BITS Pilani presentation

BITS Pilani
Pilani Campus

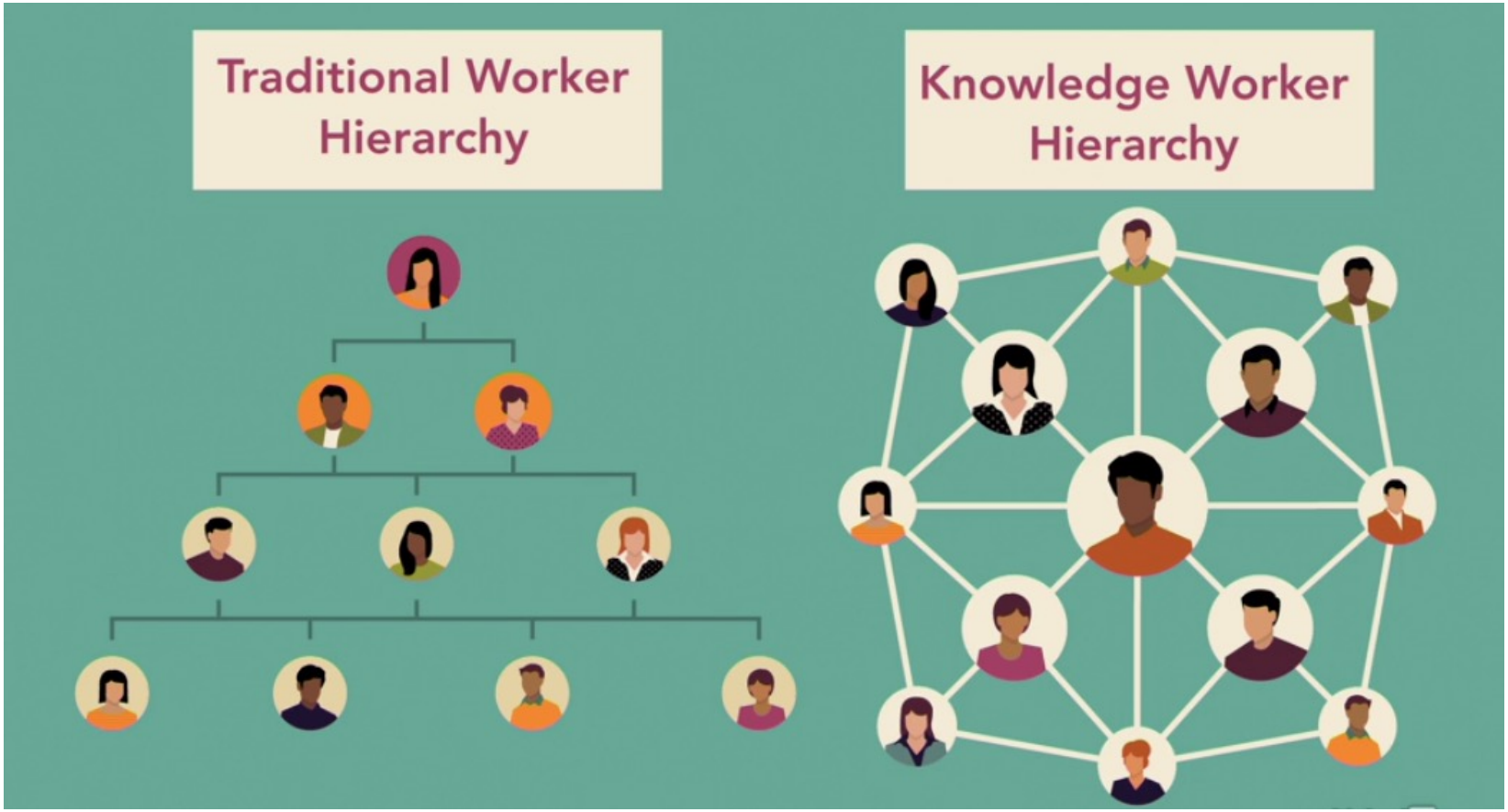
K.Anantharaman
kanantharaman@wilp.bits-pilani.ac.in

SE ZG544 – Agile Software Process
CS3 – Agile Manifesto & Agile Principles

CS3-Topics

- The rise of knowledge workers
- Agile Manifesto
- Agile Principles
- Team Motivation, Team Dynamics, Soft Skills
- Self Organizing teams, Emergent Design
- Simplicity

The rise of knowledge workers



Directive leadership style

Supportive leadership (Servant leadership style)

Source: lynda.com agile-foundations by Doug Rose

innovate achieve lead

BITS Pilani
Pilani Campus

Agile Manifesto

The Key Contributors

In Feb 2001, 17 new **methodology pioneers** met in Snowbird, Utah, USA.

To share their experiences, ideas, and practices and to suggest ways to improve the world of software development.

After Many discussions, they came up with Agile Manifesto

Image Source: https://udayanbanerjee.wordpress.com/category/agile

Agile Manifesto

Manifesto for Agile Software Development

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

Source: © 2001 www.agilemanifesto.org

- **Over the years, the Agile Manifesto has become a battle cry for organizational transformation.**

Three Perspectives

(HOT Perspectives)

- The **H**uman perspective:
  - Cognitive and social aspects, and refers to learning and interpersonal (teammates, customers, management) Process.

- The **O**rganizational perspective:
  - Managerial and cultural aspects and refers to the workspace and issues that extend beyond the team.

- The **T**echnological perspective:
  - Practical and technical aspects and refers to Technical and Coding Practices.

Ref: Hazzan Dubinsky2008_Book_AgileSoftwareEngineering

Agile Manifesto Principles (Year 2001)

innovate achieve lead

- https://agilemanifesto.org/

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 Process and tools (H)
- Working software over comprehensive documentation (T)
- Customer collaboration over contract negotiation (HO)
- Responding to change over following a plan (OT)

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

- Each principle supports and supported by other principles
- Redefined roles for Developer, Manager, Customer
- No "Big Upfront" Steps
- Iterative Development
- Negotiated and limited functionality
- Focus on Quality – Achieved through testing

Ref: Meyer2014_Book_Agile

Agile Principles

12 Agile Principles(Principles behind Agile Manifesto)

Satisfy Customer

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

Embrace Change

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

Frequent Delivery

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

Customer collaboration

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

Support & Trust

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

F2F Communication

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

Working software

07 Working software is the primary measure of progress.

Sustainable Phase

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

Technical Excellence

09 Continuous attention to technical excellence and good design enhances agility.

Keep it simple

10 Simplicity – the art of maximizing the amount of work not done – is essential.

Self Organization

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

Inspect & Adapt

12 At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

https://nicocasel.net/blog/why-i-feel-natural-with-agile-principles

Augmentation of Agile Manifesto

- Agile Manifesto was not actionable
- To support teams making Agile transitions, actionable items were needed.
- The original signatories of Agile Manifesto augmented the four values with 12 Agile principles behind the Agile Manifesto

- **Differentiate between Value, Principle, Practice**
- **4 Agile values ,12 Agile Principles and many Agile Practices**

Agile Principles (Not Official)

Organizational

1 Put the customer at the center.

2 Let the team self-organize.

3 Work at a sustainable pace.

4 Develop minimal software:

- 4.1 Produce minimal functionality.
- 4.2 Produce only the product requested.
- 4.3 Develop only code and tests.

5 Accept Change

Technical

6 Develop iteratively:

- 6.1 Produce frequent working iterations.
- 6.2 Freeze requirements during iterations.

7 Treat tests as a key resource:

- 7.1 Do not start any new development until all tests pass.
- 7.2 Test first.

8 Express requirements through user stories or scenarios.

Ref: Meyer2014_Book_Agile

Agile Practices

- Agile Manifesto → Agile Principles → Agile Practices
- Agile Practices → Project Outcome

- Agile Practices
  - Sprint Planning, Product Backlog, Sprint Review, Planning Game, Frequent Delivery, Retrospective
  - Definition of Done
  - Whole Team, Osmotic Communication, Daily Scrum
  - TDD, Pair Programming, Continuous Integration, 10-minutes Build
- Agile methods/methodologies
  - Scrum, XP, Kanban, Crystal

Quiz

Q1

Principle 1

Agile Principle#1

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

- Releasing software early
  - Shipping a working version of software as early as possible. by choosing the features and requirements that will deliver the **most value**. This is the best way to get customer feedback.
- Delivering value continuously
  - The team that truly collaborates with customer has option of making necessary changes along the way. That's what continuous delivery means. --- (vs CCB)
- Satisfying the customer.
  - Plan short iteration, Deliver highest value, Early feedback, Incorporate feedback in next iteration. Collaborate with customer.

Source: The Agile principles By Andrew Stellman and Jennifer Greene

Agile Principle#2

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

- Requirements changes due to emerging/new opportunities
- Respond to change instead of tight alignment to plans
- Change satisfies the customer's latest needs and provide the customer with a competitive advantage
- Agile teams embrace change by treating project changes as positive and healthy developments for the project
  - Nobody gets in "trouble" when there is change, We don't sit on the change until it is too late, We're all together.

Agile Principle#3

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

- By using time boxed iterations to *deliver working software frequently.*
- Shorter Iterations (~2 wks.) (Smaller releases means fewer chances of bugs , frequent feedback)
- Agile teams constantly adjust the project so that it delivers the most value to the customer
- *We no longer regard a release cycle of "a couple of months" as agile. The industry has evolved to daily or weekly releases.*

Source: https://www.plutora.com/blog/12-agile-principles

Agile Principle#4

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

- **Business people** is referred to a Product Owner, or anyone who is a proxy between customer and team.
- Emphasize here to have a shared responsibility and accountability, 'work together' stresses on total commitment on both sides.
- Catching misunderstandings early, **clarify requirements just-in-time** and to keep all team members 'on the same page' throughout the development helps in producing successful outcomes.

Some list of customer satisfaction issues and Agile Approaches

Customer dissatisfaction issues	Agile Approaches
Product requirements misunderstood by the development team	The customer is able to provide feedback just-in-time and also at the end of the sprint, not before it's too late at the end of the project.
The product wasn't delivered when the customer needed it.	Working in sprints allows agile project teams to deliver high-priority functionality early and often.
The customer can't request changes without additional cost and time.	Agile teams can accommodate change in requirements, and shifting priorities with each sprint, by removing the lowest-priority requirements –offsetting cost.

Source:T1

Agile Principle#5

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

Motivated and Talented individuals

Source: Agile Foundations - Principles, practices and frameworks by Peter Measey
McGraw-Hill Education ACP Agile Certified Practitioner Exam by Klaus Nielsen

Motivated individuals

5th Agile Principle states "Build projects around motivated individuals ….."

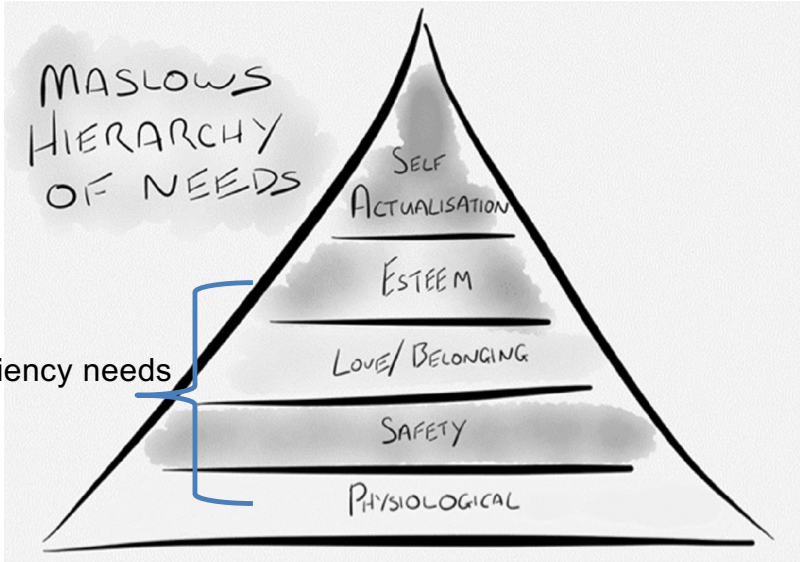
- **Motivation** releases energy and creativity and is an essential component of **high performance.**

- We will look at a few **different approaches to understanding what motivates individuals** and the vital role that talent plays in achieving high performance.
- **We will look at psychological models**

Maslow's hierarchy of needs (Motivation theory)

innovate achieve lead

- Maslow's hierarchy of needs is a theory in psychology proposed by Abraham Maslow in his 1943 paper still relevant today in 2020.



Deficiency needs

- Maslow suggested that, once our basic needs are met, our behavior will be driven by meeting higher-level needs.
- He assumed that human beings have the natural propensity to move towards self-actualization (Fulfillment of one own full potential/talents) by satisfying preceding needs.

Physiological : Human survival requirements, such as food, water and air, etc., comprise physiological needs. The absence of these will create various psychological symptoms, such as hunger, thirst, discomfort.

Safety: Physical safety, economic security, employment, health and wellbeing, and protection against accidents/illness.

Love/Belonging: The sense of belonging and acceptance, for instance in working groups, congregations, professional bodies and sports teams, can foster creativity and motivation.

Self Esteem/Respect :Sense of contribution, achievement, recognition, freedom and attention.

Self-actualization: 'the desire to accomplish everything that one is capable of becoming'. Once individuals have achieved self-actualization they can provide their support to others.

Management's attitude determines motivation

innovate achieve lead

McGregor's Theory X and Theory Y

Theory X managers believe that employees...	Theory Y managers believe that, given the right conditions, employees...
Hate work	Like and need work
Seek money and security	Seek to be involved and realise their potential
Have to be forced to work	Drive themselves and work effectively
Prefer to be told what to do	Take initiative
Are rarely creative	Are naturally highly creative
Are selfish	Commit themselves to larger goals

- An Agile leadership style should be in alignment with McGregor's Theory Y, which views employees in a positive light.
- As in Agile, that puts individuals and teams first, McGregor's research outcomes prove that teams under Theory Y management showed better performance in comparison to Theory X teams.

- Whether employees display Theory X or Theory Y behavior is a consequence of how management treat them.
- This means that generally managers will get what they expect – if they expect and manage for Theory X behavior, they will typically get employees displaying Theory X behavior.

Some factors only demotivate

- A researcher Herzberg proposed a refinement to Maslow's and an addition to McGregor's approach
- Hygiene factors comprise:
  - Pay, company policy, quality of supervision/management, working relations, working conditions, status and security.
- Motivators comprise:
  - Achievement, recognition, responsibility, advancement, learning, type and nature of work.

Daniel Pink,2010 - Motivation comes from autonomy, mastery and purpose

Autonomy – people's desire to direct their own lives and to gain control over some (or all) of the four main aspects of work: what, how, when and with whom.

Mastery – becoming better at something that matters to an individual. This can be achieved by taking on tasks that allow people to develop skills further. Mastery is fostered by an environment where learning is encouraged and mistakes are tolerated.

Purpose – fulfilling a natural desire in people to contribute to a cause greater than themselves.

Talent

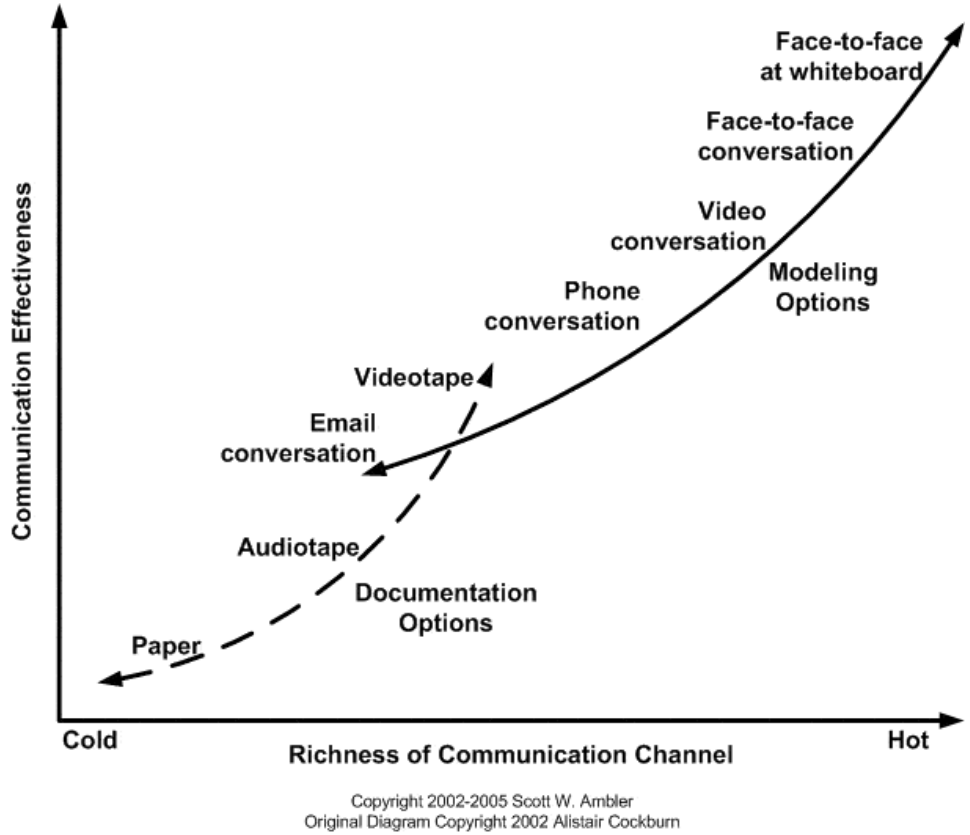
- To achieve high performance, motivation needs to be coupled with **talent**.
  - So where does talent come from? Are people born with it or can they acquire it? Matthew Syed argues for the latter, and states that (Syed, 2011):
- Talent comes from purposeful practice

- Practice needs to be purposeful.(Master at something)
  - Not all practice is useful. People only develop when they repeatedly try things that are just out of reach and get quality feedback on their performance.
  - The paradox of excellence is that it is built on necessary failure
  - The learning process is often best facilitated by an expert coach.

- Engaging in purposeful practice leads to high performance – and the opposite is also true.

Agile Principle#6

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



Remote teams:

- Skype and Hangouts allow us to have remote face-to-face talks.

- One-on-one communication has it challenges
- Number of communication channels: N × (N–1)/2.
- 7 member team = 7*(7-1)/2=21

Agile Principle#7

"Working software is the primary measure of progress."

- Working software is better than progress reports, because it's the most effective way for the team to communicate what they've accomplished.

- Finished analysis, complete models, or beautiful mock-ups have may be necessary, But have little meaning if they aren't converted into working software.

Agile Principle#8

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

- Agile work is intense.
- Regular weekends, night outs, overtime not sustainable
- The entire team is responsible for maintaining sustainable phase
  - Business owners or Product owners
  - The development team
  - Team Lead/Scrum master
  - Organization/Sponsors

Source: https://soulofscrum.com/blog/f/keeping-a-sustainable-pace

8. Sustainable Phase …

Product Owners (and other interested parties outside the Scrum Team):

- Don’t make commitments to the business that don’t come from the Development team.
- Don’t expect for the Development team to commit to anything longer term than the upcoming Sprint.
- All prioritization comes through the Product Owner…respect that.
- Keep in mind the cone of uncertainty when developing your Product Road Map.

https://soulofscrum.com/blog/f/keeping-a-sustainable-pace

8. Sustainable Phase …

Development Team:

- Beware of estimating what you can get done in a sprint
- By taking on too many backlog items into the Sprint Backlog you endanger having the time for creative solutions.
- Cross training skills between team members
- Worst of all the temptation of accepting lower quality for the Product Increment.
- Collaborate with Product owner and prioritize the work

Source: https://soulofscrum.com/blog/f/keeping-a-sustainable-pace

8. Sustainable Phase …

Scrum Masters:

- If you see the team overextending, ask them about this in the Retrospective.
- Make sure there are no external pressures on the team.
- Habitual or frequent overtime indicates an issue that you should investigate and remove as an impediment to team health.

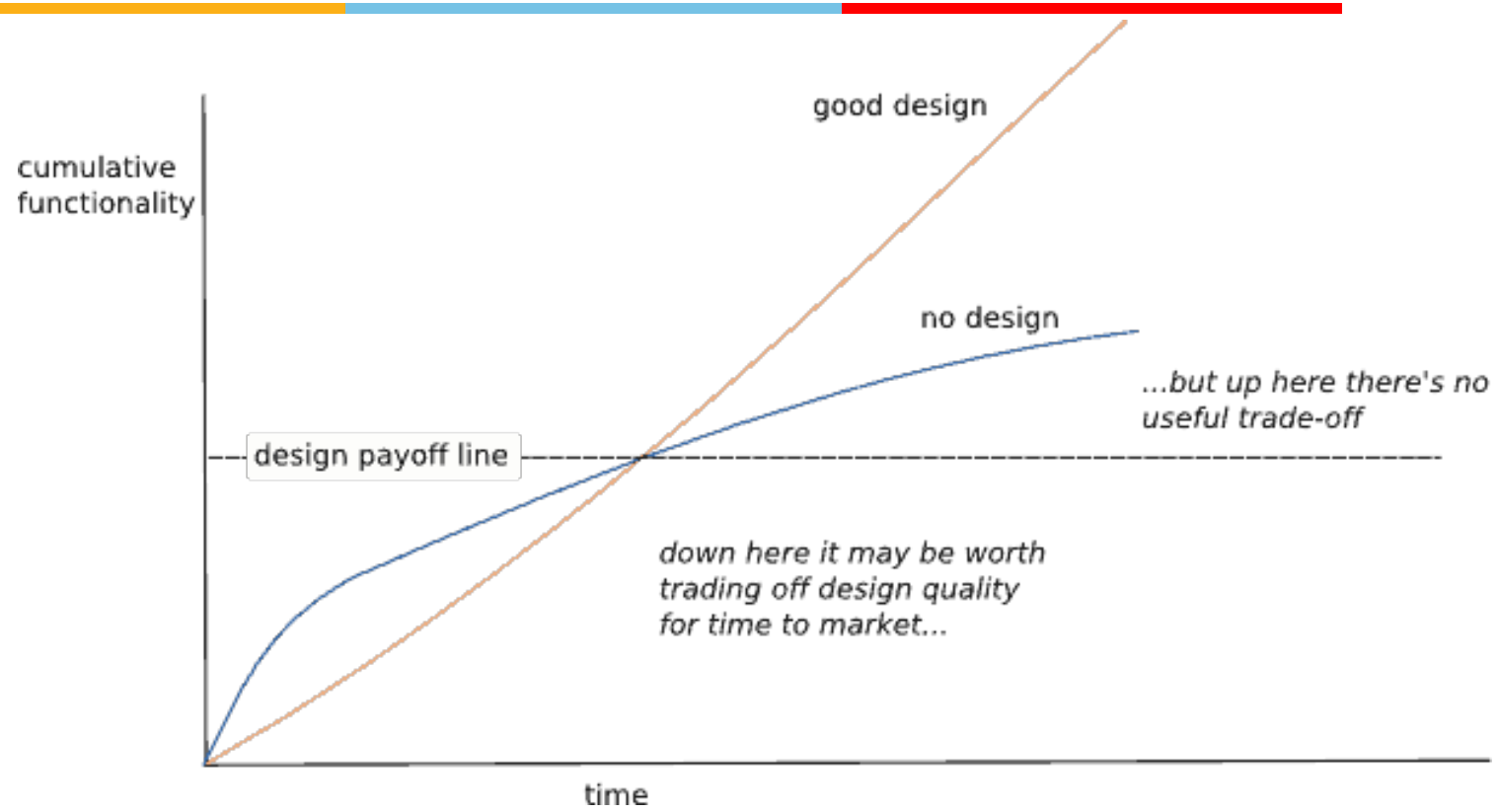
Source: https://soulofscrum.com/blog/f/keeping-a-sustainable-pace

Agile Principle#9

"Continuous attention to technical excellence and good design enhances agility."

- Object Oriented design, Design patterns, Decoupled service-oriented architectures, Containers, Cloud technology and other innovations an tools bring technical excellence to product.
- Well designed code is easy to maintain and extend.
- Constantly lookout for design and code problems and take time to fix those problems.
- Use TDD and pay back Technical Debt

Technical Excellence



Technical debt is a term first coined by Ward Cunningham. It describes the **accumulation of poor design** that crops up in code when decisions have been made to implement something quickly. Ward described it as Technical Debt because if you don't pay it back in time, it starts to accumulate. As it accumulates, subsequent changes to the software get harder and harder. What should be a simple change suddenly becomes a major refactor/rewrite to implement.

Martin Fowler created a pseudo-graph to visualize this:

Q2

innovate achieve lead

BITS Pilani
Pilani Campus

Simplicity

The art of maximizing the amount of work not done - 10th Agile Principle

Simplicity in Agile context

This means:

1. Focusing on ensuring that only the **simplest, leanest and fit-for-purpose product** is delivered, especially when considering lifecycle-driven documentation, and only producing what adds value.

2. Focusing on **maximizing the amount of work not done** when creating the product, i.e., focusing on **simplicity of delivery**

Fit-For-Purpose product

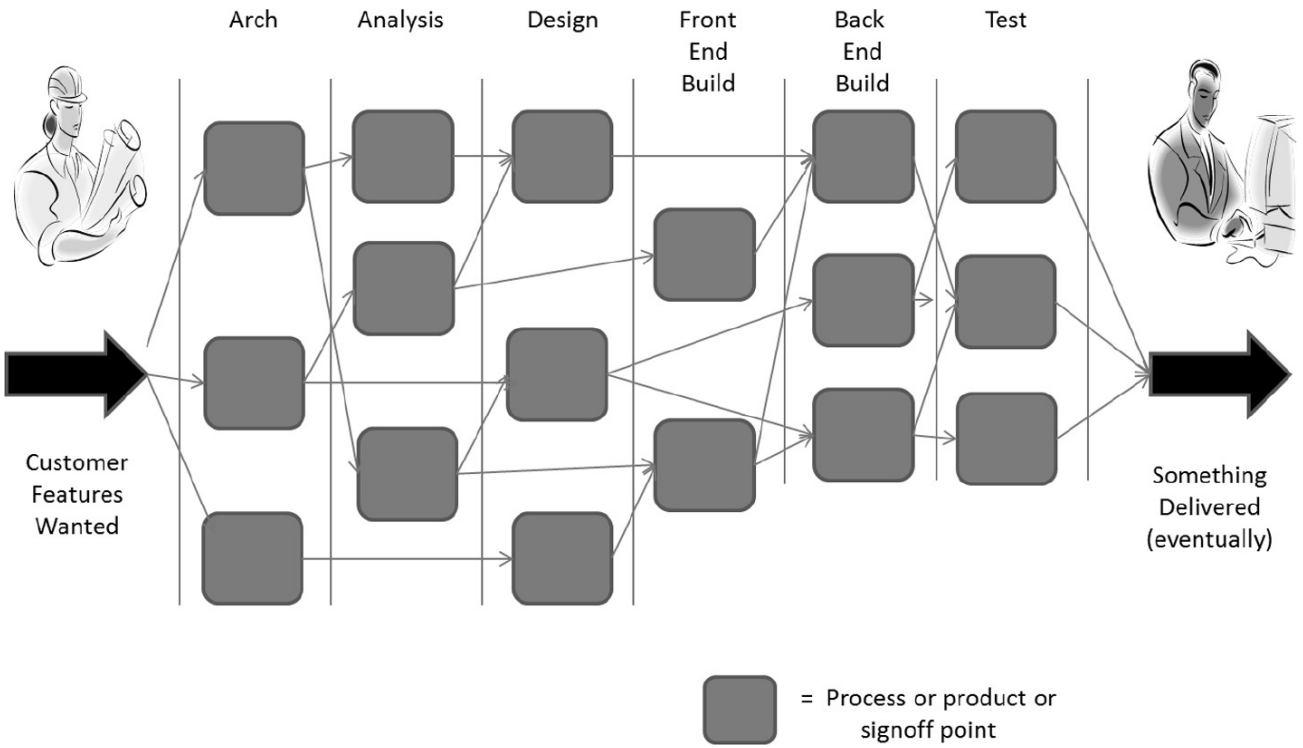
- This principle is therefore about reducing clutter and keeping the backlog focused on whatever needs to be delivered first
- Breakdown of features that are actually used in a typical delivered system (Standish Groups 2002).
  - **Features always used – 7% → Deliver these features first**
  - Features often used – 13%
  - Features sometimes used – 16%
  - Features rarely used – 19%
  - Features never used – 45%
- Agile frameworks have the concept of producing technically fit-for-purpose products.

Fit-For-Purpose Delivery

- This essentially means following Lean Software Development:
  - Eliminate Waste, Build in Quality, Create knowledge, Defer commitment, Deliver fast, Respect people, Optimize the whole
- For example,
- Eliminate waste:
  - Extra stories, stories constantly changing and the buffers created by crossing organization boundaries.
- Build in quality
  - If defects are routinely found in the verification process, the development process is defective.
- Defer commitment :
  - Abolish the idea that development should start with a complete specification
- Optimize the whole:
  - Viewed across the whole value chain - Brilliant products emerge

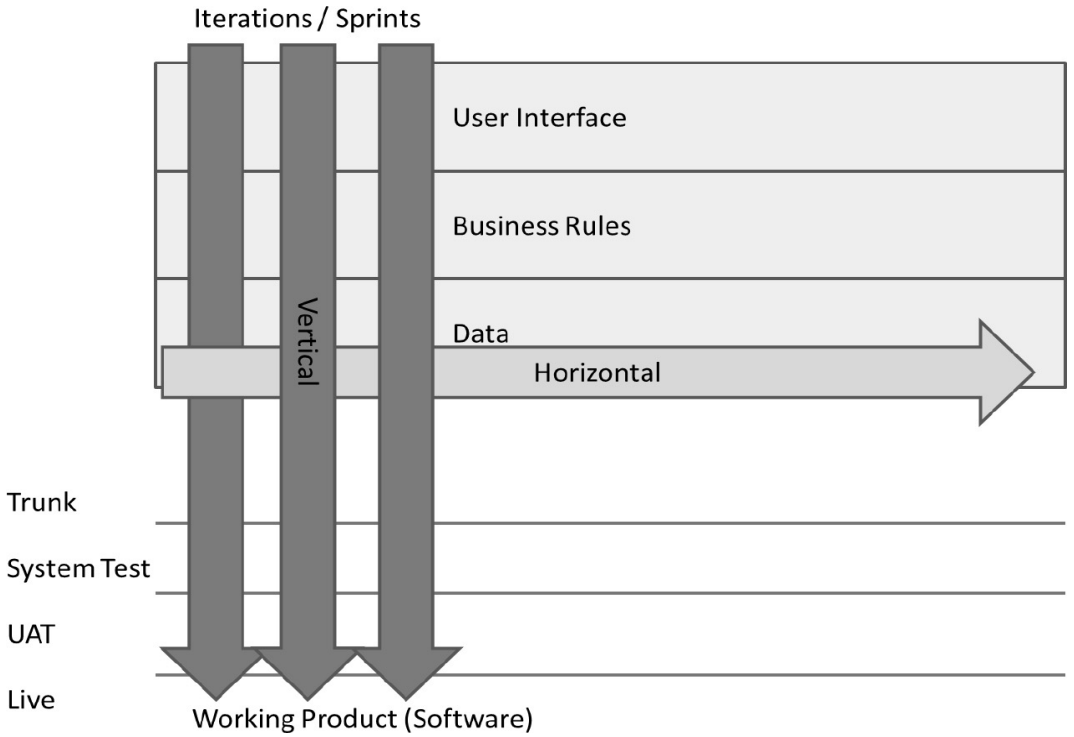
Silo Delivery

- In a team working with a silo mentality , these unseen boundaries can easily become barriers to effective communication and delivery



Vertical slices

- In an Agile delivery teams focus on producing the highest value stories in vertical slices down the architecture.
- Water fall thinking results in Horizontal slicing

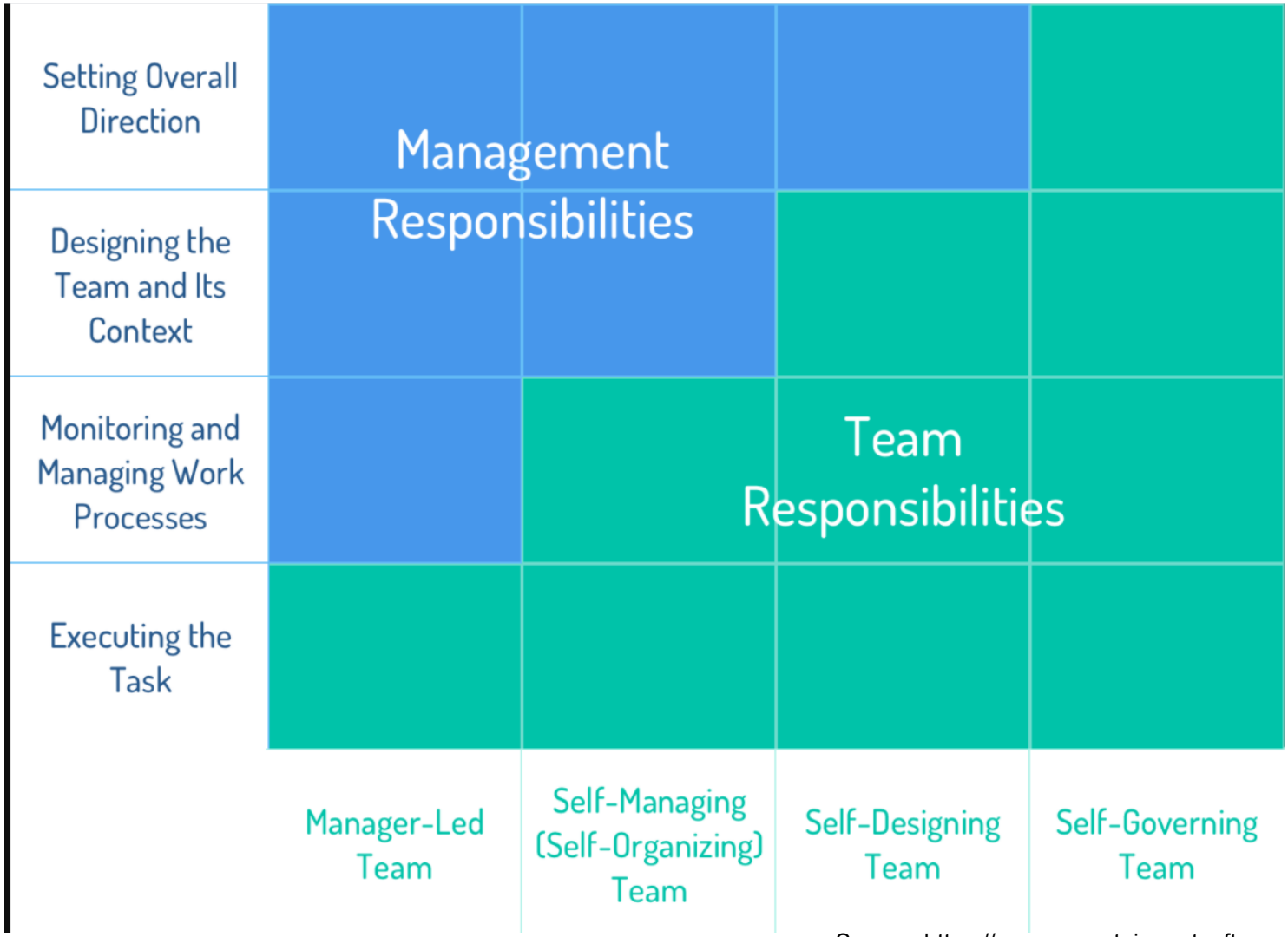


Agile Principle#11

"The Best Architectures, Requirements, and Designs Emerge from **Self-Organizing Teams**."

Source: https://www.mountaingoatsoftware.com

Type of teams



Source: https://www.mountaingoatsoftware.com

Self Organizing teams

- Not every agile team will choose to organize themselves the same way
  - Some teams will decide that all key technical decisions will be made by one person on the team.
  - Other teams will decide to split the responsibility for technical decisions along technical boundaries
  - Still other teams may decide that whoever is working on the feature makes the decision but has the responsibility of sharing the results of the decision with the team.
- Making use of the **collective wisdom of the team** will generally lead to a better way of organizing around the work than will relying solely on the wisdom of one personnel manager.

‘Emergent design’ – why is it important?

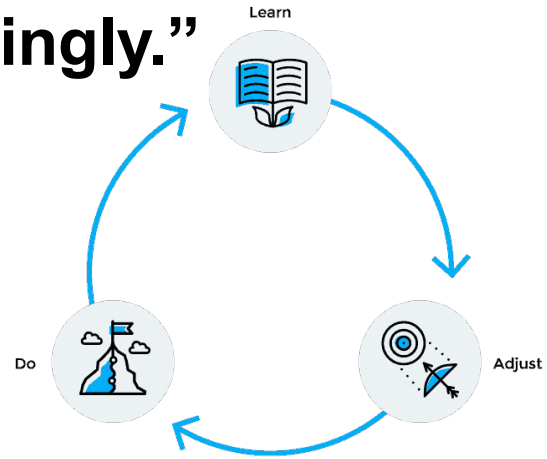
- BDUF: Big Design Upfront
- EDUF: Enough Design Up-front (**Agile team implement this Approach**)
- Upfront design - restrict opportunities to change and improve the design as the product is being developed.
- Wait until the **last responsible moment** to make design decisions
  - You can make decisions based on evidence that is identified as the system is being built. **This typically means that decisions are of a higher quality, because they are not just theoretical decisions based on little evidence**.
  - Example: Ordering a Hardware before lead time. **The last responsible moment is governed by that lead time**, and we have to find out enough about the hardware to make the order by that time, do some experimentations, gather required knowledge etc.

Self-organizing teams and emergent design

- Self-organizing teams are empowered, within agreed boundaries, to deliver **fit-for-purpose products, in a fit-for-purpose way, within the most effective time scale.**
- Self-organizing team in relation to emergent design is that there will be some **overarching design principles that teams must or should align to the timescale.**
- If teams are **forced to align to an externally defined detailed design** they are unlikely to 'go the extra mile' to try and identify or implement any opportunities to make the design better (**opportunistic design**).
  - It is likely that the only people who can effectively make the right detailed design decisions are team members. Nobody else will understand the evolving design as well as the team does.
  - This ties in with the concept of **'real options**' (Matts, 2007), which means keeping your options open for as long as you possibly can and making a decision when you are in the best position to make it with confidence.

Agile Principle#12

"At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly."



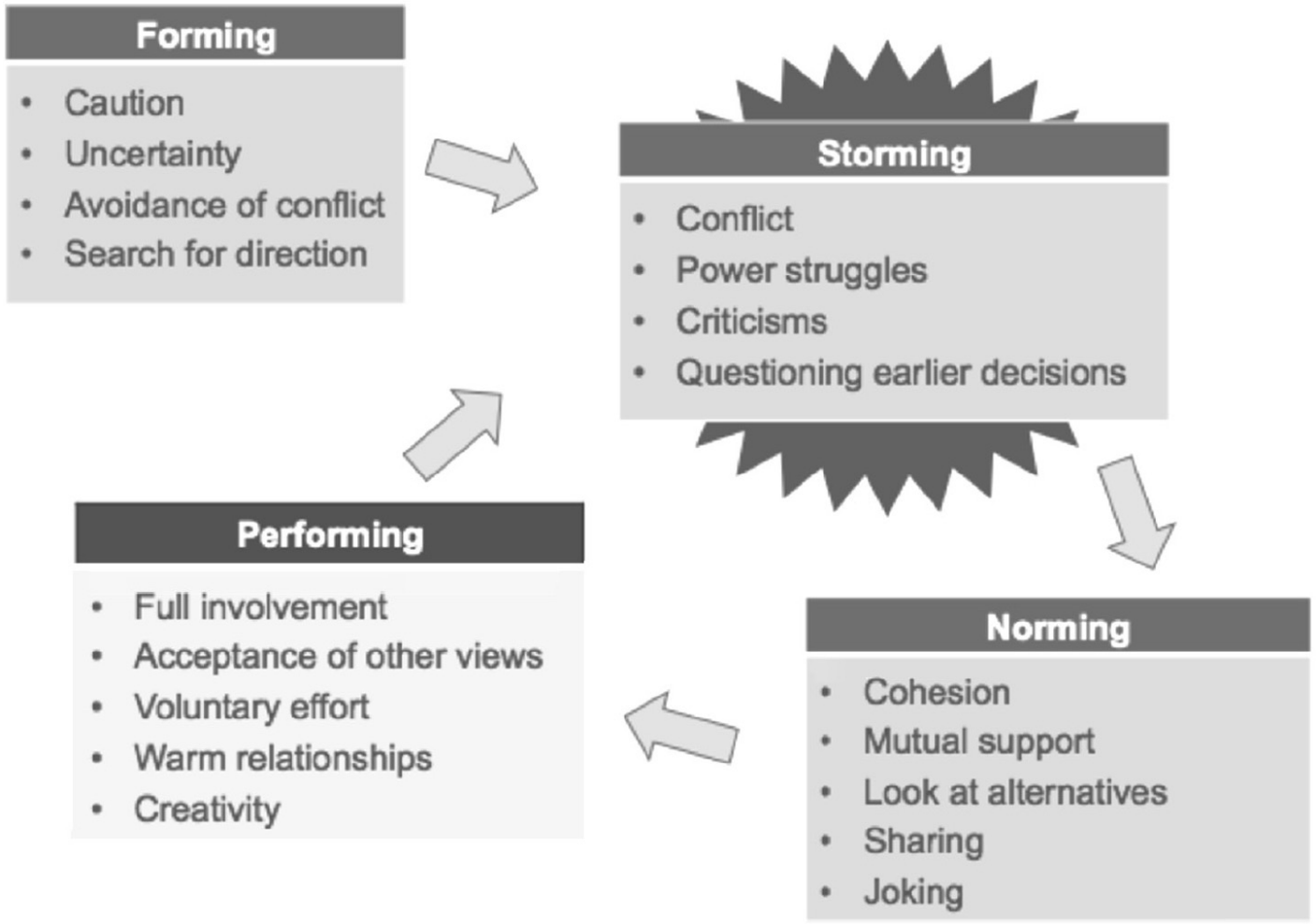
- At regular intervals, self-organizing teams should take the time to look at the way they work and adjust accordingly. No team runs perfectly.
- A mature agile team can identify issues with respect for each other and then take action to improve the process.

https://www.plutora.com/blog/12-agile-principles

Emergent Design From Self-Organizing Teams

Team Dynamics and Interpersonal skills

Tuckman's theory of team evolution



Lencioni – the five dysfunctions of teams

Dysfunction team

Functional high performance team

Interpersonal Skills – Key terms

- **Adaptive leadership** A type of leadership that deals with changes and problem solving
- C**ollaboration** The action of working with someone to produce something
- **Conflict resolution** Method(s) to solve conflicts
- **Emotional intelligence** Focused on people and forging strong and supportive relationships
- **Negotiation** Process of reaching the best results
- **Servant leadership** A philosophy and set of practices that enriches the lives of individuals, builds better organizations and, ultimately, creates a more just and caring world

Emotional Intelligence

- Emotional intelligence, in an Agile team, provides the team with the tools to **make things work and the ability to perform well**.

Mixed Model by Daniel Goleman

	Self	*Other*
	Personal Competence	***Social Competence***
Recognition	***Self-Awareness***	***Social Awareness***
	Emotional self-awareness	Empathy
	Accurate self-awareness	Service-oriented
	Self-confidence	Organizational awareness
Regulation	***Self-Management***	***Relationship Management***
	Self-control	Developing others
	Trustworthiness	Influence
	Conscientiousness	Communication
	Adaptability	Conflict management
	Achievement-driven	Leadership
	Initiative	Change catalyst
		Building bonds
		Teamwork and collaboration

- The higher the emotional intelligence of the Agile team, the greater are the chances for being successful in a people-oriented environment.

The Emotional Intelligence Skills Assessment (EISA) framework

innovate achieve lead

- Provides a strong fundamental assessment of emotional intelligence in project managers and Agile team members

Factor	Comments
Perceiving	The ability to recognize, acknowledge, and attend to the emotions of one's own self and other team members
Managing	The ability to express emotions in a controlled manner
Decision making	The ability to apply emotions effectively in decision making
Achieving	The ability to generate the emotions that will motivate oneself toward the pursuit of a desired goal
Influencing	The ability to motivate others in the pursuit of a goal, by evoking similar emotions in others as well

Collaboration

innovate achieve lead

- Why do we collaborate and what factors contribute to effective collaborations?
  - **We collaborate in order to deliver software (deliverables)**
  - **To foster progress by making decisions (decisions)**
  - **To learn (knowledge).**
- Cockburn and Highsmith (2002) identified collaboration as a combination :
  - **Interpersonal (trust, participation, commitment, respect)**
  - **Cultural (values and principles)**
  - **Structural (organization, technology, and practices) values**.
- Agile practices encourage collaboration and coordination through:
  - Daily stand-up meetings, Daily interaction with the product team, Stakeholder coordination

Adaptive Leadership

- Adaptive leadership that deals with changes and problem-solving.
- Adaptive leadership focuses on team management from building self-organized teams for developing servant leadership style.
- Adaptive work, such as that embodied by Agile, requires adaptive leadership. Different situations call for different responses, there is a technical or routine response, and there is also an adaptive response

Situation	*Technical or Routine*	*Adaptive*
Direction	Define problems and provide solutions	Identify the adaptive challenge and frame key questions and issues.
Protection	Shield the organization from external threats	Let the organization feel external pressures within a range it can stand.
Orientation	Clarity roles and responsibilities	Challenge current roles and resist pressure to define new roles quickly.
Managing Conflict	Restore order	Expose conflict or let it emerge.
Shaping Norms	Maintain norms	Challenge unproductive norms.

Negotiation

Fisher, Ury, and Patton (1991) call their approach "principled negotiation." Their book, *Getting to Yes*, contains four key elements:

- Separate people from the problem
- Focus on interests, not positions
- Invent options for mutual benefit
- Use objective criteria

Conflict resolution

innovate achieve lead

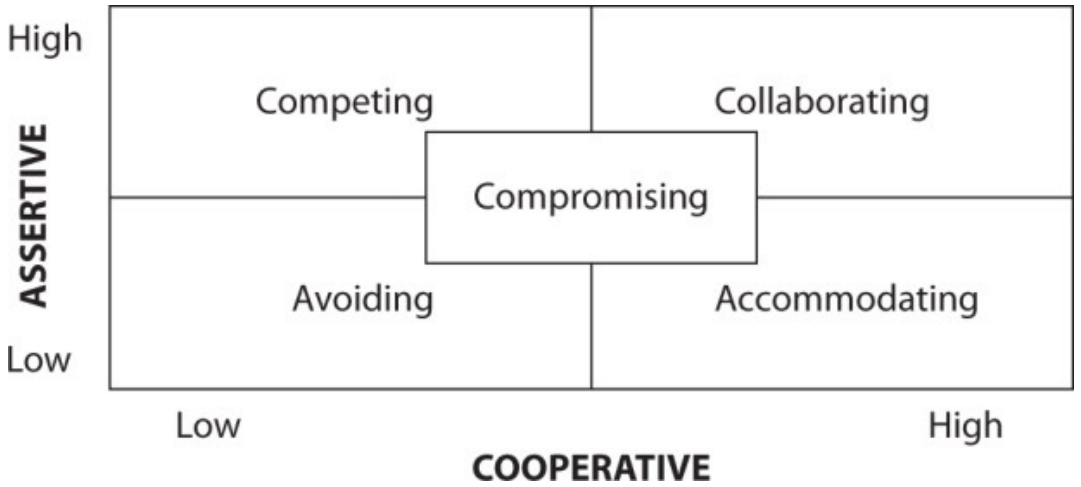
” The steps in conflict management may involve the following activities:

- Conflict identification
- Conflict analysis (who, what, why, when)
- Conflict resolution

Conflict Resolution Techniques

The Thomas-Kilmann conflict mode and the five general techniques for solving conflicts are fairly similar. His five general conflict resolution techniques are:

- Withdraw/avoid
- Smooth/accommodate
- Compromise/reconcile
- Force/direct (compete)
- Collaborate/problem-solve



Conflict resolution

The book, Coaching Agile Teams (2010) by Lyssa Adkins, is popular in Agile circles and contains the following

Level 5	"World War" is when talks stop and the parties simply seek to destroy each other. We should do whatever is necessary to solve this conflict before someone gets badly hurt. Desperate measures may be needed and often outside help is employed.
Level 4	"Crusade" is really bad. Protecting one's own group becomes the focus and language is ideological. It is us or them. Winning is the only option. A safe environment must be created to allow time and, hopefully, some shuttle diplomacy to figure out a solution to the conflict.
Level 3	Things may turn worse with a "Contest" conflict where the winning trump resolves the issue. Language may include personal attacks. In this situation we need to accommodate and accept the other party`s demands and, hopefully, end the conflict with a negotiation.
Level 2	Conflicts may worsen or start at "Disagreement" where personal protection trumps collaboration and language is guarded and open to interpretation. The situation may be handled by giving the participants needed support and ensuring the empowerment of the participants to find a good solution without fearing for their safety. We need a cool, calm environment.
Level 1	The lowest level of conflict is "Problem to Solve," which we all know. Information sharing and collaboration takes place as language is open and fact based. The way to deal with these kinds of regular conflicts is with collaboration by seeking consensus and the win–win situation.

Conflict level	Successful response options
Level 1: Problem to Solve	Collaboration. Seeking a win-win situation. Consensus. Learning where every team member's head is with regard to the issue and, in time, arriving at a decision everyone can back.
Level 2: Disagreement	Support. Empowering the other to resolve the problem. Safety. Anything that restores a sense of safety, such as collaboration games or regrounding in the team's shared values.
Level 3: Contest	Accommodate. Yielding to the other's view when the relationship is more important than the issue. This is a successful short-term strategy only and becomes a liability if used often over the long term. Negotiate. When the "thing" the conflict is about is divisible, such as the use of a shared resource, negotiation can work. Negotiation will not work when the issue revolves around people's values. Values are not divisible, and one person giving in to another in violation of their own values feels like a sellout. Get factual. Gather data about the situation to establish the facts.
Level 4: Crusade	Establish safe structures again. Use "shuttle" diplomacy, carrying thoughts from one group to the other until they are able to de-escalate and use the tools available at lower levels of conflict.
Level 5: World War	Do whatever is necessary to prevent people from hurting one another.

Conflict resolution skills

The win–win approach	How can we solve this as partners rather than opponents?
Creative response	Transform problems into creative opportunities.
Empathy	Develop communication tools to build rapport. Use listening to clarify understanding.
Appropriate assertiveness	Apply strategies to attack the problem, not the person.
Cooperative power	Eliminate "power over" to build "power with" others.
Managing emotions	Express fear, anger, hurt, and frustration wisely to effect change.
Willingness to resolve	Name personal issues that cloud the picture.
Mapping the conflict	Define the issues needed to chart common needs and concerns.
Development of options	Design creative solutions together.
Introduction to negotiation	Plan and apply effective strategies to reach agreement.
Introduction to mediation	Help conflicting parties to move toward solutions.
Broadening perspectives	Use the three articles on running meetings in conflict resolving mode.

Conflict resolution an important factor of emotional intelligence, which more and more teams foster to improve the teamwork.
Communication, active listening, negotiation skills, and soft skills all play an important part in Agile methodologies, as teams are self-governed and empowered.

Servant Leadership

The core characteristics of being a servant leader are:

Listening

- Empathy
- Healing
- Awareness
- Persuasion
- Conceptualization
- Foresight
- Stewardship
- Commitment to the growth of people
- Building community

The emphasis here is on four factors

- Get the right people into the team.
- Trust team members rather than requiring them to prove themselves trustworthy

.• Let the team select the project approach for project success

- Stand back and let the team do their work.

Anti Patterns: Agile Manifesto

- The tool makes us Agile, Relentless automation
- Hierarchies
- Over-standardization
- Proxy customers (Business Analysts, Architect acting as customer)
- Considering plans and roadmaps as commitments
- Expecting too much detail
- Not engaging stakeholders

Q3

Anti Patterns: Agile Principles

- Out of sight, out of mind - Stakeholders
- Requiring additional documentation or reporting, "We will need this later", Documentation as collaboration, Write only documentation
- One size fits all approach towards team management
- Chasing the metrics
- Ignoring the environment
- Multiple deployment environments
- Detailed story descriptions, Fixed standards or Process, Aiming for Small stories on the backlog
- Restricting who can talk to the customer
- Not considering cultural differences
- Lacking collaboration skills
- Over-complicating things/Future proof everything
- Insisting on Sign-off Process
- "Just in case" development
- Management focus on individuals
- Iterations planned in advance
- Focus on the tasks not the value