IAPM

WHITE PAPER

HYBRID PROJECT MANAGEMENT

iapm

INTERNATIONAL ASSOCIATION OF
PROJECT MANAGERS

IAPM

INTERNATIONAL ASSOCIATION OF PROJECT MANAGERS

In 1997 the IAPM was still a fledgling association. It started out as a loosely structured international network for project managers who shared the objectives of promoting and modernising project management and providing young project managers with the tools to work effectively and successfully. Since this time, the IAPM has held annual International Project Manager Meetings (IPMM). Back in 1998 the IAPM published the precursor to the PM Guide 2.0, the IAPM By-laws of Project Management. These by-laws were completely revised and adapted to modern requirements and real-life project management scenarios in the PM Guide 2.0, which was published in 2010. In the same year, the IAPM was completely relaunched. The Scrum Guide 1.0, the current Agile PM Guide 2.0's predecessor, was published in March 2011. In 2013 the International PM Guide 2.0 was published and in 2017 the Hybrid PM Guide 2.0

The IAPM launched two awards in 2012, the Project Manager of the Year award and the Book of the Year award.

The Project Manager of the Year award is very special to the IAPM because it is presented to people for outstanding achievements in project management. Award recipients may have been instrumental to the resolution of a crisis situation, they may have successfully implemented a complex development project or they may simply have performed exceptionally in various areas of project management for many years.

The Book of the Year award honours books on the subject of project management that are published in both German and English. These books may communicate experience and knowledge in an innovative way, be (auto)biographical works or textbooks providing an introduction to the subject of project management.

The IAPM is an independent certification body which examines the knowledge and competence of the certification candidates with a comprehensive, fair and neutral online examination system. The certification system is therefore tailored to the challenging world of project management in the 21st century.

INHALT

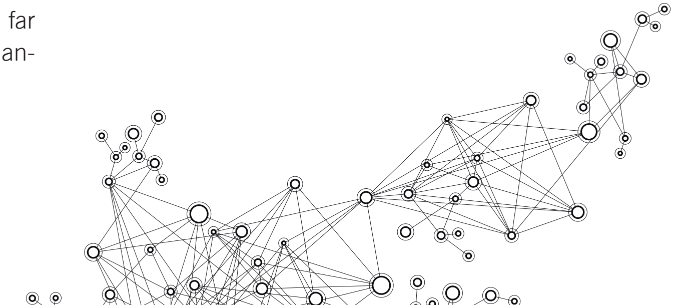
HYBRID PROJECT MANAGEMENT

hy·b·ri̱d: something combined, mixed, or created by crossing different elements

The subject of Hybrid project management is examined in the *Hybrid PM Guide 2.0*, which explains its origins, types, potential and limitations.

1. AN INTRODUCTION TO HYBRID PROJECT MANAGEMENT

Generally speaking *Hybrid project management* is understood as a way of using Agile approaches in a classic project environment that is managed in phases. However, this only describes one sub-area among a number of different Hybrid varieties. A number of combinations are derived from three types of management approaches; Classic project management, Lean Management and Agile project management. All these approaches have their own charms and also their own limitations. A given combination can be used by companies in a simple way to create an appropriate and company-specific project management culture. Even in the manufacturing sector, which has so far been the domain of Classic project management, Agile manufacturing concepts are now proving to be a workable and complementary alternative. The motivation to use Hybrid project management or to take its use into consideration is often due to a desire to acquire more efficient and more effective tools, but it also often occurs in companies that are organised on traditional lines, which are not able to get used to the idea of transforming into an Agile organisation, or which prefer to follow a controlled and phased process. First and foremost, the different project cultures and their characteristics will be compared so that the possible combinations can be then examined.



2. THE THREE MOST IMPORTANT PROJECT MANAGEMENT APPROACHES

2.1 CLASSIC PROJECT MANAGEMENT (WATERFALL MODEL)

In Classic project management, the whole project's life cycle is broken down into sequential phases. The completion of every phase constitutes a milestone, which involves carrying out a review to check, and if possible approve, the completion of specified tasks within each phase that have been implemented, ensuring that they were delivered on time, on budget and in keeping with planned performance. The next phase can only begin once the previous phase has been successfully approved. Feedback between successive phases is only possible in a limited way and potentially results in higher costs. As the project timeline lengthens, potential change-related costs often rise exponentially as a consequence of the sequential approach. To address this problem, a greater amount of effort is expended during the planning stage of the project to ensure that the planning is as comprehensive and accurate as possible. The fact that it can be necessary to freeze the agreed requirements at the beginning of the problem can often prove to be problematical. In the case of long project timelines, there is a risk that although the scope of the project correlates with the requirements profile at the beginning of the project, it actually becomes obsolete by the time the project is completed. Even though remedial action results from the comprehensive set of agreements with their associated change concepts, this will, however, not avoid cost and schedule variances, if such cases arise.

2.1.1 POSSIBLE PROJECT PHASES OF A DEVELOPMENT PROJECT

Requirements Phase

- Develop a business case
- Carry out a stakeholder and project environment analysis
- Create a requirements analysis and requirements prioritisation
- Develop and handover a comprehensive requirement specification document, including all requirements submitted by the customer

Design Phase (Planning)

- Define the type of project
- Agree the internal project organisation
- Define the communication structure
- Allocate roles
- Define the project's objectives and prioritisation
- Create a performance specification document with the contractor
- Approve the implementation concept
- Develop the specific project phases
- Set milestones, including the acceptance criteria for the phase transitions
- Develop a project structure plan, including critical paths
- Risk analysis
- Measures to avoid or detect errors
- Define change management
- Specify development tools, technologies and working environments
- Specify the documentation strategy

Development Phase

- Potentially in several implementation stages (prototype, end product), including a downstream test phase in each case
- Implement all requirements listed in the performance specification document
- Review the project objectives and compliance with the specifications
- Proactive stakeholder management
- Requirements are frozen – changes must be substantiated and documented
- They require customer approval
- Create the required documentation

Test Phase and Finalisation

- Design test scenarios
- End product testing
- Detection, evaluation and management of errors
- Create a product presentation
- Create user handbooks, installation instructions etc.
- Internal project review

Handover/Deployment/Operation

- Handover the product to the customer
- Product presentation
- Joint review
- Define maintenance strategy

2.1.2 PROJECT ORGANISATION AND PROJECT ROLES

Four roles can be identified in classic project management, which have different levels of authority and responsibility depending on the structure of the organisation.

1. Project manager
2. Project team
3. Steering committee
4. Stakeholders – all those who are affected by the project or the project outcomes to a greater or lesser extent

Classic project management uses three different types of organisation structures:

1. **Line management organisational structure:** In this case the project manager only has a very limited freedom to act and no managerial authority. This position remains unchanged. His role is therefore limited to informal communication and he is consequently also unable to assume any responsibility.
2. **Matrix organisational structure:** In the matrix organisational structure managerial authority overlaps the lines of authority assumed by the project lead. This means that project participants always have two supervisors with the same status: the project manager and the technical manager.
3. **Autonomous organisational structure:** A completely autonomous project organisational structure that operates independently from the rest of the organisation. The project manager leads the project team that reports directly to him, implements the project and is responsible for reporting the project results to the steering committee.

2.1.3 SUMMARY OF THE ADVANTAGES AND DISADVANTAGES OF CLASSIC PROJECT MANAGEMENT

Advantages

- Greater reliability with planning in terms of costs, deadlines and deliverables
- Clear demarcation of project phases
- Easy to process control scenarios
- Classic project management offers an effective operational framework with its solid and clear requirements and familiar implementation methods.
- Clearly defined responsibilities
- Simple scaling

Disadvantages

- Rigid project stage boundaries are often unrealistic
- Planning is difficult with unfamiliar technical implementations
- Inflexibility in the event of changes
- At the beginning of the project there is often a degree of uncertainty about costs and at the end of the project costs spiral as a result of changes
- The customer is forced to submit comprehensive requirements at an early stage without being able to align project outcomes or ideas.
- As responsible parties, the project management team and the steering committee are expected to be on par with their team in technical matters or they will make important decisions without the appropriate background knowledge.

2.2 LEAN MANAGEMENT

2.2.1 THE PRINCIPLES OF LEAN MANAGEMENT

The primary objective of *Lean Management* was originally to optimise the manufacturing process of physical products on all levels. Since then the Lean mindset has continued to develop and it is now being used in every sector and for every product. Unlike Agile and Classic project management, Lean Management is, however, not a management system that has been created for project work, rather it is a collection of methods and strategies, which can be applied to project management.

The most important ones are:

- just-in-time, the right thing at the right time, in the right quality and quantity, at the right place
- Kanban (labels) to ensure just-in-time and implementation of the pull principle
- Avoidance of Mura (unevenness), Muri (overburden) and Muda (waste)
- Kaizen as an evolutionary, continuous improvement process in small steps
- Kaikaku, a revolutionary, radical improvement process that facilitates cultural change within companies
- Jidoka, autonomous automation for error detection
- Heijunka, reduction of production batches to gain production levelling and flexibility
- The 5 W questions – posing the question why a number of times (not necessarily 5 times) in order to appropriately investigate why something has happened
- Andon, the generation of simple, visual status controls (traffic light colours)
- Genchi genbutsu - “going and having a look” as an effective aid for defining problems

The main objective can be summarised in the following statement:

“Generating the greatest possible customer benefits through an optimum value chain, which avoids any unnecessary costs or effort.”

Lean principles are generated from these measures, which soon point the way towards Agile project management and are applicable in the project environment:

- Focussing on people (customer, colleague)
- Avoiding waste
- Transferring knowledge
- Reflecting on your own behaviour
- Transparency at work
- Continuous learning
- Agreeing joint goals
- Striving for the best possible quality
- Restricting work
- Keeping work flowing

Despite their many similarities, it is important not to mix up Lean and Agile because there are in fact a few, but significant, differences in their fundamental focus.

1. Lean aims for standardisation; agile aims for individualisation/flexibility
2. Lean focuses on the production process; agile focuses on the product
3. Lean aims to avoid errors and thus waste; agile regards errors as a part of the learning process

2.2.2 LEAN METHODS FOR PROJECT MANAGEMENT

2.2.2.1 KANBAN

Kanban is a Japanese term meaning “card” or “record”. It is a control system based on the “pull principle”, which aims to improve manufacturing productivity. Kanban is mentioned here because is forms the basis of IT Kanban, which is used in project management today. However, there are considerable differences, which means that IT Kanban is described in the following sections on Agile because it is virtually a pure form of Agile.

2.2.2.2 MVP – MINIMUM VIABLE PRODUCT

MVP stands for a product that could be sold with a minimum amount of specifications and features. The objective is to place a product that only has the most necessary features with the customer as quickly as possible. The customer's feedback is then used to develop the product or evaluate the customer's level of interest in general. All advanced functions and features are therefore meant to focus on the purpose of the product. Anything that is superfluous and not related to the product's purpose is avoided. It is especially the case in start up companies that there is often a manageable level of capital combined with a high level of risk for the return on investment (RoI). MVP ensures that it is possible to very quickly identify and occupy gaps in the market.

MVP minimises the project timeline by limiting the focus to only the most necessary factors. In a rapidly changing market, long product development cycles always run the risk of developing products that are out of touch with customers' needs.

Advantages

- Short timeline
- Low risk
- Easy to measure a product's or service's sales potential

Disadvantages

- Limited to less complex products and services
- Customer is the tester who determines if the minimum has already been achieved
- Product develops with the customer

2.2.2.3 LEAN STARTUP

Lean Startup is based on Lean Management methods and describes an approach for founding businesses with the least amount of risk possible. This is made possible by minimising the amount of cost and effort required to bring a product or service to market. This does not just involve carefully examining specific deliverables, but also the business model, the distribution channels, the pricing structure and customer acquisition.

Advantages

- Low risk
- Easy to measure a product's or service's sales potential
- Company growth scalable to RoI

Disadvantages

- The customer is the tester of products and beta versions
- Failures and the resulting learning process are components of the start up concept

The following factors are critical:

- Close involvement and communication with the customer (if there is one)
- Lowest possible use of capital or resources
- Short development cycle without lengthy preliminary planning
- If possible, obtain customer feedback very early on in the process using beta versions or prototypes in the market and set out the findings in a product update.
- Incorporate feedback from end users and continuously obtain customer feedback
- Learning by doing and iterative development of products and services
- Often in conjunction with the MVP

2.3 AGILE PROJECT MANAGEMENT

2.3.1 THE FOUNDATIONS OF AGILE PROJECT MANAGEMENT

Agile project management is often erroneously compared with Scrum. Nowadays Agile project management is actually a mixed bag of different practices, methodologies, tools and sets of rules of which Scrum is one of the most well known. The symbolic starting point of Agile project management as a counterpoint to the waterfall model took place in 2001 with the creation of the *Agile Manifesto*, or more precisely the *Agile Software Development Manifesto*.
Principles were formulated in the Agile Manifesto in order to get to grips with the increasingly complex field of software development within the environment of the New Economy after it was found that phase-driven projects were often failing.

The Agile Manifesto states:

- Individuals and interactions are more important than processes and tools
- Functioning software is more important than comprehensive documentation
- Collaboration with the customer is more important than contractual negotiations
- Reacting to change is more important than following a plan

The *Agile Manifesto* should not be misunderstood because the statements on the right hand side should never be completely superseded by the values on the left hand side; it is simply the case that more attention is consciously placed on the statements on the left hand side.

Although the family of Agile methodologies have continued to develop and expand into different versions since the publication of the *Agile Manifesto*, this is where all its fundamental ideas are held. Some of the most well known Agile processes are described in the following chapter.

2.3.2 THE MOST IMPORTANT AGILE PROJECT MANAGEMENT PROCESSES

2.3.2.1 SCRUM

Scrum is not a method, but a framework. Similar to sets of rules used in different types of sports, it sets out operational boundaries and establishes rules. It does not make any statements and it does not specify how something should be developed. It also does not include a strategy that could form a point of reference to work from. Its objective is to create or manage a finished and deliverable product in short iterative steps with the aid of self-organising teams. Single iterative development cycles are referred to as Sprints. The fundamental values of commitment, courage, focus, openness and respect form the basis of Scrum. They form the foundation for the three pillars which uphold Scrum:

Transparency

The important aspects of the working process need to be transparent and clear for all participants, in order to be able to share a common understanding of the methods and the results.

Inspection

The working processes, artefacts (produced by people or modified things) and the results in terms of the achievement of joint goals are regularly reviewed the project participants.

Adaption

Deviations or impediments that have been identified in the processes or results are immediately rectified or eliminated in order to minimise the impact on cost, performance or time.

Scrum consists of three groups of elements, which are closely linked with rules.

A. The Role of the Scrum Team

1. Product Owner

- Is responsible for RoI and is fully responsible for representing the customer
- Establishes what is being developed and why it is being developed
- Approves the deliverables in a Sprint review

2. Development Team

- Interdisciplinary, self-organising team with a maximum of 9 members
- Decides how the deliverables are created
- Reviews its working processes in the Sprint Retrospective

3. Scrum Master

- Ensures that the Scrum rules are adhered to
- Eliminates impediments in the working process
- Facilitates and supports the Scrum team

B. The Artefacts

Artefacts are working documents, documents, charts and results.

1. Product Vision

The product vision serves as a navigational light for all project participants, which highlights the project's goals and purpose.

2. Product Backlog

The product backlog is a list that is prioritised and is usually incomplete, which is organised into themes and aspirations that are to be implemented in the product. In Scrum jargon they are referred to as user stories. It is often the case that user stories are re-prioritised, added, removed or replaced in the product backlog and this process enables development to be adapted to suit the changing conditions.

3. Sprint Backlog

The Sprint backlog includes the tasks that are necessary to implement high priority user stories selected from the product backlog in a product increment in the next Sprint.

4. Product Increment

At the end of every Sprint there is a deliverable and tested product increment that must be approved by the product owner.

5. Sprint Burndown Chart

The progress of the work is visualised via the Sprint burndown chart.

6. Release Burndown Chart

The release burndown chart documents the progress of the work from sprint to sprint and in doing so facilitates release planning for the product owner.

7. Impediment Backlog

A list of impediments that is to be generated by the Scrum master for the Scrum team, which he works through.

C. The Scrum Events (4 Meetings & Sprint)

All events in Scrum are time boxed, i.e. the event needs to be completed during a fixed period of time.

1. The Sprint

The Sprint is a development period which lasts no longer than 4 weeks, during which time the product increment is created. The Sprint backlog serves here as the basis of the work.

2. Sprint Planning

A meeting which involves establishing an understanding between the product owner and the development team regarding the user stories that are to be implemented in product backlog. A second step then involves breaking down an agreed amount of user stories from the development team into tasks for the next Sprint and placing them in Sprint backlog.

3. Daily Scrum

A 15 minute daily stand up meeting during which members of the development team in the Sprint join together to both give a short formal status report.

4. Sprint Review

In a Sprint review the development team presents the product increment to the product owner. Using the criteria agreed in Sprint planning, the product owner verifies the product increment that has been created and approves it when the requirements have been fulfilled.

5. Sprint Retrospective

Following the Sprint review the development team meets with the Scrum master to check the work they have collaborated on and, if necessary, to include suggestions for improvements, which may be implemented at a later date.

D. The Scrum Elements in the Process

The product owner creates an initial product backlog based on a product vision. This is discussed with the development team in the first part of Sprint planning and modified, if necessary. In the second part of the Sprint planning, the high priority elements are then transferred into the Sprint backlog, broken down into tasks and worked through in the following Sprint. The members of the development team who meet in the daily Scrum change throughout the Sprint on a daily basis. Deliverables and working methods are reviewed and modified, if necessary, at the end of the Sprint in the Sprint review and in the Sprint retrospective. The cycle starts again from the beginning with the next Sprint planning meeting. This process is repeated until the product owner declares that the task has been performed or the product backlog has been fully implemented.

E. Advantages and Disadvantages of Scrum

Advantages

- Fewer rules, which are easy to understand
- Clear communication channels
- Greater flexibility and easy to adapt to changing requirements
- Effectiveness due to the self-organising development team
- Greater transparency of processes and results
- Quick implementation of product ideas enables the customer to receive early feedback
- Continuous learning and improvement process

Disadvantages

- Staff must take on more responsibility, but not everyone will agree to this
- Scrum does not provide any recommendations for action
- Scrum is easy to understand, but difficult to implement
- When scaling Scrum, certain rules are often disregarded and the advantages of Scrum are therefore potentially lost. For example, collecting product increments over several Sprints in order to achieve a synchronised, bundled delivery with a different development team.
- Scrum is potentially difficult to reconcile with traditional company structures and would require the introduction of new organisational practices (e.g. employee assessment for independent teams)

2.3.2.2 IT-KANBAN

IT Kanban combines the original Kanban concept with principles from the areas of Lean Production and Lean Development, and also the Theory of Constraints, which states that the throughput of a system is only determined by a limiting factor. This results in a process model for software development in IT companies, which involves displaying the value chain on a Kanban board. By demarcating parallel work in progress (WiP), a continuous and optimum workflow is achieved. Transferring work from station to station takes place in pull mode, i.e. if a station has work capacity available then it acquires a new task provided that the specified WiP level has not yet been reached. Although this was originally devised for IT development, IT Kanban can be very easily adapted to any type of sector or environment. This is possibly one of the reasons why IT Kanban is generally described simply as Kanban in the literature and why the distinction between IT Kanban and the original Kanban has become blurred.

A. THE PRINCIPLES OF IT KANBAN

Visualisation of the Value Chain

The Kanban board is the key component of the workflow visualisation and the identification of bottlenecks. The To Do column right on the left hand side contains all pending tasks. A task then goes through the different stations from left to right in accordance with the necessary process steps, such as analyse, design, coding, testing etc. Each station is also divided again into two columns: in progress and done. Each station can pull new tasks from the done column from its preceding station (pull principle).

Restricting Work in Progress (WiP)

The tasks which are allowed to be processed by each station simultaneously are restricted. Possible bottlenecks are therefore dealt with and the work keeps flowing. This measure, along with the pull principle, ensures that efficiency is maintained at a high level.

Controlling the Workflow and Analysing the Metrics

Prioritising tasks enables tasks with different degrees of urgency to be sent across the Kanban board. The system can be optimised in conjunction with the analysis of the metrics and planning can be developed in a more reliable way. Typical metrics are:

- Lead time (the time that a task takes to go through the complete value chain)
- Cycle time (the time that a task takes to get from point A to point B)
- Throughput (the average throughput of tasks in a given time)
- Cumulative flow diagrams
- Queues
- Blockers (potentially tasks that require external support)
- WiP (work in progress)

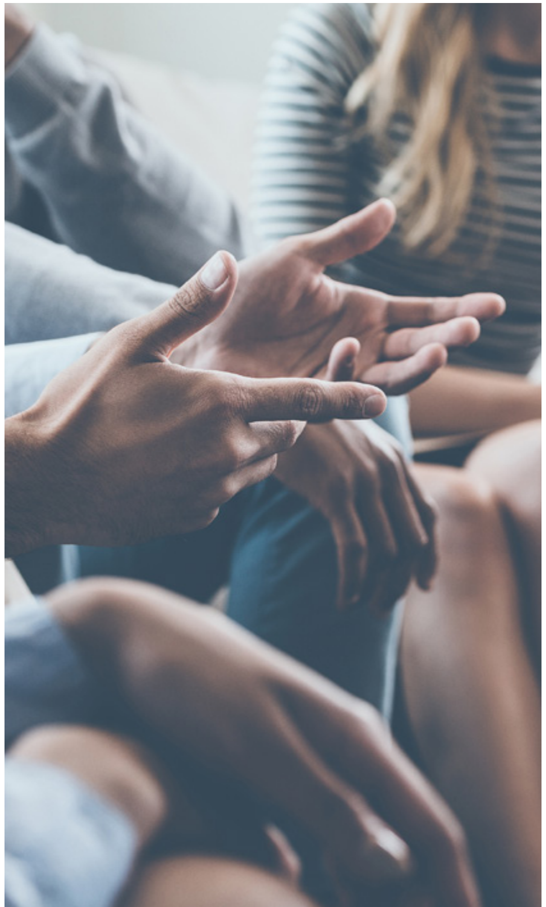
Formulation of Explicit Process Rules

The aim is to create an objective basis to ensure that all process participants are working with a common understanding and are following the same rules. The following agreements, for example, are part of this process rules:

- When is a task classed as done?
- How are tasks pulled?
- According to which rules are new tasks pulled (priority)?
- Who chooses the task?
- How are returns due to incorrect implementation managed?

Promoting Leadership Qualities and Continuous Improvement

Continuous improvement requires the input of *on site* employees who are affected. In order to facilitate communication between the different levels of the organisation, it is necessary to promote genuine leadership qualities and to welcome communication on an equal footing. The ability to learn and to implement what has been learned is an essential component of the improvement process (known as Kaizen).



B. THE INTRODUCTION OF IT KANBAN WITH FLIGHT LEVELS

Flight Level 1

IT Kanban is used in a small organisational unit, e.g. in a cross functional team in order to improve the process in the team. It does not have any influence on external interfaces. The stakeholders that direct the team with their tasks do not come to an agreement and do not prioritise their tasks. The whole system therefore only improves in a limited way.

Flight Level 2

In the second flight level the stakeholders agree with the team in what is termed as queue replenishment meetings. The pending tasks are prioritised and they are assigned service classes with strict rules, which determine how quickly a task is supposed to be completed and how much attention will be given to it. This leads to an increase in effectiveness because results that are necessary to achieve the objectives are favoured.

Flight Level 3

In the next flight level, the IT Kanban system is extended to other teams and team collaboration becomes a prime focus. Bottlenecks become visible by optimising the whole value chain incrementally. When the bottlenecks are resolved waiting times at the interfaces are reduced and the workflow progressively improves.

Flight Level 4

In the final flight level 4, the company portfolio is managed through the use of IT Kanban. The value streams from several products and projects are optimised and what is *achievable* is balanced out with what is required.

C. ADVANTAGES AND DISADVANTAGES OF IT KANBAN

Advantages

- Easy to implement in any environment and processes, even with the smallest teams
- Direct visualisation of the process
- The Kanban board is a good communication tool for team synchronisation
- The WiP limit prevents unconstructive multitasking
- Unlike with Scrum, the size of the tasks is not limited (Sprint length)

Disadvantages

- A physical board is only workable with small, local teams
- In large projects the physical Kanban board becomes confusing
- Documentation and archiving make it necessary to use software solutions
- Kanban only develops to its full potential if the entire supply chain is consistently incorporated from the requirements stage right through to delivery (continuous delivery)

2.3.2.3 EXTREME PROGRAMMING (XP)

XP is an iterative process used for developing software. In the original version it consists of interrelated elements, namely values, principles and practices. XP is suitable for small to medium sized teams of up to 15 people and is a strict and disciplined development process, which places a strong focus on customer involvement.

A. ROLES IN XP

The roles in XP are not set in stone and merely serve as a guide during the initial phase. The objective is to ensure that every member of the XP is able to contribute to the overall success of the project within his capabilities and regardless of his role.

1. Coach

The coach promotes the fundamentals of XP, helps the team to select the most effective tools and ensures that the values, principles and practices are adhered to.

2. Tester

The testers provide automated tests in accordance with customer requirements and are responsible for the quality of the finished product.

3. Interaction Designer

Interaction designers form the bridge between developers and users. They support the customers with the creation of user stories and support their implementation.

4. Architect

Architects are responsible for the internal structure of the system and optimise this as required.

5. Project Manager

The project manager organises communication between the XP team and the customer. He is responsible for progress tracking and for planning the next tasks.

6. Executives

The executives are the project sponsors and have responsibility for the project.

7. Technical Writer

The technical writer describes the system functions and gathers information from discussions with users about their experiences in order to integrate them into the end product.

8. User

Users create system requirements in the form of user stories. Ideally, users represent the broad spectrum of future users.

9. Programmer

Programmers develop the system and transfer the user stories into functioning software and programme functions.

B. VALUES, PRINCIPLES AND PRACTICES IN XP

Values

The values of communication, courage, feedback, respect and simplicity are of prime importance in XP and form the foundations upon which the principles are built.

Principles

- There are 14 principles, which form a bridge between the values which are more abstract and the practices which will be directly used:
- Accepted responsibility
- Assume simplicity – both with the development process and the product
- Concrete experiments (targeted experiments highlight errors at an early stage and also confirm workable approaches)
- Embracing change (whoever resists change cannot become agile and adapt to constantly changing conditions)
- Honest measurement (avoid talking up results)
- Incremental change
- Local adaptions (XP should be adapted to local conditions, which means both the working methods and the end product)
- Open, honest communication
- Play to win
- Quality work (not quick and dirty)
- Rapid feedback (keeps costs low and avoids delays)
- Small initial investment
- Teach learning (continuous learning is an essential component of Agile approaches)
- Travel light (avoid unnecessary and unwieldy tools and methods)
- Work with people's instincts, not against them

Practices

- On-site customer
- Planning game (at the beginning of the development iteration in order to agree the scope of work and the technical implementation)
- Short releases (in order to gather customer feedback)
- Coding standards (jointly agreed coding standards)
- Continuous integration
- Collective ownership (responsibility is held jointly for the system that has been created)
- Metaphor (user stories are described in a way that everyone understands)
- Sustainable pace (avoid overburdening team members through overly heavy workloads)
- Pair programming (programming in pairs with regular role reversal)
- Refactoring (a deliverable is continuously enhanced in small steps whilst still maintaining functionality)
- Simple design
- Testing

C. ITERATION LOOPS WITH XP

An XP project begins with the formation of the project team that needs to be set up in accordance with the subject of the project in a multifunctional way. The user works with the team to develop his requirements in the form of user stories. The architect generates an initial system design. After the release has been planned the iteration loops begin in the next step in order to develop the system. The releases are thoroughly checked, errors rectified and submitted to the user.

1. **Second by second:** Pair programming enables the code that is generated during development to be scrutinised by a second programmer and errors or design weaknesses can be rectified immediately.
2. **Minute by minute:** Development in XP is test-driven. The UNIT tests that are generated before the code facilitate ongoing high test coverage.
3. **Hour by hour:** The practice of continuous integration enables the developed components to be integrated into the executable system several times a day. This approach enables errors to be detected at an early stage and avoids expensive long term effects.
4. **Day by day:** Similar to the daily Scrum that occurs as part of the Scrum framework, a daily stand up meeting also takes place with XP during which the development team discuss project progress and reflects on problems. As the meetings take place standing up, which is not the most comfortable of situations, participants tend to focus more intensely in order to ensure that the meeting is completed in a short a time as possible.
5. **Week by week:** Executable system versions are made available to customers at frequent intervals in order to demonstrate that their requirements are being faithfully represented. Faults can therefore potentially be avoided and counteracted in the early stages. Any additions to the requirements can also be included for later implementation.
6. **Month by month:** In contrast to phase-driven project models, releases are not delivered on one occasion only at the end of the project, but are delivered on a monthly basis, if possible. This makes a part of the receivables financially available to the customer at an early stage.

D. ADVANTAGES AND DISADVANTAGES OF XP

Advantages

- The techniques are easy to combine with other Agile approaches
- Greater flexibility during implementation
- High quality results

Disadvantages

- As a complete package, it is more suitable for small to medium sized IT teams
- Promotes effective customer integration, which is not always easy represent
- Less suitable for dispersed teams
- Very IT focussed and therefore difficult to transfer to different business sectors

2.3.2.4 DESIGN THINKING

In its approach Design Thinking goes way beyond classic models and, in contrast to the technical disciplines where feasibility is of paramount importance, the user and his requirements take centre stage. The multidisciplinary development teams put themselves in the position of the user and try to analyse the problem through his eyes. Design Thinking is therefore not confined to one business sector, but is suitable as a systematic solution strategy for many complex problems and for developing services and innovative products from the most varied of fields. Solutions and ideas that are identified should be made into prototypes as soon as possible, in order to obtain appropriate feedback from the user. The results that are achieved and the innovative solutions finally unite the three essential components; technological feasibility, cost effectiveness and the ability to fulfil a desire or a need.

THE THREE ELEMENTS OF THE WORK ENVIRONMENT

The success of Design Thinking is primarily based on a collaborative thinking and working process, which is facilitated by the following three elements:

Multidisciplinary Teams

Several multidisciplinary heterogeneous teams each comprising five to six people work on a single subject with the support of a coach trained in the methodology. The objective is always to achieve tangible and comprehensible results and to share them with other teams in order to obtain as many different perspectives as possible that help solve any problems.

Flexible Work Spaces

Work spaces and furniture are designed in a flexible way in order to support the creative process. A variety of presentation options (white boards, flip charts, post-it notes etc) and tools to build the prototypes (paper, Lego, materials etc.) provide plenty of scope for team members to exchange thoughts and to generate and implement ideas in a playful way.

The Design Thinking Process

Design Thinking has its origins in Silicon Valley, which is an environment that is characterised by its pragmatic and dynamic entrepreneurial activities. In Design Thinking an idea is swiftly turned into a prototype that includes all the necessary requirements. This takes place iteratively in six phases enabling feedback to be generated from the preceding phases. Phase 1 is the *comprehension phase* in which the team members attempt to put themselves in the user's shoes using surveys in order to be able to follow *observations* in phase 2, in which the team observes how the user deals with the specific issues. Viewpoints are then defined in the third phase based on these observations. The findings derived from this process are then collated, consolidated and various user perspectives are mapped out to provide a comprehensive 360 degree view of the problem. The fourth phase is the *brainstorming* phase, which links to the preceding *empathy* phase. The work environment and its ability to facilitate creative processes plays a significant role in this search process. In the penultimate phase, phase 5, the team focuses on the most suitable ideas and transfers these ideas into prototypes that require the minimum cost and effort. Finally, during the sixth phase, the prototypes are assessed to determine how useable they are.

ADVANTAGES AND DISADVANTAGES OF DESIGN THINKING

Advantages

✓

- Prototyping at an early stage enables conclusions to be drawn quickly concerning the usability of a solution
- Design Thinking enables many different creative techniques to be integrated
- Multidisciplinary teams provide an extensive range of perspectives about the issue

Disadvantages

⛔

- It is essential that the facilitator is experienced
- Creative processes that are used during brainstorming are not always tangible and results are not mandatory
- Expensive facilities are required and it is time consuming

2.3.2.5 FDD FEATURE DRIVEN DEVELOPMENT

The FDD Agile process does not enjoy the same level of popularity as Scrum and XP, however, FDD has advantages which make it interesting when considering Hybrid project management. Small to medium sized teams made up of highly qualified specialists are the domaine of Scrum and XP. For this reason, scaling is difficult and it is often achieved at the expense of the original agility. From the outset FDD was conceived for large projects and it manages to incorporate both small and large teams without demanding restructures and adaptation processes. Unlike its Agile counterparts Scrum and XP, FDD includes a more distinct planning phase leading up to the iterations and is therefore often, unjustifiably, compared with a phase-driven project management approach.

A. ROLES IN FDD

1. Project Manager

He coordinates the allocation of resources and is responsible for scheduling the project. The project manager does not play a part in the actual programming work. He is responsible for the administrative tasks in the project.

2. Chief Architect

He keeps track of the architecture of all the software and the central models. The chief architect supports developers and customers throughout the process of collaboratively developing new software components. In small projects with a small team, the roles of the chief architect and the chief programmer are merged into one role.

3. Development Manager

He manages the day-to-day business and resolves resourcing issues. In certain projects the role of the development manager is linked with that of the chief architect or the project manager.

4. Chief Programmer

In larger projects experienced chief programmers lead the individual developer teams, create feature lists, plan and manage their implementation, and are responsible for the operational activities that are assigned to them.

5. Class Owners

Class owners are developers who look after the implementation of features in small teams and who are involved in their technical planning with their chief programmer and chief architect. They are assigned to the key classes.

6. Domain Experts

Domain experts are users, customers, sponsors, business analysts or a combination of the above. They have a thorough knowledge of the task (domaine knowledge), that the product to be developed must have.

7. Supporting Roles

In addition to the roles detailed above, there are other roles, which can play a supporting part, as required: language guru, release manager, build engineer, system administrator and tool smith.

8. Zusätzliche Roles

Testers, deployers, technical writers.

B. THE PROCESSES IN FDD

1. Process: development of an overall model

- Formation of the modelling team from chief programmers and domain experts
- Circulate an overview of the domain areas to be formed by the domain experts
- Review of available requirements documents and references
- Development of models in small groups
- Amalgamate the small models into one team model
- Iterative improvement of the team model
- Setting out in writing important notes relating to the team model
- Identification of the key classes and verification of the model in the form on an assessment

2. Process: creation of a feature list

- Formation of the feature list team from the chief programmers from the modelling team
- Creation of feature lists for the individual system areas led by the chief programmer
- The results from process 1 are firstly subdivided into major feature sets (subject areas) and then into business activities (feature sets). Each step in an activity is identified as a feature. Hierarchically arranged, categorised feature lists are the result of this analysis.
- Implementing the features is not permitted to take longer than the anticipated time frame of two weeks, otherwise the feature must be broken down into sub features.
- Verification of the feature lists in the form on an assessment.

3. Process: planing the features

- Formation of the planning team to include the project manager, development manager and the chief programmers
- Specifications for the sequence of development
- Assign the business activities (feature sets) to the chief programmers
- Creation of a class owner list in which the developers are identified as the supervisors of the known key classes
- Creation of an implementation plan, including scheduling, taking into account dependencies and the complexity of the features and the availability of resources.
- Verification of the results in the assessment format.

4. Process: designing the features

- Formation of the feature team by the chief programmer
- The domaine expert provides a summary of the features that are to be implemented via the domaine
- Review of available requirements documents and references
- Development of the sequence diagrams
- Creation of the features models
- Inspection of the design

5. Process: building the features

- Implementation of the features that have been set up from process 4
- Code inspection
- Unit tests

Processes 4 and 5 are managed in loops until all the features are implemented and the project is completed. If necessary, more features can be added in step one based on findings from the implementation of the overall package.

C. ADVANTAGES AND DISADVANTAGES OF FDD

Advantages

- Scalable for larger projects and teams
- Better for teams in which team members can contribute different types of experience
- Offers defined project tracking facilities
- Supports several teams working in parallel

Disadvantages

- Even though FDD constitutes a project with an agreed functional scope, it carries a certain amount of uncertainty in relation to complex project issues. Despite careful preliminary planning and assessment, previously unforeseeable problems or risks can arise during development, which invalidate the performance planning, scheduling and resource planning which have been carried out so far.
- The programme code is not assigned to a team (shared ownership), but to an individual.
- The iteration steps are not defined.

3. THE BASELINE FOR HYBRID PROJECT MANAGEMENT

3.1 THE BASIC TYPES OF HYBRID PROJECT MANAGEMENT

Hybrid project management results from developing an existing framework and adding new elements, or selecting various concepts and introducing combinations of these concepts.

The following considerations are applicable to the three most important basic Hybrid formats:

1. The Classic project management culture is developed using Agile concepts (Classic-Agile)
2. A combination of Agile concepts (Agile-Agile)
3. Lean Project Management supplemented with Agile elements (Lean-Agile)

3.2 THE CORRECT SELECTION OF COMPLEMENTARY ELEMENTS

To create a suitable Hybrid system, it is necessary to focus on the components' potential options and requirements and to gain sufficient knowledge about them. In the following section, the selection criteria are described in closer detail to simplify the choice. In addition, an XP principle should be adhered at this point, namely travel light i.e. limit the focus towards only what is necessary and only combine several systems with each other that are absolutely necessary to tackle the task.

3.2.1 SIMPLE, COMPLICATED OR COMPLEX?

Project forecasts are often subject to a high degree of uncertainty. The concept of the *Cone of Uncertainty* estimates the probable range of the actual project costs by up to four times the amount estimated before the beginning of the project. This is even more evident the more the project's performance is tied to research and development. The reason for this is the inadequate planning capability of dynamic and complex projects. Classic project management requires pre-planning of the project before the actual work begins. This is achievable if the problem is clearly defined and if the project's performance target is relatively static. Even complicated problems can be solved with the appropriate knowledge and planning efforts in a Classic project environment without a high degree of uncertainty. This is different if complex and/or dynamic project objectives are involved. Whilst complicated issues can be distinguished through linear relationships, complex themes have interacting elements, which make far-reaching predictions impossible and do not enable the issues to be captured through planning. If the project goal is also *flexible*, i.e. the project goal can only be mapped accurately over the project timeline, it is only possible to always plan the project *on sight* and work through it incrementally. In doing so, Lean Management or Agile project management approaches are followed.

Decision Matrix to Select an Approach for a Project or Project Component

Project objective	Implementation of the project objective	Favourable project management approach
broadly static	simple	classic
broadly static	complicated	classic or agile
broadly static	complex	classic, agile or lean
dynamic	simple	agile or lean
dynamic	complicated	agile or lean
dynamic	complex	agile or lean

3.2.2 REQUIREMENTS RESULTING FROM THE PROJECT, THE BUSINESS AND THE TEAM

1. PROJECT TYPE AND BUSINESS SECTOR

Whilst Classic project management covers all the different types of projects in all business sectors, including organisational, investment and development projects, the origins of many of the Agile equivalents have been laid down specifically in development projects in the IT sector. It is therefore necessary to consider whether using specific Agile methods is feasible and worthwhile. In many cases transferability is completely unproblematic and in other cases adaptation processes are absolutely necessary. However, sometimes it is still not possible to use certain approaches outside of their *home business sectors*.

Example 1: XP practices such as pair programming which are devised to be used exclusively in software development and their basic ideas can at best be adopted by other business sectors. They therefore no longer correspond with the original and the user benefits of this approach must be evaluated on an individual basis. Examples of the use the pair programming method can now be found in the legal field (pair contracting) and the design industry (pair designing).

Example 2: events such as the daily Scrum used in the Scrum methodology are easy to transfer to other business sectors and facilitate improvements in knowledge sharing in any environment.

2. PROJECT SCOPE AND TEAM SIZE

Every project management approach has its own optimum field of work and should also be deployed as part of a Hybrid structure appropriately. This should not mean that a chosen approach will not function outside its normal comfort zone, however, certain limitations and additional costs can certainly arise.

Example 1: Classic project management requires extensive organisational investment, which is unnecessary and therefore inefficient for micro-projects.

Example 2: Scrum rules are very suitable for small to medium sized tasks and teams. Scrum must be scaled for large projects. There are various approaches to scalability, which in reality often result in a loss of agility (e.g. with the delivery of increments) due to potentially unmanageable channels of communications and challenges posed by the synchronisation of individual Scrum teams and their results.

Suggestion Matrix for Selecting an Approach for a Project or Project Component

Project size and team size	Favourable project management approach
small - medium	Lean or Agile
medium - large	Classic, Lean, Agile
large - very large	Classic, FDD, Kanban or Design Thinking

3. COMMUNICATION CULTURE AND CONNECTING STAKEHOLDERS

Disbursed project teams and stakeholders pose an additional challenge. Agile methods in particular focus very heavily on integrating the communication channels used by the participants and on more regular and intensive information sharing within the teams, which is also mandatory for customers and stakeholders.

In addition to an assortment of components, a range of appropriate communication platforms need to be provided when designing Hybrid project management methods in order to ensure that teams communicate with each other, and also the customer and other stakeholders.

4. WHAT IS REQUIRED? A TOOL, A METHOD OR A COMPLETE STRUCTURE?

Irrespective of the problem that needs to be resolved, it is important to establish whether a complete project management set of rules is required or whether a method, a philosophical approach or simply a tool to support the existing structure is perfectly adequate. Making parts of the existing project management approaches available and integrating them into the particular project landscape is entirely feasible. This does, however, require a broad knowledge of the different options that are available.

Example 1: in a Classic project environment interaction and communication problems can arise, which delay the project life cycle. The project manager decides together with his team to introduce IT Kanban in flight level 1 in order to visualise the value chain and to facilitate mutual understanding in the team. Irrespective of the success of the Hybrid approach, the inclusion of additional teams is taken into consideration.

Example 2: in a Scrum project the development team decides to use XP techniques to carry out the Sprints in order to ensure a high level of quality for the software that is produced.

3.2.3 THE ORGANISATIONAL ENVIRONMENT AND THE CULTURAL ASPECT OF CHANGE

Usually processes and rules within a business are designed in such a way that they strengthen the internal structure of an organisation. At the same time, the interplay that occurs within the Hybrid structure is closely linked with elements embedded within the corporate culture.

Whilst Classic project management structures can be very well integrated into hierarchical structures, Lean elements depend on strong involvement from project team members. Ultimately, many Agile components almost completely abandon hierarchical structures, demand self organisation and depend on leadership rather than management. Agile working cannot be dictated ("be Agile!"), but develops if a suitable project environment is provided.

With this mind, structural adaptation processes need to be started in order to use the selected Hybrid elements in an optimum way.

1. Classic project management

- Hierarchical organisational structure
- Top-down assignment of project participants
- Project team members are directed by the Human Resources Manager
- Line managers must assign and work and collect the results
- In the matrix organisation, project team members have more than one line manager

2. Lean Management

- Hierarchical organisational structure
- Top-down assignment of project participants
- Project team members are directed by the Human Resources Manager
- Tasks are specified collaboratively with the line manager
- Opinions and suggestions for improvements from the team members are discussed with line managers and implemented if approved.

3. Agile Project Management

- Self organising teams
- Top-down assignment of project participants should be avoided
- Teams are guided by targets
- Team members must be in agreement with working in self-organising teams and agree to take on the associated responsibility that is involved.
- Team members make the work they perform transparent
- The use of facilitators to support team members with their work and to resolve any conflicts.
- New internal processes need to be set up, which in hierarchical organisational structures are managed by line managers who are responsible for disciplinary matters. Examples of this are staff evaluations, professional development, additional training and holiday planning.

3.3 THE MOTIVATION TO USE HYBRID PROJECT MANAGEMENT

The motivation to use Hybrid project management shapes the transformation processes that need to be managed and the achievable goals in a significant way.

Generally it is important that the objectives are directly connected to the opportunities which a Hybrid approach, or the addition of certain elements, can offer.

1. Resolution of an acutely problematical situation in the Classic project environment

- The current project approach is not able to achieve the project objective in compliance with the guidelines. Psychological strain has already resulted.
- A conflict situation between the customer and the contractor relating to project progress or results has intensified.
- It is clear to the project participants in advance of a project that the project is difficult to plan in its entirety or in phases.
- The customer finds it difficult to formulate the project goal in a precise and complete way.

Objective of the Hybrid method: the use of Lean and Agile elements to help the customer to become more effectively integrated, to use incremental development to drive forward high priority sub-goals for the customer and, following consultation with the customer, to perhaps initially limit activities to the most important goals.

2. Hybrid project management as a transition process from Classic to Agile project management.

- The business expects significant benefits from a new company structure and an Agile approach to project management.
- Jumping in at the deep end is not a suitable approach for the company to choose because it is not possible to estimate the possible risks associated with a re-structure.

Objective of the Hybrid method: This approach provides a degree of certainty and gives the participants space to adjust to the new processes and rules before the transition is finally and fully completed. The Hybrid intermediate points must be established in advance and, following their introduction, they must be assessed to determine whether they are appropriate.

3. Improvements in customer integration in a Classic project environment

- Current communication with the customer is not always appropriate to ensure that customer wishes are adequately represented in the project results.
- This is due to several reasons: different languages are spoken between the customer and the contractor/developer and the customer's wishes are vague.

Objective of the Hybrid method: inclusion of practices and communication tools to be able to respond to the customer more effectively without losing the centrally planned project management approach. Possible division of the project method into an Agile pre-development process and a traditional implementation across the board.

4. Follow a trend and introduce Agile elements in an existing Classic project environment.

- Agile project methods are appealing due to the range of advantages they offer compared with existing Classic structures.
- Reports have been published in many publications describing the positive experiences companies have had that have introduced Agile project methods.
- The constant repetition of Agile terms underpins the feeling of *being involved*.

Objective of the Hybrid method: Being able to keep pace in project management with new management methods and the changing working environments.

It is important to note: certain adjectives are repeatedly associated with Agile project methods, which are not appropriate in reality. Before a Hybrid approach is chosen, it is imperative that sufficient knowledge is acquired, along with a broad overview of the possible partner elements, in order to safeguard the organisation from any undesirable developments. Agile does not mean quicker, instead it means more manoeuvrable and more adaptable!

5. Performance improvements

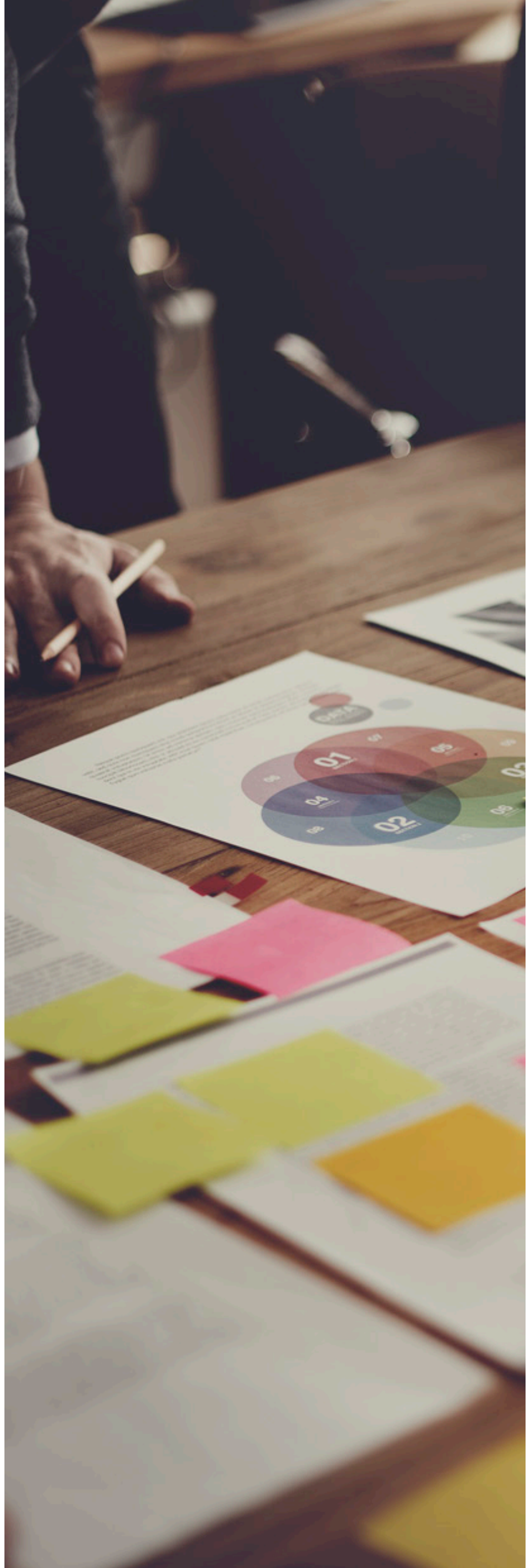
- The existing system does not appear to be performing adequately
- Quality defects
- Communication problems and the silo mentality
- Suitable tools, methods and procedures have not been defined

Objective of the Hybrid method: introduction of elements, which make an organisation's own structure more effective and efficient, increase quality and cooperation and make a working environment available.

6. Docking on to other project cultures

- Occasionally partner companies or customers work with other project approaches, which are incompatible with your own project culture.
- Statutory provisions and regulations recommend or demand compliance with certain project schemes.
- Collaboration with colleagues, partners and customers from the international arena explicitly demands a culturally appropriate type of project management.

Objective of the Hybrid method: set up docking points in order to offer a suitable project interface to the partner and to ensure that guidelines are observed. The project management approaches that are chosen should be selected in a culturally acceptable way in order to enable international colleagues to be integrated effectively and ensure that the expected results are achieved.



4. THE HYBRID PROJECT MANAGEMENT PATH

There are no ready-made solutions and no best practice blueprints. At best there are concepts which can and must be empirically tested and fine tuned. Every organisation, its employees and its projects are unique and require an individual, tailor-made solution.

4.1 HYBRID PROJECT MANAGEMENT STRUCTURES IN THE PROJECT CONTEXT

The options that are available for an individual project are considered and also the organisational structure in terms of a project pool or an environment for multi project management.

1. Incubator

A separate protected project environment within an organisation in order to enable a new project management approach to be tested in an unbiased way. The aim is to benchmark project outcomes outside the incubator. When a project is successfully completed, the project team can create a nucleus to help with the transfer of applied sets of rules, methods and tools into other projects.

2. Container Solution

Some of the project approaches create a framework of rules or requirements, which can still be filled with content. In the container solution a second system is embedded in the existing system in which both systems are independent from each other and do not affect each other.

Example 1 (Classic-Agile): in a project environment characterised by Classic methodologies, IT Kanban is introduced in order to improve communication, control WiP and reduce risk.

Example 2 (Agile-Agile): a development team in a Scrum project decides to use XP in their Sprints in order to be able to draw upon a pool of common software development techniques.

3. Partial Assimilation

An existing project management concept is expanded through the use of components from a second concept. The new structures that have been added are completely integrated into existing project management methods.

Example 1 (Classic-Agile): in a Classic project approach self reflection and team communication are intended to become stronger. The project leader decides to hold *reviews and retrospectives* with the team at two week intervals in order to examine both the project results and the working methods in a critical way.

Example 2 (Classic-Agile): a company that is influenced by Classic methods enters into an agreement with the customer to design the customer requirement specification with part of the project team to ensure well in advance that the customer's wishes are better understood and potentially to show the customer alternative ways of doing things.

4. Synthesis

Unlike the container solution, two different project management approaches are used here and are merged to create a new hybrid in order to rule out any possible deficiencies in the individual system.

Example (Agile-Agile): one of the most well known systems is Scrumban, which is a combination of Scrum and IT Kanban. In Scrumban the cyclical structure is transferred into a continuous workflow. Planning for the Sprint and storing up subjects in the product backlog is no longer carried out at specific points in time, but is based on actual requirements. The usual Scrum burndown charts are superseded by Kanban metrics.

5. Parallel Worlds

Before the project starts, sub-projects or work packages are assigned appropriate project approaches depending on requirements and are then dealt with on an individual basis. This assumes that the modules that will be worked on can be developed independently from each other.

Example (Classic-Agile): in a project, hardware and software components are developed in different ways. Certain interface specifications that needed to be followed for the integration of the module at the end of the project were defined in advance for this purpose. Subsequently, the different development strands are managed separately. Hardware development on traditional lines and software development using an Agile approach. The results are compiled after project completion.

6. The Interlocked Solution

During the course of the project there are sub-projects or work packages which are dependent on other sub-projects/work packages, however, they need to be handled differently from a project methodology perspective. This lends itself to an interlocked approach. The Hybrid approach is similar to the "parallel worlds" approach, however, synchronisation measures need to be taken to ensure that the results from the subsystems are compatible.

Example (Classic-Agile): in a project, hardware and software components are developed in different ways. As the two subsystems are closely linked to each other, there are employees who are integrated into both teams and whose task it is to ensure that the project outcomes are structured synchronously. In doing so, the different development strands are managed loosely together and the results are checked for consistency at certain synchronisation points. Hardware development is run on traditional lines and software development makes use of an Agile approach. The results are compiled after project completion. This hybrid approach only makes sense if it can be assumed that the scope and number of changes on the hardware development side do not put too much strain on the traditional approach.

7. Coupled Systems

Coupled systems are an additional way of grouping project approaches that belong to a new Hybrid model. Whilst some Hybrid approaches are intended to feel their way towards creating a new project management culture, the coupled approach is a very precise tool, which is intended to achieve specific goals. To achieve this end, accurate knowledge of the difference project approaches is essential.

Example (Agile-Lean-Agile): the aim is to create a low risk, fast and Agile start up process for a new product. A complete set of rules from the various types of project management are intended to safeguard these requirements and accompany the process from the development of the product idea right up to the point when a series of product features have been brought to the market. The solution is a coupled system made up from Design Thinking, Minimum Viable Product (MVP) and Scrum. Design Thinking helps to covert a product idea into an initial testable prototype, which is aligned with customer requirements. The use of MVP enables the prototype to be transformed into a first product that has been reduced down to the necessary minimum and to be tested to determine if it is ready for market. A Scrum approach enables the product range to be incrementally widened with the success of the market launch.

4.2 CHALLENGES POSED BY THE HYBRID CULTURE

The introduction of a new project management culture initially brings many complications and, alongside the positive effects that can be expected, it is important that the associated effort and costs are not forgotten. It is a lot like when a patient recovers from an illness. It often gets a lot worse before it gets better. Changing to a new project management approach and the potential change in the corporate culture that will also result from this will take time. Organisations often resist change.

Staff need to buy into the process and require training. Unfortunately, it may also sometimes be necessary to lose certain members of staff. Not everyone will feel welcome and needed in a newly structured project environment and organisational structure. The legal framework needs to be checked, internal data amended and existing quality standards need to be updated (e.g. ISO 9001).
Finally, it needs to be ascertained whether the hybrid system satisfies the specified requirements, or whether subsequent improvements or an alternative approach are required.

5. CONCLUSION – WHAT COMES NEXT?

Ultimately, it is important to check whether the whole Hybrid concept is more than just its sum individual components because only then was success achieved with its implementation. What follows next depends specifically on the company's executive board. Do you choose to transition slowly and gradually from a project domain based on the Classic approach to an Agile/Lean project domain or do you complete the whole paradigm shift in the form of a revolution? Do you take incremental steps from one Hybrid solution to the next in order to incorporate the best approaches you have discovered into your toolkit, or do you consolidate the newly discovered Hybrid project management approach? It is probably becoming increasingly impossible to avoid continual changes in our project culture because our working environment is also in a state of constant change that occurs in ever shorter cycles. Hybrid project management will therefore provide a valuable contribution to enable us to sufficiently keep up with this rate of change in our projects.

IMPRINT

www.iapm.net

1st edition
Copyright © IAPM 2017

Photography: www.istockphoto.com

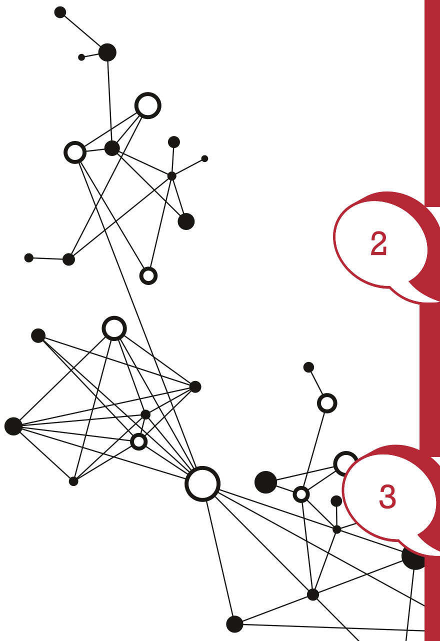
Edited by IAPM
International Association of Project Managers™ in Liechtenstein

ISBN: 978-3-941739-30-7

Quality Management System
The quality management system used by the
IAPM International Association of Project Managers™
satisfies the requirements of ISO 9001.

Trademark Protection
IAPM International Association of Project Managers™
is a protected EU trademark – no. 9539354 –

The IAPM is an ideally association headquartered in Liechtenstein.
Nr. FL-0002.353.470-6



THE BENEFITS

HOW YOU CAN BENEFIT FROM IAPM CERTIFICATION

1 Competitive advantages & career launching pad

- Proven agile project management competence
- Competitive advantages for organisations and individuals
- Standardisation of terms and methods with the Hybrid PM Guide 2.0
- External, objective verification of knowledge and experience

2 Online examinations

- No travel expenses
- No pressure of time to prepare
- Exams can be taken on any PC

3 No re-certification necessary

- No certificate expiry date
- No new costs

4 Fair fees

- The fees depend on the GDP of the country in which the certificate candidate has citizenship.

5 Anonymous Certification

- No subjective evaluations of the certification holder
- No “fail quota“
- No discrimination of the certification holder