The Leading Reference for technology-based products

Product Management Journal

Volume 7 £10 / €10 / $14

Contents

Agile

And its impact on product managers

product
focus

Training

Product Management and Product Marketing

Industry-leading public and private courses

Focus on technology-based products

Tools, templates and checklists

Learn best practice

Certification

PRODUCT MANAGER product focus CERTIFIED

product
focus

★ Trustpilot

+44 (0) 207 099 5567
+1 (877) 290-6240
www.productfocus.com

Welcome

Leading the way for product managers

So what does Agile and Scrum mean for product managers?

You may be working in a software company that already uses Agile development approaches like Scrum. You might be encountering Agile for the first time as it's introduced by your development team or you might have heard about Agile and wonder what value it can bring to your business. Our aim in this journal is to give you a basic understanding of what it's all about and importantly what it means to you as a product manager.

Agile is now established as the dominant approach when developing software products. Whilst there has been widespread adoption and undoubted benefits, the traditional waterfall approach is still used in a vast number of organisations. We want you to sift the hype from the reality and help you talk sensibly about Agile.

Who's who?

The *Product Management Journal* is published by Product Focus as an independent publication for product managers with technology-based products. Product Focus was founded and is run by Ian Lunn (top) and Andrew Dickenson.

The founders continue to deliver many of Product Focus's training courses and reviews alongside their team of senior consultants.

To get all our previous journals, and receive the latest copy, sign-up at **www.productfocus.com**

Agile

Everything you need to know about Agile

Agile is a generic term for software development approaches designed to handle uncertainty and change. This is done by delivering iterations quickly and often. The goal is to produce working solutions that can evolve based on customer feedback and changing requirements.

It is often contrasted with more traditional Waterfall development methodologies where development flows through a series of phases. The next phase can only begin after the previous one is complete. At a simplistic level, requirements are signed-off at the beginning of a project, and are developed to deliver the final product.

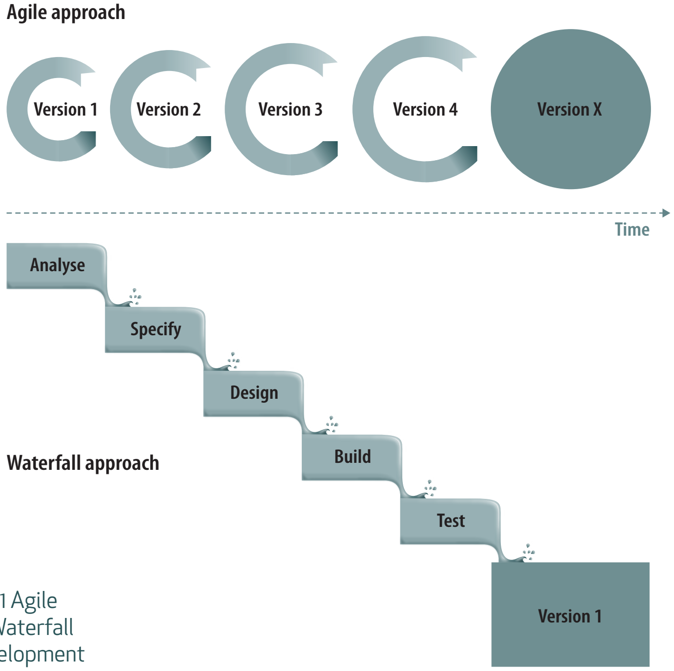


Fig. 1 Agile vs Waterfall development approaches

The attraction of Agile

Waterfall methodologies are rooted in the world of manufacture and mass production. You provide all the requirements up front and identical products roll off the production line. Success requires detailed specifications, accurate resource forecasting and top-down planning.

The problem with developing products using the Waterfall approach is that you have to know what you want at the beginning and wait until the end to see the results.

Unfortunately businesses, clients and users are often not sure what they want at the beginning of a project. They struggle to articulate their requirements. They change their minds as their understanding improves and the market context changes around them. Waterfall attempts to deal with this through change control processes but these can be complex to manage and difficult to feed into the development cycle.

Software development is inherently more flexible than physical manufacture. It's usually easy to create prototypes and develop new versions which makes an Agile approach possible. In a survey (Oct 11) by Ambysoft on IT Project Success, respondents were asked whether projects were seen as successful, challenged or failures by their organisation. Iterative and Agile projects were reported as around 15% more successful in comparison with projects that took a traditional or ad-hoc approach.

"Product sponsors like Waterfall – they have the perception they know up front what they're going to get and with Agile they won't – however Agile shows the sponsor tangible progress and lets them shape the deliverable from the outset."

James Salmon, Group Product Manager, Mobile Self Care, Vodafone Group Online

However, Waterfall has not gone away

In most larger organisations a Waterfall approach to new product development has been implemented and embedded across the whole company. This is the only way to manage and control multiple large-scale projects. These are often known as Stage-Gate processes where projects can't proceed to the next stage until the previous gate has been passed.

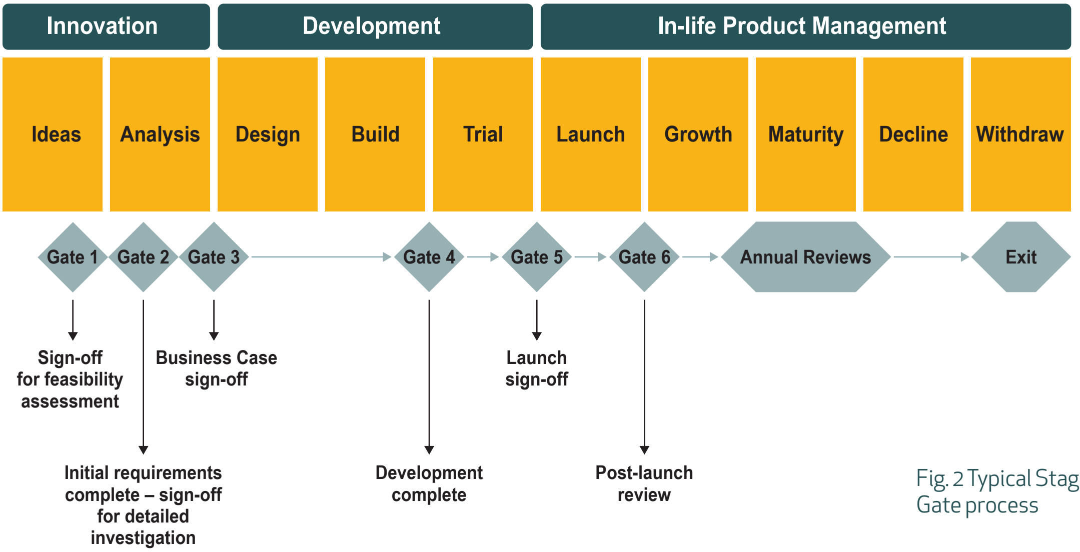


Fig. 2 Typical Stage-Gate process

Agile is not the answer to everything

A pure Agile approach doesn't work all the time. In the Telecoms and IT industries most development is about more than '*cutting brand new code*'. It's about knitting together complex software systems, hardware and service components. Offshore development requires co-ordination with other development teams in different locations, time-zones and companies. Other departments such as marketing typically have long lead-times so need to know a launch date and release contents months in advance. For all these reasons the top-down planning of more traditional approaches is still needed.

Agile development methods explained

Kanban means 'signboard' in Japanese and describes managing a process by making the workflow publicly visible to all. Developers regulate their work by pulling tasks from a queue when ready.

Lean software development is a set of principles that help software businesses optimise their processes and development methods. The aim is to get products to market faster and with higher quality by delivering the minimal set of requirements.

Scrum is the most widespread and popular Agile software development approach. It is a project delivery framework based on short, time-boxed, incremental, development releases (see article on p.8).

XP (Extreme Programming) is an approach based on short development cycles and taking various software development practices to extreme levels, e.g. writing the test first and then coding and two developers working at the same workstation (pair-programming).

Water-Scrum-fall

There are very few software developers who are unfamiliar with the concepts in Agile however adoption by business is not universal. From our 2018 annual survey we know that 49% of companies use pure Agile, 41% use both Agile and Waterfall (sometimes mixed together, e.g. Water-Scrum-fall) and 10% just Waterfall.

The report goes on to state that when developers are asked about which approach is used they often say 'I'm using Scrum, but...' This is because most customise and adapt Agile practices such as Scrum to fit with their organisation's existing processes and to fit the type of projects they are working on.

This matches our experience where the reality is a mixture of traditional and Agile approaches. For example, an upfront piece of work on analysis and requirements, development using Agile and a final phase with customer trials and a launch. The term that often gets used to describe this is 'Water-Scrum-fall'.

The Agile Alliance and the Agile Manifesto

The Agile Alliance is a non-profit organisation committed to advancing Agile development practices. It operates on the principles of the Agile Manifesto which were pulled together by 17 leading software developers at a Ski Resort in Utah, USA in Feb 2001 (see www.agilealliance.org).

If you read through the manifesto and particularly the 12 principles that underpin it, you will get a pretty good idea of what Agile software development is all about.

The Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

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

The 12 principles behind the Agile Manifesto are listed below:

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

Scrum

The rules and its impact on the product owner

Scrum is a project management framework. As the most popular and widely-used Agile approach it has become synonymous with Agile software development.

Scrum does not dictate how everything should be done on a project but leaves many details to the development team. However it does have a set of defined roles, terminology and tools. The following section describes a pure Scrum implementation. In practice this is adapted to suit the organisation, project and development team involved.

Scrum explained

Scrum is based on a series of short, focused development projects called sprints. These are time-boxed which means they must deliver by a set date. If necessary the scope is reduced to deliver on time. The deliverable from each sprint should ideally be something that a customer could use, i.e. a potential shippable product with incremental functionality.

Fig. 3 Scrum is named after the close-knit shoulder-to-shoulder formation used in the sport of rugby

There are 3 defined roles in the delivery of Scrum projects, the product owner, the Scrum master and the team.

The product owner is the voice of the customer with responsibility for ensuring the final product delivers customer value. Product owners compile and prioritise product requirements to create a product backlog. This creates a list of things to be developed ordered by their priority. Scrum is designed to cope with change so this is an ongoing and iterative process. As new requirements come along or the focus shifts, the product owner can re-prioritise, which is known as 'grooming the backlog'.

The Scrum master champions and facilitates the Scrum process, coaching the team and tackling any impediments, (i.e. blocking issues) that get in the way of product delivery.

The team is a cross-functional group of, typically, 5 to 9 people who deliver the product (effort estimation, analysis, design, development, testing, documentation and progress tracking). Purists argue the team should be self-organised but in practice we've seen the need for experienced members to provide insight into estimation and work allocation.

Before the start of a sprint, a sprint planning meeting is held to decide on the requirements that will be included for development – these are known as the sprint backlog. The team then commits to delivering these within the sprint cycle.

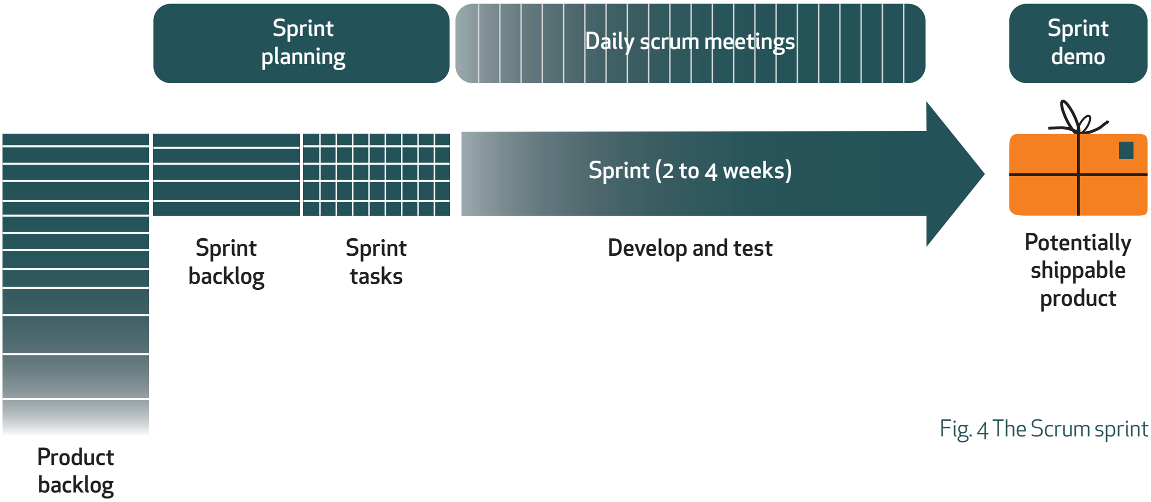


Fig. 4 The Scrum sprint

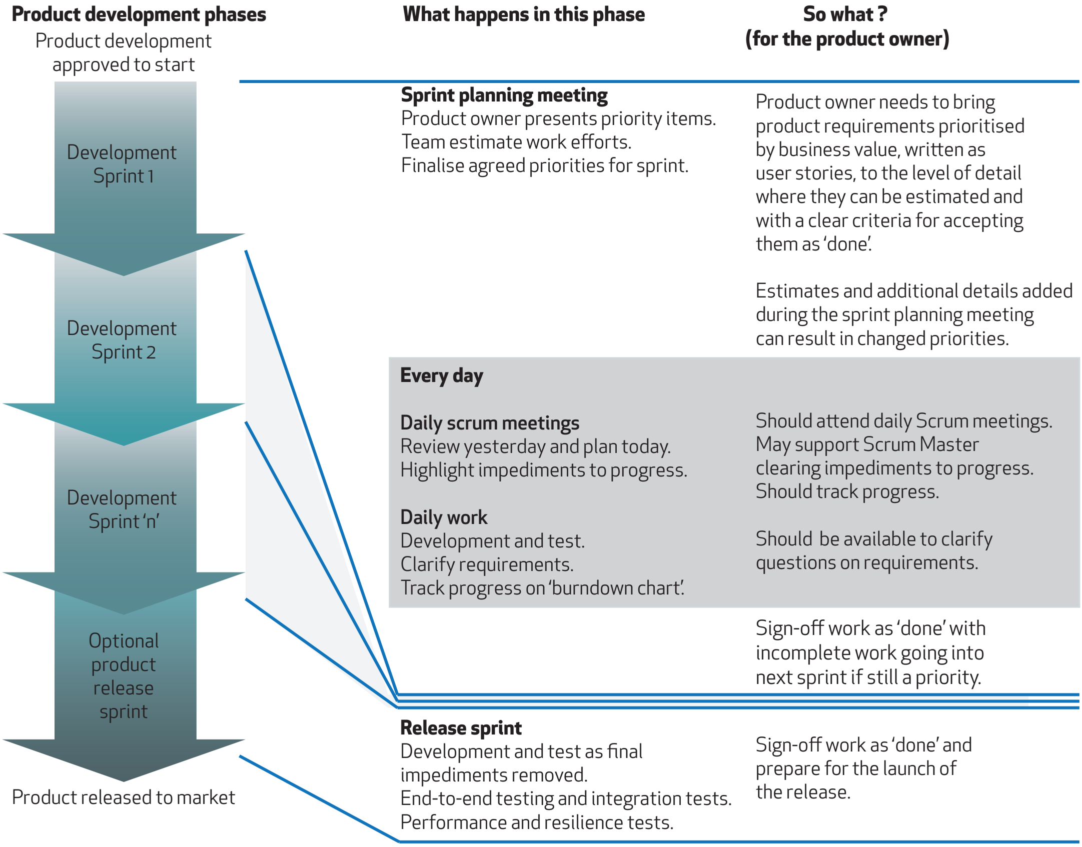
Every day the Scrum master runs a short (e.g. 15 min) daily Scrum meeting. This is to share the status of what's being developed and identify any potential issues. To keep them short they are often run as stand-up meetings where everyone stays standing for the duration. Attendees should be able to answer the questions. What have you done since the last meeting?, What will you do between now and the next meeting and is there anything that will stop you from doing what you have planned?

Once the sprint is finished a sprint demo is organised to show the various stakeholders what has been delivered and get sign-off from the product owner. Following this a sprint retrospective meeting is held to review how things went, whether the forecasts for work completion were accurate and to identify any other issues. This is used to improve performance in future sprints.

The product owner and Scrum

A product manager is often assigned the role of product owner and so in the rest of this article we describe the impact of Scrum on the product owner.

A key role for the product owner is to present the priorities for each sprint. This is done by constantly managing the product backlog and prioritising based on customer insight and development estimates.


Product development phases
Product development approved to start
Development Sprint 1
Development Sprint 2
Development Sprint 'n'
Optional product release sprint
Product released to market

What happens in this phase	So what ? (for the product owner)
Sprint planning meeting Product owner presents priority items. Team estimate work efforts. Finalise agreed priorities for sprint.	Product owner needs to bring product requirements prioritised by business value, written as user stories, to the level of detail where they can be estimated and with a clear criteria for accepting them as 'done'. Estimates and additional details added during the sprint planning meeting can result in changed priorities.
Every day	
Daily scrum meetings Review yesterday and plan today. Highlight impediments to progress.	Should attend daily Scrum meetings. May support Scrum Master clearing impediments to progress. Should track progress.
Daily work Development and test. Clarify requirements. Track progress on 'burndown chart'.	Should be available to clarify questions on requirements.
	Sign-off work as 'done' with incomplete work going into next sprint if still a priority.
Release sprint Development and test as final impediments removed. End-to-end testing and integration tests. Performance and resilience tests.	Sign-off work as 'done' and prepare for the launch of the release.

Fig. 5 The product owner and Scrum

Writing requirements

Requirements are documented in a variety of ways depending on their priority. To start with all requirements are described at a high level, termed an Epic, which describes the theme for particular functionality. For an email application this might be to send, forward and reply to emails, file and sort emails and use contacts. Those requirements of medium priority are then broken down into 3 to 5 key features to give sufficient insight to the development team so that they understand where the development might be heading.

The highest priority requirements, i.e. to be built in the next sprint, are described in far more detail. For these, rather than write a long abstract document, the product owner is encouraged to write user stories. These are short, simple descriptions of a feature told from the perspective of the person that wants it (typically a user). As a *<type of user>*, I want to *<some goal>* so that *<some reason>*. For example, as a user of email, I want to change my password so that I can maintain the security of my account. They are typically written on sticky notes or small cards.

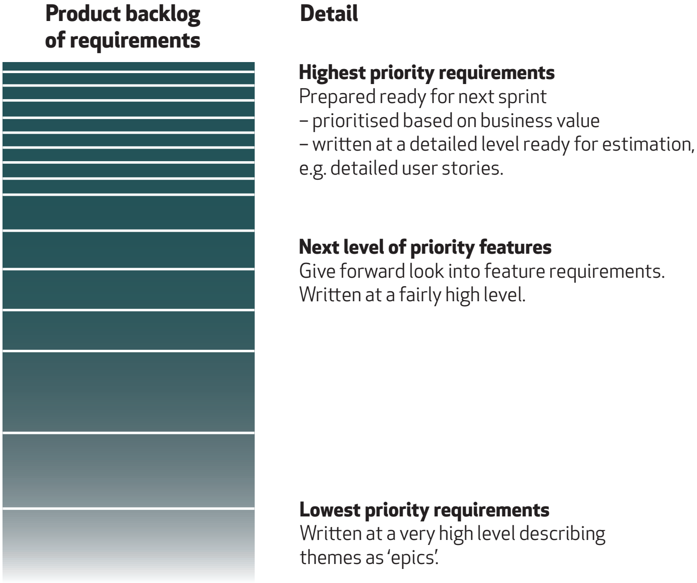

Product backlog of requirements
Detail
Highest priority requirements
Prepared ready for next sprint
– prioritised based on business value
– written at a detailed level ready for estimation, e.g. detailed user stories.
Next level of priority features
Give forward look into feature requirements. Written at a fairly high level.
Lowest priority requirements
Written at a very high level describing themes as 'epics'.

Fig. 6 The product backlog

These sprint backlog requirements should also be written so as to:

- Minimise dependency on other requirements, enabling them to be developed in any order (and to minimise problems if some do not get developed at all)
- Be valuable to an end user
- Be small enough to fit into a sprint
- Be clear enough to be estimated
- Be testable.

Wise Words
"One should always play fair when one has the winning cards."
Oscar Wilde, Irish poet and playwright

It's important to write requirements at a 'product level' so that all aspects of the product necessary to support the features are built before it can be confirmed as finished. For example, a user story describing how a user must be able to change their password needs to drive the development team to create the user experience, the user help screens, the database interaction, the email confirmation of the password change etc. Writing at this level makes it far easier for the product owner to prioritise requirements.

While the product owner is responsible for presenting their priorities during the sprint planning meeting, they don't, at that stage, know the effort required to build the feature. The sprint planning meeting provides that insight and the product owner will choose to change priorities if, for example, 2 requirements have identical business value but 1 takes twice as much effort to build.

Tracking and estimation

One of the most common things that is used in Scrum is a task board to manage and track the status of work in a Sprint (see photo on p. 19). Typically a row is allocated to each user story with columns representing To Do, Underway, To Verify and Done.

Estimation in Scrum is simplified by only evaluating development tasks relative to each other. This is done by awarding points for the size of each task. Scrum doesn't use man-days estimates. As each sprint is the same length of time, the team learns how many total points can be delivered in a single sprint. This is known as the team's *velocity*. The velocity of development is tracked using a burndown chart. This is updated regularly as new stories are

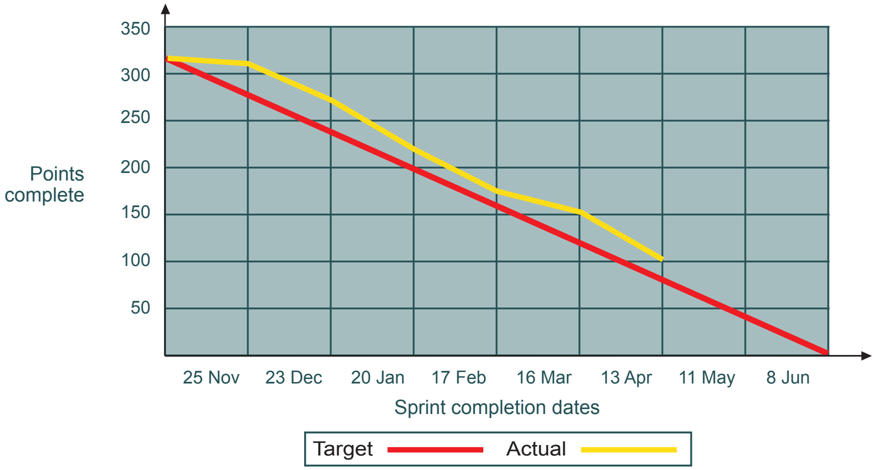


Fig. 7 Example release burndown chart with 4-week sprints

added or stories are completed. A burndown chart records user stories completed and shows the team if they are on-track to deliver the volume of work previously forecast.

Scaling Scrum

Scrum is great for standalone development projects where goals are changing and the scale can be handled by a single team. However many projects have a number of teams working in different locations, on multiple systems or on fundamental architecture changes. This is a common challenge in large projects and the impact is to drive hybrid development models such as the one shown in Fig. 8. and methodologies like the Scaled Agile Framework. As the product lead you need to co-ordinate with your colleagues and across many development teams. The challenge is maintaining consistency of priorities, aligning user stories and scheduling inter-dependent developments.

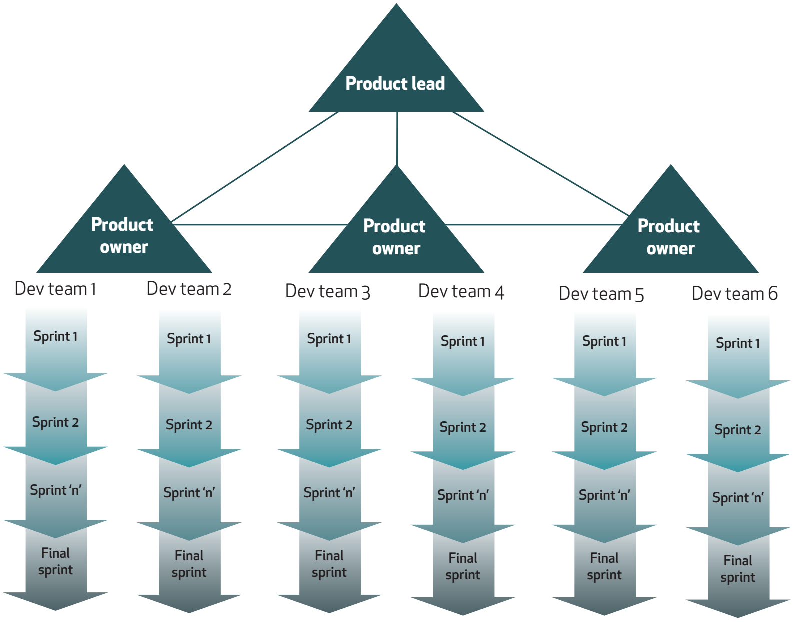


Fig. 8 Managing multiple development teams with Scrum

Greater responsibility

As the product owner you set the development priorities through the product backlog. The Agile approach means that you can change direction to respond to market and competitor moves or as the understanding of what's needed evolves. This places a heavy responsibility on your shoulders as, more than ever, the success of the product is down to you.

Checklist

Survive and prosper with Agile

If you're a product manager wondering how to get the most from Agile and Scrum we've compiled this list of 5 key areas to help you understand what you should be focusing on.

1. Product manager vs product owner?

One of the key aspects of Agile is working very closely with someone who understands what the customers want for the product. The role is critical to drive what is developed. In Scrum this role is called the *product owner* and involves prioritising the backlog of features, answering day-to-day questions, providing sign-offs and feedback. Done properly it takes a large commitment of effort and time.

Product management is set up by companies in different ways. However a fundamental part of the role is gathering customer requirements and defining the products that the company sells. So product managers are ideally placed to be product owners. The challenge is that most product managers have a range of other responsibilities that keep them very busy, e.g. in-life product management, product marketing, supporting sales.

In our view, as a product manager you must not give up control of what is being developed for your roadmap and must find a way of taking control of the product owner role.

One way is to have one or more *backlog managers*. These may be business analysts or team leaders who take on the product owner role for the duration of the project. The key is that they have at least a dotted line reporting into you so you're involved in the prioritisation and sign-off of work.

Another option is to split the product management role so that one person takes on the inbound activities of working with the Agile team to define and deliver the product and someone else takes on the outbound role of helping the business market the product.

> "I think the Scrum term product owner is a misnomer and worry at the 'scope creep' that seems to be promoted by some Scrum advocates that gives more and more product responsibilities to this project-based role, e.g. budget, strategy. To my mind, the product manager owns the product and I much prefer the term backlog manager for the Scrum role."
>
> **Saeed Khan, Industry expert and founder of Transformation Labs**

Whatever happens you need to avoid taking on the full product owner role without first getting your job definition and resourcing sorted out.

2. Manage your stakeholders

Development may be releasing new versions of the software every 4 weeks but marketing won't want to launch a new product each month and most customers won't want to receive and deploy it.

In product management we have roadmaps for good reasons. When selling to businesses the sales process can take a long time so it's important to start discussions early about when products or features will become available. Another reason is that we often need to plan in advance to meet specific launch windows such as the Christmas rush or a major industry event. And finally our organisations, suppliers and customers all need to be able to plan ahead to work efficiently. So as product managers we like to plan the future and be in control.... smooth releases with a full set of features, delivered on time to an expectant market.

With the iterative and evolutionary nature of Agile you might ask "How can I market a product when I have only a limited view of what will be in it or when it will be available?"

The trick is the careful management of what's being delivered and communicated to your organisation and customers. You need to control what is released, when to release it and when to talk about it. Setting the right expectations and ongoing stakeholder management are critical.

3. Find customers to work with

The beauty of Agile is how changes in product development direction can be easily accommodated by re-prioritising the product backlog for the next iteration. If at all possible you should be setting up an early-adopter programme to recruit customers who are keen to get their hands on new features in return for providing feedback.

"Should you assume the product owner role as a product manager? It's a great question. Although as a product manager you often have the knowledge, experience and leadership skills – you probably don't have the time! However the alternative can be someone too junior (who needs lots of hand-holding), or someone from Development who has little understanding of your customers and an undoubted conflict of interest."

Derek Britton, Independent Product Management professional and Agile commentator

"Agile gives product managers great opportunities to offer early-adopter versions to customers and get good, instant feedback – prototyping and user validation help make great products."

Derek Britton

It can be a challenge finding customers to work with so be careful to make sure the customers you choose are representative of the market as a whole.

4. Build credibility with development

Whether you're in daily contact with the development teams as the *product owner* or working with a *backlog manager* you need to build your credibility with development. Development needs to have faith that you understand the market, have thought through the product direction and can lead the prioritisation of requirements, otherwise they might just go off and do their own thing.

You can do this by making sure you've clearly communicated your vision for the product, responding to questions in a timely fashion and by making yourself aware of the main development issues and any potential impacts.

"In my experience, one of the biggest challenges implementing Agile is the cultural change. It's not that hard to change processes, tools or even skills – but changing hearts and minds can be a huge challenge."

Derek Britton, Independent Product Management professional and Agile commentator

5. Embrace Agile to get the benefits

Agile does bring huge benefits to software development and as a product manager you can reap the benefits of an Agile approach if you strive to make it work. Remember that as a product manager who is also a product owner, Agile holds you to account! You have the freedom to direct the development teams, respond to changing markets and customer requirements – there is no excuse for launching with a product the market doesn't want.

However for Agile to be successful the whole organisation needs to embrace it. This takes training, process development and ongoing support. So, as a product manager, don't be afraid to ask for some Agile training.

Tools

Software to help you manage requirements and projects

Many companies using Agile and Scrum rely on dedicated 'War Rooms' with notes stuck to the wall to view the sprint tasks and track progress. But what if you're trying to track the work of multiple teams in multiple locations or as a product manager, how do you manage and share your requirements backlog?

There are a large number of software tools available to help with Agile and Scrum processes. Some focus on managing the whole project, others focus on requirements, however most have functionality that is useful for the product manager/product owner.

You can download our free report on the top 10 product management tools from the resources section of our website. There are tools specifically for generating ideas and others specifically for prioritising requirements that help in the early stages of the roadmapping process. Some of them can take you all the way through from creating ideas to generating your roadmap.

We've no affiliation with the top 10 vendors below and most offer a free trial so you can check them out for yourself.

product focus TOP 10

PRODUCT MANAGEMENT TOOLS

Accompa Aha! Confluence Hansoft ProdPad

JIRA ProductPlan RALLY roadmunk StoriesOnBoard

Scrum this!

From the Cranky Product Manager's blog

Fig. 9 The Cranky Product Manager

There are some excellent product management blogs around and one of the best is from the Cranky Product Manager.

The Cranky Product Manager claims to be the fictional, snarky alter-ego of a mild-mannered software product management professional. She (and we think it's a she) publishes anonymously which is just as well given the responses she got from certain members of the Agile community when she first posted the blog entry below. Reprinted with kind permission – enjoy.

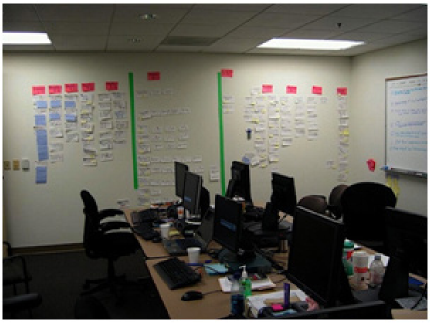
Hey You! Mr. Release Manager!

The Cranky Product Manager appreciates that you're trying to do this Agile Scrum thing by the book. And that it is hard for you. Because before this Agile tsunami came crashing down you mainly just tracked the progression of different release documents (Is the PRD done? Check. Is the Functional Spec done? Check. Is the Design Doc done? Check.)

OK, that's not fair of the Cranky Product Manager. You did more than that. You also ran hurried release meetings once a week that tried to bury issues instead of surface them (OK everyone, here's the status of all the documents. Anyone have any issues? None? OK, let's adjourn.) You also organised two or three Fantasy Cricket leagues plus your wedding during working hours, and boy it all took a lot of time.

But now, in the Scrum era, things are different. It' s not easy. You once had a private office, but you now spend the bulk of your day tethered to a communal table in a stifling hot "War Room," inhaling the body odour of The Veteran software engineer trying to tune-out the grandstanding arguments between two nimrod hotshot coders ("My idea is the most elegant...", "No it's not. It's trivial. You'd have to refactor it immediately."), and listening to the documentation writer bitch and moan that she can't write the doc by Friday if the product keeps changing every hour. It's really hard to organise fantasy leagues or surf the web with so little privacy. Plus the porn shui of the War Room is completely off.

So it sucks to be you, Mr. Release Manager, and the CPM is sorry for you.



But just because you are stuck in that War Room doesn't mean the Cranky Product Manager should have to join you. You argue that in Scrum the product manager is the same as the Product Owner, and therefore the Cranky Product Manager needs to be constantly available to the team in order to make on-the-spot decisions within minutes of the asking. Ergo, you demand the Cranky Product Manager sit in that sticky-note-encrusted, windowless tomb with you all damn day.

Uh, no way. Not gonna happen.

Why not? Because the Cranky Product Manager needs to be the Voice of the Customer and the Voice of the Market. How is she to do that without actually VISITING some customers and prospects? And VISITING means that she actually needs to leave the office, hop on airplanes, and fly far, far away. She cannot answer questions from the dev team within 5 minutes if she's on a plane, or in a meeting, or on the phone with a customer. Not that the CPM wouldn't LOVE to hear debates about Iron Man or whether that Star Wars cartoon is "canon" or not – all day, every day, for hours on end. Who wouldn't?

Wise Words
"Do not condemn the judgement of another because it differs from your own. You may both be wrong."
Dandamis, philosopher

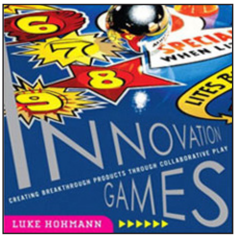
And your response, Mr. Release Manager? You argued that perhaps the Cranky Product Manager should not visit so many customers and should spend more time in the War Room.

Anyone else see the irony? The Voice of the Customer should have less interaction with customers? All so she can make on-the-spot customer-facing decisions more quickly?

The Review

Reviews and feedback for product managers

Book Review



Innovation Games
by Luke Hohmann
(Addison Wesley, 2006)

Whatever development process you are using, discovering customer requirements is a key part of the product management role. Most of us take the standard approach of customer meetings, market research and talking to sales and support. However getting real customer insights is usually a challenge.

Innovation games are a fun and effective way of working with customers to understand their needs, wants and desires. They are an alternative to simplistic questioning which tends to influence the discussion, or costly market research where an agency interprets the responses for you. Customers often have a hard time articulating their problems or envisioning solutions and these games draw out insights and nuances that can be extremely valuable.

The book's tagline summarises its goal of creating breakthrough products through collaborative play. It contains 12 different games with a description of each, why it works, how to prepare for it and how to process the results.

One of the games is called Product Box. You ask customers to imagine they are selling your product at a trade show. You give them some blank cardboard boxes, pens and coloured paper and ask them to design the box packaging. The box might show marketing slogans, features, pricing or pictures. When they're finished they have to present the box back to the room and persuade the other players of the features and benefits. This gives a way for customers to tap into their latent preferences and requirements and express them as they sell your product back to you.

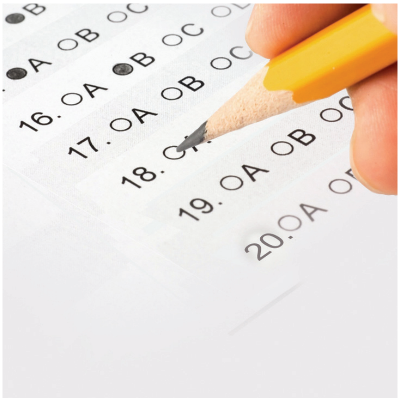
Of course, one of the potential drawbacks of these games is getting hold of the customers to participate. And the next thing to consider is whether you feel confident to run the innovation games yourself or want to bring in some outside consultants to help. However once you've addressed these issues, the games are a low risk and engaging way of getting high quality insights into your customers' requirements.

Wise Words

"You'll know a game is going really well when your customers don't want to stop.... it's this deep level of engagement that gets past any barriers to communications and produces the most honest and useful feedback."

Luke Hohmann

Have your view



In September we did a survey to ask about which topics you would like to see covered in future journals. We got 130 responses and nearly 130 different answers!

We think this shows just how varied the product management and product marketing challenges are in our industry.

After some more detailed analysis, the top 10 topics are listed below. Let us know if you have a favourite by emailing **journal@productfocus.com**.

1. In-life product management (including customer retention)
2. Effective approaches to competitive research
3. Product strategy and managing a portfolio of products
4. Product managing services and solutions (as compared to products)
5. How to drive product innovation
6. Persuading senior execs of the value of product management
7. The impact of social media on product management
8. Effective approaches to roadmapping
9. Getting and validating customer requirements
10. How to effectively manage stakeholders, meetings, workshops and your time

Annual Survey

We use our survey to benchmark product management each year. Let us know if you'd like to take part. You can **download the latest results** from our website.

AREN'T YOU OFF TO YOUR FIRST SCRUM STAND-UP MEETING?
TO DO
○ ROADMAP
○ BUSINESS CASE
○ CUSTOMER MEETING
○ PRESENTATION
A MEETING
SITION..
SCRUM
WWW.PRODUCT FOCUS.COM
NY STATE
SEX BOMB
© COPYRIGHT PRODUCT FOCUS 2011
SPOON-FEED DEVELOPMENT, AVAILABLE WHENEVER THEY NEED ME.... HOW WILL I GET ANYTHING ELSE DONE?

The Insight

The dangers of Agile hype

Over the past few years lots of people have been cashing in on the success of Agile and Scrum. Some have even reached cult status so that people say *"I was trained by xxxx one of the founders/gurus".* So is there an Agile/Scrum bandwagon ... an overstating of the benefits?

Hard facts on the success of Agile are thin on the ground. When someone sponsors its introduction their reputation is *on-the-line*. No one wants to talk about failure after spending large amounts of money on training and so much upheaval.

So, is Agile better than Waterfall? In our opinion, **yes** when it comes to the iterative development of software features and online software, **no** when it comes to large, complex, multi-element solutions. Do Agile and Scrum give a clear framework for development and continuous improvement – **yes**, but so do other systems. Does prototyping and a series of short Waterfall-type projects get you many of the benefits of Agile – well quite possibly, **yes**.

That leads to the reality in most organisations, which is to use whatever approach seems most appropriate for the project in hand– be it Agile, Waterfall or a mixture of the two.

But from a product management perspective it doesn't matter!

Whatever development approach you use, early customer feedback from prototyping or an Agile iteration really helps but you've still got to understand, prioritise and get the requirements right.

And whatever development approach you use, it still comes down to the quality of the people. Good people will do good work whatever process they're using.

And after you've developed the perfect product you still have to launch and market it. If no one knows your product exists or understands what it can do, it isn't going to be successful.

Don't get us wrong – we think Agile is a great thing, but it's not the only thing. **Don't get sucked into the hype, dogma and sometimes even religious-like zealotry ... Scrum is not the one and only true way.**

Wise Words

"If you hype something and it succeeds, you're a genius... if you hype it and it fails, then it was just hype."

Neil Bogart, American record executive

Product Management Leadership Forum

1-day workshop
for product management leaders

Learn how to build and run a world class product management function

Discuss issues with peers and industry experts

Get unique tools, insights and knowledge

product
focus

+44 (0) 207 099 5567
+1 (877) 290-6240
www.productfocus.com

Helping you and your business deliver world class product management

product

focus

+44 (0) 207 099 5567
+1 (877) 290-6240
www.productfocus.com