

# The DSDM Agile Project Framework for Scrum



Produced by:  
**Andrew Craddock, Keith Richards,  
Dorothy Tudor, Barbara Roberts, Julia Godwin**  
for the DSDM Consortium

## Contents

<b>Introduction .....</b>	<b>2</b>
Introducing the DSDM® Agile Project Framework.....	2
Introducing DSDM .....	2
Introducing Scrum .....	3
<b>The DSDM Agile Project Framework for Scrum .....</b>	<b>4</b>
Philosophy.....	4
Values.....	4
Principles .....	5
Variables .....	8
Lifecycle.....	10
Roles .....	12
Artefacts .....	17
<b>DSDM Agile Project Framework Techniques .....</b>	<b>19</b>
Introduction .....	19
Facilitated Workshops.....	19
Modelling .....	20
MoSCoW Prioritisation applied to the Product Backlog .....	20
DSDM Timeboxing applied to a Sprint.....	20
<b>In Summary .....</b>	<b>22</b>
<b>About the Authors.....</b>	<b>22</b>

## Introduction

### Introducing the DSDM® Agile Project Framework

The DSDM Agile Project Framework is an abstraction of DSDM® Atern® designed to complement other Agile approaches in an environment where a defined project approach is either expected or will add value. This pocket book describes a version of the Agile Project Framework tailored specifically to complement Scrum. It incorporates DSDM's project-focused principles, together with DSDM's rich set of roles and responsibilities that are ideally suited to a corporate project environment. It also offers the robust and fully Agile DSDM techniques for establishing and demonstrating control in a project. At the delivery level, the Agile Project Framework uses a simplified version of the DSDM process, in order to work alongside and complement the Scrum product delivery process without the need for a change of approach. The DSDM Agile Project framework brings together the strength of DSDM at project level and the streamlined simplicity of Scrum at the delivery team level.

One or more aspects of the DSDM Agile Project Framework may be used to supplement Scrum on a project-by-project basis or an organisation may choose to implement some or all of it as a standard. Regardless, aspects of the DSDM Agile Project Framework should be used where they make the use of Scrum easier, more effective or simply make it more accessible to project and organisational stakeholders.

### Introducing DSDM

#### DSDM Atern

Atern is the current version of the Dynamic Systems Development Method (DSDM). DSDM was initially created in 1994 through collaboration of a large number of project practitioners across many blue chip companies who were seeking to build quality into Rapid Application Development (RAD).

DSDM is a proven framework for agile project management and delivery, helping to deliver results quickly and effectively. It concentrates on strategic goals and incremental delivery of real business benefits while keeping control of time, cost, risk and quality.

Agility is enabled through the encouragement of self- directed, empowered teams working together in a supportive and collaborative manner. Since 1994, DSDM has been effectively applied to a wide range of projects from small software developments all the way up to full-scale business process change.

## DSDM agility

DSDM advocates that projects should do just 'enough design up front' within a Foundations phase in order to understand and clarify the structure of the overall solution and to create an agile plan for delivery of the project. This puts in place the foundations for successful development and delivery.

It is important to understand that the Foundations phase of a DSDM project is very different to the Analysis and Design steps in a traditional 'waterfall' approach. In a DSDM project, analysis and design activity carried in Foundations covers the full breadth of the project but deliberately avoids going into detail. Substituting traditional 'big design up front' with DSDM's 'enough design up front' promotes Agility in developing the required solution whilst avoiding the risk of 'no design up front' that makes many larger and more strongly governed organisations so nervous.

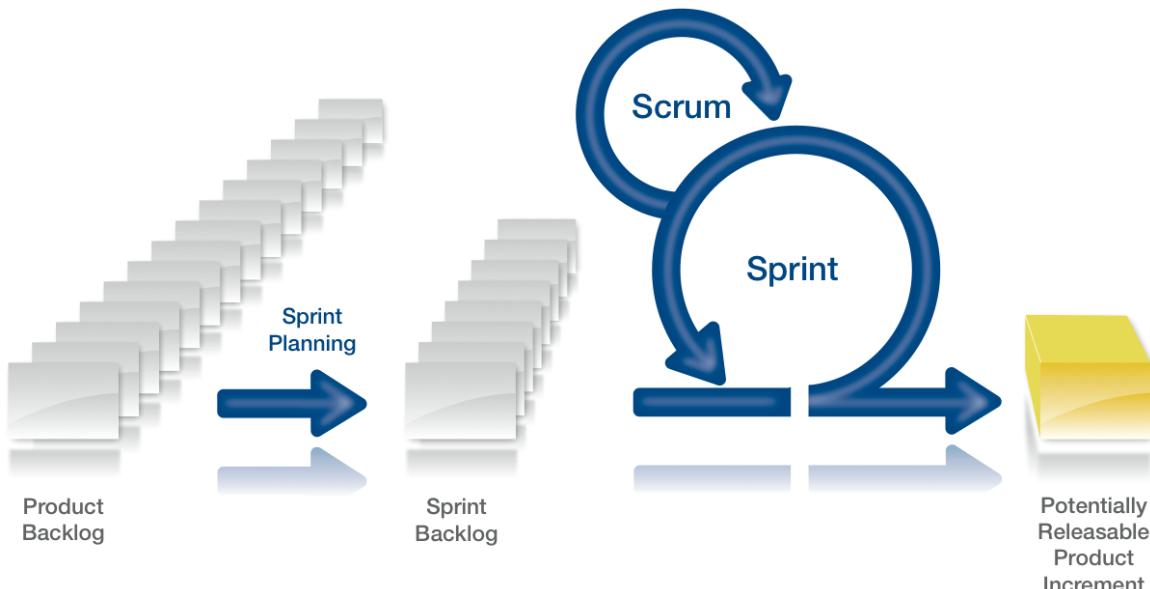
Where required, DSDM can be used to complement other project management disciplines such as PRINCE2® or PMI® without conflict or duplication of effort.

(DSDM® and Atern® are Registered Trade Marks of Dynamic Systems Development Method Limited)  
(PRINCE2® is a Registered Trade Mark of the Cabinet Office in the United Kingdom and other countries.)  
(PMI® is a registered trademark of the Project Management Institute)

## Introducing Scrum

Scrum takes an empirical, product-focussed approach to the delivery of products (predominantly software). From a process perspective it is both simple and elegant. Throughout this paper, the authors have tried to use terminology and definition of *what Scrum is* that is consistent with The Scrum Guide 2011 (July 2011, Jeff Sutherland and Ken Schwaber, [www.scrum.org](http://www.scrum.org)).

### The Scrum Process



- A Product Owner defines an ordered set of requirements for the product to be built. The Product Backlog is a dynamic artefact that evolves over time in line with the changing needs of the business for which the product is being built
- The Scrum Team (comprising the Product Owner, the Development Team and the ScrumMaster) use their knowledge, experience and understanding of the product and the requirements to agree a number of the most important items from the Product Backlog to be addressed in a Sprint
- In the process of Sprint Planning, the Scrum Team collaborate to work out how they are going to represent the items they selected from the Product Backlog in the Potentially Releasable Product Increment delivered at the end of the Sprint. The output of the Sprint Planning Meeting is the Sprint Backlog which defines at varying levels of granularity the work needed to deliver the Increment
- The duration of the Sprint is agreed at the outset and, every day, the team hold a 15 minute meeting called a Scrum. The primary purpose of the Scrum is to allow the Scrum Team to synchronise activities, identify impediments and agree a plan for the next 24 hours

- The Sprint ends with a Sprint Review Meeting where the Product Increment is demonstrated and a Retrospective in which the team reflect on the effectiveness of the way they worked
- A ScrumMaster is present throughout to help keep the Scrum Team focussed on their goal and to help them get the most out of the Scrum Events (Sprint Planning Meeting, Daily Scrum, Sprint Review and Retrospective)

Scrum has a product focus rather than a project focus so more emphasis is placed on incremental release of a product in the context of a product lifecycle than is placed on formally ending development work after an agreed period of time. Effectively, development ends when the value to be delivered by the next product increment can not be justified from a business perspective.

## The DSDM Agile Project Framework for Scrum

### Philosophy

The DSDM Agile Project Framework embraces the project delivery values at the heart of DSDM and fully aligns with the product development philosophy inherent in Scrum.

The DSDM Philosophy is that any project must be aligned to clearly defined strategic goals and focus upon early delivery of real benefits to the business.

This is best achieved when all stakeholders understand the business objectives, are empowered to an appropriate level, and collaborate in order to deliver the right solution. This solution will be delivered in the agreed timescale, according to the priorities driven by the business. As the project progresses, the stakeholders must accept that change is inevitable as the understanding of the solution deepens.

### Values

DSDM covers more than just software projects. However, like DSDM and Scrum, the Agile Project Framework fully adopts the values laid out in the Manifesto for Agile Software Development specifically valuing:

#### People and Interactions over Processes and Tools

In an Agile project, great emphasis is placed on the individual in the context of the project team. Every individual is expected to be ready, willing and able to play their part in the project, carrying out their role with competence and professionalism. Every member of the team is expected to work collaboratively with everybody else, using his or her knowledge, experience and judgement to shape a project outcome that best meets the need of the sponsoring business. Processes and Tools play an important part in any project but much less emphasis is placed on these in an Agile environment. Agile processes need to be light touch and serve to guide and support rather than dictate what individuals and teams should do and how they should do it. The assumption is that the team themselves are best placed to understand what needs to be done and to work out the best way of doing it. The Agile Project Framework provides appropriate light touch guidance, whilst keeping the emphasis at all times on the people and their interactions

#### Working Software over Comprehensive Documentation

The choice of words used here reflects the origins and primary focus of Agile – that it is for software delivery. However, changing a single word - changing “software” to “solution” - elevates this value from delivery of a software product into the broader context of business change projects. DSDM has been proven to work equally well for non-software projects.

The message behind this value is to break the illusion of security and stability that comes from document-driven processes. Specification of every detail of requirements, solution design, plans etc. in documents that get ‘signed off’ by stakeholders before work is allowed to progress is both wasteful and ineffective. The Agile Project Framework embraces the need for high-level versions of these artefacts in the early phases to frame development and delivery projects and to support governance. After the Foundations phase in a project, the framework employs collaborative techniques with active business engagement to ensure that the right solution is delivered. The framework also advocates light and timely documentation to support the solution in production, beyond the end of the project.

## **Customer Collaboration over Contract Negotiation**

This value encourages project teams and the sponsoring business to work collaboratively at all times. Typical commercial contracts assume that a traditional waterfall process underpins development and 'a fixed price for a fixed specification' is the standard for project contracts. Agile projects emphasise collaboration, and therefore contracts need to reflect this.

Looking more broadly at what constitutes a contract – effectively any document signed by those responsible for sponsoring the solution and those delivering it - it is important to ensure that, where created, all parties follow the principle of such documents being 'light touch' and 'guiding' rather than being 'detailed' and 'prescriptive'. By this definition, the Product Backlog may represent a contract, effectively defining the scope of a project. But it is cast at a high level and requires customer collaboration with less formality to flesh out the detail of requirements throughout the iterative development of the solution during the project lifecycle.

## **Responding to Change over Following a Plan**

This value emphasises the fact that the world around a project is rarely frozen in time. The pace of change in the world of business is now so fast that adopting an approach to building solutions that does not accommodate, or ideally embrace, change is unlikely to lead to a successful outcome. With the level of change that is a fact of life in the modern world, creating detailed, long-term plans becomes a waste of time. In this modern world, the high-level 'light touch' and 'guiding' plans as advocated by the Agile Project Framework better meet the need.

The DSDM Agile Project Framework specifically embraces the last sentence in the Manifesto that clearly states, in the context of the values above "That is, while there is value to the items on the right, we value the items on the left more." It is important not to shun Processes, Tools, Documentation, Contracts and Plans but instead to ensure that they are only created where they add value, and only to the level of detail that adds value. They should be created in a form relevant to and taking full advantage of the Agile philosophy.

## **Principles**

The eight DSDM principles underpin the Agile Project Framework and support the Philosophy. They bring the Values to life by guiding the attitude that must be taken and the mind-set that must be adopted in order to deliver consistently whilst still remaining flexible. Compromising any principle undermines the basic philosophy and introduces risk to the successful outcome of the project.

The eight Principles are:

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**

Every decision taken during a project should be viewed in the light of the overriding project goal, which is to deliver what the business needs, when it needs it, and always remembering that the project is simply the route to achieve the end goal – delivery of a working solution.

In order to fulfil this principle, project teams need to:

- Understand the true business priorities
- Establish a sound business case
- Seek continuous business sponsorship and commitment throughout the project

Specific business roles (described later), in conjunction with the work completed in the Foundations phase, and supported by time-boxed delivery and appropriate prioritisation of requirements, enables teams to fulfil this principle.

## **Principle 2 - Deliver on time**

Delivering products on time is a very desirable outcome for a project. It is quite often the single most important success factor.

Late delivery can undermine the very rationale for a project, especially where market opportunities or legal deadlines are involved.

In order to fulfil this principle, teams need to:

- Timebox the work into short delivery Sprints
- Focus on business priorities
- Always hit deadlines

Working in short focussed Sprints, typically of 2-4 weeks, and flexing the scope of what is delivered in that timeframe allows teams to implement this principle and build a reputation for timely and predictable deliveries. Incremental, timeboxed delivery of valuable solution features in short timeframes forms the basis for control over the longer-term delivery of the project.

## **Principle 3 – Collaborate**

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

Collaboration encourages increased understanding, greater speed and shared ownership, which enable teams to perform at a level that exceeds the sum of their parts.

In order to fulfil this principle, teams need to:

- Involve the right stakeholders, at the right time, throughout the project
- Ensure that team members are empowered to take decisions on behalf of those they represent
- Actively involve the business representatives, especially the Product Owner
- Build a one-team culture

The Product Owner role, complemented by the other business roles defined in the Agile Project Framework, brings the appropriate subject matter experts into the project so they can contribute to the solution. The ScrumMaster is responsible for facilitating a high level of collaboration between all Scrum Team members. The Business Analyst role also helps ensure collaboration between business and delivery roles is effective.

## **Principle 4 - Never compromise quality**

The level of quality of the solution to be delivered should be agreed at the start. All work should be aimed at achieving that level of quality. No more and no less.

A solution has to be 'good enough'. If the business agrees that the features in the Minimum Usable Subset of the solution have been provided adequately, then the quality of the overall solution should be acceptable. At a detailed level, if a story meets its acceptance criteria, then the story is good enough.

In Scrum, it is normal to agree a 'Definition of Done' which applies generically to stories, by default all stories are considered to be 'done' when these criteria and the criteria specific to the story are met in full.

In order to fulfil this principle, teams need to:

- Set the level of quality before development starts
- Ensure that quality does not become a variable
- Design, document and test appropriately
- Build in quality by constant review
- Test early and continuously

The work of the Foundations phase of the project, together with regular reviews throughout the project lifecycle, will help the team to build a solution of the appropriate quality.

In an Agile project everything is tested as early as possible. Test-driven techniques result in a test being written before the deliverable is actually produced. Ensuring that appropriate review and testing activity is carried out in each Sprint ensures that quality of the overall solution is not compromised

## **Principle 5 - Build incrementally from firm foundations**

In order to deliver real business benefit early, all Agile approaches advocate incremental delivery. This encourages stakeholder confidence and is a source of feedback for use in subsequent Releases.

Releases that are deployed into operational use may lead to early business benefit.

In establishing firm foundations, it is important to understand and agree the scope of the business problem to be solved and the scope of the proposed solution, whilst deferring the detail until later

In order to fulfil this principle, teams need to:

- Do enough design up front 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

Teams implement this principle using the high level process defined by the Agile Project Framework, to deliver 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**

In order to converge on an accurate business solution Agile approaches use iterative development. The concept of iteration is embedded throughout the Agile Project Framework - from the overall lifecycle of the project down to the lowest level of feedback cycles within a Sprint.

It is very rare that anything is built perfectly first time and projects operate within a changing world. The Agile Project Framework advocates a pragmatic approach to change that relies on iteration in order to embrace change and produce a better solution.

In order to fulfil this principle, teams need to:

- Be creative, experiment, learn, evolve
- Embrace change - the solution will evolve as the team learns more about it
- Take an iterative approach to building all products
- Continually confirm the correct solution is being built
- Converge on an accurate solution

Change is inevitable. The Agile Project Framework allows for change and harnesses its benefits.

Within the constraints of fixed time, cost and quality, change is actively encouraged in order to evolve the most appropriate solution. Iteration and constant review ensure that what is finally delivered is what the business really needs.

## **Principle 7 - Communicate continuously and clearly**

Poor communication is often cited as the biggest single cause of project failure. Agile techniques are specifically designed to improve the effectiveness of communication both for teams and between individuals.

In order to fulfil this principle teams will:

- Hold Scrums (short stand-up team meetings) each working day
- Employ rich communication techniques, such as modelling and prototyping
- Present iterations of the evolving solution early and often (as a minimum, the end of each Sprint)
- Keep documentation lean and timely
- Manage stakeholder expectations throughout the project
- Encourage informal, face to face communication at all levels
- Use Facilitated Workshops wherever appropriate

Agile approaches emphasise the value of human interaction. The roles and responsibilities defined in the Agile Project Framework and techniques encourage team and stakeholder collaboration. Modelling and prototyping allow scrutiny of early impressions of the solution.

These techniques are far more effective ways of communicating than relying on large textual documents, which in themselves provide no financial value to the project or the business.

## Principle 8 - Demonstrate control

It is essential to be in control of a project at all times. The team and wider stakeholders need to be proactive when monitoring and controlling progress, ensuring the project stays aligned to the plans and Business Case agreed in the Foundations Phase.

In addition, in many corporate and/or governed environments, it is not enough simply to *be* in control. You need to be able to *prove* it.

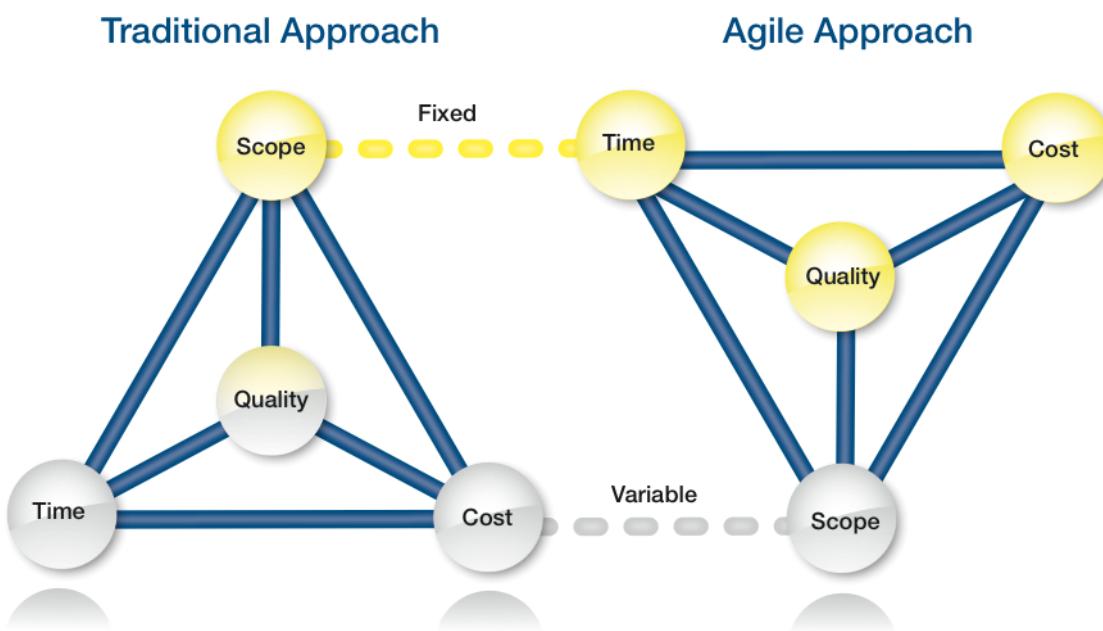
In order to fulfil this principle, the Scrum Team and the Project Manager will:

- Use an appropriate level of formality
- Be able to demonstrate that the project is in control at all times
- Manage proactively
- Make plans and progress visible to all
- Measure progress through focus on delivery of products (rather than completed activities)
- Evaluate continuing viability of the project, based on the business objectives

The use of well planned incremental delivery, which incorporates incremental review and acceptance, and based on sound but not detailed planning in the Foundations phase helps all project participants follow this principle.

## Variables

For most projects, Project Management involves balancing three key parameters – time, cost and quality. As with DSDM, the Agile Project Framework splits the traditional ‘quality’ parameter into two – technical quality and the scope of features to be delivered as part of the solution. Trying to fix all four parameters at the outset is impractical and is the cause of many common project problems.



In the traditional approach to project management (left hand diagram) the feature content of the solution is fixed whilst time and cost are subject to variation.

If the project goes off track, the delivery date is often extended or more resources are added. It is recognised that adding resources to a late project just makes it later. A missed deadline can sometimes be disastrous from a business perspective and could easily damage credibility. A missed deadline is bad enough, but the loss of predictability around delivery dates may be even worse.

Quality often becomes a casualty as projects struggle to deliver the fixed scope in a constrained timeframe, causing problems in addition to those related to late delivery and increased cost.

With proper planning, any three of the four project variables can be fixed provided one is allowed to vary. In an Agile project the greatest predictability and the most successful outcome for a project usually comes from fixing time, cost and quality and allowing the scope of the features delivered to vary.

By default, the Agile Project Framework follows DSDM; time, cost and quality are fixed as part of establishing firm Foundations for the project. To reach the point where this can be achieved, an understanding of the high level features is required, sufficient to provide a sensible estimate for those aspects of the project that are fixed. At the same time, it is normal for a subset of the features to be identified as mandatory. This is on the understanding that if the solution does not include these features then it will have either no value or that value will be so severely compromised the project would be considered an outright failure.

When working this way, the contingency within the project comes from the lower priority features, which are progressively re-scoped and, if necessary, de-scoped to protect the commitment to time, cost and quality. The most common options for prioritisation include Absolute Prioritisation where required features are ranked from 1 to n and Relative Prioritisation using a technique such as DSDM's MoSCoW Prioritisation.

Solid predictability of end-date for a delivery becomes a reality in projects that use the DSDM Agile Project Framework.

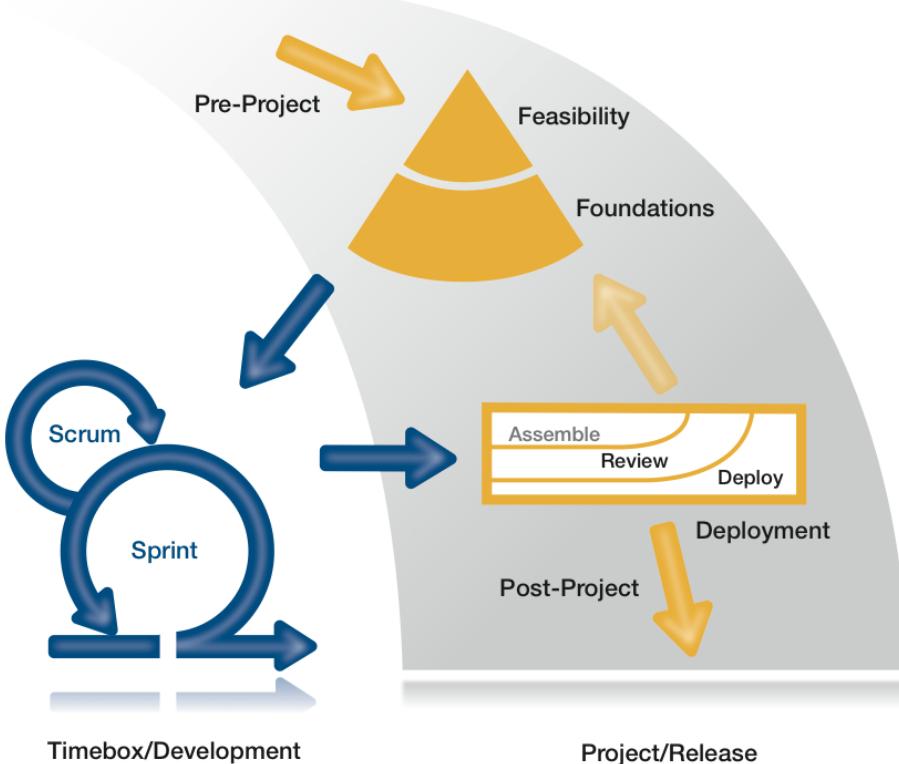
## Lifecycle

Every project has a lifecycle that starts with the identification of a potential need and ends at a point when that need is either met or has been rejected. Rejection can occur at any point if the project becomes non-viable.

Traditional project approaches follow a predefined lifecycle, typically executing a sequential pattern of Analyse, Design, Build, Test and Deploy. Larger projects following this approach may execute that pattern a number of times as a means of breaking the project into smaller, more manageable chunks, in order to deliver some of the business value early.

Although Scrum has the concept of a Product Roadmap to provide strategic direction and follows an incremental approach to achieving this, by default it does not capture the full spectrum of the project lifecycle that may be important to many projects and/or organisations where, for example a project encompasses more than just the development of software. Although Scrum can, and often does deal with products beyond software, the Agile Project Framework may make these aspects easier to manage in a project context. The DSDM Agile Project Framework takes account of all the work required from initial concept to final measurement of the business benefits realised post-project.

The Agile Project Framework has two elements; one focussed on *project management*, the other on *product delivery*. Scrum deals well with the product delivery and can be readily employed for that aspect of the overall framework. DSDM deals well with Agile project management. When considering Scrum in the context of the DSDM Agile Project Framework, it is important to acknowledge the overlap between the project and product delivery elements. The overlap is represented by Scrum's Product Roadmap and Product Backlog which need to reside within the project delivery element of the overall framework which is also responsible for any deployment activity required to get the Product Increment delivered from a Sprint into live use.



The phases in the Agile Project Framework are:

- Pre-Project
- Feasibility
- Foundations
- Evolutionary Development
- Deployment
- Post-Project

The following sections briefly describe these phases.

## Pre-Project

This phase formalises a proposal and prioritises it in the context of other work being carried out by the organisation in line with its strategic goals.

## Feasibility

Feasibility provides the first opportunity to decide whether a proposed project is likely to be viable, from both a business and a technical perspective. It involves very a high-level investigation of potential solutions, costs and timeframes.

Beyond Feasibility, project viability should be continually assessed, to ensure that the benefits predicted are still justified, based on the cumulative cost of the project to date and the estimated cost to deliver what remains.

## Foundations

Foundations establishes a firm and enduring foundation of understanding for the project. The three essential perspectives of business, solution and management are combined to provide a robust and flexible project focus.

The level of Foundation detail is strictly limited to avoid unnecessarily constraining the way in which a solution evolves. However, by the end of Foundations, it must be possible to demonstrate how the project delivery will meet the business needs by the end of the project.

## Evolutionary Development

Evolutionary Development takes an iterative and incremental approach to developing the solution as a whole. In the context of a project, the solution will include both the *Product* (often software) and any associated *changes* within the *business* wanting to exploit that product. Detailed requirements, normally expressed as *Stories* are investigated, further analysed and translated into a viable product engineered to full operational readiness.

Iterative Development techniques, either simultaneously or sequentially, address both functional and non-functional requirements including areas such as performance, capacity, security, supportability and maintainability.

Throughout, continued involvement of the Product Owner provides an on-going opportunity to guide development and to validate fitness for purpose of the product. The business roles from DSDM (the Business Sponsor, Business Visionary and Business Advisor(s)), will often be valuable in a complex corporate environment where it is not feasible for a single individual (the Product Owner) to understand all the detail of all aspects of requirements for a software product as well as understanding the full implications of the way the business may need to change in order to exploit it. The perspective and input from the other DSDM roles may well be valuable to the Product Owner and the rest of the Scrum Team who can rely on the Business Visionary to provide high level context and handle the business change aspects of the wider project and on Business Advisors to provide real depth and detail around specialist requirements such as compliance with specific legislation or working practice.

## Deployment

The focus of the Deployment phase is on getting the solution into operational use or ready for market. The number of passes through the Deployment phase will depend the number of Releases scheduled as part of the project. This is, in turn, driven by the business imperatives. Three aspects of deployment considered in this phase are final assembly (configuration/packaging) of the product, a final review of what has been delivered and the actual deployment of the product into operational use.

## Post-Project

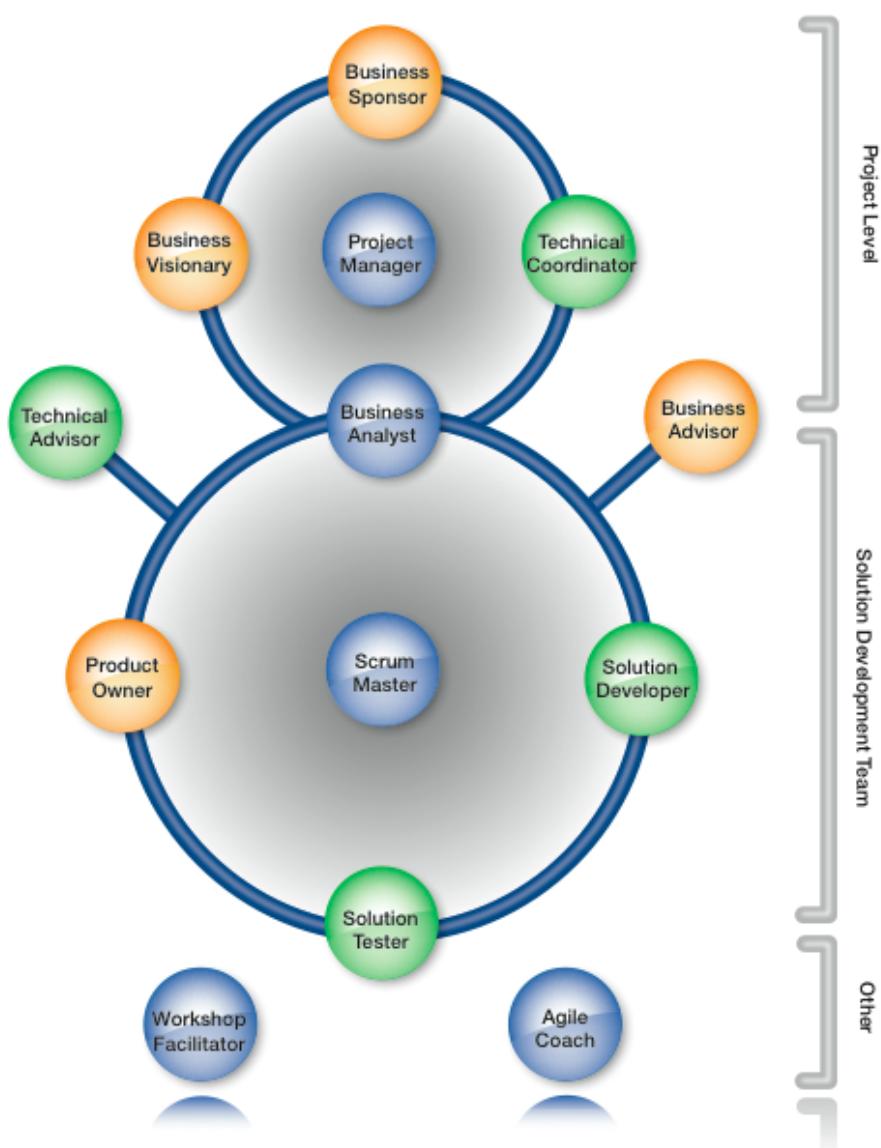
Post-Project takes place after the last planned Deployment. It assesses the project performance in terms of the business value actually achieved.

## Roles

Roles in the DSDM Agile Project Framework fit into three categories: Project Level, Solution Development Team and Other. In a Scrum context the Solution Development Team is the Scrum Team.

Each role is focused on one of: Business, Solution, or Management/Process aspects of the project. The focus is colour coded on the diagram below with orange for business, green for solution and blue for process. In the diagram below the two key Scrum roles of Product Owner and ScrumMaster have been introduced in the place of the standard DSDM Business Ambassador and Team Leader roles that are broadly equivalent

The DSDM roles of Solution Developer and Solution Tester collectively make up the deliberately generic *Development Team* defined in Scrum. In an ideal Agile environment the Development Team will be fully multi-skilled and the differentiation between development and testing roles may detract from the collaborative ‘whole team’ ownership of the work. But in many larger, more complex organisations and in projects dealing, for example, with the challenge of interfacing new applications with legacy systems created using old or just different technology the differentiation may well be valuable or even a necessity.



Project Level roles are responsible for strategy and business perspective for the project as a whole. They are collectively responsible for overall project direction, coordination and governance. For larger projects, the Project Level roles are expected to oversee a number of individual Scrum teams, ensuring that each team remains focussed on the overall business success criteria for the project.

Solution Development (Scrum) Team roles in the diagram above are responsible for the detailed work involved in defining, developing, testing and deploying the solution. The core of the team are engaged throughout the project whereas others, such as Business and Technical Advisors, may only be needed from time-to-time as the project progresses.

The “Other” roles provide independent guidance and support for the project, in areas such as Facilitation or the Agile process and Agile thinking. In a Scrum environment at least part of the Agile Coach role, and often the Workshop Facilitator will form part of the ScrumMaster’s responsibilities.

Roles do not necessarily equate to individuals, nor relate directly to job titles. One person may cover multiple roles, or a single role can be shared between several people. Roles organised as described in the Agile Project Framework (above) allow the project to fit more naturally into the broader organisational context. For example, it is unlikely that the person sponsoring the project (typically a senior manager / budget holder) would also be the person using the end product of the project on a day-to-day basis.

The Agile Project Framework roles and their associated responsibilities are described below.

### **Business Sponsor**

This is the most senior project level business role. The Business Sponsor acts as project champion and is committed to the proposed solution and the approach to delivery. This role owns the solution once delivered and is responsible for realising the associated benefits.

A Business Sponsor needs to have sufficient authority within the organisation so they can ‘force open closed doors’ and resolve major business issues. They make the financial decisions and are responsible for enabling progress throughout the project.

Normally, only one person holds this role. They are not typically engaged every day or even every week, but they should be available throughout a project’s duration, to provide a clear escalation route.

Responsibilities:

- Owns the Business Case
- Ensures on-going project viability
- Makes sure that funds and other resources are made available as needed
- Guarantees a fast and effective decision-making process
- Responds rapidly to escalated issues
- Takes ultimate responsibility resolving conflicts

### **Business Visionary**

This is a senior project level business role, which requires more active involvement than the Business Sponsor.

The Visionary is involved throughout the project to provide strategic direction, and to ensure that the solution delivered will provide the benefits as described in the Business Case.

The Visionary is responsible for interpreting the needs of the Business Sponsor and ensuring these needs are properly represented in the Business Case and properly communicated to the team or teams involved in developing the solution.

In a pure product context this role will probably be the Product Manager

Responsibilities:

- Owns the wider implications of any business change
- Defines the Business Vision
- Communicates and promotes the Business Vision to all interested parties
- Monitors progress in line with the Business Vision
- Contributes to vision-critical requirements, design and review sessions
- Approves changes to high level requirements in the Product Backlog
- Ensures collaboration across stakeholder business areas and availability of resources
- Promotes translation of the Business Vision into working practice

## **Project Manager**

The Project Manager is responsible for business, technical and delivery aspects of the project throughout its lifecycle and provides high-level management direction and co-ordination.

In line with the Agile concepts of empowerment, the Project Manager leaves detailed product delivery planning and control to the Scrum Teams. The Project Manager role is focussed far more on ensuring that the environment the team is working in is right than it is on the work the team are carrying out.

Responsibilities:

- Liaises with senior management and governance authorities
- Plans and schedules at the high level, but not detailed task planning
- Monitors progress against baselined high level plans
- Manages overall project configuration
- Motivates teams to meet their objectives
- Monitors business involvement within the Scrum Teams
- Resources specialist roles as required
- Manages risk and handles escalated problems
- Encourages Scrum Teams to handle difficult situations where sensible (rather than simply stepping in and taking ownership of issues that the team should be able to resolve for themselves)

## **Technical Coordinator**

The Technical Co-ordinator is the project's technical design authority and is responsible overall for the technical quality of the solution.

The Technical Co-ordinator ensures that Scrum Teams work in a consistent way within a project and deliver compatible output and also ensures that the project meets the desired technical quality standards appropriate to both the project and, where applicable, the wider product or organisation.

This role provides the glue that holds the project together, providing advice and guidance on technical decisions and innovation. A framework of empowerment ensures that this is a guiding role rather than a commanding one, with the Development Team empowered within defined boundaries to collaborate, innovate and manage for themselves how they go about building the solution.

The Technical Co-ordinator performs the same function, from the technical perspective, as the Business Visionary does from a business perspective.

Responsibilities:

- Agrees and controls the technical architecture
- Determines technical environments
- Advises on and co-ordinates cross-team technical activities
- Identifies and manages technical risk
- Makes sure that non-functional requirements are achievable and subsequently met
- Ensures adherence to standards of technical best practice
- Controls technical configuration of the solution
- Manages technical aspects of the solution's transition into live use

## **Business Analyst**

This role provides an additional perspective to the concept of the Scrum Team. The Business Analyst facilitates communication between the Solution Developers, Solution Testers and the business roles, in particular the Product Owner. Using analysis and facilitation skills and techniques, the Business Analyst ensures that the business needs are properly analysed and are correctly reflected in the approach the team adopts in order to generate the envisioned solution. The Business Analyst also typically assesses the potential impact of the evolving requirements on other areas or systems. It is important to understand that the Business Analyst does not act as an intermediary between the Product owner and the rest of the Scrum Team.

Responsibilities:

- Supports unambiguous and timely communication between Business and Technical participants
- Manages information related to business requirements
- Supports the Product Owner in thinking through the detail of stories in the Product Backlog
- Ensures that business implications of day-to-day decisions are properly thought through

Critically, the Business Analyst must not become a proxy customer who becomes or stands between the Product Owner or other business roles and the Development Team.

## **ScrumMaster (Team Leader)**

The ScrumMaster replaces the DSDM Team Leader role. The ScrumMaster ensures that the Scrum Team follows an iterative, collaborative approach to product development, helping to keep the team focused on meeting its objectives. The ScrumMaster typically facilitates the Scrum events and the planning and co-ordination of all aspects of product delivery at the detailed level. The Scrum Master acts as a 'servant leader' for the whole Scrum Team, working to remove impediments that hinder progress.

Responsibilities:

- Helps the team focus on on-time delivery of the agreed product increment within the Sprint
- Encourages full participation of team members in Scrum events and product development work
- Helps ensure that the iterative development process is properly focused and controlled
- Encourages discipline in scheduling and execution of required testing and review activities
- Manages risk and removal of impediments at Sprint level, escalating as required
- Facilitates learning within the team and encourages continuous improvement of the team and development processes

## **Product Owner (Business Ambassador)**

The Product Owner provides the business perspective for all decisions relating to the way the solution's fitness for business purpose is defined and implemented. In the Agile Project Framework the more 'high level' perspective that the Product Owner normally assumes is covered by the Business Visionary role.

The Product Owner generally comes from the business area that will be the beneficiary of the project and/or represents an end-user of the solution who is outside the organisation running the project. The Product Owner provides detailed business information on any matter from the perspective of those who will ultimately use the solution.

As a true ambassador, the role is responsible for the informal day-to-day communication channels between the team and the wider business users and beneficiaries of the solution.

Responsibilities:

- Ensures that the product delivered by the team has optimal business value:
  - Shapes the Product Backlog, in line with the Business Vision, to ensure the most valuable aspects of the solution are developed and delivered as soon as is practical
  - Contributes to requirements, design and review sessions
  - Provides a business perspective for day-to-day decisions
  - Describes business scenarios to help define and test the solution
  - Provides day to day assurance that the solution is evolving correctly
- Ensures that the transition to live use of the solution is safe and effective
  - Communicates with business users outside the project.
  - Organises business acceptance testing as required
  - Develops business user documentation as required
  - Ensures adequate user training is carried out

## **Business Advisor**

This role is additional to the typical Scrum Team roles.

The Business Advisor supports the Product Owner by providing specific and often specialist input to requirements detail, development and testing. They may be an intended user or beneficiary of the solution or they may provide guidance on areas of compliance.

The Business Advisor supports the Product Owner through the provision of detailed or specialist input and advice with regards to:

- Requirements, design and review sessions
- Business perspective for day-to-day decisions
- Business scenarios to help define and test the solution
- Assurance that the solution is evolving correctly
- Business acceptance testing
- Development of business user documentation
- User training

## **Development Team**

### *Solution Developer*

This role provides a development perspective within the context of a collaborative Scrum Team, recognising that in many corporate environments, a developer is a skilled specialist distinct from a tester. The Solution Developer interprets business requirements and translates them into a deployable solution that meets functional and non-functional needs.

Responsibilities:

- Works with Business roles and Solution Testers
- Undertakes iterative developments of the deployable solution
- Records and interprets the detail of changes to requirements and their consequences
- Adheres to technical constraints laid out in the Solution Foundations
- Participates in quality assurance to ensure products are fit for purpose
- Tests own output prior to independent testing

### *Solution Tester*

This role provides a testing perspective within the context of a collaborative Scrum Team, recognising that in many corporate environments, a tester is a skilled specialist distinct from a developer. The Solution Tester designs, builds and performs all types of testing during development and deployment of the solution, focusing predominantly on the technical aspects of testing.

Responsibilities:

- Works with Business roles to define test scenarios for the evolving solution
- Carries out technical testing
- Reports test results to the Technical Co-ordinator informed of depth / coverage of testing (for Quality Assurance purposes)
- Assists Product Owner and Business Advisor roles to carry out important business related testing

## **Technical Advisor**

This role is additional to the typical Scrum Team roles.

The Technical Advisor supports the Scrum Team by providing specific and/or specialist technical input to the project often from the perspective of those responsible for operational change management, operational support, on-going maintenance of the solution etc.

From their specialist technical perspective the Technical advisor may help support:

- Requirements, design and review sessions
- The operational perspective for day-to-day decisions
- Identification of operational or support scenarios to help define and test the solution
- Assurance that the solution is evolving correctly
- Operational acceptance testing
- Development of support documentation
- Training of operations and support staff

## **Workshop Facilitator**

This role manages the workshop process, ensuring the necessary workshop preparation and communication happens. A Workshop Facilitator is responsible for workshop context, but not for the workshop content. The Workshop Facilitator should have no stake in the workshop outcome.

Responsibilities - for each workshop:

- Agrees scope with the workshop owner and plans the workshop
- Becomes familiar with the subject area
- Engages with participants to understand any major areas of interest or concern
- Helps the workshop meet its objectives

## Agile Coach (part of the role of ScrumMaster)

Where appropriate, i.e. where a team has limited experience of using the Agile Project Framework, the Agile Coach helps the team to grow their understanding of Agile in the real world, and so provide the greatest benefit for the organisation.

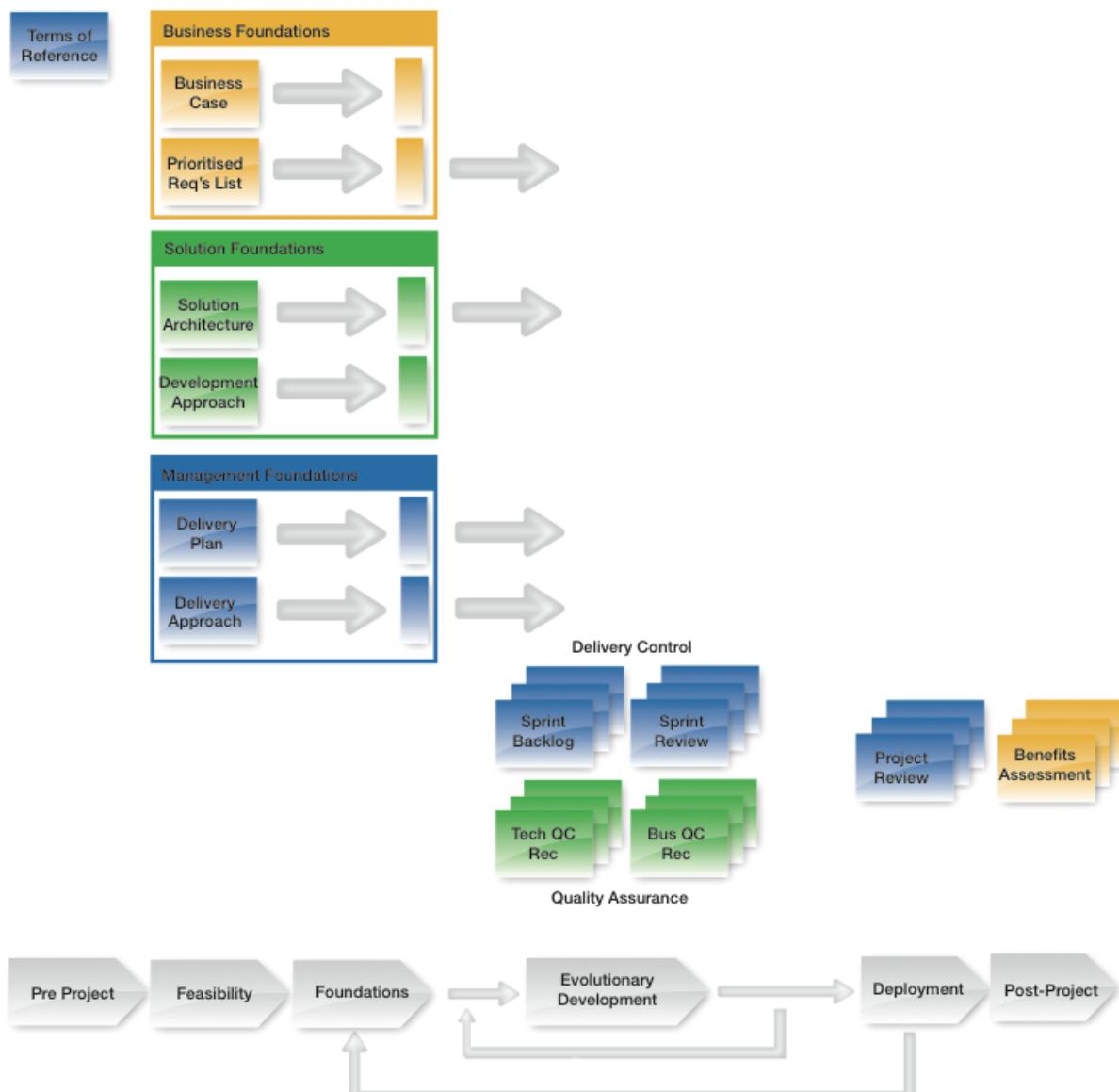
If something in the project environment runs contrary to Agile philosophy, values or principles or where an Agile technique is blocked, then the Agile Coach will help by coaching individuals, influencing the environment or, where appropriate, adapting the techniques being used.

### Responsibilities:

- Provides detailed knowledge and experience to team members on the Agile Project Framework, Scrum and the Agile practices being used within them
- Tailors the Agile Project Framework and the Scrum process to fit specific project constraints
- Helps the team and others to make full use of Agile techniques to realise their value
- Encourages the collaborative and co-operative working essential to success of an Agile project

## Artifacts

The Agile Project Framework defines a total of 8 fundamental work products in addition to the solution being created. The use and formality of these artefacts may vary from project-to-project but all 8 should at least be actively considered for every project. Some projects in some organisations may need to supplement this set with a variety of other artefacts as required.



## **Terms of Reference**

Defines at a very high level the objectives and business drivers for the proposed project.

## **Business Foundations**

The Business Foundations are created in outline during Feasibility, and refined to provide firm and enduring foundations for the project as a whole by the end of the Foundations phase without becoming too detailed and restrictive. The Business Foundations normally comprises a business case and the Product Backlog

### ***Business Case***

Describes essential business considerations that justify the project, and then are used to assess the viability of the project moving forwards.

### ***Product Backlog (Prioritised Requirements List)***

The Product Backlog equates to the DSDM Prioritised Requirements List. It provides a set of high-level stories that the project needs to address, indicating their priority with regards to meeting the overall project objective.

## **Solution Foundations**

The Solution Foundations are created in outline during Feasibility and refined to provide firm and enduring foundations for the project as a whole by the end of the Foundations phase without becoming too detailed and restrictive. The Solution Foundations normally comprises a definition of the solution architecture and the approach to developing the solution.

### ***Solution Architecture Definition***

Provides an overview and architectural framework for both business and technical aspects of the potential solution. This will evolve as the project proceeds.

### ***Development Approach Definition***

Defines the standards and practices to be adhered to and provides guidance on how the solution should be evolved as the project proceeds.

## **Management Foundations**

The Management Foundations are created in outline during Feasibility and refined to provide firm and enduring foundations for the project as a whole by the end of the Foundations phase without becoming too detailed and restrictive. The Management Foundations normally comprises a high-level delivery/release plan and a definition of the approach to managing the project.

### ***Delivery/Release Plan***

Provides an initial high-level schedule of Sprints, Releases and other activities for development, testing and deployment of the solution. For larger projects a single high-level Delivery Plan will deal with coordination of the efforts of multiple Scrum teams. This plan is constantly reviewed and revised as the project progresses to reflect the latest business demands and predicted outcomes in terms of timescales and delivered scope

### ***Delivery Approach Definition***

Describes the approach to the set-up and management of various aspects of the project, including how the project will be organised and governed. It also describes the approach to managing Change, Configuration, Communication and Risk.

## **Delivery Control**

The artefacts created to define and control the work of a Sprint include the Sprint Backlog and the Sprint Review record.

### ***Sprint Backlog***

The Sprint Backlog defines the work the Development Team will perform to turn Product Backlog items into a "Done" Increment. It is a forecast by the Development Team about what functionality will be in the next Increment and the work needed to deliver that functionality. Like the Delivery Plan, the Sprint Backlog is kept under constant review and revised as the Sprint progresses.

### *Sprint Review record*

Where useful, the Sprint Review record captures what has been achieved and any feedback, which will influence plans moving forward. Any outstanding issues are considered in the context of the Delivery/Release Plan and future Sprint plans. If review points are built into the Sprint, for example as part of the DSDM Timeboxing technique, the review record should be used to capture feedback at each review point and will therefore evolve as the Sprint progresses. This may be particularly useful in more regulated environments where it is important to demonstrate compliance with specific legislation or process controls.

### **Quality Assurance**

Scrum advocates that a Definition of Done be provided to help guide the Scrum Team to a successful realisation of a Story in the Product Increment. The quality review records described below may provide a useful record of the reviews undertaken to ensure that this is the case

#### *Technical Quality Review Records*

Where useful, this artefact provides a record of essential technical review and testing activities and any traceability to resolution required for issues and defects. These need to be to the depth and detail appropriate to the individual project, organization and legislative environment in which the project exists.

#### *Business Quality Review Records*

Holds any records related to business review and acceptance of the solution as it evolves, as well as formal acceptance of its fitness for purpose and readiness to be deployed from a business perspective

### **Project Review**

An evolving artefact updated at the end of every Release to reflect project performance and learning.

### **Benefits Assessment**

Describes how the benefits predicted in the Business Case have actually accrued as the Deployed Solution has been used.

## **DSDM Agile Project Framework Techniques**

### **Introduction**

The following DSDM practices are proven techniques that may be used by the project team to help fit practically and predictably into a governed, corporate project, programme or portfolio management environment without compromising project Agility. This allows Agile to provide the appropriate level of scalability and rigour, whether the project is small and simple or large and complex.

Facilitated Workshops and Modelling represent techniques that may be usefully employed in any project, Agile or otherwise. Both these techniques promote and support communication and collaborative working.

MoSCoW Prioritisation offers a project focussed ‘wrapper’ for the simple ordering of a Product Backlog. Timeboxing offers a structural overlay to a Sprint that may help improve predictability of delivery and also provides a structure that may help where Product Owner availability is limited

### **Facilitated Workshops**

Facilitated Workshops encourage collaborative working and enable high quality team-based decisions to be made in a shortened timeframe. People brought together as a group communicate more effectively and generate more creative solutions.

A well-run Facilitated Workshop also delivers an outcome with a high degree of buy-in and ownership from those people who have taken part.

Ideally, workshops are independently facilitated by someone external to the project. At the very least, the Workshop Facilitator should be independent of the workshop result, to ensure that all ideas and

contributions are given equal weight. A trained facilitator creates an environment that allows full participation.

Effective workshops follow a well-defined and carefully thought-out process. This process should include defining the objective, identifying appropriate participants, creating an agenda, managing the logistics and distributing any pre-reading to participants.

Facilitated Workshops are particularly valuable when applied to activities such as requirements identification and refinement, prioritisation, Release or Sprint planning, risk analysis, problem solving, product reviews and retrospectives.

## Modelling

Modelling is a technique for collaboratively evolving diagrams and pictures that define the problem or intended solution. They are used in Agile projects to improve communication through visualisation.

How often models are used and their formality depends on the nature of the project and the team's level of skill and experience in modelling techniques.

Models will also vary depending on the type of project, prevailing standards and best practice. In deciding what models should be created and when, the simplest rules to follow are:

- Be able to justify the value of the model for enhancing understanding of the given subject
- Use an approach that works for you and your organisation
- Do enough and no more so that the purpose of the model is achieved.

## MoSCoW Prioritisation applied to the Product Backlog

Delivering on a guaranteed date to a fixed cost means that some work originally planned for a delivery may have to be deferred. It may also be necessary to include work not originally identified.

Essential work must be completed and it is only less critical work that may be omitted from a delivery.

MoSCoW Prioritisation is a straightforward technique that can be used to achieve this, using these MoSCoW rules:

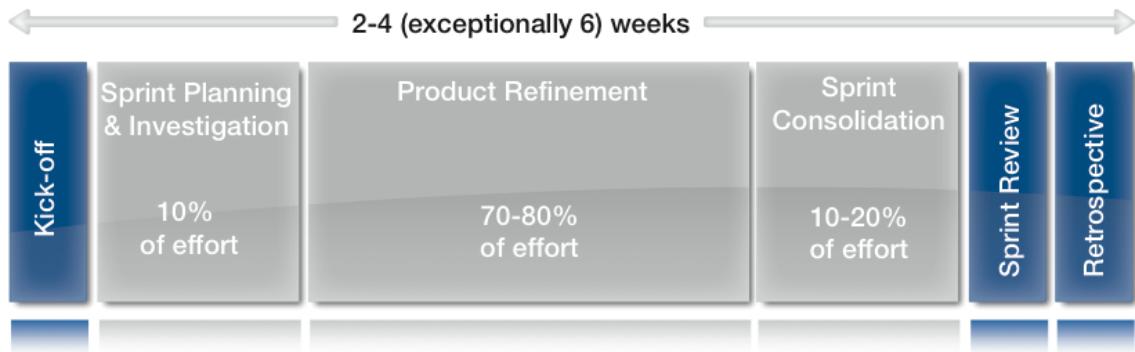
- Must Have: requirements that are fundamental to the solution. Without these the solution will be unworkable and useless. Must Haves define the Minimum Usable SubseT which an Agile Project Framework project guarantees to deliver
- Should Have: important requirements for which there is a short-term workaround. Normally classed as mandatory when more time is available, but without them the business objective will still be met
- Could Have: for requirements that can more easily be left out
- Won't Have this time: for requirements that can be included in later development. Won't Haves are excluded from plans for the current delivery

MoSCoW Prioritisation can provide the basis for decision making about project team activity at all levels. The advantage of MoSCoW is that it allows business expectations to be set at the Project level, in the knowledge that the team will definitely deliver the Must Haves, they are likely to deliver most or all of the Should Haves, and they may deliver some Could Haves.

## DSDM Timeboxing applied to a Sprint

The following is a representation of the DSDM Timebox applied to a Scrum Sprint. Applying some or all aspects of the DSDM Timebox may help the team better control their Sprint and help them translate a delivery aspiration for a Sprint into a commitment and to deliver what they promised when they promised it i.e. a potentially releasable increment of the product to the agreed level of quality and meeting the most valuable of the requirements identified for it.

All of the activity described in the steps below is required for any Sprint. The steps provide some structure that may help the team control the work they do. It is important for the team to consider the challenges they face and to decide which, if any, of these distinct steps need to be put in place and how much effort should be associated with each.



### ***Kick-Off***

In the Kick-off session, the Team agrees what will be taken from the Product Backlog for this Sprint.

### ***Sprint Planning and Investigation***

This step provides a firm foundation for work to be carried out during the Product Refinement that follows. At this time, the Scrum Team achieves an understanding of the detail of the requirements to be met and how they will be addressed as part of the Evolving Solution. The Sprint Backlog of detailed work to be completed is baselined at the end of the this part of the Sprint and the team commits to deliver one or more products by the end of the Sprint, based on their detailed investigation and their estimates of the work required to do this.

Where requirements are already well understood – perhaps because they have been explored as part of the work of the previous Sprint – the effort will be in line with the standard equating to 2 hours per week of Sprint. Where little or no understanding of the requirement exists, the team may spend double that (10%) or perhaps even more of the duration of the Sprint in this step.

At the end of Investigation, all Scrum Team members and, where it will be useful, other stakeholders should attend a review. The purpose of the review is to ensure that the Team and any stakeholders attending the review have a shared understanding of what the team plan to do to address the requirements for the Sprint

### ***Product Refinement***

This is where most work is carried out in the Sprint and development work and associated testing should be virtually finished by the end of this period.

At the end of Product Refinement there is a major review to look at which deliverables have been created and see what amendments will be needed to satisfy the acceptance criteria. The Product Owner and anybody else involved in product acceptance should attend in order to provide their feedback and help prioritise the work to be completed in Sprint Consolidation.

When Product Refinement is complete, no new work should be started and any remaining work to complete in-progress deliverables should be carefully prioritised for Sprint Consolidation.

### ***Sprint Consolidation***

Actions agreed at the end of the Refinement review are carried out together with any final work required to satisfy organisational or project standards. Final testing is completed and any product failing to pass its tests is not considered to be delivered. The effort associated with Sprint Consolidation will depend on how well honed technical and quality practices are within the team and in the complexity of the overall solution configuration. E.g. when delivering a straight forward ecommerce application using modern technologies such as .Net, with good technical disciplines within the team Consolidation may just be an hour or two. Where the team is challenged with legacy integration issues or where technology, skill or environment doesn't readily support Test Driven Development and Continuous Integration up to 20% of the Sprint duration may be needed if a potentially releasable product increment is to be delivered.

### ***Sprint Review***

The Sprint review often takes the form of a demonstration of the product increment delivered by the Scrum Team. It looks formally at what the Team delivered in the Sprint and either acknowledges that the increment of the product delivered is fit for purpose or highlights where it

is not. Any requirements that were intended to be addressed but were for some reason de-scoped are also noted.

Anything planned for delivery but not actually delivered may be considered for inclusion in a later Sprint at the next planning session.

#### *Sprint Retrospective*

The Sprint Retrospective provides the opportunity for the Team to explore the effectiveness of the way they worked in the Sprint. Particular attention is paid to:

- Any issues with quality or anything else that lead to the solution or aspects of it deemed not fit for purpose.
- Understanding the reasons behind why any requirements were de-scoped

Where appropriate, a summary of what was delivered and formal acceptance of it, along with any notable shortfalls or deficiencies may be captured in a Sprint Review Record.

## In Summary

The DSDM Agile Project Framework for Scrum offers teams and organisations a range of options to help with some of the challenges of scaling and governance for Scrum projects. The over-arching process framework, roles, products and techniques have all evolved over time as part of DSDM and have been repeatedly proven in corporate environments. Far from being heavy and prescriptive, the framework as it applies to Scrum is intended to provide a 'toolkit' for teams aimed at making their lives simpler and easier. This is achieved by offering a fully Agile structure to support larger projects and projects in organisations accustomed to a more formal approach to project management and governance.

## About the Authors

Andrew Craddock is currently the Technical Director of the DSDM Consortium and is the lead author of the DSDM Agile Project Framework for Scrum. Co-authors Keith Richards, Dorothy Tudor, Barbara Roberts and Julia Godwin are all members of the Technical Team at the DSDM Consortium responsible for the shaping and development of the DSDM family of methods.

All the authors are actively engaged in consultancy, training and coaching of Agile methods including DSDM and Scrum. Collectively they have decades of experience with the application of Agile in both public and private sectors and have worked with organisations ranging from software houses employing just a handful of people to some of the biggest and most successful multi-national companies in the world.

Andrew Craddock <a href="mailto:andrew.craddock@nlightentraining.com">andrew.craddock@nlightentraining.com</a>	Barbara Roberts <a href="mailto:barbara@roberts66.co.uk">barbara@roberts66.co.uk</a>
Keith Richards <a href="mailto:k.richards@agilekrc.com">k.richards@agilekrc.com</a>	Julia Godwin <a href="mailto:jdg@be-consulting.co.uk">jdg@be-consulting.co.uk</a>
Dorothy Tudor <a href="mailto:dottudor@tcc-net.com">dottudor@tcc-net.com</a>	

Thanks also for feedback and review from members of the Scrum Community:

Jose Casal-Gimenez ([jose.casal@eveningcoders.com](mailto:jose.casal@eveningcoders.com)), Nick Clare ([nick.clare@ntlworld.com](mailto:nick.clare@ntlworld.com)),

Mischa Ramseyer ([mischa@pragmatic-solutions.ch](mailto:mischa@pragmatic-solutions.ch))