IFB295 – IT Project Management

DSDM Agile Project (DAP) Framework

Lecture 4

DSDM Philosophies, Principles, Lecture, Roles

Lecturer - Prakash Bhandari

Agenda

1. Overview of Agile
2. Agile SW Development Manifesto
3. What is DSDM
   - Philosophy, Principles & Lifecycle
4. Preparing for Success
5. Assessment 1 Discussions

Agile Overview - What is Agile?

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**

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

But Agile is not just about delivering software; it applies to all types of projects.

Agenda

1. Overview of Agile
2. Agile SW Development Manifesto
3. What is DSDM
   - Philosophy, Principles & Lifecycle
4. Preparing for Success
5. Assessment 1 Discussions

Manifesto for Agile Software Development Twelve Principles

A **Manifesto** is:
"*a written statement declaring publicly the intentions, motives, or views of its issuer*"

A **Principle** is:
"*a comprehensive and fundamental law, doctrine, or assumption*"

Definitions retrieved from:
https://www.merriam-webster.com/dictionary

Hillary, M. (2015). History of Agile Manifesto. Retrieved from http://margarethillary.com/history-of-agile-manifesto/

Manifesto for Agile Software Development
Twelve Principles

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

Manifesto for Agile Software Development Twelve Principles

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

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

Manifesto for Agile Software Development Twelve Principles

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage

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

Manifesto for Agile Software Development
Twelve Principles

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale

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

Manifesto for Agile Software Development
Twelve Principles

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale
4. Business people and developers must work together daily throughout the project

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

Manifesto for Agile Software Development Twelve Principles

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale
4. Business people and developers must work together daily throughout the project
5. Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done

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

Manifesto for Agile Software Development
Twelve Principles

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale
4. Business people and developers must work together daily throughout the project
5. Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation

7. Working software is the primary measure of progress

Manifesto for Agile Software Development
Twelve Principles

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale
4. Business people and developers must work together daily throughout the project
5. Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation
7. Working software is the primary measure of progress

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

Manifesto for Agile Software Development Twelve Principles

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale
4. Business people and developers must work together daily throughout the project
5. Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation
7. Working software is the primary measure of progress
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely

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

Manifesto for Agile Software Development Twelve Principles

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

Manifesto for Agile Software Development Twelve Principles

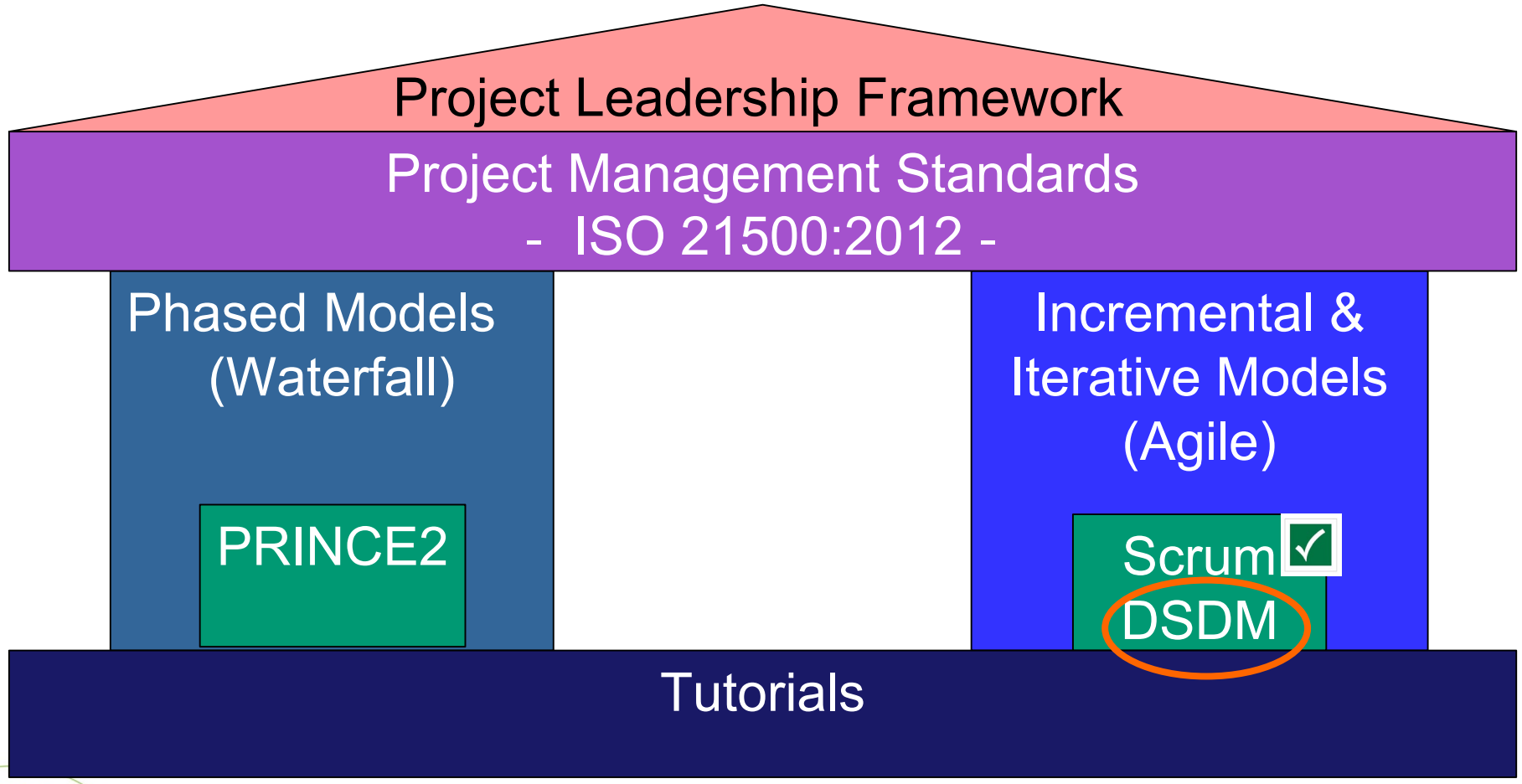
1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage
3. Deliver working software frequently, from a couple of weeks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

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

Manifesto for **Agile** Software Development Twelve Principles

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

Unit Themes


Project Leadership Framework
Project Management Standards
- ISO 21500:2012 -
Phased Models (Waterfall)
PRINCE2
Incremental & Iterative Models (Agile)
Scrum
DSDM
Tutorials

Agenda

1. Overview of Agile
2. Agile SW Development Manifesto
3. What is DSDM
   - Philosophy, Principles & Lifecycle
4. Preparing for Success
5. Roles & Team Performance

What is DSDM?

Dynamic System Development Methodology (DSDM)

DSDM Agile Project (DAP) Framework

- The DSDM Agile Project (DAP) Framework Handbook (2014 Onwards)
- DSDM is free to view and use online (see Blackboard readings)
- Leading, proven, Agile approach
- DSDM - the oldest established Agile approach
  - Originally launched in 1995
- Owned by The DSDM Consortium
  - A not-for-profit collegiate organisation
  - https://www.agilebusiness.org/
- Established and proven integration between *DSDM®, Atern®, and **PRINCE2®

*DSDM, Atern, and AgilePM are registered trade marks of Dynamic Systems Development Method Limited in the United Kingdom and other countries..
** PRINCE2® is a Registered Trade Mark of the Office of Government Commerce in the United Kingdom and other countries.

Why DSDM?

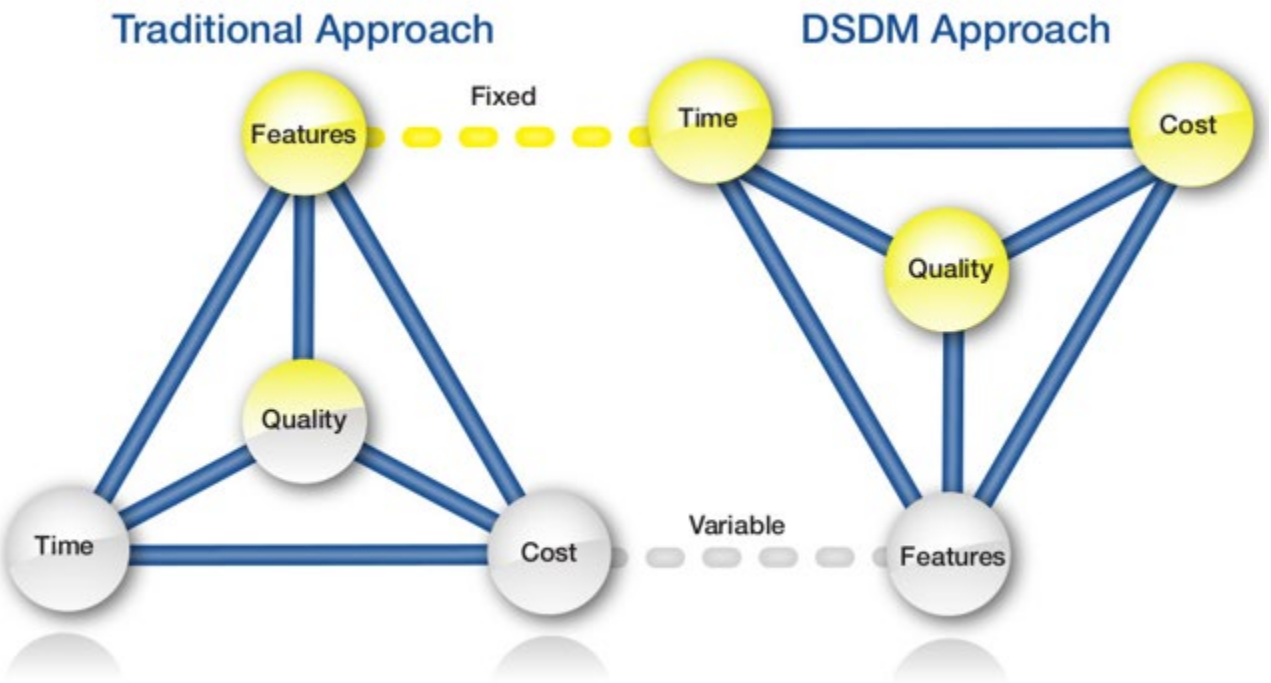
DSDM has:

- A broader focus than most other Agile approaches in that it deals with projects rather than just the development and delivery of a product (typically software).
- A long track record of successful Agile project delivery in all types of corporate environments
- Proven to be fully scalable, working effectively in small simple businesses, large and complex organisations, and in highly regulated environments.
- Shown to be equally effective for both IT and non-IT projects, for example business change projects.

DSDM takes a pragmatic approach, recognising that it often needs to work alongside existing standards and approaches such as PRINCE2 .

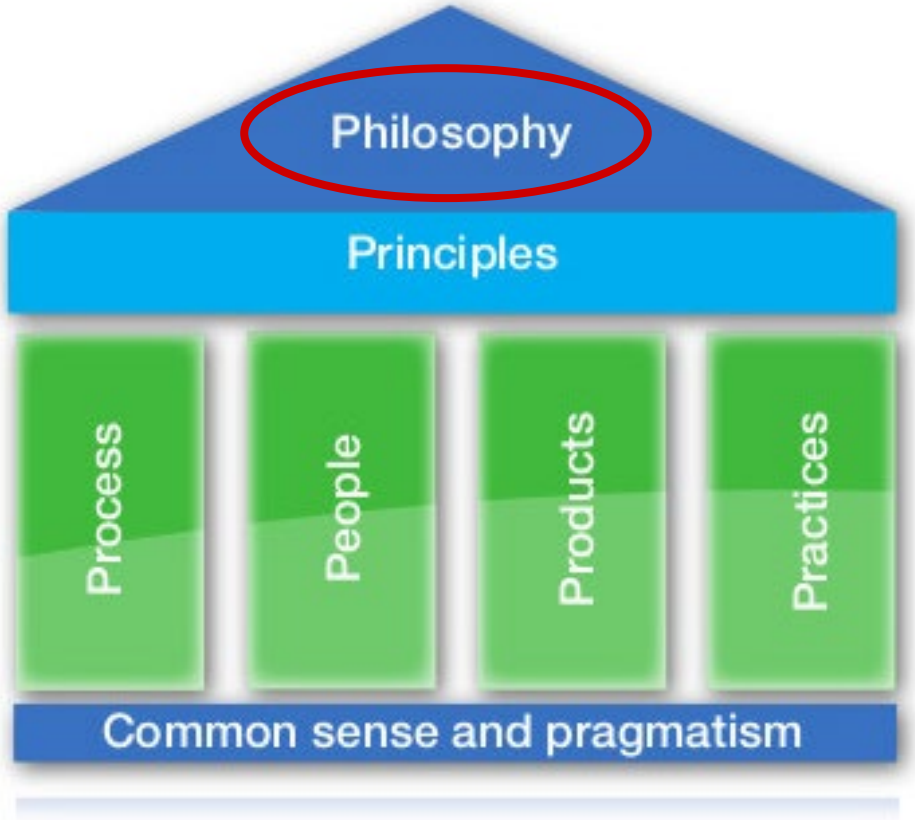
DSDM vs Traditional (Waterfall)

What is the difference?



PARADIGM SHIFT

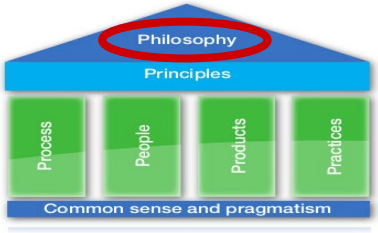
DSDM Philosophy



"best business value emerges when projects are aligned to clear business goals, deliver frequently, and involve the collaboration of motivated and empowered people"

DSDM Philosophy

"best business value emerges when projects are aligned to clear business goals, deliver frequently, and involve the collaboration of motivated and empowered people"

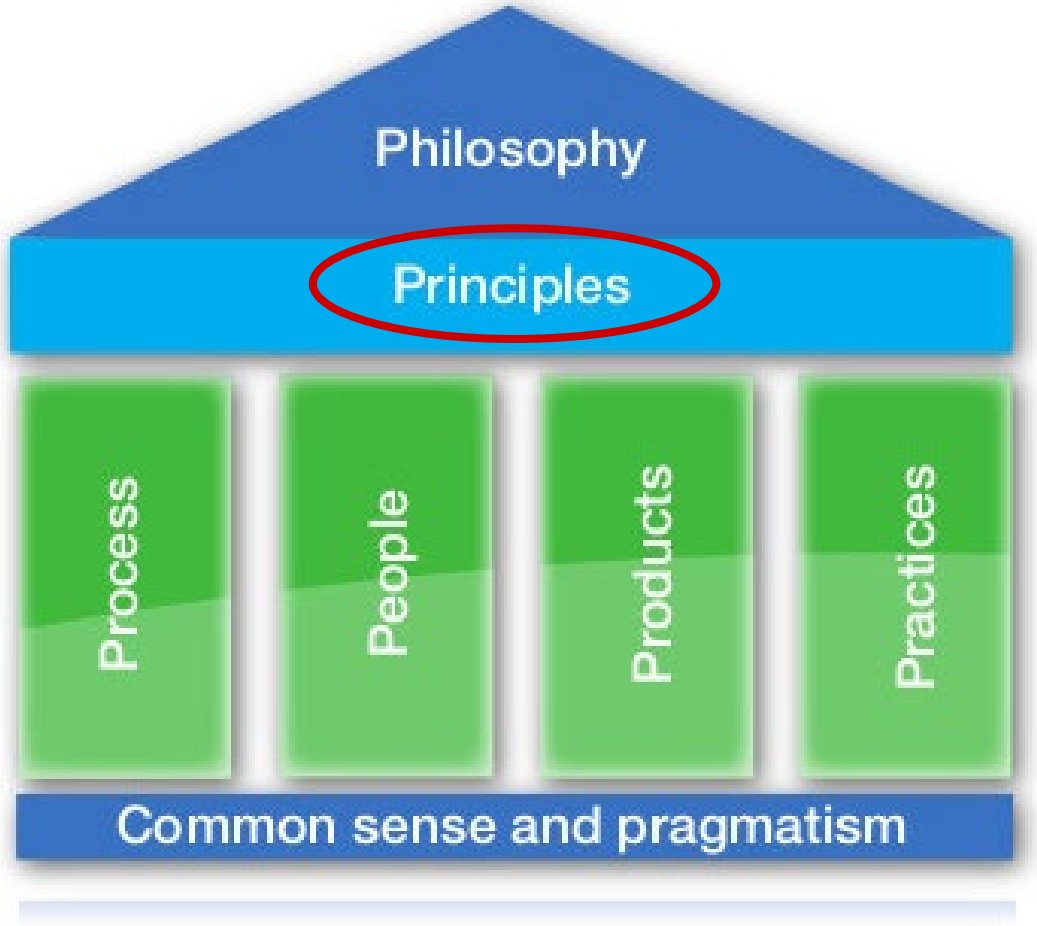


This is achieved when all stakeholders:

- Understand and buy into the business vision and objectives
- Are empowered to make decisions within their area of expertise
- Collaborate to deliver a fit for purpose business solution
- Collaborate to deliver to agreed timescales in accordance with business priorities
- Accept that change is inevitable as the understanding of the solution grows over time

Stakeholders encompass everybody inside or outside the project who are involved in/or affected by it.

DSDM Principles



DSDM Principles

- Principles support the philosophy
- Highlight attitude and mindset needed by team
- Compromising any principle undermines philosophy
  - And introduces risk
- Applying all 8 principles ensures maximum benefit
- Collectively principles enable organisations to collaboratively deliver best value solutions

The 8 DSDM Principles

The eight principles of DSDM support DSDM's philosophy that:

"best business value emerges when projects are aligned to clear business goals, deliver frequently, and involve the collaboration of motivated and empowered people".

1. Focus on the business need
2. Deliver on time
3. Collaborate
4. Never compromise quality
5. Build incrementally from firm foundations
6. Develop iteratively
7. Communicate continuously and clearly
8. Demonstrate control

Principle 1

Focus on the business need

- Decisions based around project goal
  - To deliver what business needs it to deliver, when it needs to be delivered
- Requires the team to:
  - Understand true business priorities
  - Establish sound business case
  - Seek continuous business sponsorship and commitment
  - Guarantee Minimum Usable Subset

- Supported by:
  - Business roles
  - Business Products agreed at Foundations stage
  - Key techniques - MoSCoW prioritisation and Timeboxing

Every decision taken during a project should be viewed in the light of the overriding project goal - to deliver what the business needs to be delivered, when it needs to be delivered.

Principle 2

Deliver on time

- Requires team to:
  - Timebox the work
  - Focus on business priorities
  - Always hit deadlines
  - Build confidence through predictable delivery
- Supported by:
  - Key techniques: Timeboxing and MoSCoW
    - To build a reputation for timely and predictable deliveries

Principle 3

Collaborate

- Requires team to:
  - Involve the right stakeholders at the right time, throughout the project
  - Ensure it is empowered to make decisions on behalf of those they represent
  - Actively involve business representatives
  - Build one-team culture

- Supported by:
  - Business roles
  - Key technique: Facilitated workshops

Teams that work in a spirit of active cooperation and commitment will always outperform groups of individuals working only in loose association.

Principle 4

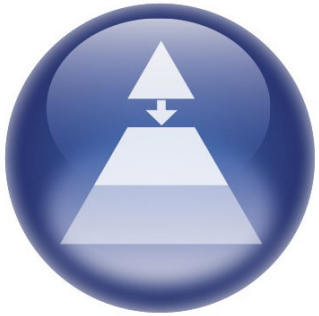
Never compromise quality

- Requires team to:
  - Set level of quality at the outset before work starts
  - Ensure quality does not become a variable
  - Test early, continuously, and to the appropriate level
  - Build in quality by constant review
  - Design and document appropriately
- Supported by:
  - Testing products
  - Early and integrated testing
  - Regular reviews throughout lifecycle
  - Key techniques: MoSCoW and Timeboxing

Principle 5

Build incrementally from firm foundations

- Requires teams to:
  - Carry out appropriate analysis and *enough design up front* (EDUF) to create strong foundations
  - Strive for early delivery of business benefit where possible
  - Accept that most detail emerges later rather than sooner
  - Evolve more precise estimates as the project progresses
  - Formally re-assess priorities, and informally re-assess ongoing project viability with each delivered Increment
- Supported by:
  - Creating a solid base of knowledge during Feasibility and Foundations phases. This forms the base for developing incrementally thereafter, with incremental deliveries from one or more Sprints typically contributing to a formal Release



Principle 6

Develop iteratively

- Iterative development allows team to converge on accurate solution
- Rare that anything built perfectly 1st time
- Requires team to:
  - Build products using an iterative approach
  - Continually confirm the correct solution is being built
  - Accept that most detail emerges later rather than sooner
  - Embrace change – the solution will evolve as team learns more about it
  - Be creative, experiment, learn, evolve
- Change is inevitable, allow for it and harness its benefits
- Supported by:
  - Iteration and constant review ensures the evolving solution aligns with what the business really needs

Principle 7

Communicate continuously and clearly

- Requires team to:
  - Encourage informal, face-to-face communication at all levels
  - Run daily stand-up sessions
  - Use facilitated workshops
  - Use ‘Rich Communication’ – modelling, prototyping
  - Demonstrate iterations of evolving solution early and often
  - Keep documentation lean and timely
  - Manage stakeholder expectations throughout
  - Always aim for honesty and transparency in all communication
- Supported by:
  - User involvement and empowerment
  - Stand-up and Facilitated workshops
  - Clearly defined roles and user involvement
  - Models and prototypes – to make early instances of solution visible

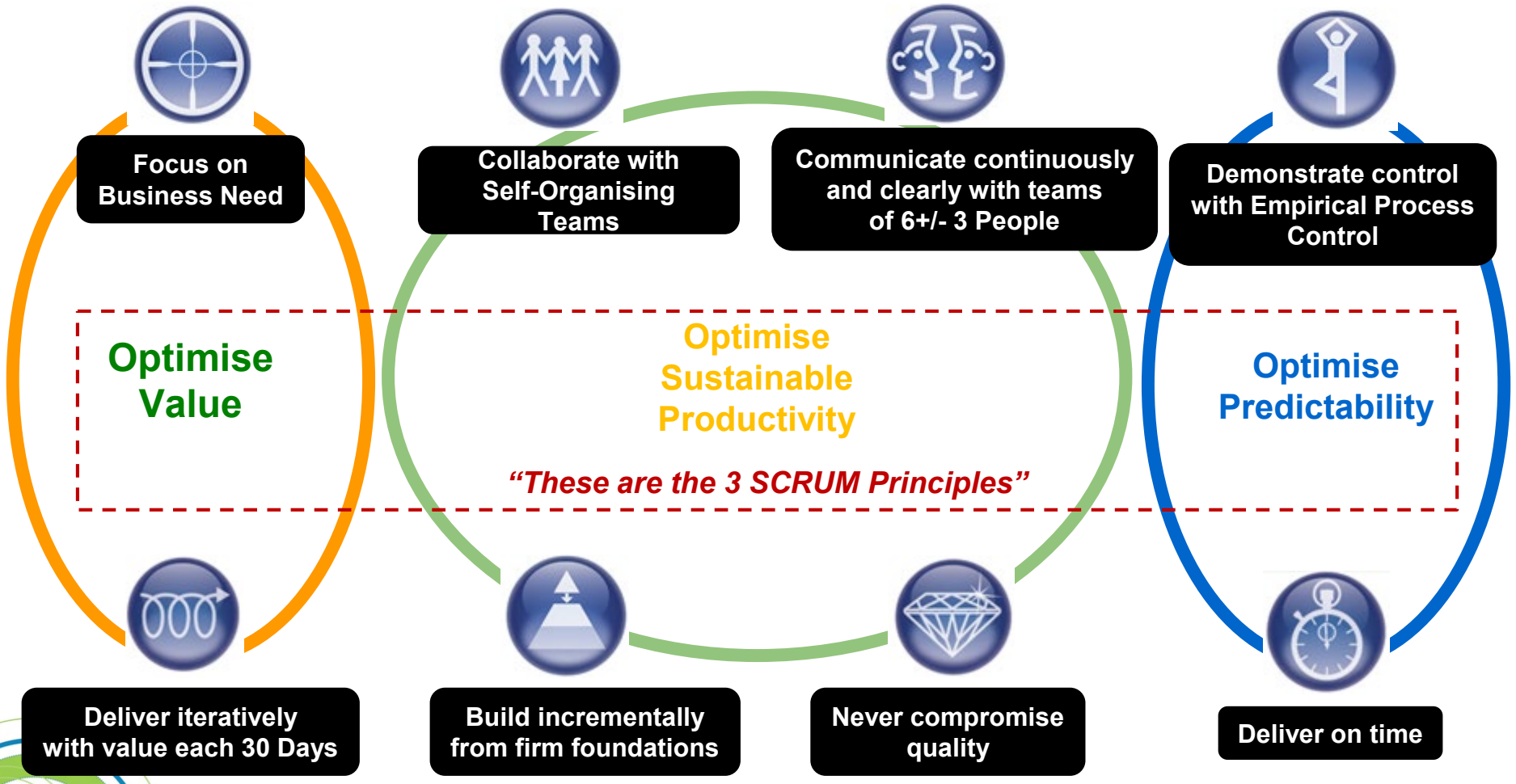
Principle 8

Demonstrate control

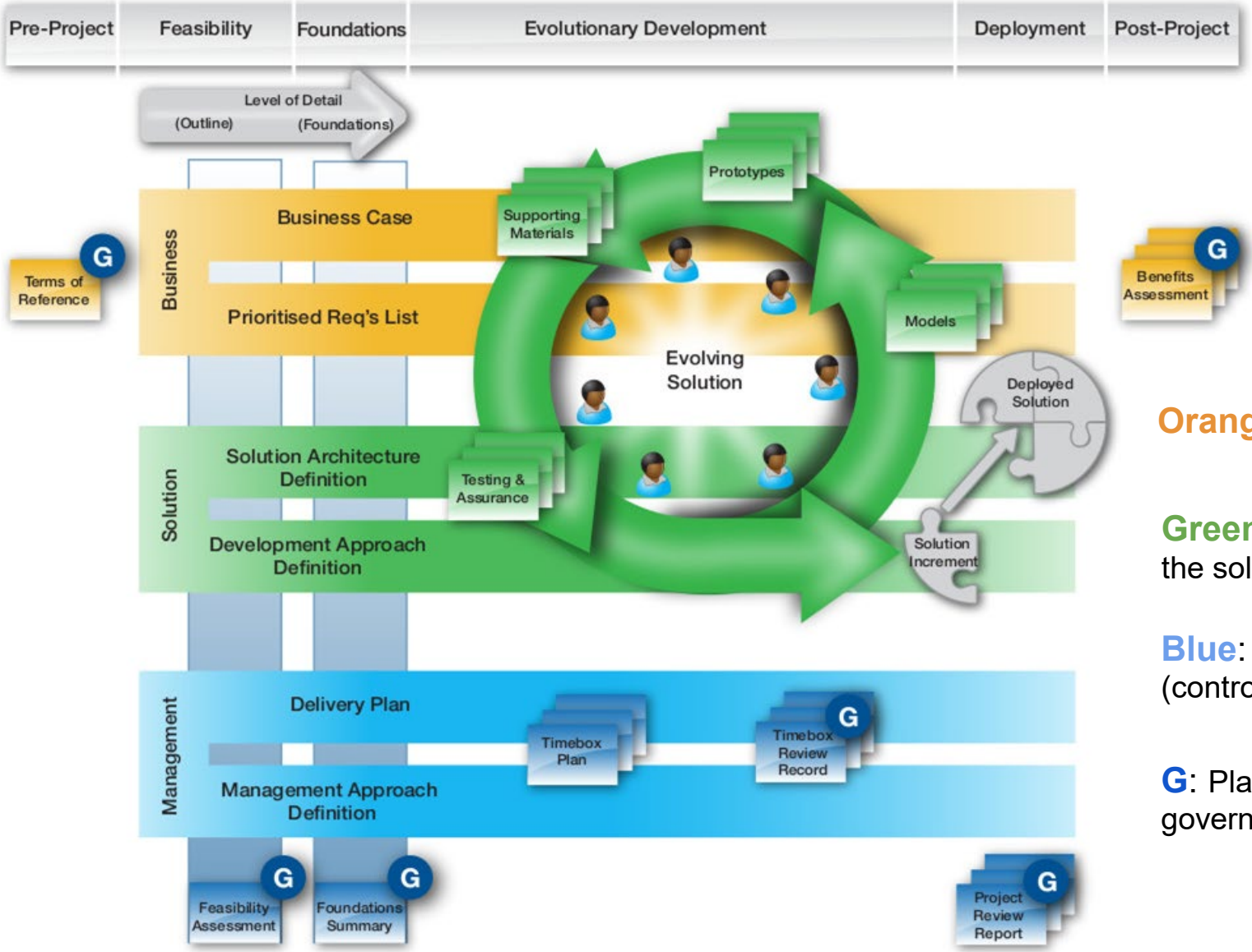
- Requires team, especially Project Manager and Team Leader, to:
  - Make plans and progress visible to all
  - Measure progress through delivery of products
  - Manage proactively
  - Continuously evaluate project viability based on business objectives
  - Use appropriate level of formality for tracking and reporting

- Supported by:
  - Key technique: Timeboxing
  - Constant review
  - Planning products
    - Management Foundations and Timebox Plans

Alignment of 8 DSDM & 3 SCRUM Principles



DSDM Lifecycle (Process, Phases & Products)



Orange: Business-focused

Green: Contributes to the solution being created

Blue: Project management (control interests)

G: Plays a part in governance processes

Agenda

1. Overview of Agile
2. Agile SW Development Manifesto
3. What is DSDM
   - Philosophy, Principles & Lifecycle
4. Preparing for Success
5. Assessment 1 Discussions

Preparing for Success

Adhere to and embrace the Principles:

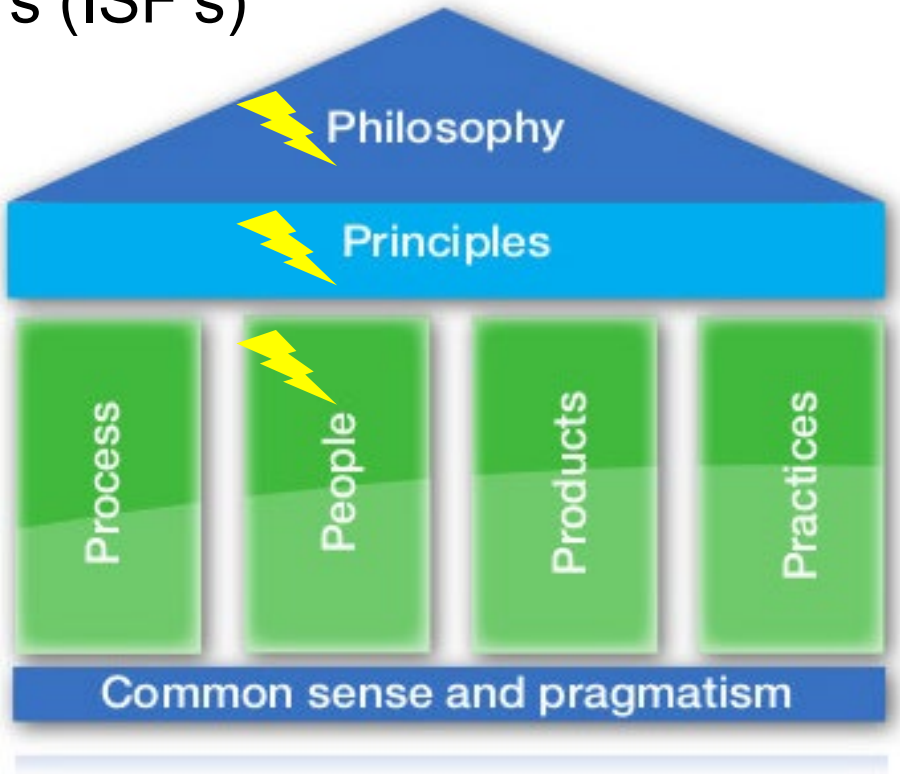
- Treat non-adherence to the principles as a risk
  - Breaking any of the principles will be a significant risk to the success of the Agile process and the success of the project
- Discuss the Principles openly with the project team at the start of the project and ensure everyone buys into them

The principles help direct and shape the attitude and mindset of a DSDM team. Compromising any of the principles undermines DSDM's philosophy, as together they deliver a collective value that outweighs their individual benefits.

Preparing for Success

Instrumental Success factors (ISF's)

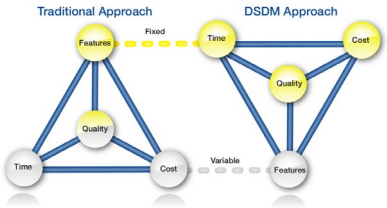
- Embrace the approach
  - Philosophy
  - Principles
- Development Team
  - Empowerment
  - Stability
  - Skills
  - Size
- Business Engagement
  - Committed
  - Involved
  - Supportive
- Iterative development, Testing and Delivery
- Transparency
- Risks

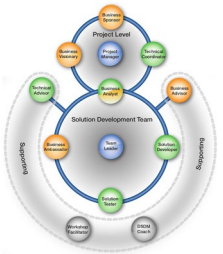


Preparing for Success

Understand Your Constraints:

- Consider the variables
  - Is there flexibility in depth and detail of features?
- Think about the people
  - Are all roles capable of, and committed to the project approach
- Consider the Principles
  - Will the organisation support this way of working?
- This is rarely a black and white (clear) decision
  - There is a tool to help decide





Preparing for Success

Utilise Controls such as Timeboxing and MoSCoW:

- **When dealing with problems ensure that**
  - Changing deadlines is not an option – Time is fixed
  - Adding resources is **not** an option – Resources/Cost fixed
  - Quality is not negotiable
  - Need to find another way to resolve problems
    - Drop a feature?
    - Revisit MoSCoW?

- **Avoid adding Musts after baseline has been agreed (changing breadth)**
  - Change of breadth requires formal change control

Preparing for Success - Self Directed Teams

Tightly Managed Teams (Traditional)	Self Directed Teams (Agile)
Take directions	Take initiative
Seek individual reward	Focus on team contributions
Focus on low-level objectives	Concentrate on solutions
Compete	Collaborate
Comply with processes, regardless of outcome	Continuously look for better ways of working
React to emergencies	Take steps to prevent emergencies

The best solutions emerge from self-organising, empowered teams.

Preparing for Success

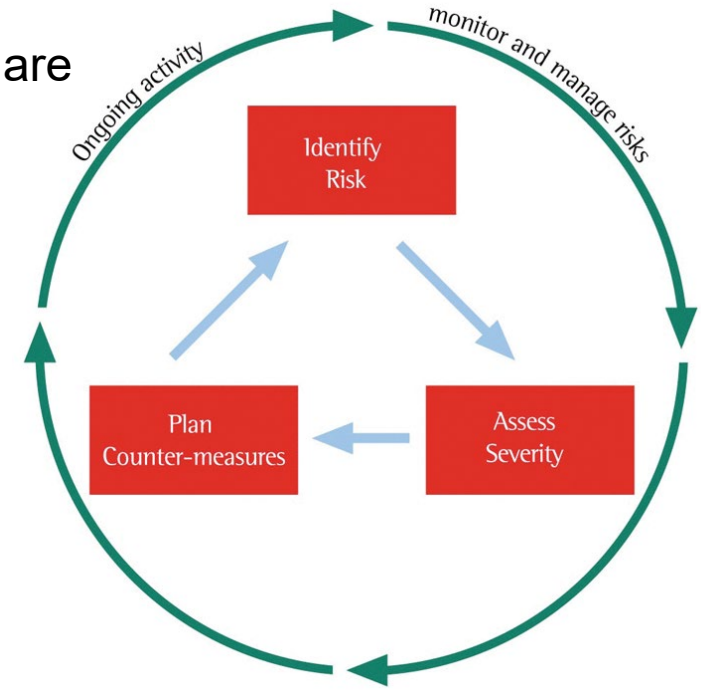
Practise Agile Style of Management (Self Organising Behaviour):

- Different style of management (compared to traditional)
  - Enabling constant change during elaboration of the detail
  - Continuously correcting course
  - Maintaining aim on target (delivering a usable solution on a fixed date)
- Monitoring progress in a different way
  - Measured by delivery of products (not by activity)
  - Sustaining the high rate of progress throughout
- Targeting and motivating empowered teams (**not** directing them)
  - Collaboration requires a no-blame culture
  - Building culture of team success/failure

Preparing for Success – know your risks

Risk management process is unchanged. Some risks are different

- **Typical traditional risks**
  - Missing deadlines
  - Assuming unknown or volatile requirements are clear and fixed
  - Delivery of wrong solution
  - User acceptance testing late in life cycle



- **Agile risks**
  - Not complying with Principles
  - Non-availability of Business roles
  - Having detailed specification up front
  - Expecting 100% solution
  - Swapping resources in and out

Agenda

1. Overview of Agile
2. Agile SW Development Manifesto
3. What is DSDM
   - Philosophy, Principles & Lifecycle
4. Preparing for Success
5. Roles & Team Performance

Assessment 1 - Team Performance

Is your team performing?

Assessment 1 Discussions

Review (Take Aways) for this week

1) **Agile Principles**
   - ❖ Individuals and interactions OVER Processes and tools
   - ❖ Working software OVER Comprehensive documentation
   - ❖ Customer collaboration OVER Contract negotiation
   - ❖ Responding to change OVER Following a plan

2) **Agile Management Style**
   - ❖ Take initiative, Focus on team contributions
   - ❖ Concentrate on solutions, Collaborate
   - ❖ Continuously look for better ways of working
   - ❖ Take steps to prevent emergencies

3) Team Performance
   - ❖ Focus on process (NOT people or product)