



DIGITAL TRANSFORMS PHYSICAL

10 Success Factors of Future-proof Requirements Management



Introduction

Requirements Management has come a long way. The first Requirements Management (RM) document may well have started out as a rough product concept sketched on to post-it notes. By today, RM has evolved into a set of cutting-edge tools and best practices that enable product development teams to efficiently build quality software.

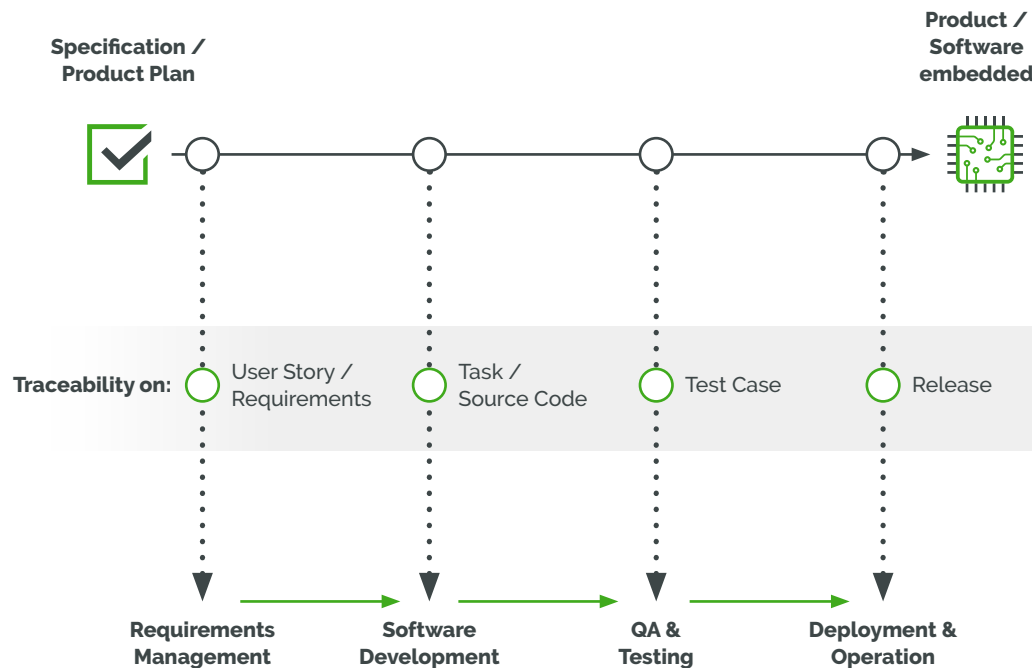
Yet a software project that perfectly fits customer needs and is delivered on time and at the planned cost level is still a rare sight. As the market environment we build products for is getting more and more competitive, developers are starting to realize the huge effect that requirements management has on their bottom line. A growing number of companies today understand that future-proofing their requirements management processes and toolchains is a crucial business need.

In this eBook you will find the 10 key success factors that contribute to a well-functioning requirements management lifecycle. Analyze your current RM processes and toolset against this checklist to see if you're ready for future challenges in developing software for a fast-paced, IoT-connected market.



01 /

Traceability



Heterogeneous toolchains are a basic trait of development environments that have evolved organically over the years. Along with the growing complexity of their products, the needs of dev teams have expanded over time, to which they responded by adding new elements to their toolbox. These new tools then had to be manually integrated with the existing toolchain, leading to a patchwork-like pattern of development tools. The result? Limited traceability, difficulties of revision control, and unnecessarily high maintenance costs.

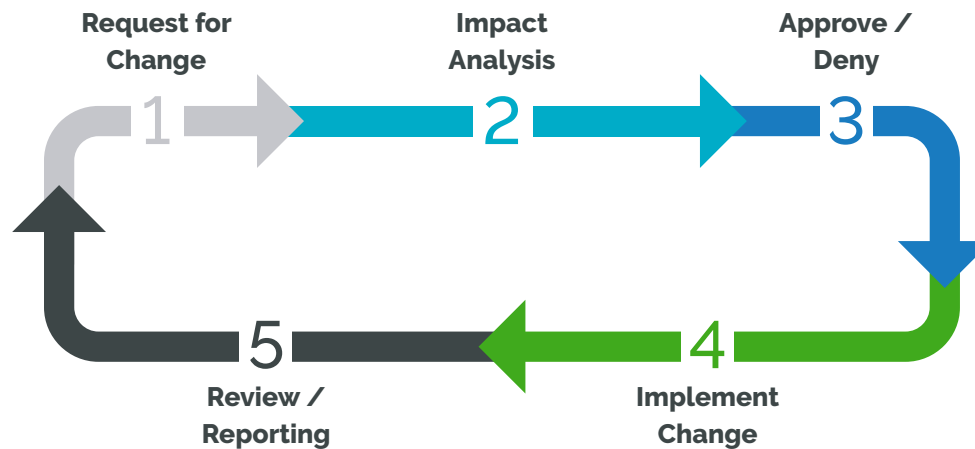
Any modern development team that has all its requirements covered by single-point solutions will have seen the difficulty of navigating such a complicated environment. As some teams are still struggling to ensure traceability across the lifecycle, the ability to trace work items from requirements down to source code, test case, and implemented feature is becoming a key enabler of productivity and quality.

Providing your teams with a scalable lifecycle-wide development platform helps ensure traceability, and positively impacts product quality, delivery times, and the transparency of development processes. All this is becoming crucial as technology increases the complexity of products (and thus, that of their development lifecycles) at a never before seen pace.

02 / Change & Configuration Management

With development lifecycles becoming more collaborative, iterative (Agile), and focused on responding to changing customer needs, change management is becoming a serious challenge. Legacy requirements management methods won't be able to support this accelerating lifecycle, as having stakeholders from various engineering disciplines edit the same set of requirements documents will eventually lead to conflicting versions.

Tracing these changes, as well as analyzing how they impact product design and quality is vital. Once you have traceability in place, creating a formal plan on implementing changes is necessary. Adequate configuration management, therefore, is becoming a key aspect of future-proof requirements management. Tracking changes enables you team to roll back mistakes, and to link code fixes or updates to defects easily.



“

An efficient change process can cut up to 33% off the typical product development cycle time.”

Source: McKinsey and Company

03 / Collaboration with all stakeholders

In a fast-paced, highly integrated, Agile development environment, providing an efficient way for all stakeholders to collaborate is about as fundamental as, say, having electricity in the building. There is simply no successful development project without it. The importance of efficient collaboration is likely to grow in the future as walls between development disciplines continue to be torn down.

When capturing and defining requirements, it's imperative that requirements engineers, but also testing & QA people can work with end users (customers) to really understand the required functionality. The requirements verification process also benefits from all disciplines being able to contribute.

With DevOps gaining traction for its many recognized benefits, the convergence and more holistic view of software development is taken to a new level. Both Agile and DevOps require disciplines to collaborate in order to build bridges across development silos throughout the lifecycle.

As a key success factor of future-proof requirements management, collaboration is as much a technical challenge as a cultural shift. It is a key management responsibility to enable and encourage the efficient cooperation of all stakeholders by adopting new processes and tools.

- 1. Satisfy the customer through early and continuous delivery**
- 2. Welcome changing requirements**
- 3. Business people and developers work together daily**
- 4. The best way to convey information is face-to-face**
- 5. The team regularly tunes and adjusts its behavior**

Source: Principles behind the Agile Manifesto

Blog post

The Top 5 Challenges of Software Requirements Management

[Read it](#)

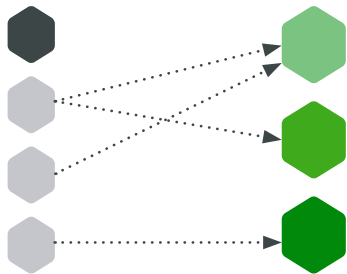
04 / Efficient re-use of assets

Rather than radical jumps ahead, much of the recent development in how we build software today aims to enhance the efficiency of processes. Reusing is hardly a revolutionary idea, but shaving a few percents off development time and costs really goes a long way in boosting your profitability.

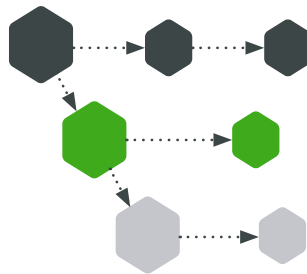
As software products are becoming more and more sophisticated, providing your developers with access to project history, and making it easy for them to leverage existing data, common assets, and shared processes can prove to be a game-changing advantage. Not to mention that reusing tested code that has already proven to be reliable contributes to the robustness of your products' quality.

When building a product line of multiple product variants, you might be able to recycle requirements, but also chunks of source code, test cases, processes, and other assets. In a scenario where complex processes are used (in safety-critical development for instance, where compliance requirements might call for the use of elaborate development practices), re-using project templates with tried and tested processes can result in significant savings.

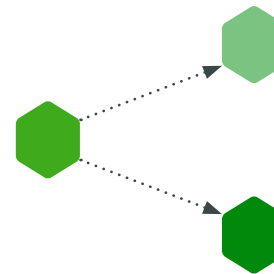
Requirements Library



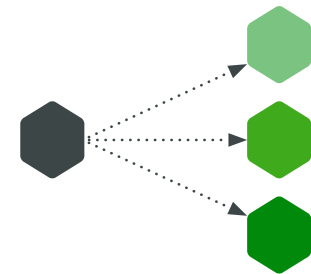
Branching - Versions



Product Variants



Project Templates



05 /

Migration & integration of different tools

Working on a central platform to provide a single source of truth for various disciplines across the lifecycle is highly beneficial. But proprietary development scenarios don't always allow this simplification of the tool ecosystem.

Overlaps and gaps in the functionality of single-point software, as well as difficulties of communication between tools contribute to tool friction. In such cases, the migration of data and the integration of processes across tools can help overcome the limitations of a fractured tool environment.

Successful requirements management and development teams work with tools that either offer built-in integrations, or allow them to simply create interfaces to enable the undisturbed flow of data and work processes across the lifecycle.

Development teams stuck with legacy suites such as IBM® Rational® DOORS®, Next Generation, or HPE ALM will need to reevaluate whether their current tooling fits their strategic purposes and scaling needs.

They might need to find solutions to migrate or integrate elements of their current product suite into a new development environment.



“Organizations tend to have a mix of tools, and as a result of the tool friction, there is a difficulty in the traceability. So we see a trend towards integrated holistic solutions that allow that traceability across the lifecycle.”

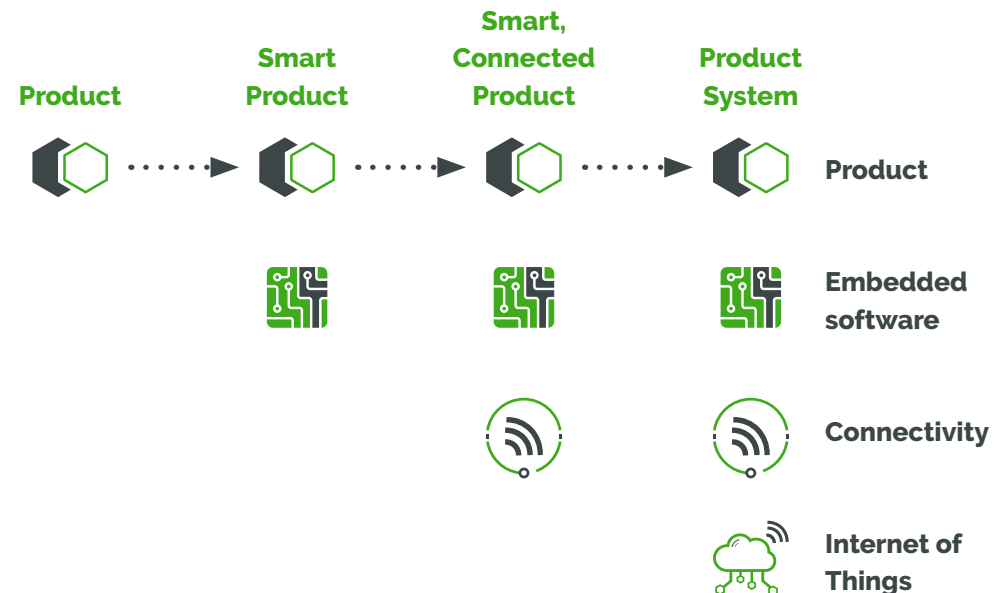
— Michael Azoff, Principal Analyst, Ovum

06 / IoT Requirements Management

Managing requirements can be a challenge in the case of Internet of Things-ready (embedded) products, where deliverables include hardware, software, and service components. Without the orchestrated management of separate development streams all using their own tools, disarray is inevitable at the confluence of hardware engineering, software development, and service innovation processes.

To alleviate resulting issues, applying a holistic view of the product lifecycle is becoming necessary. Getting your teams to start thinking in terms of a single product lifecycle can be a difficult management challenge. Provide them with a central platform that helps them see and manage this overarching lifecycle of parallel development processes.

Such an integrated approach can also contribute to the safety, reliability, and security of these IoT-connected embedded products, supporting current or future compliance needs and bolstering trust in your products.



07 / Agile Requirements Management

Requirements are at the very root of the transition process to Agile, now a defining trend even in safety-critical industries, where the adoption of Agile seemed impossible just a few years ago.

- **94% using Agile processes in some parts of the development lifecycle**
- **33% uses Agile throughout the entire lifecycle**

An Agile Requirements Management process is based on user stories developed in collaboration with all stakeholders (including end users and, ideally, representatives of all engineering disciplines across the development lifecycle). These user stories capture the required functionality from the user's perspective often in their own words, and change flexibly as feedback is gathered during the iterative and incremental process of development.

Implementing Agile is really a change in culture and processes. However, as most teams having made the transition will testify, Agile adoption can be made easier with the right tools at hand. With adequate tool support throughout the lifecycle, your team members won't experience as much of a culture shock as Agile is rolled out, enabling you to realize value from the transition sooner.

Key aspects when choosing a future-proof RM tool include smooth collaboration, automated change control, and automatically recorded artifact trails for lifecycle-wide traceability. Some other capabilities and features to consider are scalability, Kanban boards, DevOps support, and sophisticated release planning functionality, all of which are expected to become more and more important as release cycles accelerate and grow in complexity.

Use of Agile Management Tools:
Fewer respondents cited using Microsoft® Excel this year (46%) then last year (60%) and Google Docs dropped from (18%) last year to (14%) this year.

Source: 11th State of Agile Report, VersionONE

08 / DevOps

DevOps's ability to increase product quality while driving down costs is getting more and more widely recognized, propelling its adoption worldwide. DevOps works by breaking down the walls between development and operations, and by turning an open-ended development lifecycle (with product release at the end) into an infinite loop of continuous improvement.

As such, DevOps also affects requirements management. At the heart of DevOps is the

idea of the Continuous Delivery pipeline, which enables a perpetual loop of feedback and improvement to accelerate and enhance product delivery. In a DevOps lifecycle, requirements are shaped by feedback from developers, but also by real-time monitoring data from operations. Placing more emphasis on Operations-inspired non-functional requirements, and feeding back Ops monitoring insights to requirements helps build more robust software in shorter release cycles. ►

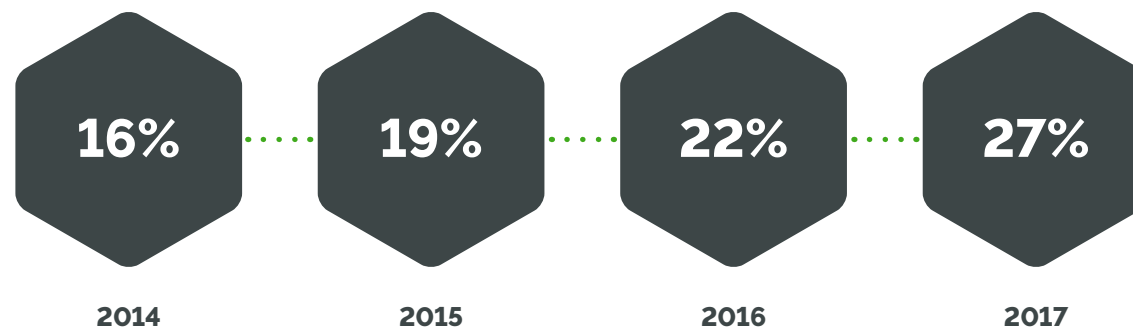
DevOps teams increased from 16% in 2014 to 19% in 2015, to 22% in 2016, to 27% in 2017.

High-performing DevOps teams have:

- **46 times more frequent code deployments**
- **440 times faster lead time from commit to deploy**
- **96 times faster mean time to recover from downtime**
- **5 times lower change failure rate (changes are 1/5 as likely to fail)**

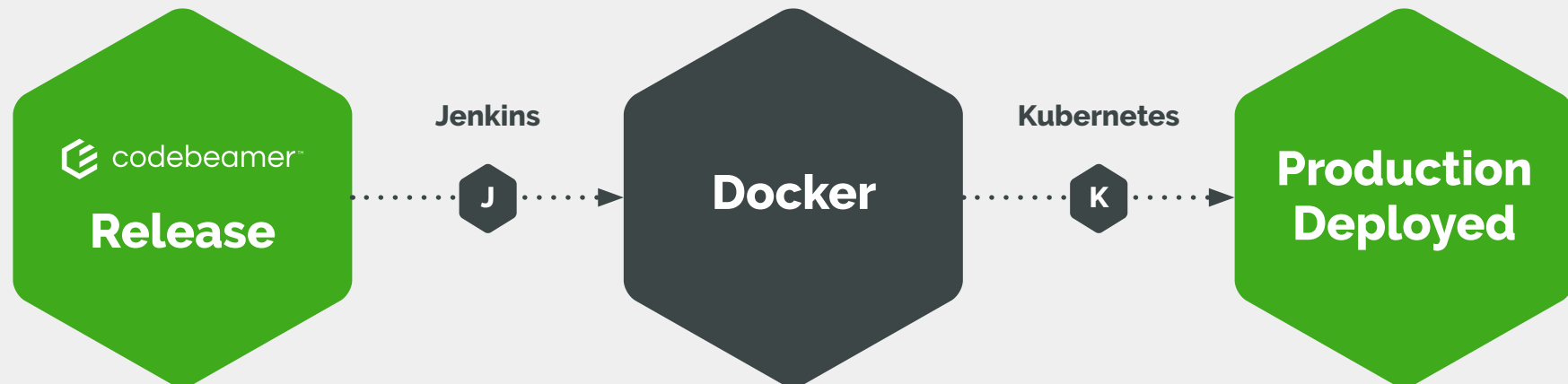
Source: 2017 State of DevOps Report, Puppet

The percentage of people working on DevOps teams increases year by year:



Evidently, these requirements of DevOps further increase the need for a single central management platform that integrates well with software tools used across the lifecycle. To support the automation of QA, testing, and deployment, the integration of tools such as Docker, Jenkins, Kubernetes and others can be expected to increase in importance. Some form of Service Desk functionality will be required to trace bugs throughout the process of fixing or updating requirements. Due to the number of DevOps tools, integration hubs such as Tasktop are also expected to gain traction in the software industry.

Continuous Delivery with PTC's Codebeamer Technology



09 / Compliance



The growing sophistication of technology products and widespread connectivity to the Internet of Things open up new possibilities for product developers, but also bring inherent risks. With the increasing convergence of medical technology and consumer products, the pace of development in autonomous driving technology, and the exchange of more and more of our sensitive personal data over the web, safety and security are becoming major concerns.

Governments and regulatory bodies the world over are designing increasingly rigorous standards to ensure the safety, reliability, and security of our everyday products. As this trend is likely to expand in all industries, compliance with these regulations is a cause of headache for a growing portion of product developers.

To build a future-proof software development process in anticipation of this tightening regulatory environment, you'll need to ensure traceability, process control, automated documentation (including change control), and advanced reporting across the entire lifecycle.

In terms of requirements, you'll have to implement adequate review and validation processes, best supported by collaborative RM tools that let you specify and enforce review rules and workflows. Ensuring (or, ideally, automating) traceability and documentation, and being able to report on all processes from requirements to release will also be vital as continuous compliance is becoming the norm.

10 / Human factors

As release cycles get shorter, products increasingly complex, and collaboration across engineering disciplines more intensive, some level of friction is inevitable. When unmanaged, however, frustration can be destructive to your team's morale and performance. Hiring skilled talent, providing your teams with best-of-breed tools and a central management platform, and ensuring the smooth flow of work across processes and software solutions are all crucial to building a well-functioning development department.

Transparency, accountability, and efficiency are all characteristics of a future-proof development

environment. Efficient work management, communication, and process management support the delivery of quality products with the stress-free involvement of all stakeholders.

What does this mean for requirements engineers and others involved in the definition of product specifications? They may have to take training in "soft skills" such as communication, and will need adequate tooling support. These enable them to crystalize user needs into product specifications, and to accurately communicate with all other stakeholders across the development lifecycle.



See for yourself if development tools really make a difference.

Try PTC's Codebeamer technology for 30 days free of charge.



**Simplify complex product
and software engineering at
scale with Codebeamer.**

Codebeamer is the most Agile Application Lifecycle Management platform on the market. It is scalable, integrated, and supports collaboration while offering powerful features across the entire lifecycle.

Ensure your product meets stakeholders' expectations. Integrate with any requirements management tool. Establish links between artifacts, and manage dependencies to achieve gapless traceability. Manage requirements for multiple product variants in a collaborative manner throughout the entire lifecycle.





DIGITAL TRANSFORMS PHYSICAL

121 Seaport Blvd, Boston, MA 02210 : ptc.com

© 2023, PTC Inc. All rights reserved. Information described herein is furnished for informational use only, is subject to change without notice, and should not be taken as a guarantee, commitment, condition or offer by PTC. PTC, the PTC logo, and all other PTC product names and logos are trademarks or registered trademarks of PTC and/or its subsidiaries in the United States and other countries. All other product or company names are property of their respective owners.

016-10-success-factors-of-future-proof-requirements-management-02-08