

Software: Driving Competitive Advantage for Engineered Products

For manufacturers, software is key to winning customer preference and loyalty. Software simply makes products better.

In the past decade, we've witnessed a profound shift in the way manufacturers deliver value to customers. Manufacturers used to view software as an afterthought, outside of core product functionality. That is no longer the case. The age of the smart, connected product has arrived. Today's manufacturers are transforming everything — from cellphones and automobiles, to toys and televisions — through the power of software.

Consider the automobile. In the past, a brake pedal activated physical systems — first mechanical, then electro-mechanical — that directly brought a car to a halt. Today's drivers interact with a layer of embedded software that sits between drivers and the physical systems that accelerate, steer, or stop a vehicle. Telematics systems are transforming vehicles into communications and entertainment centers on wheels. Through software, manufacturers have improved automotive quality, safety, and functionality — all while delivering greater consumer choice at lower cost.

A similar evolution is underway in aerospace & defense, industrial, medical device, electronics, and high tech industries. Manufacturers are increasingly embracing software to deliver competitive advantage, for very good reasons:

- **Cost:** Software innovation costs much less when you factor in the near zero incremental manufacturing cost of software.
- **Flexibility:** Software, as the name indicates, is more dynamic, allowing manufacturers to change the capability of a product years after it has shipped.
- **Differentiation:** Software enables product personalization. It also allows companies to create more product variants to pursue new markets and customers.



The most successful discrete manufacturers are 30% more likely to use electronics and embedded software to bring innovation to products.”

Aberdeen Group, 2010
Product Analytics to Engineering On Schedule and On Budget



Customers use our products in different ways today. They expect to be able to access information from anywhere. Through the use of software in our products, we are delivering a lot more innovation.”

Tom Fletcher
Vice President, Controls Engineering, Ingersoll Rand

Software Makes the Difference

In Automotive: Automatic, intelligent braking systems on many high-end vehicles react in a potential crash situation even if the driver doesn't.

- Did you know? Cars will require 200-300 million lines of software code in the near future. (Source: Frost & Sullivan)

In Aerospace & Defense: Sensors in unmanned aircraft gather surveillance data for the US military.

- Did you know? The Airbus A380 uses almost 1,000 software components supplied by more than 40 system and software companies located on 3 different continents (Source: SITA)

In Medical Devices: Software in implantable pacemakers monitors cardiac rhythms in heart patients.

- Did you know? In the last 7 years, 500 medical device recalls were related to software defects and malfunctions (Source: FDA Survey)

In Electronics and High Tech: From voice-activated texting, to wireless internet access, to privacy and security features — cell phones (and many other electronic devices) get their intelligence from software.

- Did you know? Embedded software is responsible for time-to-market. Software development now determines the delivery date (Source: Panasonic Corporation)

Chapter 1: The Challenges of Global Software Development

Software represents a huge opportunity for competitive differentiation. But as manufacturers' invest in software-intensive product strategies, challenges abound. Chief among these are the difficulties of ensuring predictable software delivery, managing product quality, and documenting compliance with regulatory requirements or safety standards.

Consider that software development poses a new and different way of developing products and bringing them to market. Instead of a well understood, physical process, software development is empirical, requiring a highly iterative and collaborative approach. Determining the state of release readiness is more difficult when you can't physically inspect software for completeness or quality. Fixing one set of defects often surfaces a new set of issues. Without a holistic view of project progress, it becomes nearly impossible to predict software release dates with accuracy.



◆ How Product Variants Increase Complexity

- Example: A single change request could affect 20 products (or more). The change request cascades from requirements documents which must be updated and authorized, to design, code and test plans which must be modified to validate the new or modified requirement. Without sophisticated branching, this single change could easily result in a 20x increase in effort.

The proliferation of product variants makes quality management especially challenging. Software can be efficiently copied and reused to create low-cost product variants. The resulting propagation of software versions and artifacts, however, only adds to the complexity. While any one defect may be easily fixed by changing a line or two of code, what if that code was used in 10 or 1,000 other product variants?

Which product versions were impacted by the defect? Which test cases and test plans were impacted? How can we validate that the right changes were made? Simple questions, yet the inability to answer them with confidence costs manufacturers millions of dollars each year in customer complaints, product recalls, and delayed product cycles.

A third set of challenges arises around documenting compliance with regulatory mandates. It's been estimated that complying with just one functional safety standard (ISO 26262) has cost the automotive industry hundreds of millions of dollars. Regulatory cost and complexity are exacerbated by the need to manage compliance across multiple regulatory bodies, products, software supply chains, and global markets.

In Chapters 3 and 4, we'll look at several examples of companies that have adopted best practices and engineering solutions to overcome these challenges. But first, let's look briefly at the current state of software engineering solutions.

Chapter 2: Current Application Lifecycle Management Solutions

For more than 20 years, Application Lifecycle Management (ALM) solutions have helped banks, insurance companies, and other organizations coordinate software development activities. Yet these IT-focused solutions were never designed to address the unique complexities of embedded software development in manufacturing environments.

These general-purpose solutions lack capabilities essential to fostering software development excellence in discrete manufacturers:

- Real-time visibility into product quality, compliance, and readiness
- Process flexibility and enforcement to right-size team practices
- Advanced variant management to make reuse practical
- ALM-PLM integration for a complete product view

These four capabilities work synergistically to enable operational excellence for software and product teams. Without real-time visibility into product quality, it's difficult to gauge release readiness. Without process enforcement and flexibility, team processes will be circumvented and data becomes untrustworthy. Without advanced variant management, reuse across product variants becomes impractical. And without ALM-PLM integration, siloed engineering organizations continue to work at cross-purposes.

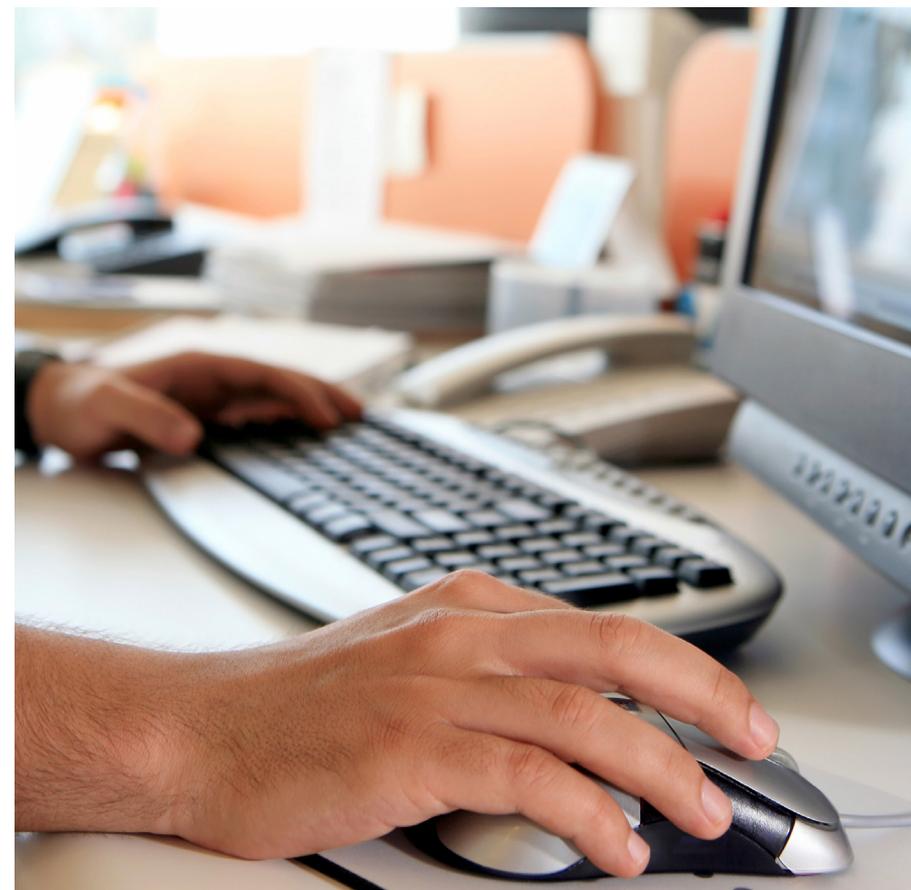
ALM solutions designed for Web and IT environments are poorly suited to software-intensive product development. For example, one IT-focused vendor provides a solution composed of separately acquired products, loosely coupled into a platform. Each product addresses only one aspect of the development cycle, such as requirements management or test management. Reporting across these point solutions requires an overly complex, choreographed sequence of actions – and yet another toolset. Delivery of a complete product view spanning ALM and PLM assets is simply not possible.

Next, let's explore the advantages of a software development platform optimized for the unique needs of organizations that build software-intensive products.



The big advantage of PTC Integrity™ is that you can put all of your effort on solving the business problem instead of configuring the system.”

Chris de Mol
Director Engineering Project Portfolio Management, Tele Atlas NV



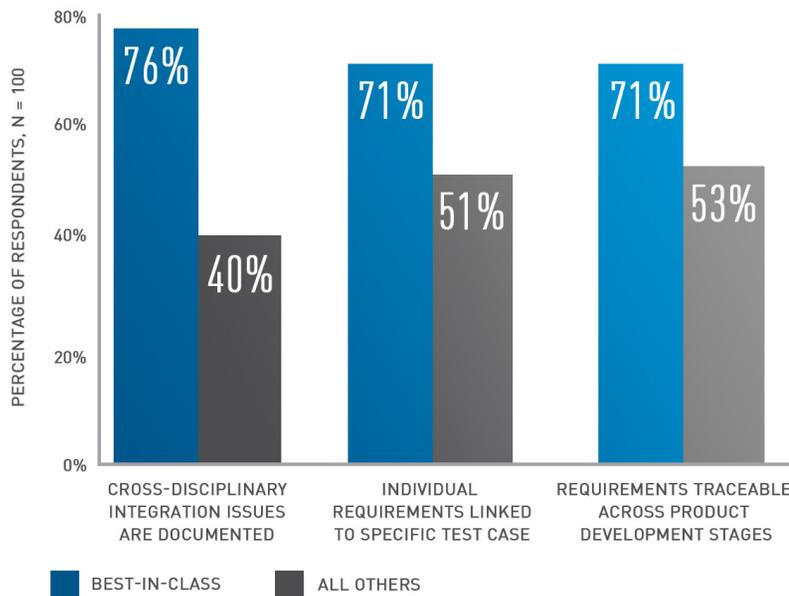
Chapter 3: ALM for Manufacturers

How can companies create a development environment that breaks down the silos between hardware and software teams, and fosters break-through innovation? For leading manufacturers, the answer is clear: by integrating software development into the product development process.

Manufacturing leaders are integrating software development with hardware development, adopting industry-specific best practices across the product development lifecycle — and extending those best practices to partners and external suppliers of product components. These leaders are increasingly relying on engineering solutions optimized for manufacturing environments to help automate and accelerate the end-to-end product lifecycle.

A recent Aberdeen report, “System Design: Get it right the First Time”, compared “best-in-class” (high-performing in terms of meeting revenue projections and product release schedules) companies to all other companies. Figure 1 illustrates how these leaders manage system knowledge.

Figure 1: How Best-in-Class Companies Manage System Knowledge



The PTC Integrity platform was the enabling tool that allowed us to transform the way we do business.”

Mark DePoyster
 Manager, ECU Core Engineering, BWI Group

Best-in-class companies are 36 percent more likely to integrate knowledge and information across disciplines. By linking requirements to specific test cases, these successful companies ensure that software not only works, but it also meets the customer’s requirements. By tracing requirements throughout development stages, change and test management is facilitated and information is readily available for compliance reports.

In order to address the challenges of embedded software development, organizations must adopt an integral solution that will unify processes throughout the product lifecycle and across product lines, improving end-to-end integration and traceability. By doing this, the enterprise can:

- Relate all engineering components
- Reuse requirements, test plans, and software components
- Orchestrate the software development process within core engineering and business processes
- Manage risk and compliance across the software lifecycle
- Provide real-time visibility into release readiness
- Collaborate across engineering disciplines and multiple organizations
- Integrate with existing engineering tools and enterprise applications
- Scale ahead of enterprise needs



Chapter 4: Accelerating Innovation in Action

The best way to ensure that a development solution is comprehensive and integrated is to adopt a single product to manage core engineering artifacts across the software lifecycle.

Rather than cobbling together several different technologies, the solution should manage all aspects of software-intensive product engineering, including requirements management, configuration management, test management, defect tracking, change management, and reporting. This built-in integration between disciplines:

- Provides a “single source of truth” for software driven product engineering, allowing instant visibility into product readiness across the entire product portfolio
- Enables collaboration across globally distributed development organizations and sharing of requirements documents, test plans, and change requests — facilitating workflows, approvals, and impact analysis
- Enables organizations to “right size” their process to team needs, ensuring rapid time-to-value. Packaged templates provide a starting point for companies still defining engineering processes and workflows

- Makes software reuse practical with advanced variant management capabilities
- Breaks down the silos between hardware and software teams by providing a complete product view of team artifacts

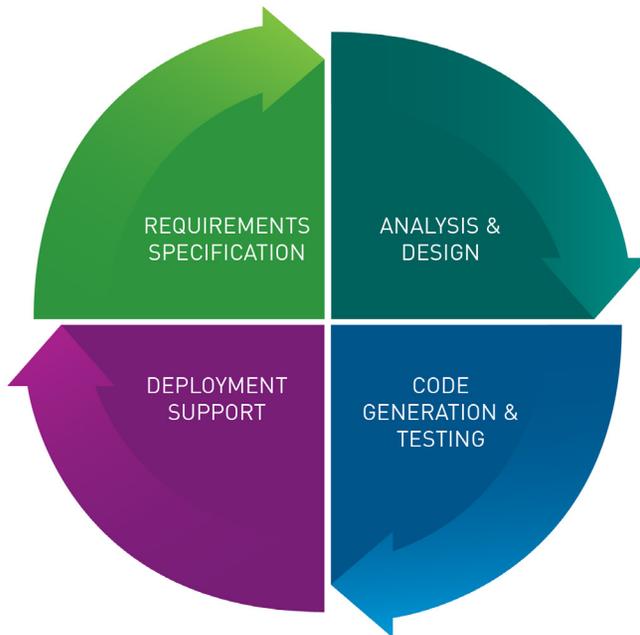
With a single solution that is automated from end-to-end, software development challenges become opportunities. Organizations can:

- Share and reuse requirements, test plans, and software code among multiple products, improving efficiency and accelerating development schedules
- Easily propagate changes and bug fixes to all models affected by a change, improving software quality
- Trace requirements through the development cycle, streamlining compliance management and reporting
- Facilitate collaboration with customers, partners, and suppliers through an open architecture that supports integration of engineering processes with other core business processes
- Improve quality and safety by enforcing process rigor and tracing hard-to-find errors across the product lifecycle

With an end-to-end automated software engineering solution, software development resources can focus on core product development and product innovation, leading to more competitive product lines and accelerated time-to-market. For companies in a range of industries, this type of solution has made a huge impact. Let's look at a few examples.

One of the world's leading automotive companies adopted an automated end-to-end engineering solution to help manage the volume and velocity of engineering change driven from software. In this company, 90 percent of product changes are software-based. By deploying a single solution for the entire product development lifecycle, the company has improved product quality, eliminated costly rework, and simplified regulatory reporting. As an added benefit, requirements and change information is now easily shared throughout the organization and with OEM partners.

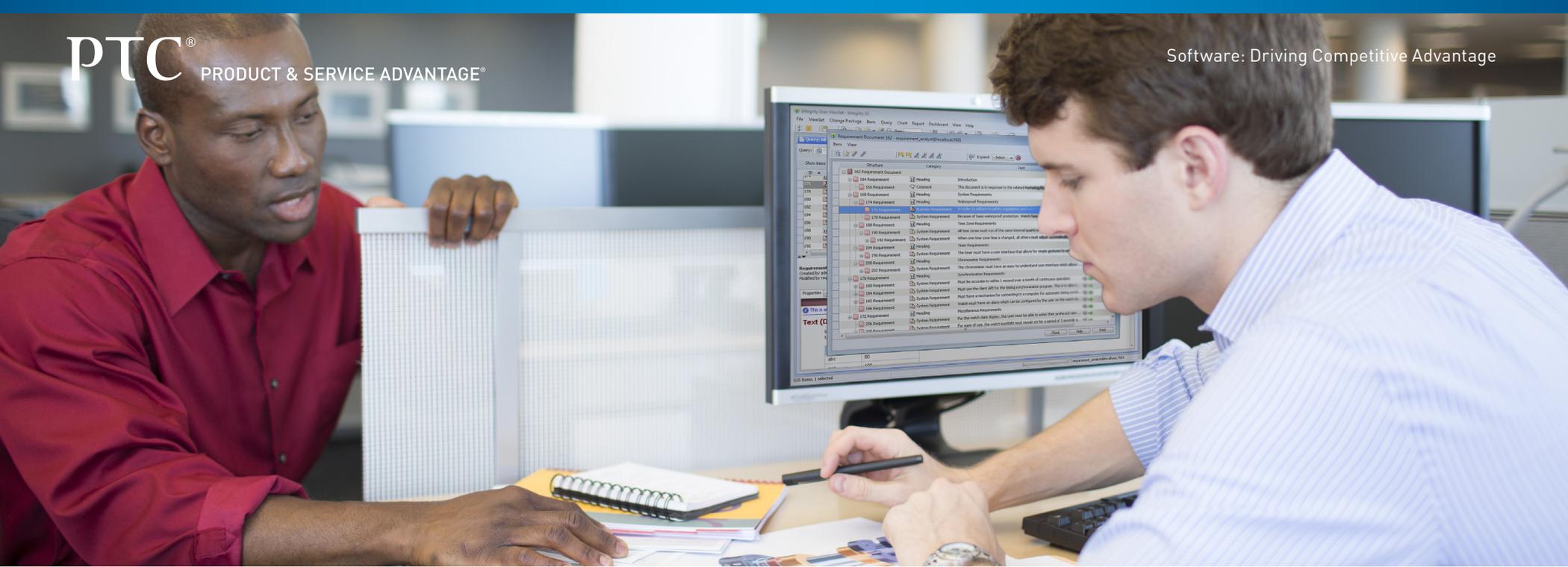
A leading supplier of in-car location and navigation services used a unified product engineering solution to create an early warning system that finds and corrects issues before schedules, quality, or costs are impacted. As a result, the first time right statistic improved from 80 percent to 97 percent. Release predictability was also improved, providing the ability to deliver as promised on-time products to customers.



A leading medical device company reduced compliance reporting efforts by 99 percent — from up to 36 man-weeks per FDA regulatory submission to mere minutes — speeding time-to-market and improving safety in product engineering.

A pioneer in unmanned aircraft systems (UAS) for intelligence, surveillance, and reconnaissance was able to move from Capability Maturity Model Integration (CMMI, a software engineering process improvement approach) Level 1 (i.e., unpredictable, reactive processes) to Level 3 (i.e., proactive processes customized for the organization) in 11 months by using a single automated solution to manage all core CMMI processes and artifacts. This company reached the 500,000 combat flight-hour milestone faster than any other UAS company.

Although these companies represent a range of industries, sizes, and growth stages, they all share common elements. They innovate through software. They employ a solution purpose built to improve and automate development of software-intensive systems. Leaders in their respective markets, they sustain this edge by bringing quality products to market quickly. In so doing, these leaders are able to focus on their core business — accelerating innovation, speeding products to market, and continuously re-inventing the future of manufacturing.



PTC Application Lifecycle Management solutions are powered by PTC Integrity. PTC Integrity accelerates innovation of software-intensive products by connecting people, processes, and artifacts throughout the software and systems lifecycle.

PTC Integrity has an open architecture that integrates disparate tools into a streamlined engineering process, allowing orchestration of engineering change and collaboration across the technology supply chain. With PTC Integrity, engineering teams improve productivity and quality, streamline compliance, and gain complete product visibility, which ultimately drive more innovative products into the market.

To learn more about the PTC global software development solution, go to: PTC.com/solutions/application-lifecycle-management

© 2014, 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, Product & Service Advantage, Creo, Elements/Direct, Windchill, Mathcad, Arbortext, PTC Integrity, Servigistics, ThingWorx, ProductCloud 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.

J3366-Software-Driving-Competitive-Advantage-EN-0014