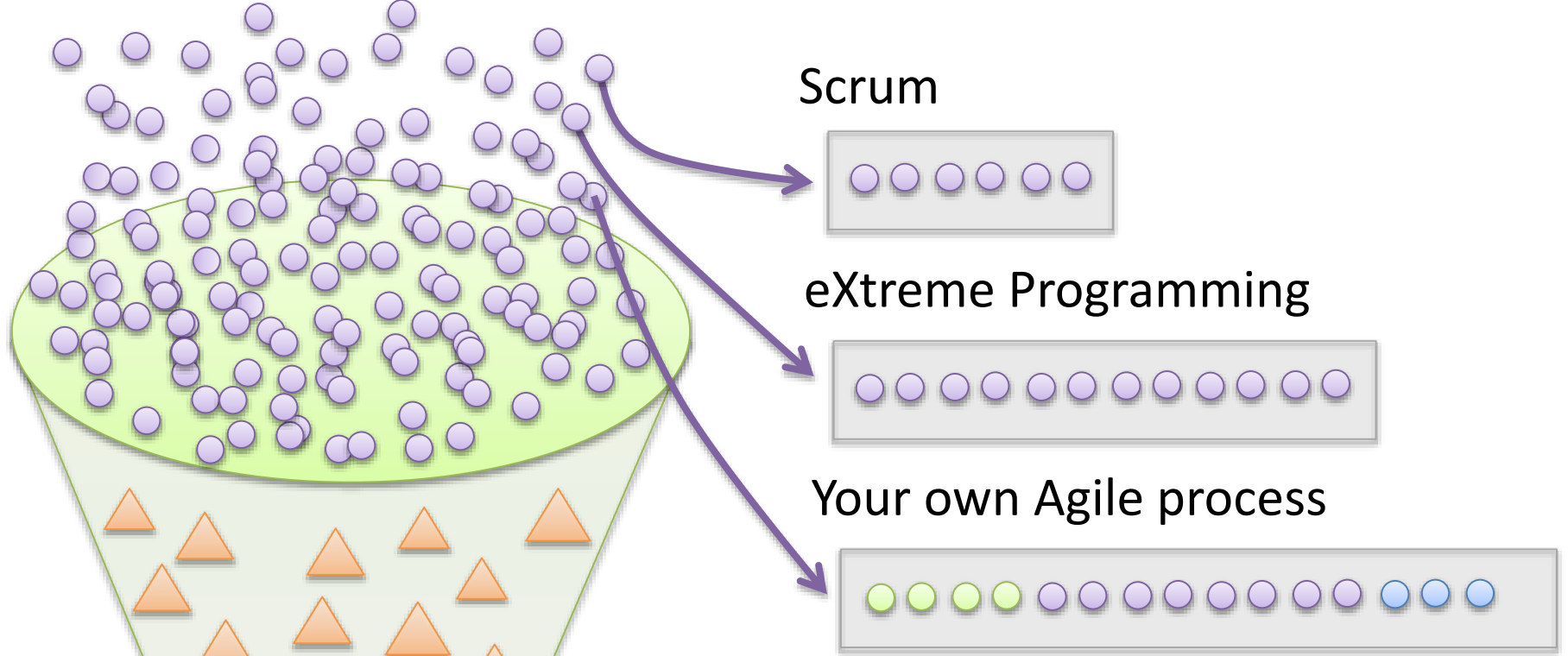
Grounded by 12 principles, &



Manifested through many many different practices

Product visioning
 Project chartering
 Affinity (relative) estimation
 Size-based (point) estimation
 Planning poker
 Group estimation
 Value-based documentation
 Prioritized product backlog
 User stories
 Progressive elaboration
 Personas
 Story maps / MMF
 Story slicing
 Acceptance tests as requirements
 Short iterations
 WIP Limits
 Early and frequent releases
 Roadmapping
 Velocity-based planning and commitment
 Iteration planning / Iteration backlog
 Release planning / Release backlog
 Time boxed iterations
 Adaptive (multi-level) planning
 Risk backlog
 Team structure of VT / DT
 Pull-based systems
 Slack
 Sustainable pace

Frequent face-to-face
 Team chartering
 Cross-silo collaborative teams
 Self-organizing teams
 Cross-functional teams
 Servant leadership
 Task volunteering
 Generalizing specialist
 Tracking progress via velocity
 Burn-up/burn-down charts
 Refactoring
 Automated unit tests
 Coding standards
 Incremental/evolutionary design
 Automated builds
 Ten-minute build
 Monitoring technical debt
 Version control
 Configuration management
 Test driven development
 Pair programming
 Spike solutions
 Continuous integration
 Incremental deployment
 Simple design
 End-of-iteration hands-on UAT
 Automated functional tests
 Automated developer tests (unit tests)
 Exploratory testing
 Software metrics



Agile is a mindset

[that in software world is]



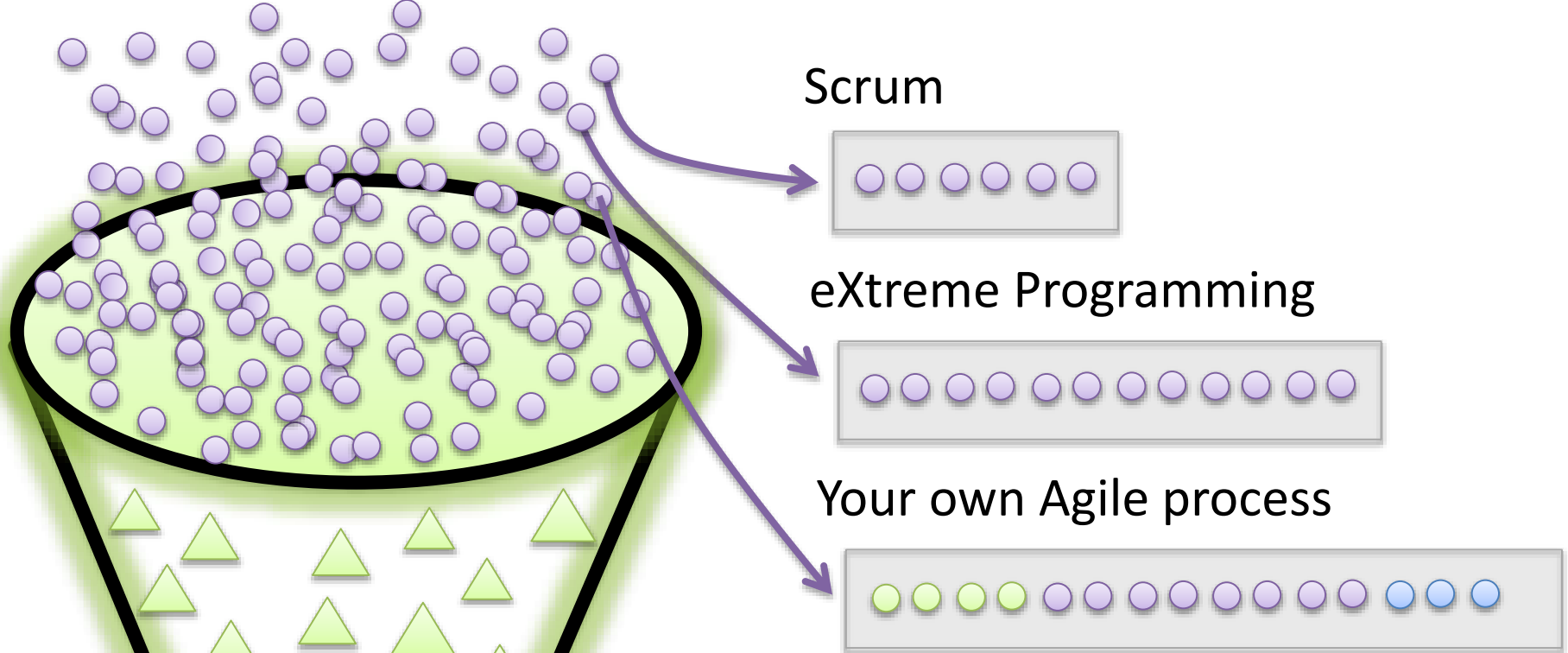
Established through 4 values



Grounded by 12 principles, &



Manifested through many many different practices



Agile is a mindset

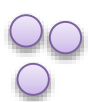
[that in software world is]



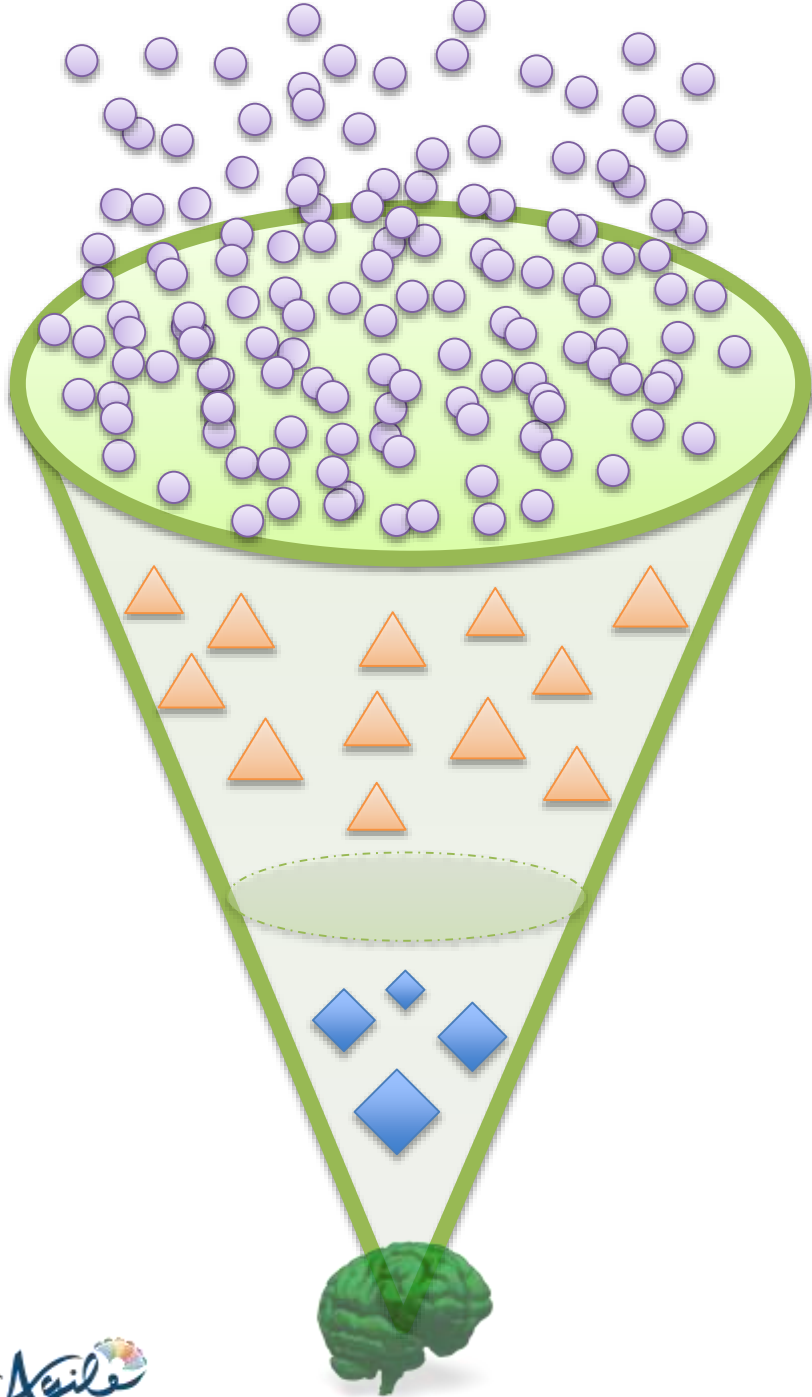
Established through 4 values



Grounded by 12 principles, &



Manifested through many many different practices



Agile as a Process and Practices

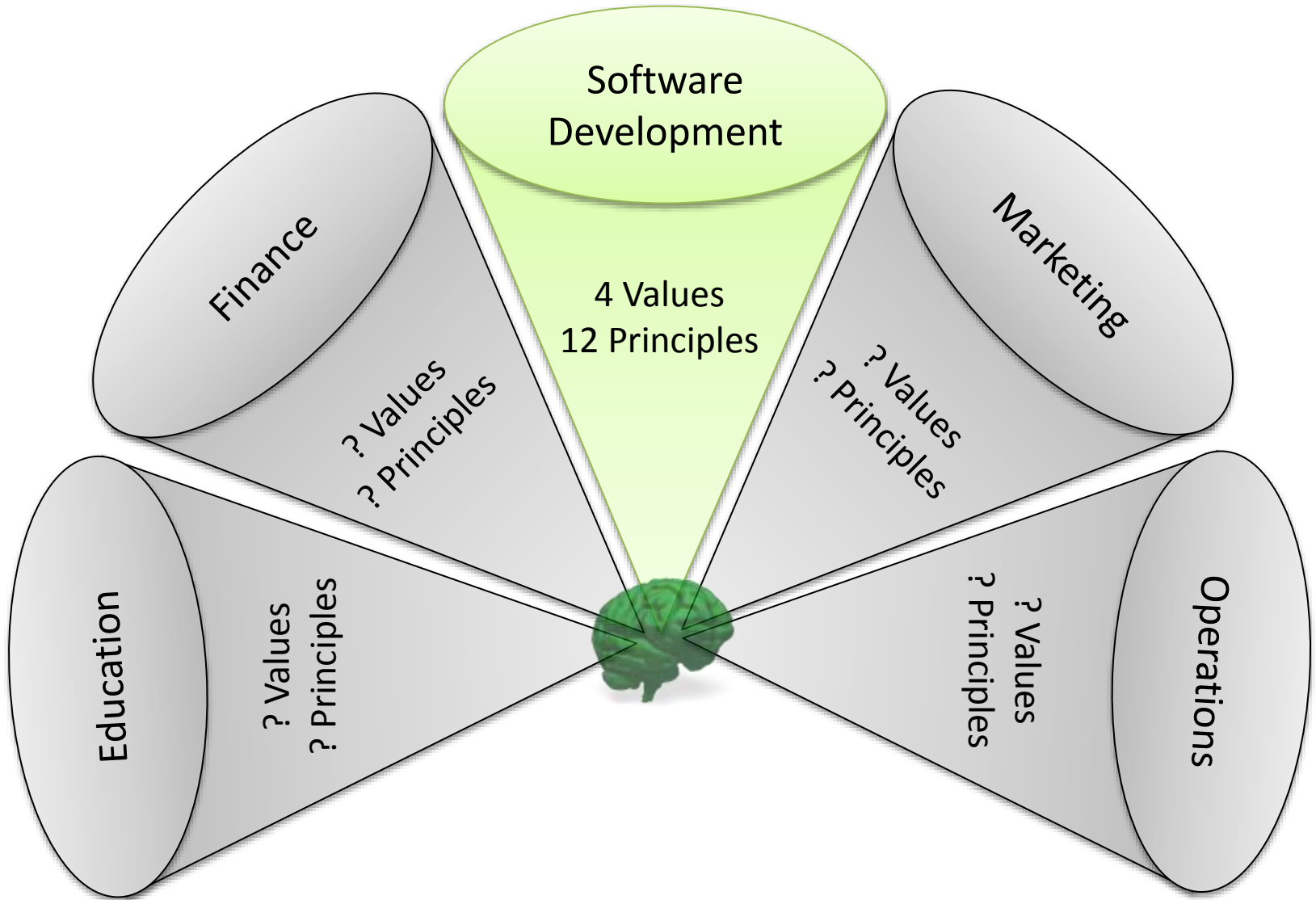
Doing Agile

Learning the practices and applying them without know the mindset and principles to know when to tailor and how to select the appropriate practices

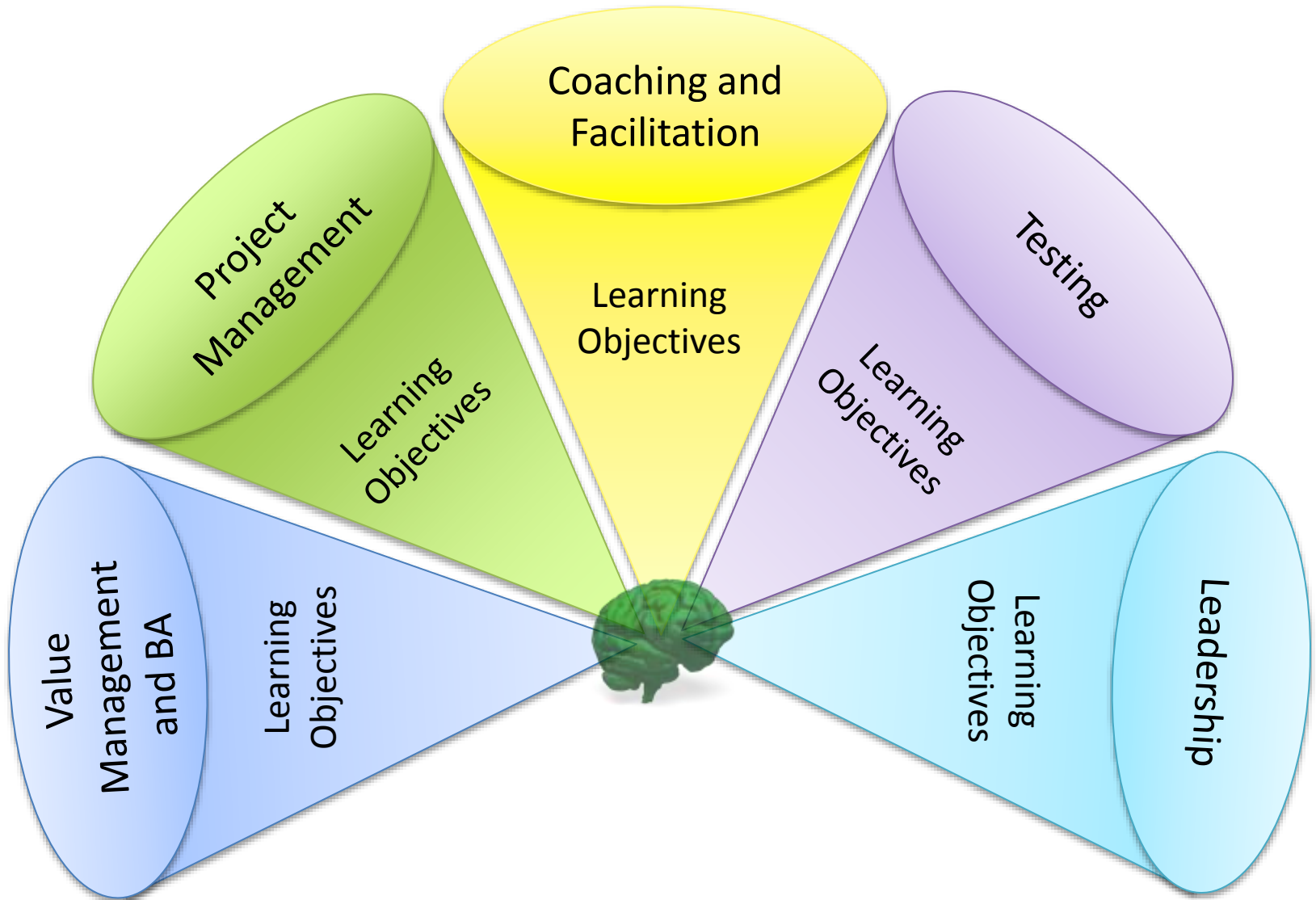
Being Agile

Internalizing the Mindset, values, and principles then applying the right practices and tailoring them to different situations as they arise

Agile as a Mindset and Culture



The Agile Mindset and other Domains



The Agile Mindset to Disciplines inside Software Development

ICAgile's Roadmap



The Agile Mindset to Disciplines
inside Software Development



Meet Jack

Company: **Future Corp**

Size: **10,000 people**

Profession: **CIO**

Size of IT: **3000 People**

Goal: **Transform organization to Agile - ASAP**

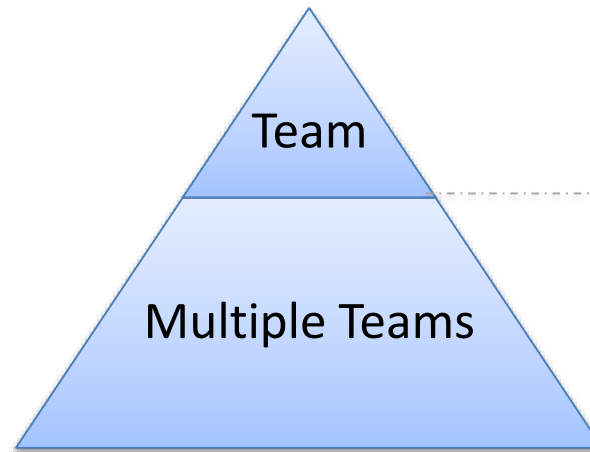
Plan: ***Something like this***

1. Start training across IT
 2. Picked a star, Stacy, in the organization and put her in charge of the transformation – in addition to her day job.
Two pilot projects were launched successfully!
 3. Memo from the CIO that says we're going to an agile process for all IT projects by the end of the year.
 4. They plan to launch five pilots every quarter. The CIO is meeting monthly to track the number of projects who are adopting the agile process.
 5. They are procuring an agile tool to help teams be consistent in their agile adoption.
- Will it Work? Will we really be Agile?**
- For how long? How Sustainable will it be?**

Process Adoption vs Culture Transformation

Process Change / Incremental Change	Organizational and Culture Transformation
Focus on Process and Technology	Focus on People
Cascading Decisions	Shared Vision
Training	Educating
Communication	Buy-in
Compliance	Commitment

Scaling Agile Spectrum



Individual Mindsets and Team (Sub) Cultures need to be aligned with Agile

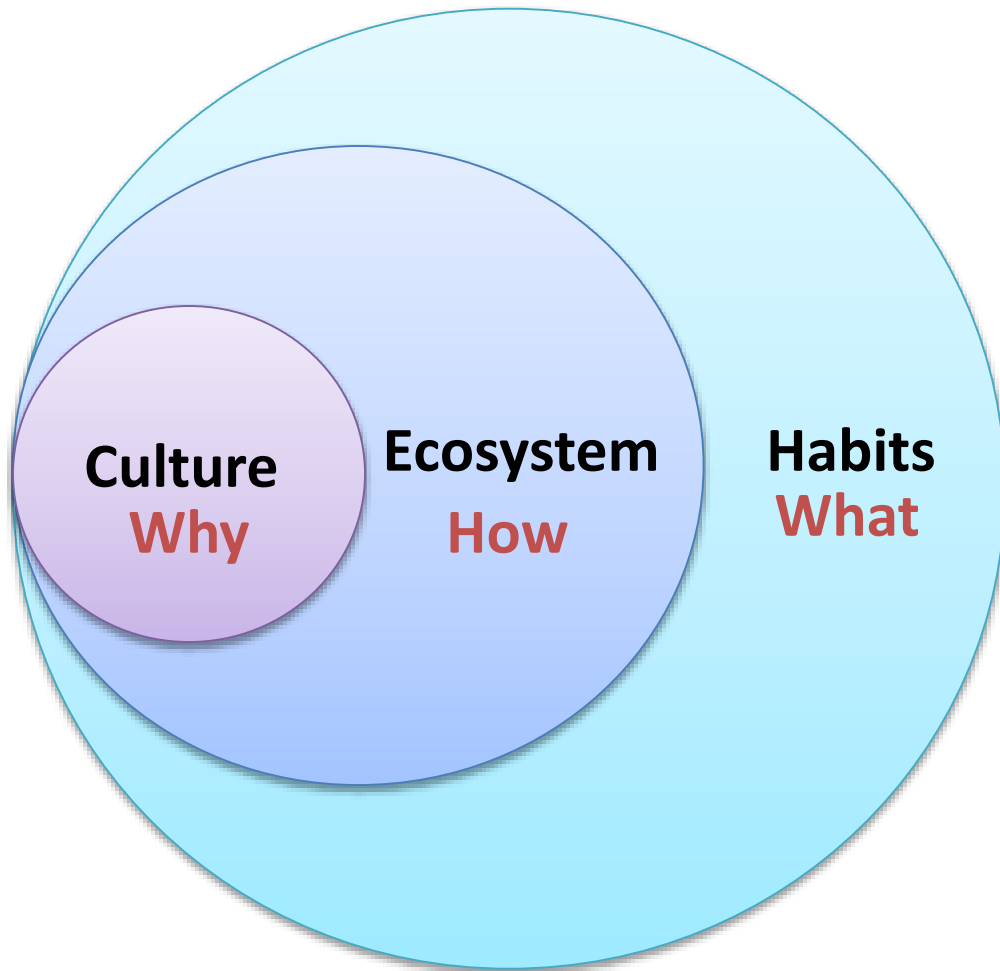


----- Chasm between Transformation and Adoption -----

Organizational Culture needs to be Aligned with Agile



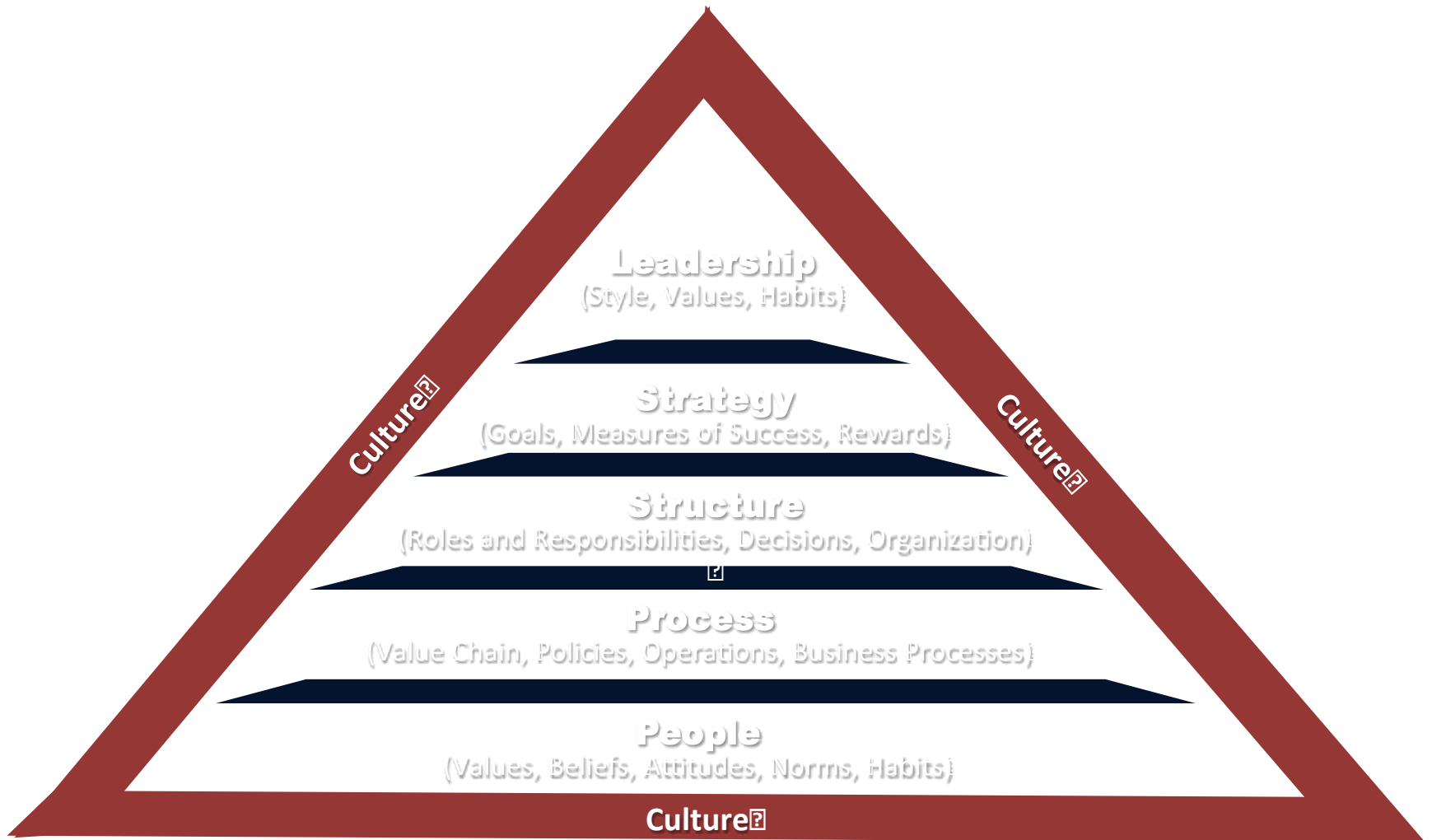
Enterprise Agility



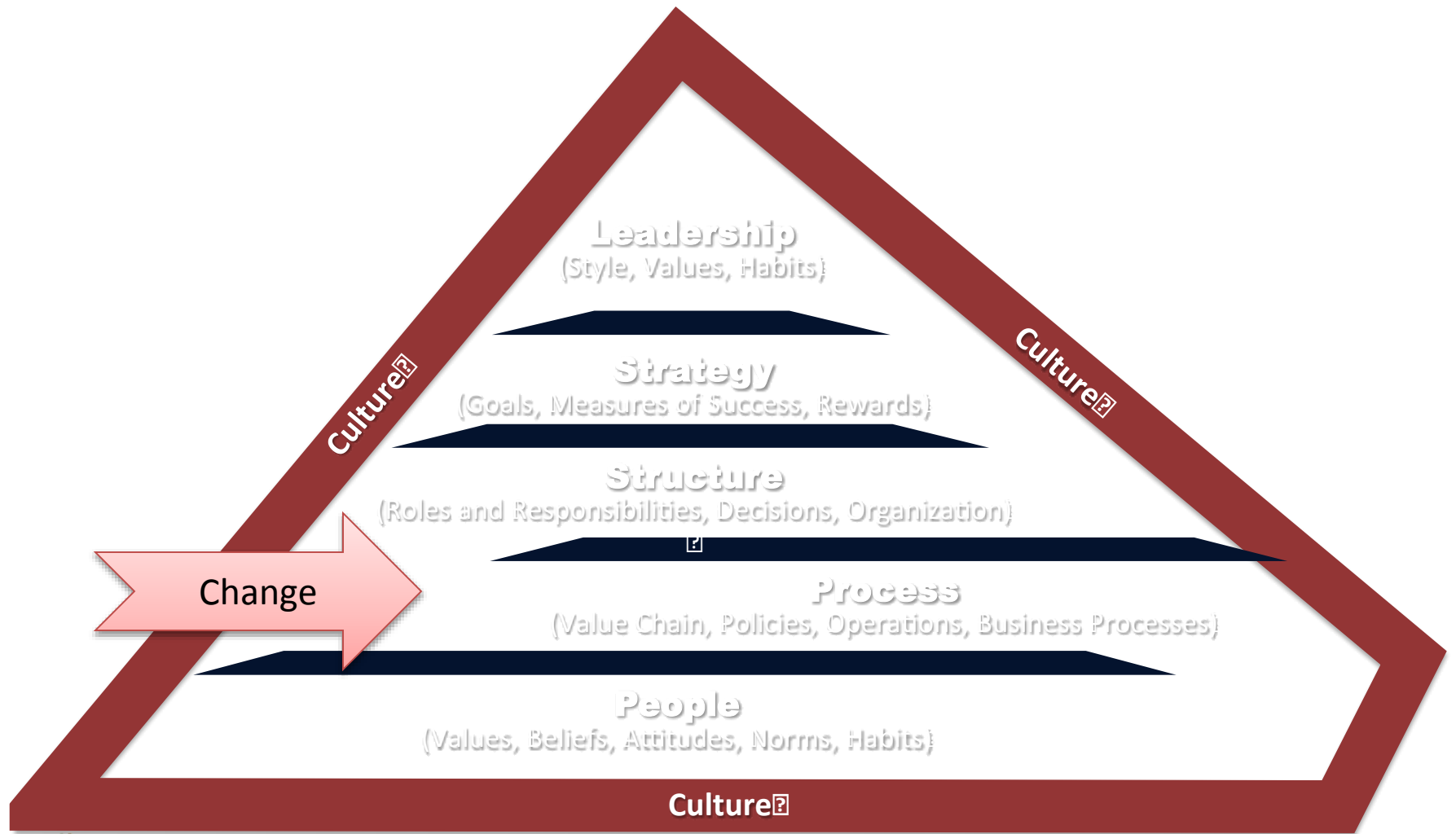
Enterprise Agile is a **culture** based on the *values and principles* of Agile, supported by the *organizational ecosystem* and manifested through *personal and organizational habits* (how work really gets done around here).

An Organizational Ecosystem consists of its:
Leadership, Strategy, Structure, Processes and People

Culture: The Organizational Ecosystem



Process-based Transformation



Culture needs to be Aligned

Collins & Porras studied:

- 18 “visionary” vs. comparisons

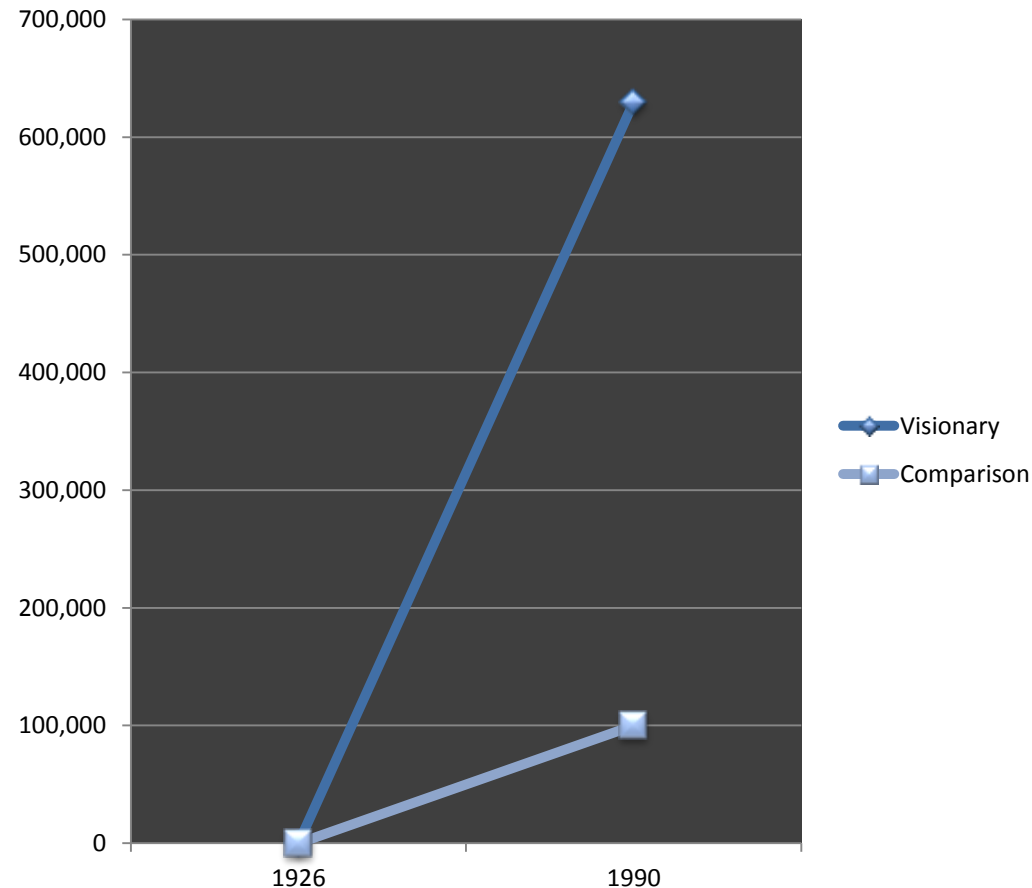
Key distinguishing factor:

- presence of a **Strong, Integrated and Consistent Culture**

Most critical differentiating factor:

- **Alignment** – where all elements of the organization work in concert

Stock Market Performance of Visionary vs. Comparison Companies (\$100 invested)



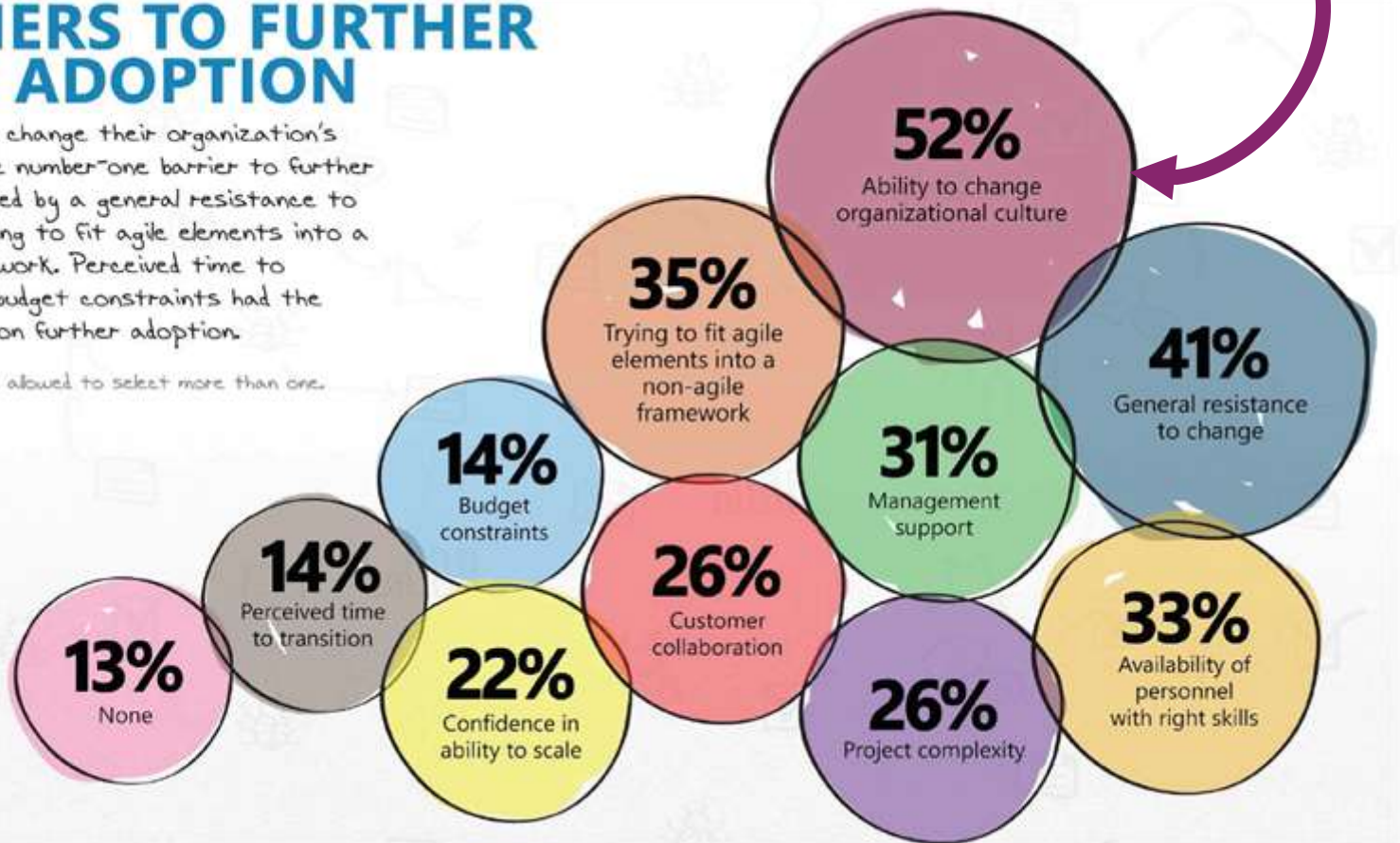
2012 Survey – Barriers to Agile Adoption

Ability to change the culture is the #1 barrier to further agile adoption
4 out of the past 6 years

BARRIERS TO FURTHER AGILE ADOPTION

The inability to change their organization's culture was the number-one barrier to further adoption, followed by a general resistance to change and trying to fit agile elements into a non-agile framework. Perceived time to transition and budget constraints had the lowest impact on further adoption.

*Respondents were allowed to select more than one.



You can't buy a culture transformation. It is hard work from within the Organization

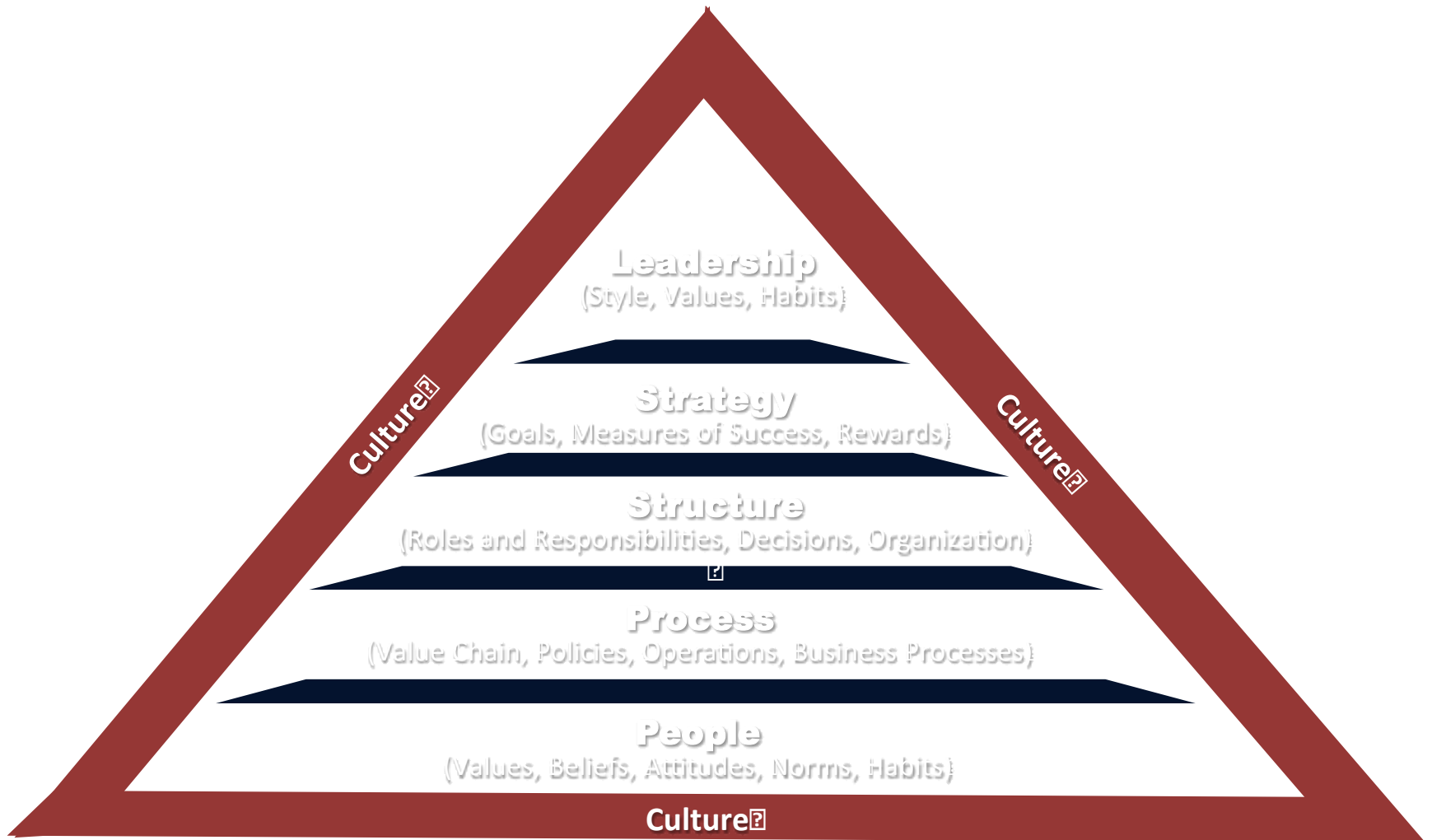


Key Question:
Do you want temporary
change or sustainable
transformation?

It's a change of lifestyle – It's
a change of Mindset



Culture: The Organizational Ecosystem

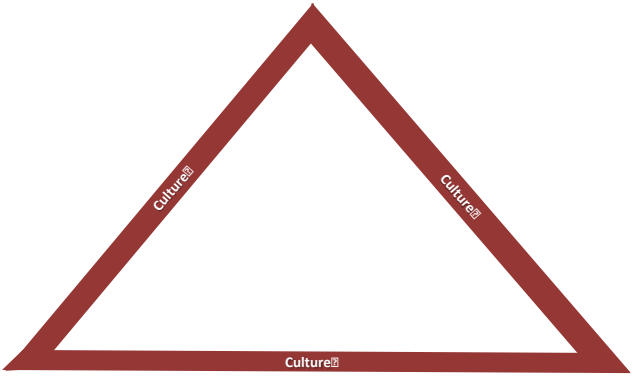


Sustainable Transformation – The Entire Ecosystem has a Common Journey



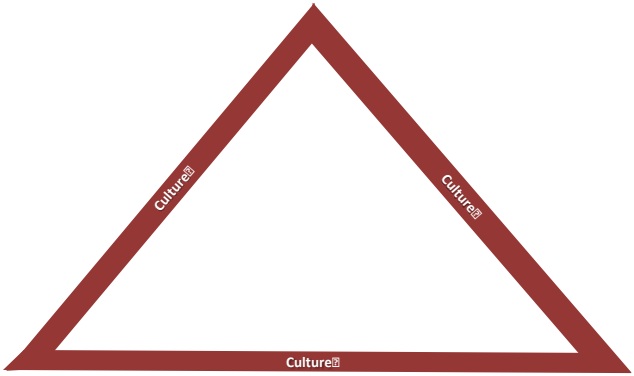
Education (ICAgile Roadmap)
Leadership Coaching
Personal Coaching

Consulting
Organizational Coaching



Monitoring Alignment
Measuring Impact
Assessing Culture and Mindset

Sustainable Transformation – The Entire Ecosystem has a Common Journey

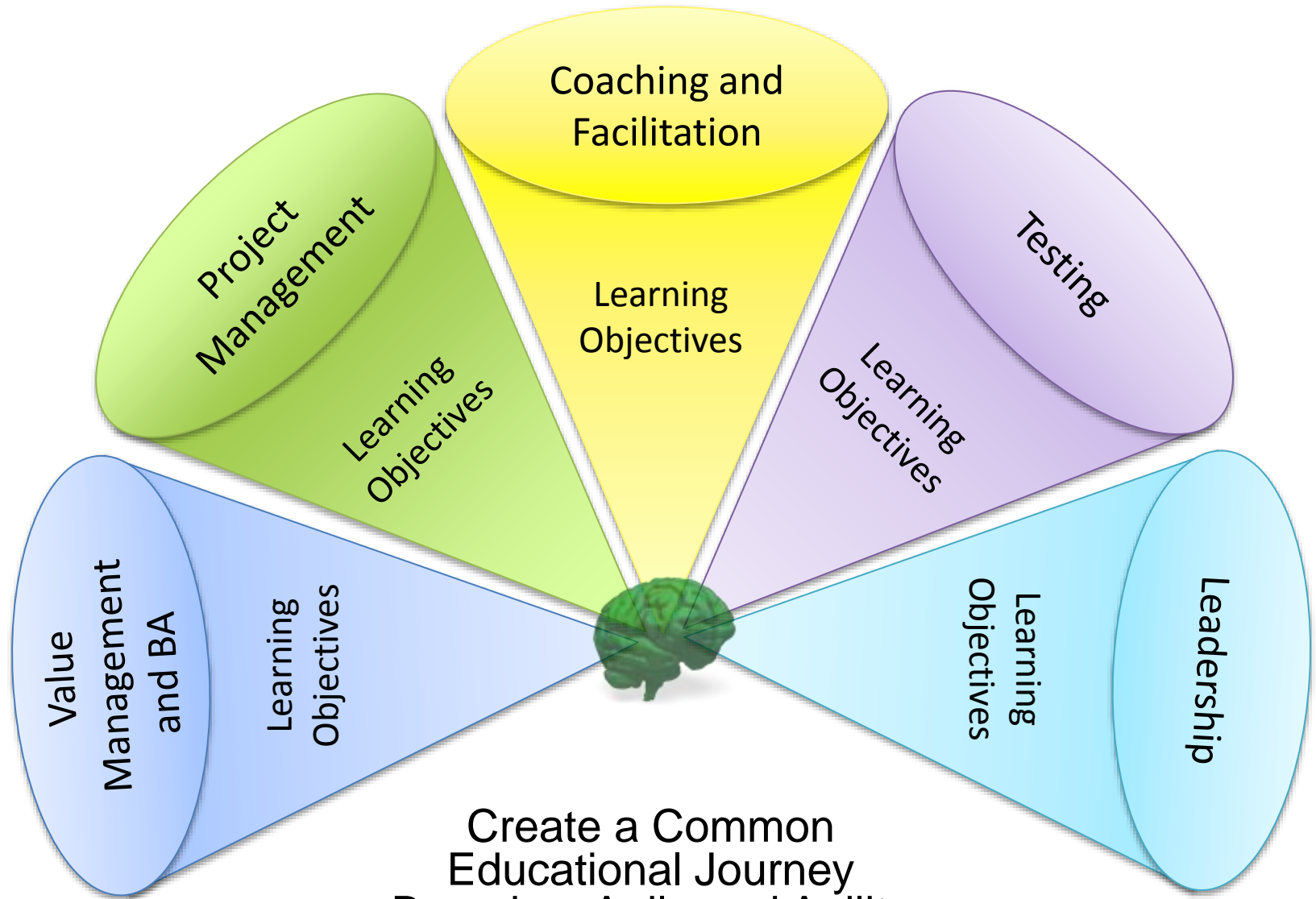


Education (ICAgile Roadmap)

Leadership Coaching
Personal Coaching

Consulting
Organizational Coaching

Monitoring Alignment
Measuring Impact
Assessing Culture and Mindset



Create a Common Educational Journey Based on Agile and Agility (not Scrum and Process)

ICAgile's Roadmap



The Agile Mindset for all Disciplines
inside Software Development

How ICAgile Helps your Educational Journey

1. Helping Define what to Learn by Engaging Agile Experts to create learning objectives for Each Discipline
2. Ensure Quality Education by Accrediting Courses for Training Organizations And Universities
3. Recognizing the education through Certifications



1. Helping Define what to Learn by Engaging Agile Experts to create learning objectives for Each Discipline

Ahmed Sidky

Alex Kell

Alistair Cockburn

Ben Butler

Bob Galen

Brian Corrales

Chris Turner

Christian Hargraves

Cindy Shelton

Claire Moss

Curt Hibbs

Dan Mezick

Dennis Stevens

Derek Huether

Elisabeth Hendrickson

Eric Jacobson

Erin Beierwaltes

Gerard Meszaros

Jeff "Cheezy" Morgan

Jeff Nielsen

Jeffery Payne

Jennifer Stone

Jon Stahl

Kevin Steffensen

Larry Cooper

Laurie Reuben

Lyssa Adkins

Marsha Acker

Michael "Doc" Norton

Michael Spayd

Michi Tyson

Mike Burrows

Mike Griffiths

Olav Maassen

Paul Mahoney

Pete Behrens

Randy Rice

Richard Turner

Sally Elatta

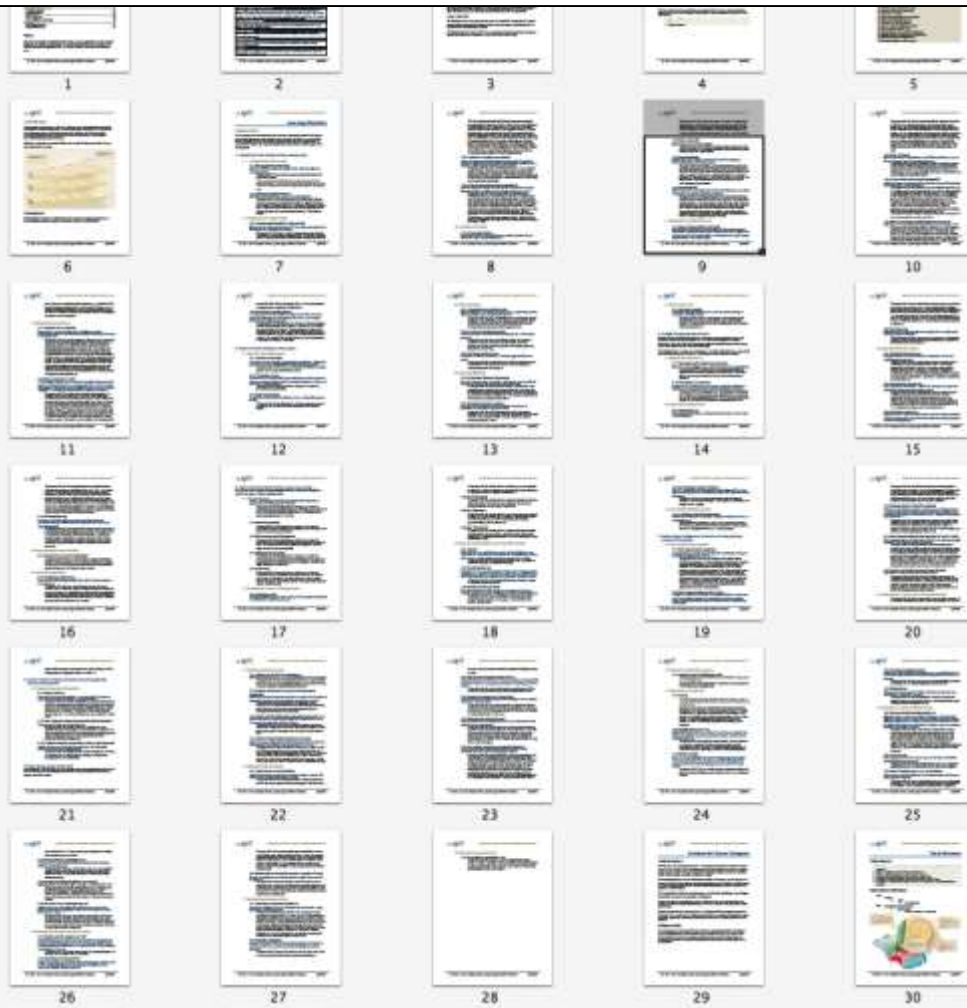
Shane Hastie

Sharon Robson

Venkat Subramanian

and many more ...

The Learning Objectives



learn practical advice on how to do so.

1.3.3 Servant Leadership

Effective Agile coaches employ a "servant as leader" style when they take up leadership with people, teams and organizations.

The purpose of this LO is to explain and make practical the concept of servant leadership, going back to the roots of the idea as originally conceived by Robert Greenleaf. Specifically, that servant leadership means more than getting the team coffee and pizza and, in fact, guides an Agile Coach's behavior in the proper application of "servant as leader" to build capacity in others, to remove one's self from the center of the action and attention and to serve the best interests of what's emerging in the environment.

1.3.4 Key Mindset shifts

Effective Agile coaches successfully make some key mindset shifts and serve as a living example of how one can thrive within these new mindsets.

The purpose of this LO is to expose the Agile Coach to key mindset shifts and help the coach see ways to live these shifts in their own lives. Key mindset shifts may include: focus on team improvement over specific results; focus on business value-driven delivery over achieving scope, schedule or budget targets; focus on the leverage in the present moment over the past or future; focus on staying curious and seeing the best in people over judging or manipulating; focus on assisting the team in achieving their commitment and learning when they do not achieve it rather than stepping in and doing it for them.

1.4 Responsibilities and Skills of the Coach

1.4.1 Roles and Responsibilities of the Coach

Effective Agile coaches know the parameters of their job. They avidly take up their responsibilities and help others take up theirs. They clearly articulate the difference (or overlap) between their role and that of others, such as product owner, project manager, program manager and functional manager.

9 | This track was developed with the generous support of Software Education.

April 2013



"Certification is the by-product; Learning is the product."

The purpose of this LO is to list the duties and skills (as opposed to the mindset) of the Agile Coach job and contrast them with the duties of other roles (i.e. product owner, project manager, program manager and functional manager) so that the coach fully understands the difference between a coach and other team members and can negotiate their role within a given context of other roles in their organization so they can healthfully coexist and help others fully take up their roles in ways that enhance the practice of Agile. Note: In some organizations the coach may fill one or more additional roles on the team, in which case the

2. Ensure Quality Education by Accrediting Courses for Training Organizations And Universities and Corporations



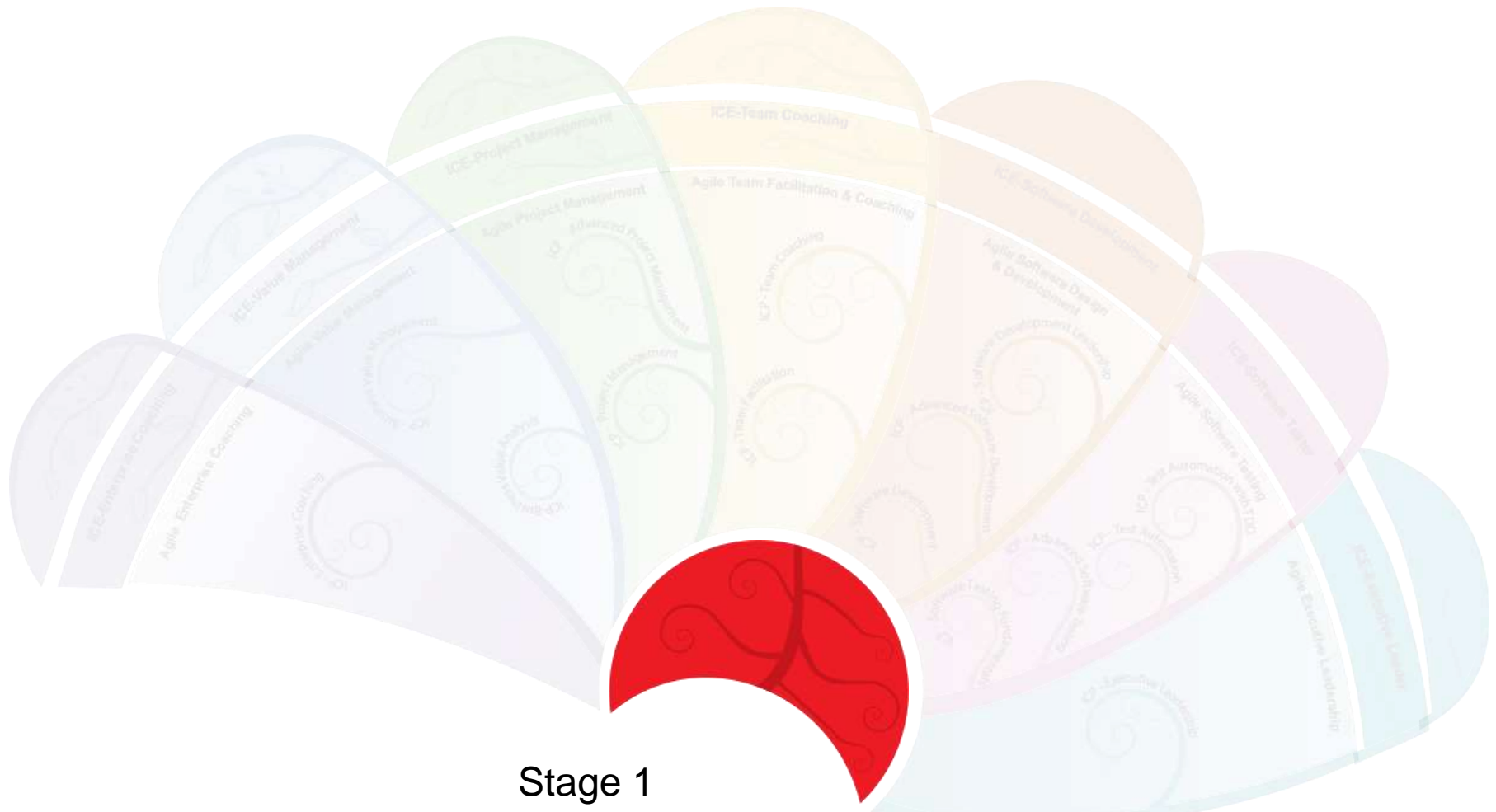
Accreditation Process

Full	2.2.3. Facilitating full participation		1) During: Opening the Session: Delivering a POWER Start (Day 1) 2) During: Opening the Session: Using Check-Ins (Day 1) 3) Information Gathering Techniques Grouping (Day 1) 4) During: Doing the Work: Engagement Strategies	All the exercise we do throughout "demo" full participation techniques. Co-leads will meta-comment to make this clear.	1) Practice "Exciting" as part of a POWER Start 2) Weather Check 3) Brainstorming Exercise 4) Evening Practice - Facilitating a Retrospective				
2.3. Facilitating Collaboration		Handouts	Slides	Exercises	Instructors Guide	Other	Explanation	Assessment	Reviewer Comments
Partial	2.3.1. Facilitating collaborative conversations		Throughout: Dealing with Dysfunction: The DiSC Model (Day 1) Other Collaborative Conversations (Day 2)		Walk the DiSC Model				Something about giving and receiving feedback
Partial	2.3.2. Facilitating team decision-making								Scope of authority is covered - decision making is not complete
2.4. Facilitating a Meeting		Handouts	Slides	Exercises	Instructors Guide	Other	Explanation	Assessment	Reviewer Comments
Full	2.4.1. Facilitating a meeting				1) Evening Practice - Facilitating a Retrospective 2) Release Planning				
3. Skillfully Facilitating the Agile Practices		Handouts	Slides	Exercises	Instructors Guide	Other	Explanation	Assessment	Reviewer Comments
3.1. Setting the Facilitation Context		Handouts	Slides	Exercises	Instructors Guide	Other	Explanation	Assessment	Reviewer Comments
Full	3.1.1. Choosing the level of facilitation intervention		On each "Meeting Rundown" slide (Day 2)						
Partial	3.1.2. Protecting the team boundary		Your Role: The Agile Facilitator Stance (day 2)		See "Learning Points" under Standup section		See notes on slide: Your Role: The Agile Facilitator Stance (day 2)		How to protect the team boundary - the words to use - signature phrases
3.2. Facilitating Chartering Activities		Handouts	Slides	Exercises	Instructors Guide	Other	Explanation	Assessment	Reviewer Comments
Full	3.2.1. Project chartering		Project Chartering: Meeting Rundown (Day 2)						
Full	3.2.2. Team chartering		Team Chartering: Meeting Rundown (Day 2)						
3.3. Facilitating Collaborative Meetings		Handouts	Slides	Exercises	Instructors Guide	Other	Explanation	Assessment	Reviewer Comments
Full	3.3.1. Facilitating release planning		Release Planning: Meeting Rundown (Day 2)		Release Planning				
Full	3.3.2. Facilitating iteration planning		Iteration Planning: Meeting Rundown (Day 2)		Iteration Planning				
Full	3.3.3. Facilitating retrospectives		Retrospectives: Meeting Rundown (Day 2)		1) Evening Practice - Facilitating a Retrospective 2) Retrospectives Standups				
Full	3.3.4. Facilitating stand-ups		Standups: Meeting Rundown (Day 2)						

3. Recognizing the education through Certifications



3. Recognizing the education through Certifications



Stage 1

Stage of Education: Intent to Learn

Certification: ICP (ICAgile Certified Professional)

Validation: Sufficient display of intent to learn agile (not Scrum, XP, Lean, etc.).
Attending a 2 or 3-day class on **Agile** is sufficient display of intent.

Meaning: The certified person has demonstrated the intent to learn and be a professional in the agile space (not only Scrum, XP, Lean, etc.)

3. Recognizing the education through Certifications



Stage 2

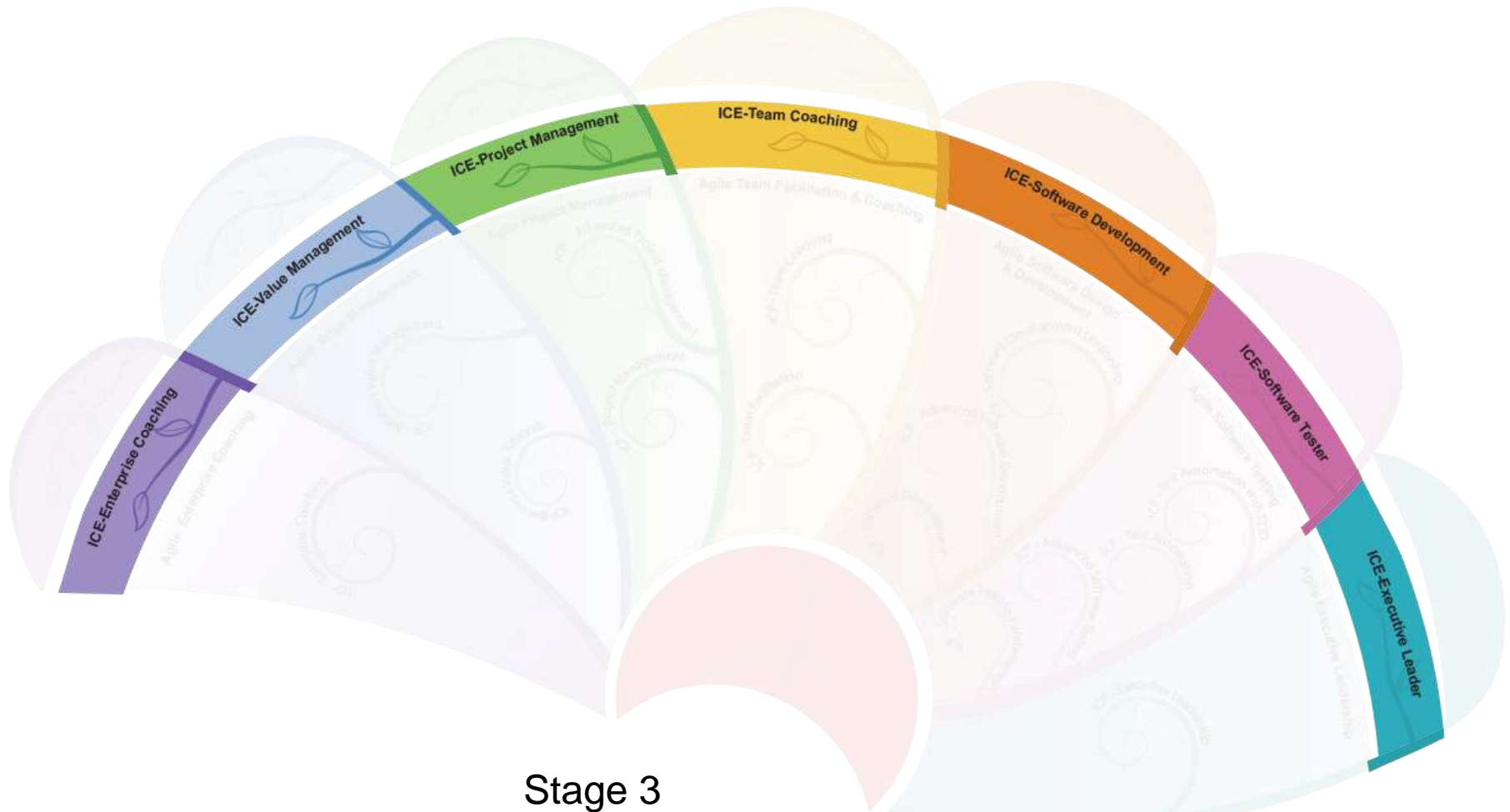
Stage of Education: Actively Acquiring Knowledge

Certification: ICP-X (ICAgile Certified Professional - Extension)

Validation: Instructor decides how to validate the knowledge acquisition. Informs ICAgile and Students and assesses knowledge acquisition during class.

Meaning: The Certified Person has extended their intent to learn and acquired agile knowledge pertaining to a specific discipline or domain

3. Recognizing the education through Certifications



Stage 3

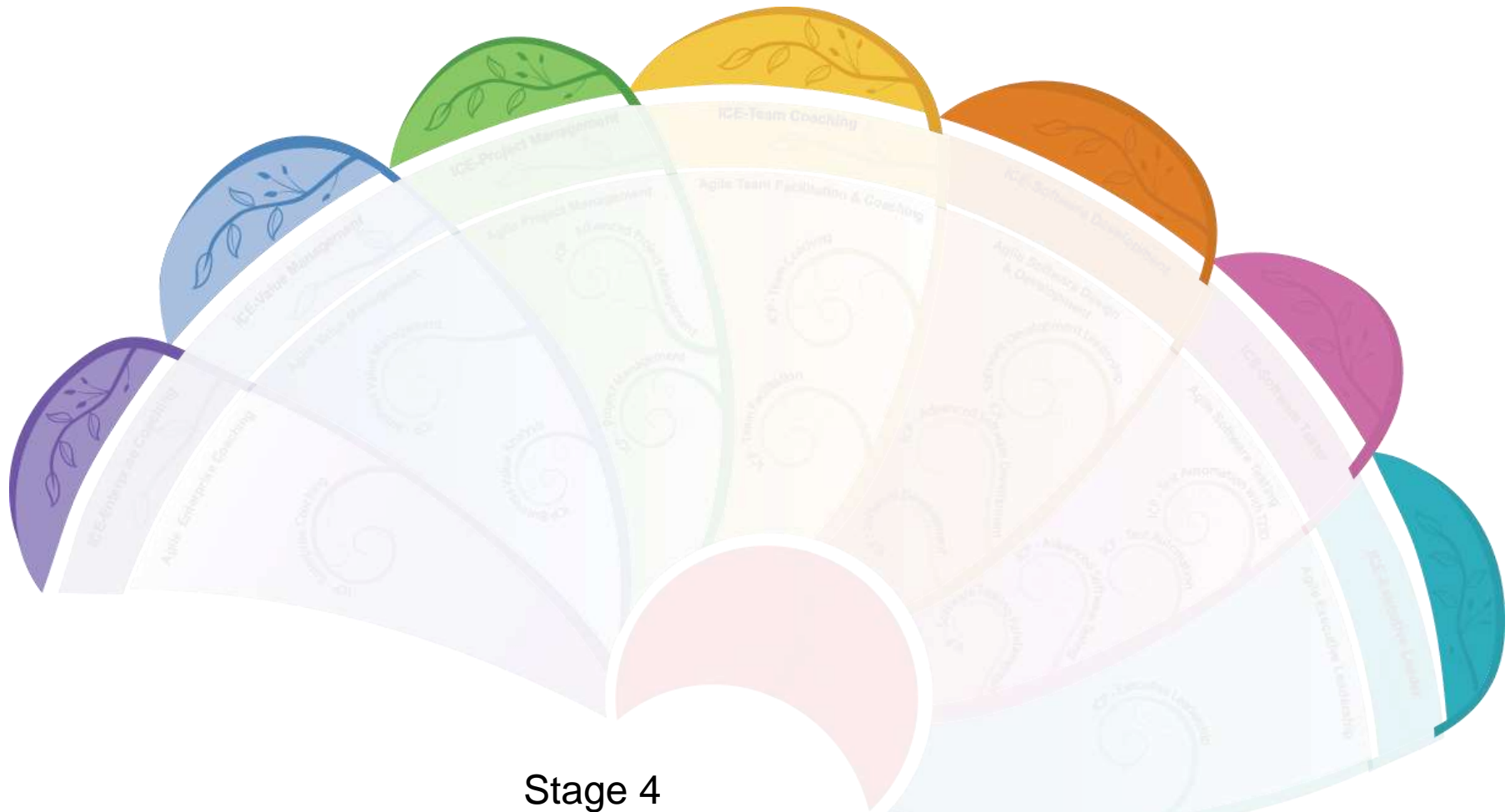
Stage of Education: Developed Knowledge into Competency

Certification: ICE (ICAgile Certified Expert)

Validation: Through a gate submission and review process entailing a presentation of knowledge coupled with an assessment of competency.

Meaning: The Certified Person has demonstrated knowledge and competency (and some experience) in a specific discipline

3. Recognizing the education through Certifications



Stage 4

Stage of Education: Maturing Competency into Proficiency

Certification: ICM (ICAgile Certified Master Agilist)

Validation: Under development: Entails an in-person display of competency, client testimonials in addition to other assessments.

Meaning: The Certified Person has demonstrated knowledge and proficiency along with extensive experience in multiple agile disciplines and domains.

3. Recognizing the education through Certifications



1. Intent to Learn → ICP
2. Actively Acquiring Knowledge → ICP-x
3. Developed Knowledge into Competency → ICE
4. Maturing Competency into Proficiency → ICM

Disciplines – not roles

- Enabling individuals with knowledge in all areas and disciplines they need to succeed, not molding them into specific roles
- Supporting the Generalizing Specialist Model



Current Tracks

- Agile Software Design & Development
- Agile Software Testing
- Agile Team Facilitation & Coaching
- Enterprise Coaching
- Agile Project Management
- Value Management & Business Analysis
- Executive Leadership

Agile Education Transcript

Timothy Meyers

ICAgile Certified Professional
Completed July 2012

- March 2011, Fundamentals of Agile by Sankar Group
ICAgile Certified Professional
- July 2012, Fundamentals of Agile PDB-ACP Free by Sankar Group
ICAgile Certified Professional

ICAgile Certified Expert in Team Coaching
Still Pending Approval

- November 2012, Facilitating Agile Teams by Sankar Group, Team Catalyst
ICAgile Certified Professional - Team Facilitation
- February 2013, Coaching Agile Teams (ICAgile Coach Certification Edition) by Agile Coaching Institute
ICAgile Certified Professional - Team Coaching

Certifications Completed

Status towards Expert Level

Transcript

Agile Fundamentals 100% Complete

History | Culture & Mindset | Creating Shared Understanding | Work in Flow | Holistic Development | Work in progress (WIP) | Including Customers and Users | Product Adaptation | Planning and Adapting | Process & Project Adaptation

Team Facilitation & Coaching 100% Complete

The Agile Team Facilitator Mindset | Development Path for Agile Coaching | The Agile Coaching Mindset | Responsibilities and Skills of the Coach | Setting Boundaries for Coaching | Facilitating on & the Fly (Coach Stories) | Facilitating Meetings | Facilitating Collaboration | Facilitating a Meeting | Setting the Facilitation Context | Facilitating Challenging Activities | Facilitating Collaborative Meetings | Designing Meetings for Team Interaction | Facilitating an Agile Practice | The Coaching Stories | The Coaching Conversation - Coaching for Action | Professional Coaching Skills | Conducting the Coaching Conversation | Mentoring Agile Roles & Travel Stories | Mentoring vs. Coaching | Teaching the Agile Service & Mindset Shift | Understanding Team Development | Setting up the Team Environment | Creating a Team Work/Off/Retreat Agenda | Characteristics of an Agile Team | Coaching the Emerging (New) High Performer | Resolving Conflict and Dysfunction within the Team | Handling Organizational Impediments | Planning in Iterations is a skill for a team

Value Management & Business Analysis 0% Complete

Value Management as an Agile Team's Goal | Agile Business and Strategy | Attributes of the Role | Thinking Skills | Deliverables | Defining Value | Delivering Value | Communicating Value | Understanding Stakeholders | Performing Analysis | Exploiting the Business | Understanding Feedback | Managing Interfaces | Managing Risks | Incentives & Profits

Testing 43% Complete

History of Agile Testing | Testing Mindset & Culture | Testing Non-Functional Requirements | Integration and System Testing | User and Product Testing | Subsequent Testing (aka Testing to Support Development) | Categories of Testing | Work on an Embedded System | Test Environments and Infrastructure | Testing (Data) Networks | Roles and Responsibilities | Test Strategy and Planning | Test to Support DevOps | Test to Support DevOps | Automated Non-Functional Testing | Automated Support for Integration and System Testing | Automating DevOps and Feature Testing | Automated UI/UX/Content Testing | Continuous Integration | Test Automation Strategy | Integrating with other Workflows in DevOps (DevOps/CI/CD) | Delivering Continuous and Policy Requirements | Organizational Structure | Culture Challenges

Software Design & Development 0% Complete

Test Driven Development | Design Thinking | Collaborating | Technical Debt | Build Decide | Programming the Work | Automated Testing | Test Activities | Collective Responsibility | Collaboration | Planning | Execution-Based Development | Learning from

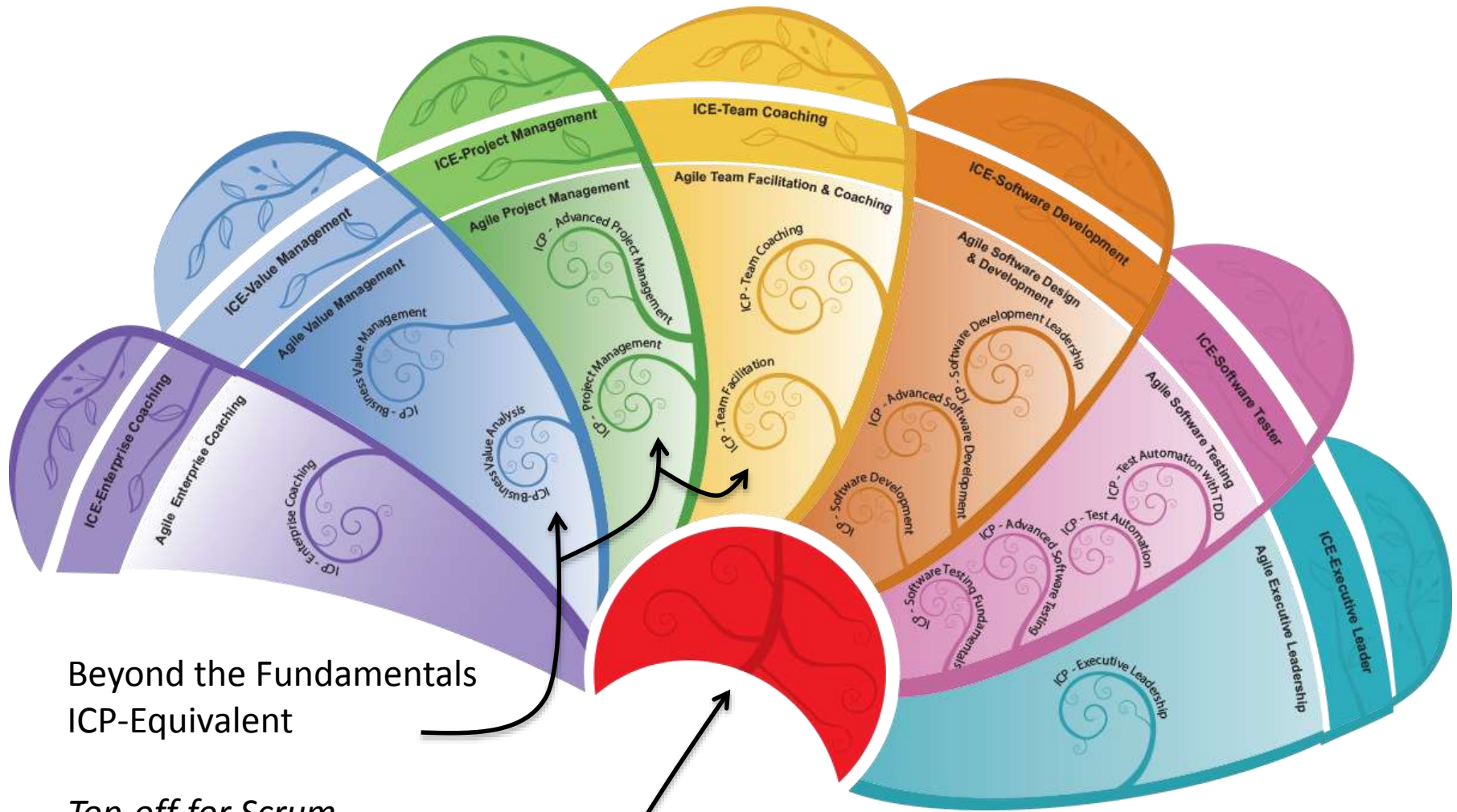
Progress within Tracks

Selected Learning Tracks

Learning Objective Completed

Learning Objective Not Yet Completed

Getting Into the Program



Beyond the Fundamentals
ICP-Equivalent

Top-off for Scrum

Start with the Fundamentals
ICP

Thank you

Questions?