

© PRINCE2 AGILE™

How to use PRINCE2® with agile ways of working





- 1. Understand the basic concepts of common agile ways of working
- 2. Understand the purpose and context for combining PRINCE2 and the agile way of working
- 3.Be able to apply and evaluate the focus areas to a project in an agile context
- 4.Be able to fix and flex the six aspects of a project in an agile context
- 5.Be able to apply or tailor the PRINCE2 principles, themes, processes and management products to a project in an agile context
- 6.To learn through the use of theory and practical exercises
- 7.To prepare delegates for the PRINCE2 Agile Practitioner exam

About yourself



- 1. Name (and company)
- 2. Role
- 3. Experience of PRINCE2
- 4. Experience of agile
- 5. Your objective for this course

About the manual



- Aligned to the PRINCE2 manual
- Early chapters
 - Basic understandings and drivers for PRINCE2 Agile.
- Middle chapters
 - Discussion and description of the Principles, Themes, Processes and Products
 - What you may find
 - What to do.
- Final chapters
 - Focus areas where PRINCE2 needs more detailed guidance when in an agile context
 - The appendices.

Exam structure



- 2.5 hour exam
- Open book
- Objective Testing Exam
- Taken on the afternoon of the third day
- 5 questions totalling 50 marks
- Pass mark is TBC.



Agenda for Day 1

- Projects and BAU
- An overview of agile
- Blending PRINCE2 and agile together
- Assumptions
- The Hexagon (incl. MoSCoW prioritisation)
- Starting Up a Project, Initiating a Project (including the Business Case, value assessment and Cynefin approach)
- Requirements and User Stories
- Organization.





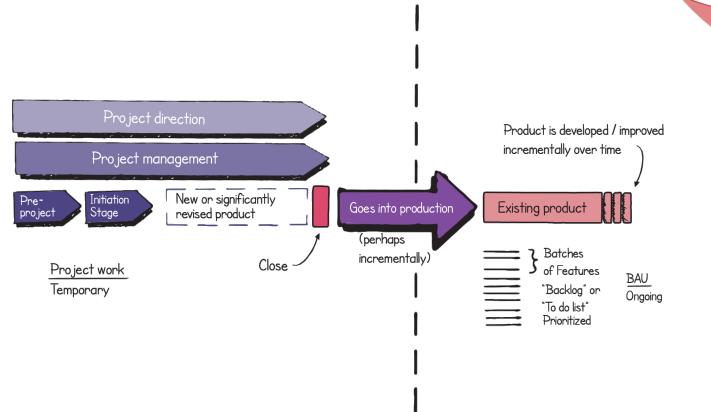
- PRINCE2 and PRINCE2 Agile are only suitable for projects
- Agile can be used on projects and for ongoing 'Business as Usual' (BAU) work
- Important to understand the difference between projects and BAU to use agile appropriately

Project characteristics	Business as Usual characteristics
Temporary	Ongoing
Team is created	Stable team
Difficult	Routine
A degree of uncertainty	A degree of certainty

Guidance reference: Section 1.2

The difference between project work and BAU work





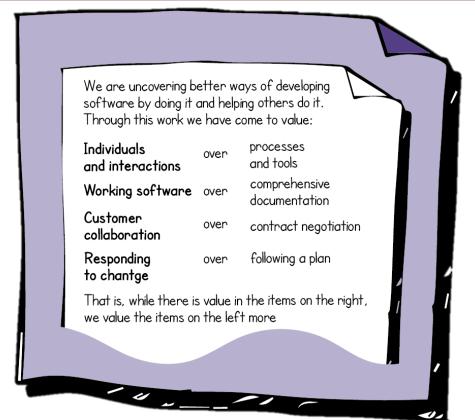




- The term 'agile' is viewed in many different ways
- Well-known frameworks referred to as 'agile ways of working'
- Well-known behaviours, concepts and techniques characterising agile
- The Agile Manifesto comes closest to a single definition it was created as an alternative to 'waterfall' processes
- Agile addressed the new demands placed on the delivery of software.

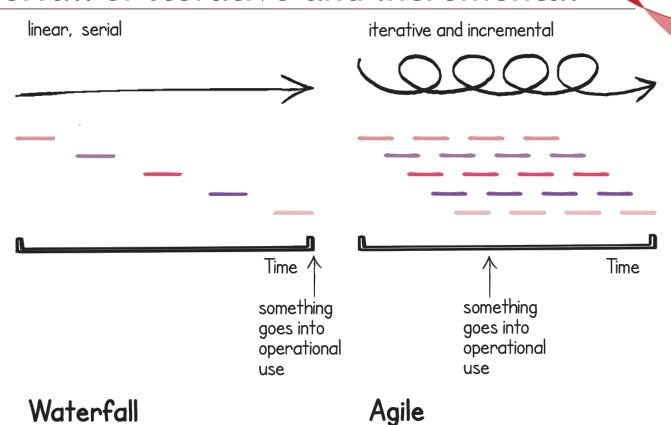








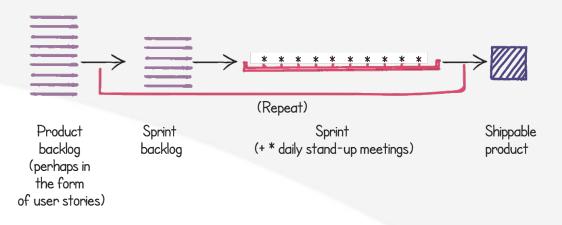
Waterfall or Iterative and Incremental



Guidance reference: Figure 2.2

Agile basics





A basic Backlog and Sprint structure is commonly used

- It can be viewed in many ways
 - Timeboxed approach for developing software
 - A collection of techniques
 - Using the Scrum framework.

Beyond a basic view



A more comprehensive view would include:

- Vision, Roadmap and Releases
- Non-IT situations
- Project work
- Flow-based working
- A wider mind-set.

Agile Frameworks



- Many frameworks are recognised as being agile
- Some are more common than others
- Some are only applicable to IT.

Scrum Kanban
Lean Lean Startup
XP SAFe DAD
DSDM/AgilePM
DevOps
FDD Crystal ASD



Agile behaviours, concepts and techniques

Along with the agile frameworks there are a variety of behaviours, concepts and techniques that are seen as being part of the agile way of working

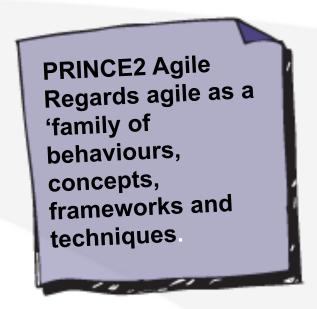
A few illustrative examples

Term	Examples	Similar terms for this
Behaviours	Being collaborative, Self-organizing, Customer focussed, Empowered, Trusting not blaming	Principles. Values. Mind-set.
Concepts	Prioritising what is delivered, Working iteratively and incrementally, Not delivering everything, Time focussed, Inspect and adapt. Kaizen. Limiting WIP	Fundamentals
Techniques	Burn charts, User Stories, Retrospectives, Timeboxing, Measuring flow	Practices. Tools.

Guidance reference: Section 2.2.2, Table 2.2





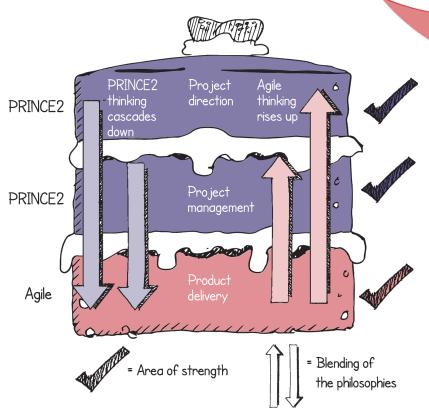


Guidance reference: Section 2.2.1

PRINCE2 Agile blending PRINCE2 and agile together

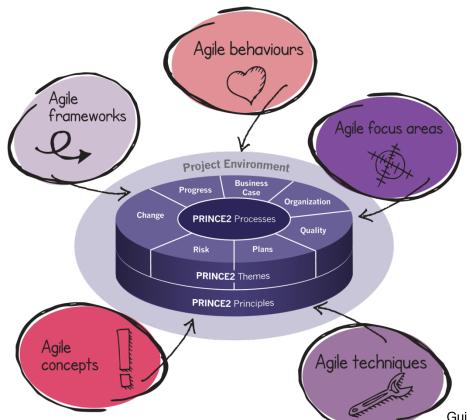


- They each have their own strengths
- Who is it for?
- When and where should it be used?





What does PRINCE2 Agile comprise of?



Guidance reference: Section 3.5, Figure 3.2





	Key point
1	PRINCE2 (2009 version) is already enabled for use with Agile
2	PRINCE2 is suitable for any style of project and is not a 'traditional' project management approach as is typically contrasted to Agile
3	PRINCE2 Agile is for any project and not just for IT projects
4	'IT only' frameworks and techniques are mentioned in PRINCE2 Agile but not extensively
5	There is much more to Agile than the Scrum framework. Agile is not Scrum.
6	The most 'commonly used' Agile approaches are Scrum and Kanban, but they are not suitable for managing a project in isolation. However, they can be effectively used in a project context.
7	The term Agile (in this manual) refers to a family of behaviours, concepts, frameworks, and techniques
8	Using Agile on a project is not a question of 'yes or no'. It is about 'how much'.

Beware of prejudice!



Control and governance allows agile to be used in complex environments.





- How PRINCE2 may look in an agile context
- Please note the word 'typically'
 - ...and 'a way' not 'the way'
- Tailoring PRINCE2 depends on the project context and may affect:
 - the level of formality
 - where the emphasis is placed
 - how it is carried out.

Pre-project

Initiation stage

Subsequent delivery stage(s)

Final delivery stage





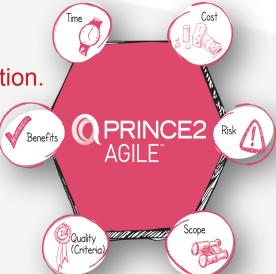


The Hexagon



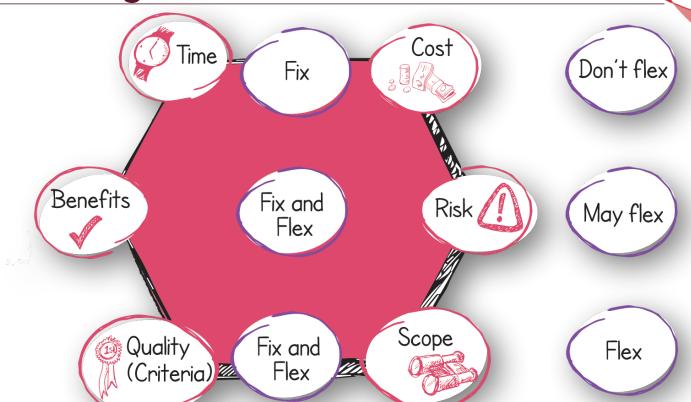
 Fundamental to PRINCE2 Agile since it involves the 6 aspects of project performance

A significant change to PRINCE2 with the 2009 edition.



The Hexagon





What to fix and what to flex



This is about tolerances and not the

aspects themselves.







Aspect	Tolerance Guidance	Summary
Time	Zero tolerance for extra time on all levels of plan	Fix
Cost	Zero tolerance for extra cost on all levels of plan	Fix
Quality	Not all acceptance criteria and quality criteria are of equal importance, so they can be prioritised Project Project Product Description Zero tolerance for the Customer's quality expectations and Acceptance criteria that are essential Tolerance may be used for the Customer's quality expectations and Acceptance criteria that are desirable but not essential Product Descriptions (in general) Zero tolerance for the Quality criteria that are essential Tolerance may be used for the Quality criteria that are desirable but not essential	Fix and flex
Scope	Not everything the project aims to create is of equal importance, so they can be prioritised Zero tolerance for Products that are essential Tolerance may be used for Products that are desirable but not essential	Fix and flex
Risk	Tolerance to be defined to the needs of the Project Board and Project Manager as this depends on the specific situation	Fix or flex
Benefit	Zero tolerance for the level that is defined as 'minimum viability' in the Business Case Tolerance may be used above the level that is defined as 'minimum viability' in the Business Case	Fix or flex

The 5 targets



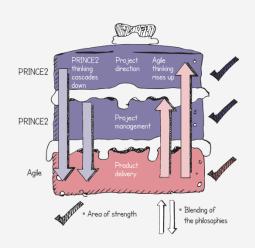
- It is essential to understand why?
- The 5 targets represent the rationale behind the hexagon
 - Be on time and hit deadlines
 - Protect the level of quality
 - Embrace change
 - Keep teams stable
 - Accept that the customer doesn't need everything.





Why?

- Early realisation of benefits
- Helps with planning
- Gives confidence
- There may be no choice
- Reduce the likelihood of cost overruns
- Improves reputation.



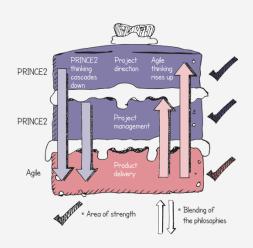




Why?

Damaging effects result from:

- Reduced testing
- Incomplete documentation
- Sub-optimal design
- Lack of appropriate training
- Non-compliance to standards.

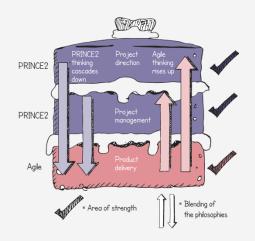


Embrace change



Why?

- It is inevitable
- A more accurate final product is more likely
- Can be handled by flexing what is delivered.



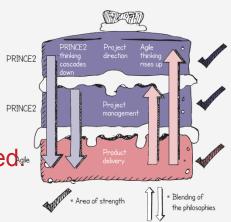
Keep teams stable



Why?

Changing team members can have a detrimental affect such as:

- Time spent bringing new team members up to speed
- Number of communication lines in the team grows exponentially
- An opportunity cost incurred to the areas providing the new people
- The team dynamics change and need to be re-established

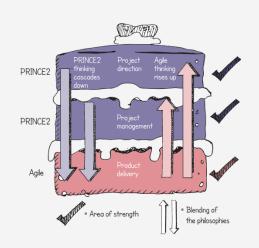




Accept that the customer doesn't need everything

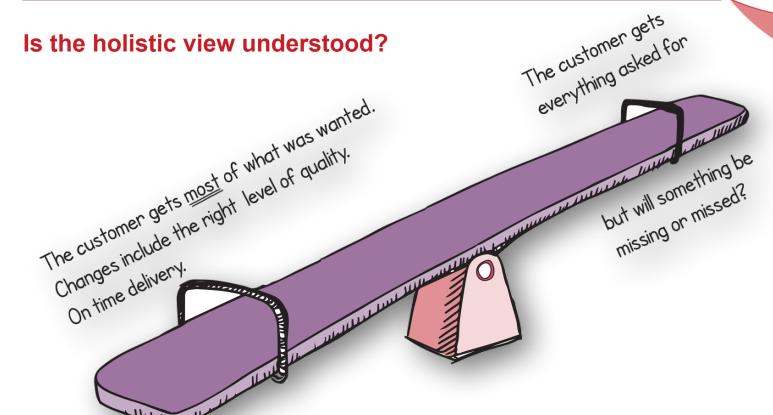
Why?

- Usually, not everything defined at the start must be delivered
- Many functions and features are rarely, or never used
- It is the safest area to compromise on
- This helps when trying to hit deadlines and protect the level of quality
- Delivers what the customer really wants more quickly.



The appropriate balance







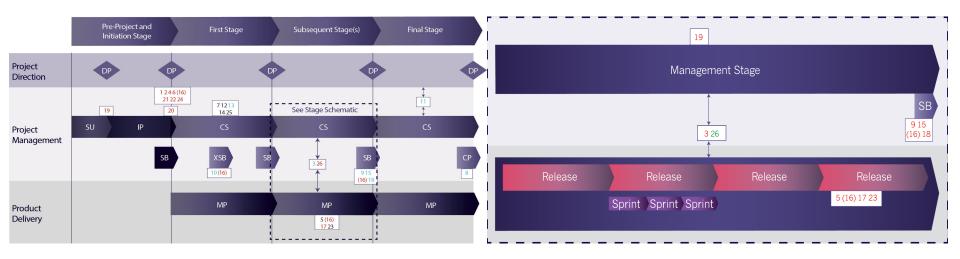


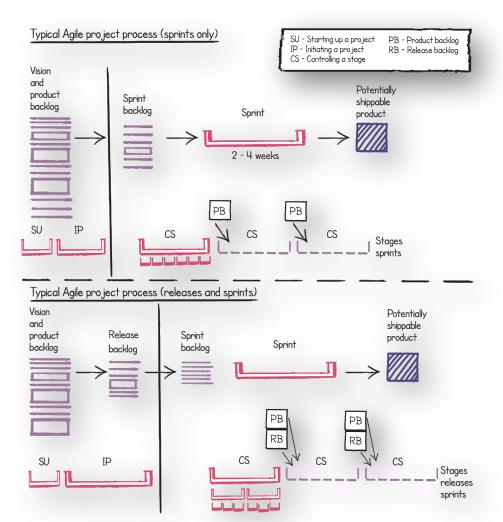
- MoSCoW Must, Should, Could, Wont have for now
- What makes a Must a Must?
- ...and a Should a Should
- ...and a Could a Could.





- Agile needs to be incorporated into all 7 processes
- The amount of agile that is appropriate to each process does vary







Relating agile processes to PRINCE2 processes

Starting up a Project and Initiating a Project



- NB: these are two distinct processes
- What you may find:
 - Up-front work how much is done?
 - emergence
 - what constitutes 'enough'
 - Visioning and chartering
 - Sprint zero, discovery
 - Starting with a backlog.

Starting up a Project and Initiating a Project



- What to do:
 - Assess the level of uncertainty
 - Cynefin
 - Define things at the right level
 - Outputs, Outcomes and Benefits
 - Project Product Description
 - High level requirements
 - Define things in the right way
 - To enable agile to work easier e.g. outcome focussed
 - Set the project up in an appropriate manner
 - Integrating with agile teams e.g. role names
 - Impact of frequent releases of products to enable and provide benefits.

Starting up a Project and Initiating a



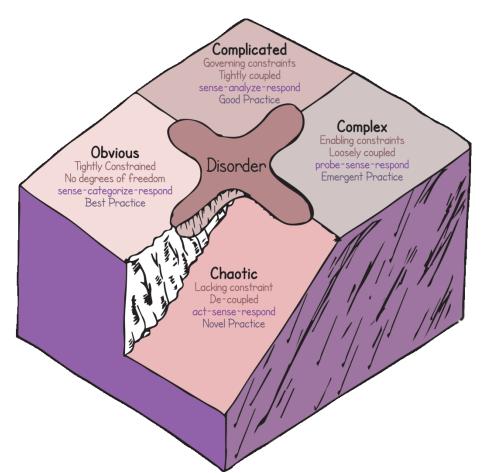


- The use of workshops
- More informal control and communication

Cynefin



- A decision making framework to help determine the level of complexity
- It describes the relationship between 'cause and effect'
- It determines how complex an environment is
- Used to help apply PRINCE2 and PRINCE2 Agile appropriately.





The Cynefin Framework

- Five domains
- Disorder is the fifth
- Can be used to assess the output, outcome or benefit
- Can be used to assess the project environment
- Collaboratively assessed to avoid people's natural tendencies.

Cynefin



- Projects will typically exist in the Complicated or Complex domains
- If work exists in the Obvious domain it will probably be handled as Business As Usual
- If work exists in the Chaotic domain it will probably be unsuitable for existing processes.





All 7 Themes need to incorporate agile

Some Themes are more prominent than others when combining PRINCE2 with agile.



Business Case - general view of agile

- It may not exist in some agile environments as there may be a focus on value assigned to features instead
- May be created at the beginning of a piece of work as part of 'sprint zero' (or similar term).





- The Business Case may be affected by the amount being delivered
- Early delivery of benefits will affect the Business Case
- The minimum viable product (MVP) will need to be clearly stated
- Project viability is not the same as the MVP
- Best-case, worst-case and expected-case will relate to the amount of features delivered
- Where there is high uncertainty this may take very little time.

Defining value



- Can often be subjective
- Easier to assign relative values
- Value can be assessed in many ways
 e.g. revenue, reducing costs, customer satisfaction
- There is a need to differentiate between output, outcome and benefit
- Outcome and benefit need to be measurable
- The Business Case should be flexible to maximise the benefit
- Focussing on benefits is helped through Collaboration.





- They need to be well defined and prioritised so that they are conducive to the agile way of working
- Many terms describe what a product does or how well it does it e.g. Requirement, Product Description, User Story
- Definitions should be at the right level and decomposed at the right time allowing then to evolve
- Boulders, rocks and gravel as a metaphor.

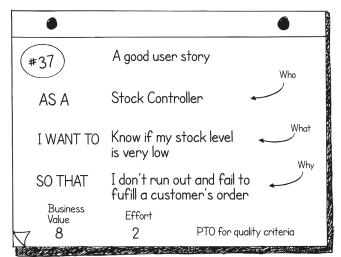
User Stories

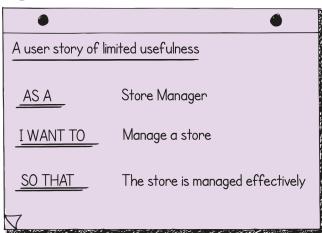


• Similar to requirements

As a <role>, I want to <function>, so that <benefit>.

- Additional information would include:
 - Acceptance Criteria
 - Effort involved
 - Value
- They are a starting point and not fully defined.







User Stories

- Takes skill to create good User Stories:
 - INVEST is used by many
 - So is a definition of 'Ready'
- Epics are a type of User Story that need to be broken down
- Technical Stories can be used for Nonfunctional requirements e.g. performance or speed.



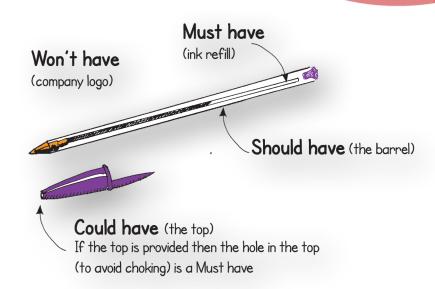


- An essential part of PRINCE2 Agile
- Two approaches frequently used are:
 - MoSCoW
 - Ordering (1,2,3,...,n)
- MoSCoW stands for:
 - Must have
 - Should have
 - Could have
 - Wont have for now.

MoSCoW and ordering

AXELOSGLOBAL BEST PRACTICE

- Is it really a Must?
 - Will it work without it?
 - Is it worth delivering it, without it?
- Can it be decomposed?
- Ordering is appropriate when:
 - There is little dependency between items
 - Items do not naturally group together.



Using prioritisation



- Can be used on:
 - Non-functional requirements
 - Quality Criteria (Acceptance Criteria)
 - Quality Review activities
- Prioritisation handles change but is it detail change or baseline change?
- Detail change can be handled dynamically.





- Agile embraces change
- Agile responds to it, welcomes it
- Change to the detail is typically viewed as positive
- Change to the baseline may not be.





- PRINCE2 could be said to be more cautious
- Blending with agile controls significant change
- Allows responsive change at the detail level
- This typifies the marriage of the two
- It is important that baselines use the correct level of detail
- Starting-up a Project and Initiation a Project can create the correct environment for this.





- Requirement definition can be binary or like a spectrum
- Good risk management can lessen the impact of change
- So can good Configuration Management
- Empowered self-organizing teams at the delivery level handle change dynamically within defined tolerances.

The Feedback Loop



- Gather feedback as quickly as possible
- Ideally 'true' feedback from the end customer
- Many forms of feedback loop exist such as:
 - OODA (Observe Orient(ate), Decide, Act)
 - PDCA (Plan, Do, Check, Act)
 - PDSA (Plan, Do, Study, Act)
 - Build, Measure, Learn (Lean Start-up).





- Self-organizing
- Two common roles:
 - Scrum Master
 - Product Owner
- Less prominence for:
 - Project Manager/Team Manager role
 - Requirements Engineer/Business Analyst (or similar) role.





- In simple terms adding PRINCE2 roles to agile delivery roles is quite straightforward
- However, how easy it is depends upon the nature of the work
- Roles need to be aligned.

Size/Scale of the work	Typical number of teams involved	Ease of synchronisation
Small	One	Probably straightforward
Large	Many (at least more than one)	Probably requires a degree of care





- Common agile concepts may need to be adjusted
- Agile teams prefer to be led and coached as opposed to managed
 - Mapping the Team Manager roles needs to be handled appropriately
- Common agile guidance refers to a single Product Owner
 - PRINCE2 Agile recommends taking a more blended view
- Common agile guidance does not have a Project Manager role
 - PRINCE2 see this role as mandatory on a project
- The Team Manager role needs to integrate into an agile delivery team.

Organisation



- How the Project Manager relates to the delivery team is key
- There are 3 options:
 - Leave the delivery team roles as they are
 - Leave the delivery team roles as they are but identify a single point of contact for the Project Manager
 - Create a Team Manager role in the delivery team
- The choice will be made according to the circumstances of the project.





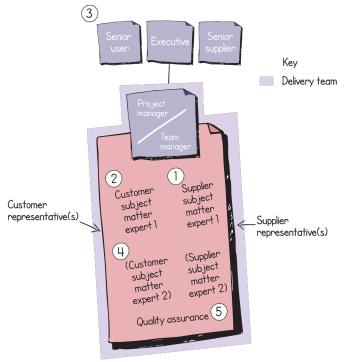
Usually represented by some or all of the following:

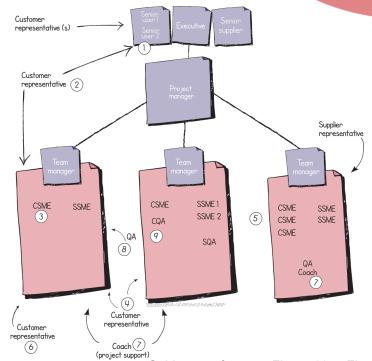
- Someone to lead the team
- Someone from the customer (or at least someone to represent the customer)
- A team to create the product
- Someone to assure the quality of the product
- Someone to coach the team (which includes coaching them in agile)
- Multi-skilled 'generalising specialists'.





All roles need to be conversant with working in an agile way





Guidance reference: Figure 10.4, Figure 10.5



Agenda for Day 2

- Servant Leadership
- Principles and behaviours
- The Agilometer and the Risk Theme
- Managing Product Delivery (incl. Scrum, Plans and Progress, estimation, burn charts, IRs and Work Packages)
- Quality (Incl. DoD)
- Controlling a Stage and Managing a Stage Boundary (Incl. Frequent Releases and Retrospectives)
- Directing a Project.

Servant Leadership



- Usually seen as best practice for leading an agile team
- The best way to lead a team is to be its servant
- Some of PRINCE2 could be said to be in conflict with this
- The focus is on:
 - Empowerment
 - Collaboration
 - Facilitation
- How much this can be used depends on context
- Nothing in PRINCE2 limits the use of Servant Leadership.





- The simplicity of the Product Owner role has limitations in a project context such as:
 - Project size
 - Size of the role
- The detailed view and the wider, higher level view needs to be reflected
- A specialist role can be used to define requirements
- Senior User is ultimately responsible
- The Product Owner operates best from inside a delivery team.





- Used to evolve the effectiveness of a team
- Typically they are made visible
- Collaboratively defined, built by consensus
- Need to be built with care.





Agile has a strong focus on principles (see Appendix E).

All PRINCE2 principles are applicable when using agile.

PRINCE2 Agile identifies 5 behaviours to be monitored.

PRINCE2 Principles - guidance



- Continued Business Justification
 - The rationale behind the MVP should be understood
- Learn from Experience
 - Many agile concepts support this and should be used
- Defined Roles and Responsibilities
 - Delivery roles will apply and should be carefully mapped
- Manage by Stages
 - Timeboxes should be integrated carefully
- Manage by Exception
 - This is at the heart of empowering people with the appropriate level of governance
- Focus on Products
 - This makes it easier to stay in control and focus on value
- Tailor to suit the project
 - Further specific tailoring is enabled by assessing the risks associated with agile.

PRINCE2 Agile Behaviours



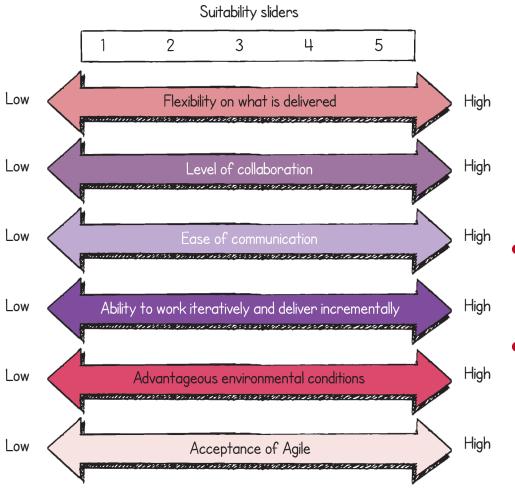
The 5 behaviours are:

- Transparency
- Collaboration
- Rich Communication
- Self-organization
- Exploration.





- An assessment tool providing guidance on risks and benefits from an agile perspective
- Assessed at the start of a project and throughout
- The Project Manager facilitates the assessment
- It is a guide to make an informed decision
- Use the sliders in isolation, do not create an average
- After the assessment, can any sliders be improved?
- The question is always 'how much' and not 'yes or no'.





The Agilometer

- The Agilometer in PRINCE2 Agile has 6 key areas
- This represents a starting point, it can be tuned.

Risk



- Generally there is relatively less prominence given to the area of risk in agile
- Agile concepts mitigate many risks associated with other approaches (e.g. waterfall)
- PRINCE2 brings a level of formality and planning to risk management
- The level of formality should be appropriate
- Address risk during stand-up meetings
- Agile has risk areas of its own (assessed by The Agilometer)
- The 5 behaviours in PRINCE2 Agile help manage risk.





What you may find:

- A strong focus in this area
 - Managing product delivery
 - Product delivery practices
- Scrum manages product delivery

Guidance reference: Section 20.2





What to do:

- Use Work Packages appropriately
- It is a vital interface/link to be managed
- Perhaps define tolerances on what is delivered at the Work Package level
- Ensure that Product Descriptions are defined at the right level
- Agree and/or give guidance on such areas quality, releases and risk.

Guidance reference: Section 20.3





How it might look

- Work Packages are collaboratively defined
- Reporting arrangements have an appropriate level of formality and transparency
- The wider project context is understood.

Guidance reference: Section 20.3

Work Packages



- A vital interface to blend to the project environment
- Collaboratively defined
- Agreed levels of detail and formality with respect to reporting, tolerances, product descriptions and quality criteria
- Guidance on behaviours, risk and the frequency of releases
- Guidance on quality checking techniques.





- A framework for developing and sustaining complex products
- A collection of roles, events, artifacts and rules
- Created by Schwaber and Sutherland (c. 1995)
- A way to assess the efficiency of your practices so that you can improve.

Scrum theory



- Founded on empirical process control theory
- Decisions based on evidence
- Covers three areas:
 - Transparency
 - Inspection
 - Adaptation.

The Scrum team



- Self-organising
- Cross-functional
- Flexible, creative, productive
- The Roles:
 - The Product Owner
 - The Development Team
 - The Scrum Master.

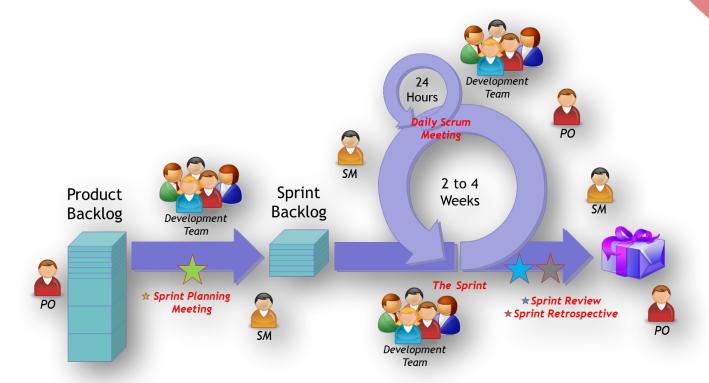
Scrum events



- Events are prescribed to create regularity
- Every event has a maximum duration
- The Sprint is at the heart of this concept
- Forces transparency
- Opportunities to inspect and adapt.

Scrum

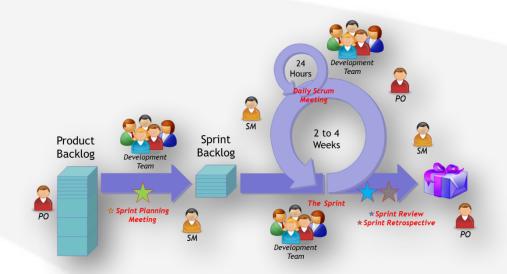




The 5 Scrum events



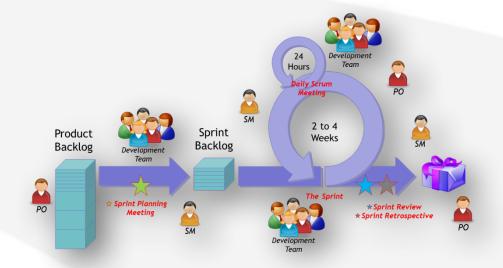
- The Sprint
- Sprint Planning Meeting
- Daily Scrum Meeting
- Sprint Review
- Sprint Retrospective.



Scrum artifacts



- Product Backlog
- Sprint Backlog
- Increment.







- Agile puts a lot of emphasis on planning
- Agile mostly uses Empiricism as opposed to Rationalism
- Planning style:
 - Based on features
 - A team-based exercise
 - Plan at the last responsible moment
- Using a points scoring system to estimate is popular although there are other ways.





- PRINCE2 supports any planning style
- A PRINCE2 project needs an end date
- Both agile and PRINCE2 accept the premise of planning horizons
- Higher level plans may more conventional but not necessarily
- Product-based planning can be used easily for any level of plan (see Appendix D)
- Planning needs to consider dependencies and grouping similar work items.

Estimation



- Using some form of points system is popular
- Estimating starts with relative estimates and is carried out as a team
- Team members estimates should not be affected by others
- Differences are discussed
- The Fibonacci sequence or T-shirt sizing are very popular systems.





- Similar to plans in that it is of particular importance
- Burn charts and WIP boards are commonly used
- The bigger picture relates to key agile values (e.g. transparency).





- What is being delivered is of the most importance
- It may take several sprints to understand how things are progressing
- In terms of 'how are things going?', it is a necessity to be in control at all times
- Many agile techniques can be used at any level of a project
- The audience for any technique will need to be comfortable with the format.

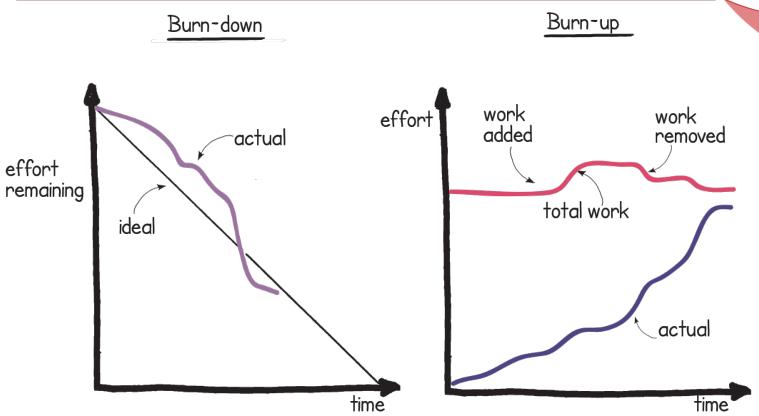
Burn charts



- A popular technique that comes in two forms:
 - a burn-down chart
 - a burn-up chart
- Displayed in the form of a graph with an x and y axis
- They show the current situation and the current rate of progress
- Burn-down charts assume the amount of work doesn't change
- Burn-up charts should be used if the amount of work is likely to change
- They focus on what has been completed.

Burn charts





Information Radiators



- Creates visual information that can be accessed immediately
- Best created and maintained manually
- Contributes significantly to transparency
- Information is 'pushed' as opposed to 'pulled'
- Can display a wide variety of information
- Holding a daily stand-up meeting by a display enables it to be updated immediately.





- Prominent techniques such as the definition of 'Done' and acceptance criteria address quality control
- Concepts such as 'test as you go' or 'test first' are used for testing and quality checking
- Evolving the definition of 'Done' is commonly used
- In some agile environments there may not be a lot of emphasis given to quality planning and quality management, during the start of a project.

Quality - guidance



- Product Descriptions are flexible (e.g. they can be User Stories)
- The Project Product Description purpose would preferably be defined as an outcome
- Quality management and quality planning includes:
 - Which tools and approaches are to be used
 - The role of the customer (an essential ingredient)
 - Assessing and costing the resources
 - Quality control considerations.





- Care needed when transferring these concepts from the software development domain
- Common agile terms include:
 - Test-driven Development (TDD)
 - Behaviour-driven Development (BDD)
 - Definition of 'Done'
 - Definition of 'Ready'
 - Refactoring
 - Technical debt.



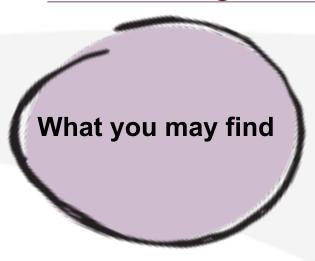


- Quality is defined by quality criteria
- Scope is defined by the products themselves.

- A reduction in scope is not seen by PRINCE2 as a reduction in quality
- Customer Quality Expectations are set and this level needs to be maintained

Controlling a Stage





- Stages may not exist as described in PRINCE2
- Container timeboxes e.g. a release
- Scrum of scrums.

Controlling a Stage



What to do:

- Plan around features
- Create flexible Work Packages in order to:
 - Make management by exception easier
 - Empower the teams to self-organise
 - Enable rich communication
- Control focusses on what is being delivered
 - Scope
 - Quality criteria
- Monitor agile risks.

Controlling a Stage



How it might look:

- Collaborative
- Visual
 - Information Radiators
 - Stand-up meetings
 - Demos
- Empirical
- Inspect and adapt.





- Frequent releases are important when using agile
- The advantages are:
 - Early delivery of benefit
 - Allows for feedback
 - Reduces risk (e.g. of delivering the wrong product)
 - Gives confidence of how the project is progressing
 - Fosters stakeholder engagement
 - Makes releasing easier
- Needs to be planned (Product-based Planning can be used)
- A release may not go into the operational environment.

Frequent releases



- Planned releases would show on Project and Stage Plans
 - Showing when they will take place
 - Showing what features are to be released
- Different views need to be taken into account when planning releases
 - The Project Board want to establish appropriate stage boundaries
 - Early benefits may need to fund later parts of the project
 - Enabling a product to fail fast, if it is going to fail
 - Products can be delivered too quickly (leading to disruption)
- Releasing into operational use is the ideal situation otherwise a staging area needs to be used which compromises agile to a degree.





What you may find:

- Stages may not exist as described in PRINCE2
- Viability decisions not usually planned in advance
- Similarity between a stage and a release (where a release is a container time box)
- How are things progressing
- How are the team and the processes working.





What to do:

- Assess how much is being delivered
- Assess the quality of what is being delivered
- What benefits have been realised?
- Is agile being use appropriately?
- Do the processes need improving?





How it might look

- Has many similarities to Controlling a Stage
 - Visual
 - Empirical
 - Inspect and adapt
- A point of formality carried out with as little ceremony as possible

Retrospectives



- Reviews the way of working (not what was produced)
- A significant technique when working in an agile way
- They need to be planned and structured (such as with a workshop)
- Covers what went well and what didn't go so well
- Improve little by little and little and often
- Keep them effective by introducing variety
- Feedback can come in the form of facts or feelings.





What you may find will vary according to the level of agile maturity.

Type of Agile Environment	Who gives 'Direction'
Basic	Product Owner
Between Basic and Mature	Sponsor
Mature	Similar to PRINCE2

Guidance reference: Section 18.2





What to do

- Manage by Exception with the emphasis on:
 - Empowerment
 - The amount being delivered
 - Rich information flows
 - Value being delivered.





How it might look

- Pulling information rather than having it delivered
- Collaborative working
 - Trusting
 - Absence of a blame culture.

Guidance reference: Section 18.3



Agenda for Day 3

- Agile Contracts
- Closing a Project
- Management Products
- Communication (Incl. workshops)
- Kanban
- Lean Start-up
- General guidance (Incl. Health Check, Transitioning, PM tips)
- Recap
- Examination preparation
- Examination.





- Contracts may need to be created in a way amenable to agile
- An issue with traditional contracts is that requirements change and someone will need to allow for this
- Trust is important as it determines the amount of risk that is shared
- Guidance on structure in order to create the right behaviours:

Focus on outcomes or throughput in preference to outputs.

Define the amount of customer involvement required in order to collaborate with the supplier in the best way.

Buy amounts of time relating to timeboxes with deliverables.

Allow for a premature end to the project.

Relate incentives to the amount delivered (value or throughput).

Avoid including detailed requirements.

Prioritize the requirements and identify a MVP.

Handle changing requirements by trading out the less important ones.

If preferred, build a contract up from the 'minimum' to start with.

Guidance reference: Section 28.3

Closing a Project

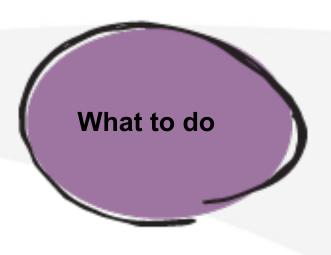




- Defined processes may only exist in mature agile environments
- Regular handovers have resulted in activities becoming second nature.

Closing a Project





- Check against the original baseline
- Evaluate the use of agile on the project
- Ensure the formality of User Acceptance is appropriate
- Finalise documentation that has been created iteratively and incrementally.

Closing a Project





- A workshop is used
- Benefits are already being delivered
- The majority of the information has already been completed.



PRINCE2 Management Products and Roles

- All 26 PRINCE2 Management Products are in appendix A
- The PRINCE2 roles are in appendix B along with the PRINCE2 Agile delivery roles.





- Can exist in a wide range of formats and level of formality
- They are not necessarily documents
- They may need to include additional information
- Some products are significant/important:











Communication problems need to be proactively addressed Effective communication is fundamental to agile Communication takes place in many ways therefore people should interact appropriately Communication channels:

- The written word
- Visualisation
- Verbally by telephone
- Verbally face-to-face.





- Face-to-face should be favoured as a faster and clearer channel
- Technology and the level of formality needs to be assessed
- There is a role for the written word but is has disadvantages
- A Project Manager (or Team Manager) needs to be aware of how a team is communicating
- Communication needs to be organised and planned.

Workshops



- Harnesses interactions and creativity
- Ideally run by a neutral facilitator who manages the workshop
- Preparation is essential
- Many tools and techniques exist
- Workshops can be used at any point on a project
- It is important to get the group dynamics right
- Correctly run workshops can create high quality outputs quickly
- This leads to clarity, consensus and ownership.



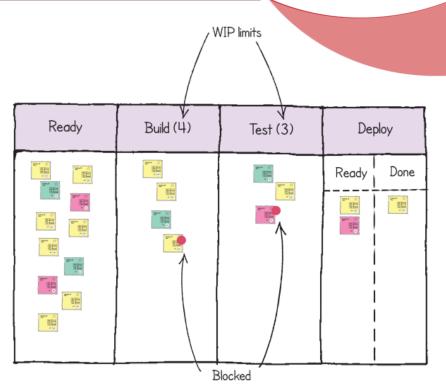


- Kanban systems are visual management systems that limit the number of work items in circulation
- Kanban should be seen as a way to increase agility through:
 - Improved day-to-day decision making
 - The deferral of commitment
 - Reduced lead times
- In PRINCE2 Agile it is applicable in a project context to time boxes.



1. Visualize

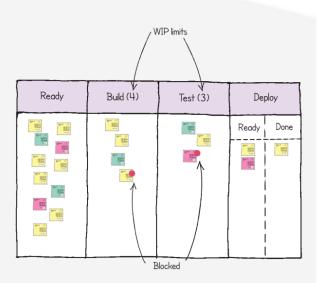
- To show how work is progressing
- To show what is still to do
- To show what problems exist.





2. Limit 'Work in Progress' (WIP)

- A fundamental concept in Kanban that may appear counterintuitive
- WIP limits underpin the 'pull' system
- Kanban avoids scheduling work at specific times
- It pulls work when capacity exists
- Reduces the impact of task switching and multitasking.



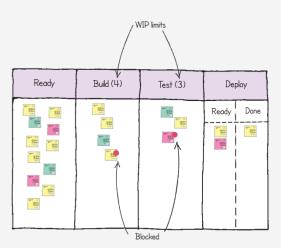


3. Manage the flow

- The team constantly looks at ways to maximise flow
- Waste is removed as quickly as possible.

4. Making policies explicit

- Boundaries need to be clearly defined about how a team works
- Policies should evolve over time.



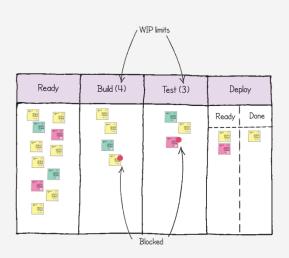


5. Implement feedback loops

- Ultimately, value being delivered is judged by the final consumer
- Quantitatively assessing this will directly affect what will subsequently be delivered.

6. Improve collaboratively, evolve experimentally

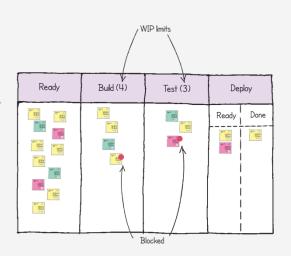
- The method builds on collaboration through experimental improvement
- Process improvement is everyone's business every day.







- Scrumban is the application of Kanban where the underlying process is based on Scrum
- Policies may exist for similar work items as flow may be more predictable
- A team may look to improve how the system works by carrying out experiments in a controlled and objective way.



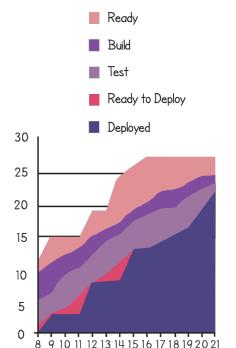




Column Counts

Day	Ready	Build	Test	Ready to Deploy	Deployed
8	2	4	4	1	0
9	4	5	3	0	3
10	3	3	5	11	3
11	2	3	4	3	3
12	4	3	4	0	8
13	3	2	5	1	8
14	7	2	4	3	8
15	7	2	4	0	13
16	7	2	5	0	13
17	5	_ 3	5	0	14
18	5	3	4	0	15
19	4	2	5	0	16
20	3	2	3	0	19
21	3	1	1	0	22

Cumulative Flow Diagram (CFD)



- Cumulative Flow
 Diagrams (CFDs) track
 work items and show the
 amount of work in each
 column each day
- In simple terms WIP is the vertical difference between the top and bottom lines whereas the horizontal difference shows the lead time.

Kanban hints



- A timebox (e.g. a stage) can be run with or without sprints
- Kanban aims to reduce the cost of delay
- Change to existing processes should be gradual
- Little's Law (in simple terms):

Average number of items in the system = average arrival rate x average time an item spends in the system

Little's Law is used for forecasting.

Lean Start-up



- Core concepts are:
 - Build, Measure, Learn
 - Create a Minimum Viable Product (MVP)
 - Fail fast
 - Validated learning.
- Lean Start-up focusses on uncertainty, learning and handling change
- It refers to 'shortening or accelerating the feedback loop'
- Learning needs to be measurable.

Lean Start-up

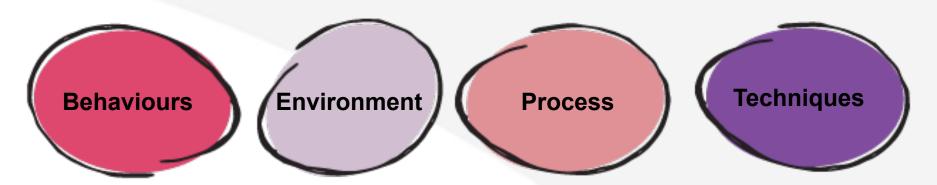


- If you are going to fail, fail fast the key point being 'the later you fail the later you learn'
- A 'pivot' is a significant change in direction as a result of something major surfacing that wasn't expected
- Minimum Viable Product (MVP) 'version of the final product which allows the maximum amount of validated learning with the least effort'.
- An MVP may not go into operational use and may be an experiment
- Releasing frequently can help with metrics.





- A checklist to assess how well a project is going from an agile perspective
- The PRINCE2 health check should be used as well
- The totality of answers will give a general indication
- Covers the following areas:







- Agile is an enabler that helps solves problems and leverage opportunities
- Create a baseline before starting the journey
- There are distinctly different types of success on a project:
 - The success of the Business Case
 - The success of the project
 - The success of agile
- Investing in agile needs to be measured in order to gauge success.

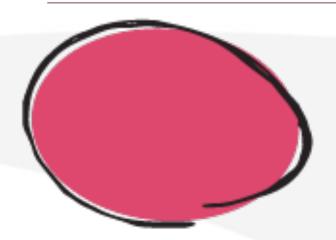




- Tips created by the team that created PRINCE2 Agile
- Covers the following areas:
 - Collaboration and self-organisation
 - Transparency, Communication and Exploration
 - Environment
 - Plan, Monitor and Control.







In summary



- PRINCE2 can be very effective in an agile context
- Tailoring is about creating an appropriate blend of the two
- PRINCE2 is already enabled to work with agile
- Agile covers a wide range of behaviours, concepts, frameworks and techniques
- Using agile is always a question of 'how much can be used according to the situation?'