

Engagement & Services Proposal

Gene and Amitai: world-class Agile consultants right here in New York.

Armed with complementary skills and experiences to meet your unique needs.



Gene Gendel is Agile Coach, Trainer and Organizational Design Agent. Gene is a proud member of the small community (about 100 people worldwide) of Scrum Alliance Certified Enterprise Coaches (CEC). Today, he is the only CEC who resides in NY State. Gene's goal is to help organizations and individual teams with improving internal dynamics, organizational structure and overall efficiency.

Gene strives to engage at all organizational levels: senior- and mid-level management, teams and individuals. Throughout his long career, Gene has served small, mid-size and large companies, domestically and abroad. Gene is a well-recognized member of global and local agile communities, where he influences people via open-space agile collaboration workshops, coaching retreats, group events and presentations. Gene is a well-recognized blogger and publisher. He is the co-author of the book Agile Coaching: Wisdom from Practitioners and Best Agile Articles of 2017.

Gene's additional references:

- Public profiles: [LinkedIn](#), [ScrumAlliance](#), [Google Search](#)
- Personal Profile: [One-Page Bio](#), [Resume](#)
- [One-Page Services Summary](#)
- [Top-3 Case Studies](#)
- [Training bundle deal with another trainer - example of how two people can combine their efforts.](#)

Amitai Schleier is a software development coach, helping teams learn to deliver more effectively by growing together. In 20 years comprising global finance, startups, agriculture, universities, and nonprofits, Amitai has managed products, projects, and people; engineered code, tests, and releases, deployed and operated production systems; provided phone and desk support; and listened to, empathized with, and improved the lot of folks in all of these roles.



Amitai has seen well-chosen, well-executed technical practices minimize waste and reduce the risk and cost of change -- and his technical coaching can guide your development teams through integrating tools and acquiring habits to more rapidly deliver value, more promptly adjust to changed priorities, and more directly contribute to your business. Amitai's highly regarded micropodcast, Agile in 3 Minutes, has helped many teams he's never met and drawn praise from many of Agile's founding figures.

Amitai's additional references:

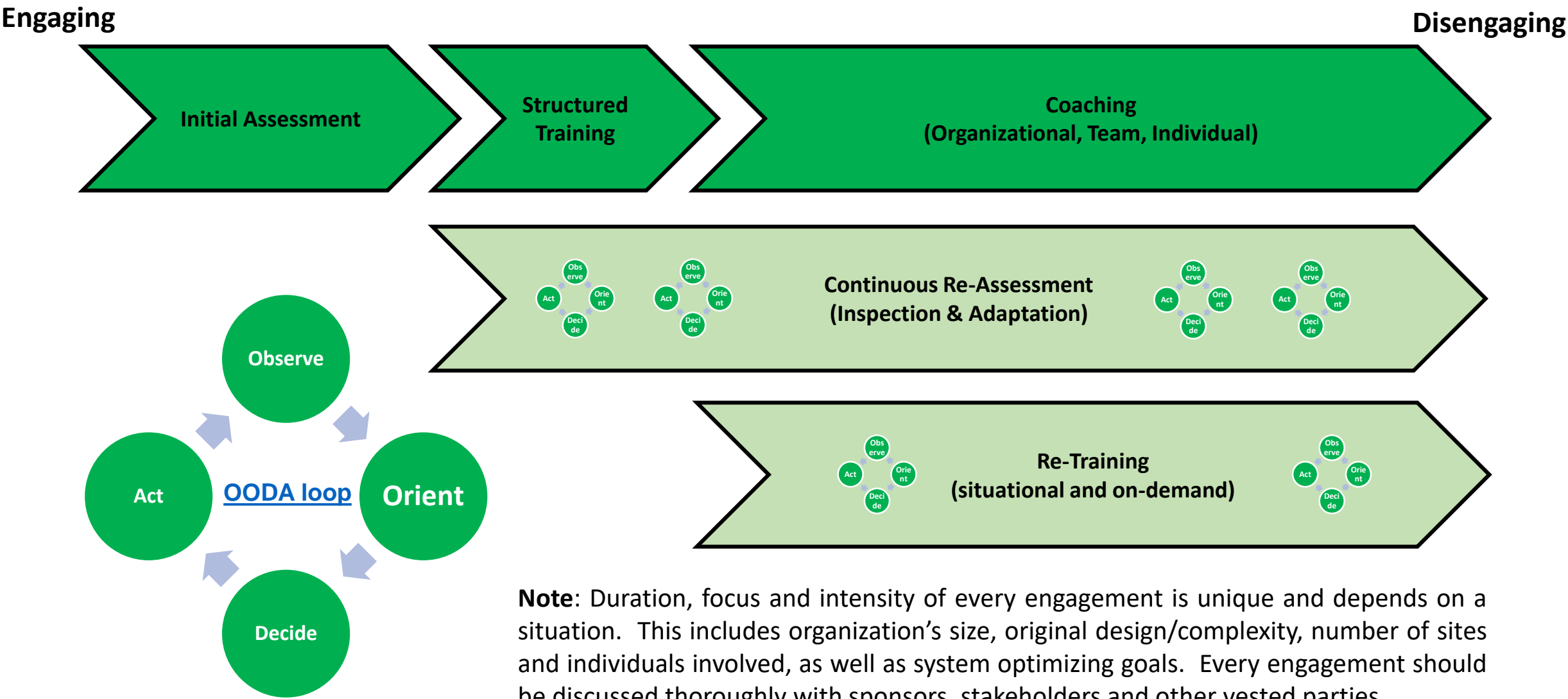
- Consulting site: <https://latentagility.com/>
- Coaching bio: <https://schmonz.com/coach/>
- Micropodcast: <https://agilein3minut.es/>
- Resume: <https://schmonz.com/resume/>
- Speaker bio: <https://schmonz.com/speaker/>
- Talks I've given: <https://schmonz.com/tag/talk/>
- Podcasts and other appearances: <https://schmonz.com/tag/not-talk/>
- Summer coding tour: <https://schmonz.com/2018/09/15/coding-tour-summer-2018-conclusion/>
- Example of open-source work: <https://schmonz.com/qmail/>

Both Gene and Amitai can be contacted via email at: gatogether@latentagility.com

ROI of Paired Coaching

- Overlapping but different personal styles
- Overlapping but different life and professional experiences
- Overlapping but different areas of focus
- Division of labor (often)
- Two heads better than one (very often)
- **Greater effectiveness in more situations, AND on average**
 - The success formula is **NOT**: Your Organization + Gene + Amitai
 - The success formula is: Your Organization x Gene x Amitai

Engagement Process (Simplified)



Initial (Re-)Assessment

Source: <http://www.keystepstosuccess.com/agile-assessment/>

Assessments (a.k.a. health checks) are an effective way to monitor organizational progress through [agile transformation](#). Properly identified and carefully monitored agile maturity metrics (AMMs) could be an effective “lever” to steer an organization towards success. However, some challenges could be caused by collecting and applying inappropriate metrics and then performing unskilled assessments that are based on misleading numbers (note: please beware of some important issues caused by [AMMs](#)). It is not uncommon for an organization to focus on metrics and other locally collected numerical attributes and miss out on much bigger picture.

Some organizations, with cultures that are driven by KPIs, scorecards and individual performance assessments, over-stress the meaning of single numbers ("checks and balances"), instead of trying to understand underlying causes of dysfunctions (e.g. organizational design, team dynamics and structure). Such organizations zoom in on trivial elements and overlook key elements: focusing too much on local indicators of success and missing out on global factors ([local optimization at expense of system optimization](#)).

Before identifying and collecting metrics to perform an assessment, organizations must answer the following questions:

1. What decisions will organization make based on metrics collected?
2. Does organization measure the right thing or the easiest thing that can be collected?
3. What is an intention of collecting a given metric?
4. Who is collecting metrics and why?
5. Who is analyzing metrics? Is analysis done skillfully, by someone with experience?
6. What happens to metrics as they travel up through multiple organizational layers?
7. What if there are some unintended consequences of collecting and/or misinterpreting a given metric?
8. Is a given metric a variable that can be measured directly or is there a proxy-variable that has to be measured instead?
9. How could metrics be “played or gamed”?
10. What bad behaviors (e.g. “cooking books”/manipulating metrics) could be expected?

Metrics can be collected and be indicative of conditions at different organizational levels. Here are examples:

- **Team level assessment** – indicative of condition at single team level
- **Multi-team assessment** – indicative of conditions across multiple teams
- **Executive Management assessment** – typically, assesses multiple organizational units and departments

“Higher-level” metrics (e.g. multi-team, executive) could, potentially, be derived by rolling up lower-level (team) metrics. However, caution must be exercised, while mixing up non-mixable units of measure. Not all metrics are comparable, unless appropriate normalization is applied. Some metrics are qualify-able but non-quantifiable and cannot be assessed "by a machine"; only an experienced person can give an objective interpretation to a non-quantifiable metric.

Below are some examples of **Team-Level** metrics that are frequently used to assess single or multi- Scrum team maturity. Some of them could be parametric (numerical), others - binary (yes/no). As time goes by weight of various metrics’ values are expected to change. (Please see various types of metrics on the next page)

Initial (Re-)Assessment

Source: <http://www.keystepstosuccess.com/agile-assessment/>

Scrum Team Dynamics and Structure:

- What is a team's "happiness factor"? What is overall team's morale and motivation?
- Is being a team member viewed as an opportunity, a career path? Or is it rather perceived, as a constraint, limitation and a burden?
- Are internal relationships among team members healthy?
- Team size – is number of individuals on a team optimal?
- Is human resource "churning" (attrition rate) high?
- Are there hierarchical relationships on a team that prevent good teaming and jeopardize individual safety?
- Is a team cross-functional: does it have all subject matter expertise necessary to perform work?
- T- Shaping individuals: are cross-functional team members present on a team?
- Are key roles well understood and supported by senior management (e.g. ScrumMaster, Product Owner, Team member)?
- Dedicated resources: are team members shared with other teams or distracted from sprint work?
- Are team members collocated? If distributed – how?
- Is a team self-organized or does it require management from outside?
- If multiple teams are involved, is there an effective multi-team synchronization?
- When impediments are discovered, how effectively are they being removed?
- Can a team effectively limit WIP (work in progress)?

Product Delivery by Scrum Team:

- Has Product Owner produced a clear Product vision and/or Strategic Product Road-map?
- Is product PBI (potentially shippable increment) produced at the end of each sprint/iteration?
- Is each PBI properly sliced (vertically) , sized and "INVEST-able"?
- Are DoR (Definition of Ready) and DoD (Definition of Done) clearly defined by a team and PO?

Work Cadence and Logistics of Scrum Team:

- Are Daily Scrums (Daily Stand-ups) effective?
- Do Product Backlog Refinement sessions happen regularly and are they effective?
- Do Sprint Showcases produce valuable customer feedback?
- Do Team Retrospectives lead to continuous improvement?
- Is a team focused only on Scrum work?
- Is sprint work feature-centric (does it produce customer value)?
- Has a team developed reliable estimation techniques and has it's ways to track progress?

Agile Engineering Practices (DevOps):

- Is Architecture flexible to accommodate potential future changes?
- Does a team use TDD or BDD?
- Is Continuous Integration in place?
- Is there full test coverage of code-base? Code refactoring?
- Are unit tests present?

(For single Scrum Team dynamics, please also refer to [Henrik Kniberg's Scrum Check List.](#))

Every metrical unit that is used for an initial assessment or subsequent maturity assessments must be clearly understood and thoughtfully applied by experienced individuals. In cases, of agile scaling, additional care must be exercised to collect and 'roll-up' metrics from multiple organizational verticals and product development areas, in ways that do not cause confusion and misinterpretation of data.

If you would like to understand some of the most commonly seen mistakes with using agile maturity metrics and deepen your personal system-level understanding of observed team dynamics, please visit [this page](#).

A good way to start an assessment is to conduct an informal [Lunch & Learn session](#). Please, use the form below to ask me a question or provide your feedback. Thank you.

Training

Source: <http://www.keystepstosuccess.com/agile-training/>

Training is a great way to deliver structured, logically sequenced education to a person or a group of individuals. Intensity of trainer-trainee interaction varies. In *less interactive training* (often used with larger audiences), trainer-trainee interaction is typically limited to Q&A. In *more interactive training* (workshop-style, used with smaller audiences, study groups, classes) trainer-trainee interaction may include games, role play, break out sessions, presentations, in-class assessments and other engaging techniques.

Below is a comprehensive curriculum of training topics that are recommend to organizations, teams and individuals in the form of a comprehensive multi-day training course or multiple, independent topic-focused workshops, spread out gradually through time. With the latter option, it is recommend to deliver topic-based training sessions in a logical order - to minimize knowledge dependency and maximize information retention.

Agile Overview (Introductory)

•Organizational Agility

- What does the word “Agile” really mean?
- [Organizational “Color”](#) (as per Frederic Laloux)
- [Organizational “Tribal Stage”](#) (as per David Logan)
- [Organizational “Motivation Version”](#) (as per Daniel Pink)
 - Frequently ignored scientific evidence
- What do companies value most?
- [Do you really want Agile Transformation?](#)

•Thinking Wide & Deep

- Lean Thinking
 - “Watching the baton, not the runners.”
 - Foundation of Lean: Managers as Teachers
 - 2 Pillars of Lean:
 - Respect for People
 - Continuous Improvement
- Lean Thinking House
- 14 Principles of Lean
- System Thinking
 - Causal Loop Diagrams (CLD)
 - Seeing System Dynamics and Local Optimization with CLDs

•The Laws”:

- [Conway’s Law](#)
- [Brook’s Law](#)
- [Larman’s Laws](#)

• Agile Frameworks

- [Scrum \(basic\)](#)
- [Kanban \(basic\)](#)
- Popular scaling solutions (high-level overview)
 - [Large Scale Scrum \(LeSS\)](#)
 - [Nexus](#)
 - [Scaling @ Scale](#)
 - [Scaled Agile Framework \(SAFe\)](#)
- Market adoption of agile frameworks
 - Market Penetration (statistics)
 - [Potential Hidden relationships](#)
 - Organizational implications of scaled solutions

Scrum

- Definition and Theory
- Scrum Roles & Responsibilities
 - Overview of Scrum Roles
 - Product Owner: responsibilities, authority/empowerment
 - ScrumMaster: responsibilities, authority/empowerment
 - Development Team: size, composition, dynamics
 - Impact on Traditional Roles: PM, BA, Manual Tester, Functional Leads
 - Business Community: SMEs, Stakeholders, end-Customers
- Scrum Events/Ceremonies
 - Sprint
 - Sprint Planning Meeting
 - Daily Scrum Meeting
 - Sprint Review/Showcase Meeting
 - Sprint Retrospective Meeting
 - Product Backlog Refinement Meetings (PBR)
 - Release Planning Meeting
 - Release Retrospective
- Scrum Artifacts
 - Product Backlog
 - Creation/Management/Maintenance
 - Prioritization/Estimation
 - Segmentation/Logical Grouping
 - Backlog “Views”: Product, Release, Sprint
 - Sprint Backlog
 - Product Increment
 - Potentially Shippable Product Increment (PSPI)
 - Minimal Viable Feature (MVF)
- Scrum “Under the Hood”
 - Agile Requirements Management
 - Triple Constraint Triangle of Conventional PM
 - Ignoring Variability and System Behaviors
 - “Cascading” Lateness, Student Syndrome
 - Velocity vs. Commitment Driven Planning (Releases)
 - Cross-Team Work Dependency
 - User Story writing
 - User Story Life Cycle: “Happy” Path
 - What User Stories Are NOT?
 - How to Formulate a Story?
 - Story Splitting/Combining
 - User Role Modelling, User Story Testing
 - Slicing Stories
 - Story Splitting, Story Sequencing & Mapping
 - User Stories vs. Use Cases, User Story Mapping
 - Important Scrum Elements
 - Card, Conversation, Confirmation (CCC)
 - Definition of Ready(DoR) / Done(DoD)
 - Visualization in Scrum
 - Scrum Board (Story Board, Task Board)
 - Viewing by Work vs. Worker
 - Viewing by worker
 - Communication in Scrum:
 - Internal & external
 - Agile Engineering/DevOps:
 - Continuous Build & CI, TDD, ATDD
 - Unit Testing & Test Automation
 - Refactoring, Technical Debt
- Scrum Tools & Techniques
 - Work Prioritization
 - Work Estimation & Planning
 - By Analogy vs. by Calculation
 - Base-lining work (using reference samples)
 - Order Magnitude (adjusting estimation scale)
 - Estimation units: Ideal Time vs. Story Points
 - Relating Hours to Points
 - When to use each method?
 - Skill set vs. Individual Capacity Management
 - Historical Velocity Trend Analysis
 - Forecasting with
 - Estimation Techniques: Planning Poker
 - Work Volume & Work Complexity
 - Effects Anchoring
 - Planning/Agile Poker
 - Historical Velocity
 - Release Planning (release burn-up)
 - Cone of Uncertainty: Optimistic vs. Pessimistic forecasting

Agile Budgeting & Finance

- Triple Constraint Triangle of Conventional Management
- Why Agilists understand Budgeting better than Financists?
- Why Project Management area is so “protected”?
- Decomposing Budget into:
 - Forecasts
 - Targets
 - Resource Allocation
- Forecasts vs. Targets
- “Rolling” Forecasts vs. Dynamic Forecasts
- KPIs: good vs. bad
- Balanced Scorecards against Budgets – what usually wins?
- “Splitting a bag of cash”
- Does Meeting a Budget Drive Individual Performance?
- What do Monetary Incentives do to People?
- Why do we need Partnership between HR and Finance?
- “Frequently ignored scientific evidence” (Daniel Pink)
- Budgeting Evolution vs. Budgeting Revolution: what is better?

Kanban

- History of Kanban
- Empire State Building
- Continuous Workflow
 - Throughput
 - Work in Progress (WIP)
 - WIP Limits
 - Cumulative Flow Diagram (CFD)
- Pull vs. Push
- Pull System
- One Piece Flow
- Principles of Product Development Flow
 - Little’s Law
 - Managing Capacity of Workers
 - Multi-tasking & Swarming
 - Managing Capacity of Part-Time Workers
 - Reducing System Variability
 - Impact of Variation on Capacity Utilization and Queue Size
 - Throughput and System Utilization
 - Queues and Servers
 - Escalation in Kanban (e.g. L1, L2, L3)
- Enterprise Kanban
 - Upstream & downstream impacts
 - Converting and Diverging work streams
- Examples of Physical Kanban Boards

Agile Scaling

- Scale or NOT to Scale?
- Common Reasons for “Customizing” LeSS
- Voices of “Elders”
- LeSS vs. LeSS Huge
- Types of Product Development
- LeSS Ceremonies
 - LeSS PBR
 - LeSS Sprint Planning
 - LeSS Sprint Review
 - LeSS Joint Retrospective
- Communication in LeSS
- Coordination in LeSS
- Requirements Management
- LeSS Roles
 - LeSS ScrumMaster
 - LeSS Product Owner
- Distribution in LeSS
- Component vs. Feature Teams
- Local vs. System Optimization
- Job Safety !=> Role Safety
- LeSS Definition of Done (DoD)
- Reducing Undone Work
- Why Does Dev Ops Exist?
- Truth about Dev Ops
- The “Contract Game”
- Team Maturity in LeSS
- LeSS Product Ownership
- Fake “Portfolio” Management
- System Feedback Loops in LeSS
- Management in LeSS
- [Causal Loop Diagrams \(CLDs\) in LeSS](#)

Agile Team Techniques

- Continuous Delivery to production
- Continuous Integration to teammates
- Pair programming with partners
- Mob programming as a whole, integrated team

Overview of Agile Tooling

- Where tools can help?
- Where tools can hurt?
- What to look for in agile tools?
- Some commercially popular agile tools
 - Jira
 - Rally
 - Version One
 - TFS/VSTS
- Use of tools in scaled solutions. Hidden relationships

Agile Code Techniques

- Test-Driven Development: express new intentions
- Refactoring: express existing intentions
- Debugging: correct failed intentions
- Improving Legacy Code: lower everyday costs and risks, enabling change

Agile Team-And-Code Outcomes

- Single piece flow
- Shared context
- Just-in-time decisions
- Your people's best work
- Your people's rapid growth
- Easy predictability
- Business continuity

Coaching:

Source: <http://www.keystepstosuccess.com/agile-coaching/>

Any experienced Agile and Scrum coach, who has enough theory and practice under his belt should have in-depth understanding of practices and principles of various agile frameworks and real-world experience of implementing it at organizations. A good coach must have a proven track record of guiding organizations through challenges of agile transformations.

An experienced coach is familiar with both: adoption successes and adoption failures, with the latter being a great “lessons learned” that a coach is not embarrassed to share. A coach with diverse experience that spans across multiple organizational systems should be able to effectively serve multiple teams, products, project cycles, environments or technologies.

Some coaches can operate equally comfortably at any organizational level: top of organization (with senior leadership), team-level/mid-level management and with individuals. But some coaches have a stronger preference for their focus: e.g. technical excellence, team-level, enterprise-level.

There are different coaching styles that can be used:

- Directive Coaching – mostly used, when:
 - Coachee exhibits low ability and inadequate subject matter expertise for contextual learning, while a coach possess strong expertise in a subject matter
 - Coachee has low level of motivation and morale
- Non-Directive Coaching – mostly used, when:
 - Coachee exhibits high aptitude, strong skill set and subject matter expertise
 - Coachee has high motivation, aptitude and morale

Some of the most important **hallmarks of Agile coaching profession** are conceptualized here: [“Agile Coaching: Lessons from the Trenches”](#).

Before engaging with a coach (Enterprise, Team or Individual level, technical), every company/client is encouraged to learn what to expect from a healthy from a coaching engagement. Recommended topics are:

- Differences and similarities between Training and Coaching
- Coaching styles: as above, plus in-depth explanations
- Coaching Specialties vs. Coaching Competencies
- Coaching maturity and ability to make influence (e.g. Enterprise vs. Team)
- Rules of coaching engagement and disengagement
- Internal (full time) vs. External (consulting) coaching
- Solo coaching vs. taking part of a coaching team
- Classic “bad smells”, commonly seen with bad coaching

***Note:** As Certified Enterprise Coach (CEC), while coaching organizations, teams and people, I am empowered to exercise my own discretion to [help individuals in achieving Scrum certifications](#). I extend this offer only to those selected individuals that genuinely support and practice values of Agile and Scrum.*

Coaches are frequently asked by customers, especially, at early stages of an engagement: "What are your objectives, as an organizational agile coach? What are you planning to accomplish?".

While personal coaching objectives may vary from one coach to another, and depend on specific needs of a client-organization, there are some basic common purposes that are shared by all coaches, in terms of what they strive to achieve, while engaging with clients. Here is the list of objectives that are shared by many agile organizational coaches: [Top 10 Objectives of Agile Coach](#).

Lunch & Learn:

Source: <http://www.keystepstosuccess.com/agile-coaching/>

Lunch and Learn (L&L) is a great way to collaborate in an informal and relaxing atmosphere, engaging people from different organizational domains and layers. One of the big advantages of L&L is that people feel more comfortable to interact with on another on par, regardless of their organizational positioning and relationships (e.g. reporting). L&L can be also tailored for specific teams, groups, departments or senior management.

L&L, typically conducted by a facilitator (usually, an experienced multi-skilled and impartial to internal dynamics coach-trainer), and can be delivered in the form of a general overview, targeted/tailored mini-training, preliminary organizational assessment or a comprehensive Q&A.

Here are some recommended L&L topics for organizations that have gone down the path of agile/adaptive transformation:

- Organizational design and system dynamics
- Values/norms, principles, practices
- Frameworks and processes
- Role, responsibilities, relationships
- Events, artifacts, Internal agreements
- Tooling, communication, interactions
- Location strategies, multi-site distribution
- Career paths, compensation, professional development
- Coding techniques to lower everyday costs and risks and enable change

Following lunch & learn session, an organization (department, group or team) may decide to either follow through with changes/improvements, by relying on its own internal capabilities or by further engaging with an expert-leader in any of the following capacities: further (in-depth) organizational [assessment](#) and/or tailored [training](#) and/or continuous [coaching](#) (with subsequent periodic reassessments).

Even in an initial Lunch & Learn session does not turn into a longer-term engagement, it should still be viewed as a great learning experience for both parties: an organization and coach-trainer.

This Page is Left Intentionally Blank