

Reducing Complexity: Synchronizing Software & Hardware Development

How to Coordinate Embedded Software with the Overall Product Development Process

Executive Summary

Rapidly advancing technologies and growing consumer demands are putting enormous pressure on high tech product development companies to develop increasingly complex products. More and more, key product features are being enabled by software due to its flexibility, its cost advantages, and its supportability over time. In fact, most of the interaction between consumers and high tech products these days is through a software-driven user interface.

However, shorter product development timeframes, more geographically dispersed development teams, and greater reliance on partners and suppliers have all contributed to making embedded software product development much more complex. This complexity results in a variety of problems, ranging from quality issues to time-to-market issues to higher costs.

New approaches are needed to reduce this complexity. Software development teams need to be better coordinated with the rest of product development. This can be enabled by a Product Development System that manages the entire product definition across disciplines – Mechanical, Electrical and Software – to minimize coordination issues, enabling a higher quality product released to market faster.

Typical Challenges in Embedded Software Development

Increased software usage comes with its share of challenges. Costs are growing as companies rapidly, and sometimes inefficiently, add developers and lines of source code. Software can now account for up to 70% of product costs for some consumer electronics products. In addition, software development has always operated in a silo, and with the trends moving towards distributed development and reliance on outside development partners, it is only getting worse. This complexity would imply that more time is needed to ensure high-quality software integration. Unfortunately, software quality is often the first to suffer as a company speeds a product to market.

Companies struggling with the complexities of embedded software development often exhibit these types of problems:

- Releasing the wrong version of software:** Software development is often quite disconnected from the rest of the product development process, which also includes Mechanical and Electrical development. Software development is also much more dynamic than hardware development, so integrating the correct version of software can often be a moving target. In addition, software release processes are often separate from hardware release processes, and are usually not as formal. Therefore, it is surprisingly common for manufacturers to release an incorrect version of software with the product. In fact, almost a quarter of manufacturers experience this problem.¹ While these errors are often found before final customer shipment, the result is far too many late-stage changes that ultimately delay the product's release to the market.
- Rewriting, rather than reusing, software source code:** Most companies recognize that reusing software is desirable, since it builds quality over time due to repeated testing, and reduces cost through the elimination of redundant effort. However, in most enterprises, it is difficult to search for existing software within a single department, let alone across departments and product lines. Therefore, most software development starts from scratch for each project.
- Struggling with distributed or third-party software development:** Companies often struggle with the inability to execute projects effectively across multiple internal and external development teams. It is difficult to gain visibility into the latest project status. It is difficult to make sure everyone has access to the most up-to-date information. It is difficult to communicate changes and understand their impact across multiple teams. This causes inefficient allocation of resources, lack of visibility into project status, and integration problems across project deliverables.
- Managing multiple, uncoordinated change management processes:** Embedded software, by definition, is closely linked with hardware. Many engineering changes that are managed by product development organizations span both hardware and software. An example is a product change that simultaneously impacts both the placement of a physical button on a consumer device, such as an MP3 player, and the software that drives the user interface display. However, software and hardware changes are typically managed using separate processes and solutions. Software uses bug tracking solutions, while hardware is governed by enterprise change management processes. It is difficult to link the two, track their progress, and make sure that both are in sync when the product is released.
- Releasing error-plagued and costly end-user documentation:** As mentioned above, software is often the key to customer adoption and to locking customers into a product line long-term. Customers often rely on documentation to determine how a product works, and it often represents their first impression of the product. However, technical publications are often the last thing companies focus on, and they typically aren't created until after the product design is done. This often results in products being shipped without documentation, or documentation that has significant gaps or errors due to the last-minute haste to release the product.

When taken in combination, these problems can result in a significant drag on corporate performance, resulting in quality problems, time-to-market challenges, and excessive product cost.

Samsung Electronics Leverages Windchill to Improve Software Reuse.

In their Windchill-based Software Information System (SIS), Samsung has connected thousands of software developers across divisions and geographic locations to improve development costs and quality through increased software design reuse. The first key component of this successful project was to integrate multiple IBM Rational ClearCase software configuration management systems to Windchill using PTC's Ready for Rational certified Integration to Rational ClearCase, enabling visibility to software designs across divisions. Second, Windchill provides a standardized approval process to turn software designs into reusable software components. Lastly, the solution provides a searchable catalog of reusable software components, allowing engineers to find reusable designs easily, instead of duplicating work done by other divisions. The system can track how many times each component has been used, allowing greater management visibility into the success of their reuse initiatives.

Exacerbating Complexity: Disconnected Systems Managing Each Discipline

All of the problems discussed above stem from the fact that embedded software teams are typically managed independently from hardware teams. Each discipline—Mechanical, Electrical and Software—uses a separate discrete system to control data and manage changes. With no way to consistently share information between these systems, companies struggle to coordinate development teams, resulting in unneces-

sary design changes, delayed product launches and increased costs. Figure 1 shows a typical customer environment with multiple systems supporting each different area of the product.

This kind of scattered IT infrastructure does not allow for efficient communication between teams and does not readily support cross-discipline design. This increases complexity. The solution is to link software development into a single system that manages the overall product development process.

Typical Electro-mechanical Design Process

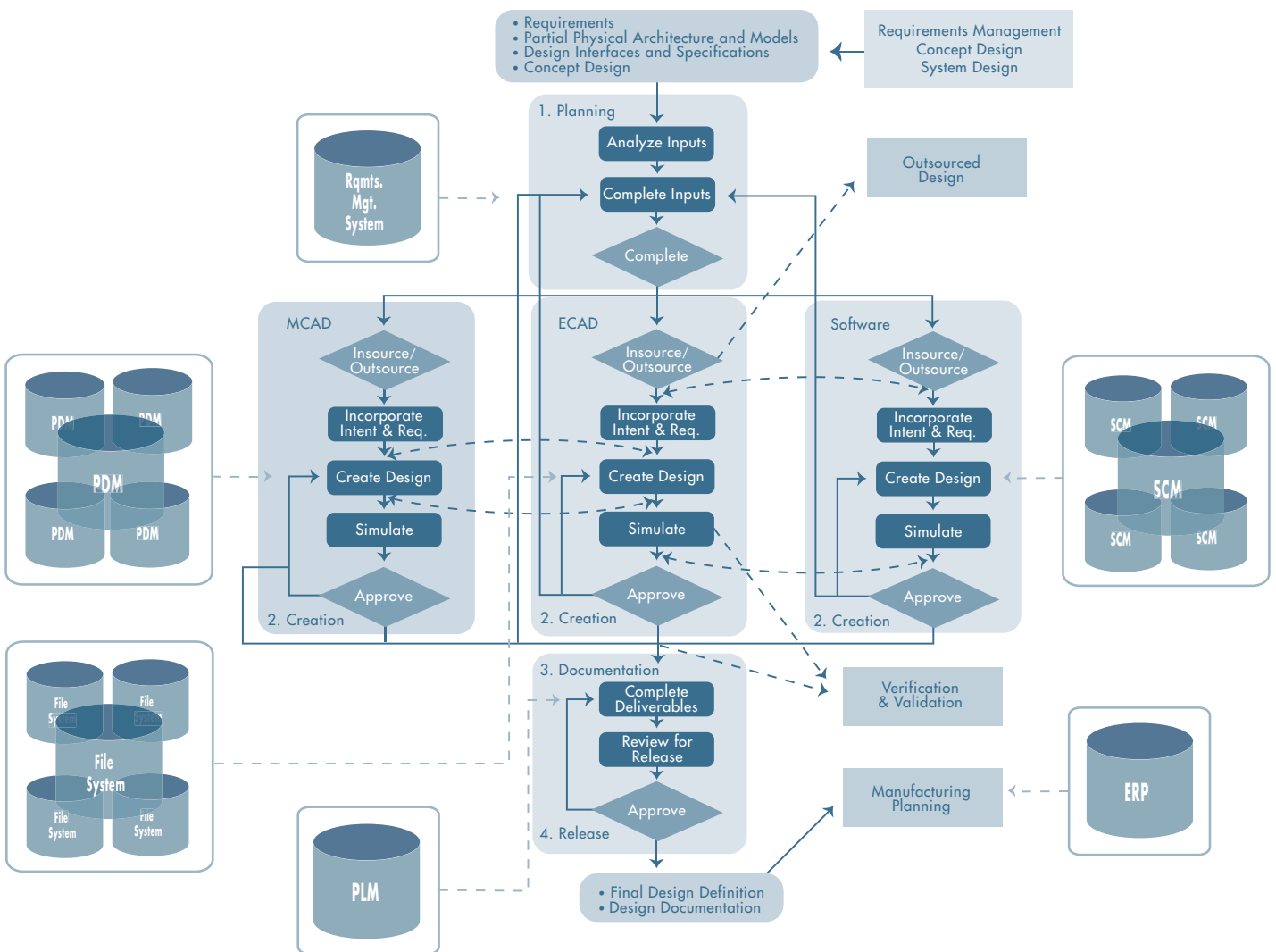


Figure 1: Product development companies typically operate with several systems in place to manage each functional discipline. PDM = product data management, PLM = product lifecycle management, SCM = software configuration management.

Product Development System from PTC: Coordinating Hardware and Software

High tech manufacturers are realizing that new approaches are needed to better coordinate software development with the rest of the product development process. New approaches are also needed to coordinate different software development teams with each other. These new approaches are enabled by the Product Development System that manages the creation and evolution of the entire product definition across disciplines—Mechanical, Electrical and Software. This system also manages the processes that span these disciplines—such as change and configuration management—to minimize coordination issues, enabling a higher quality product released to market faster.

The Product Development System, or PDS, as shown in Figure 2, changes the way companies think about and execute software development. Instead of having multiple, disconnected systems spread throughout the enterprise, the PDS enables a tightly synchronized environment that minimizes the number of systems required and the troublesome touch points between them. PTC works in combination with its strategic partners such as IBM Rational, Cadence, Mentor Graphics, etc., to provide the PDS. As a result, the PDS is able to link together the different design disciplines, and to connect engineering with the extended enterprise, such as procurement, manufacturing and suppliers. The PDS is designed to automate best-practice processes and to enable these linkages.

Product Development Processes for Electronic and High Tech

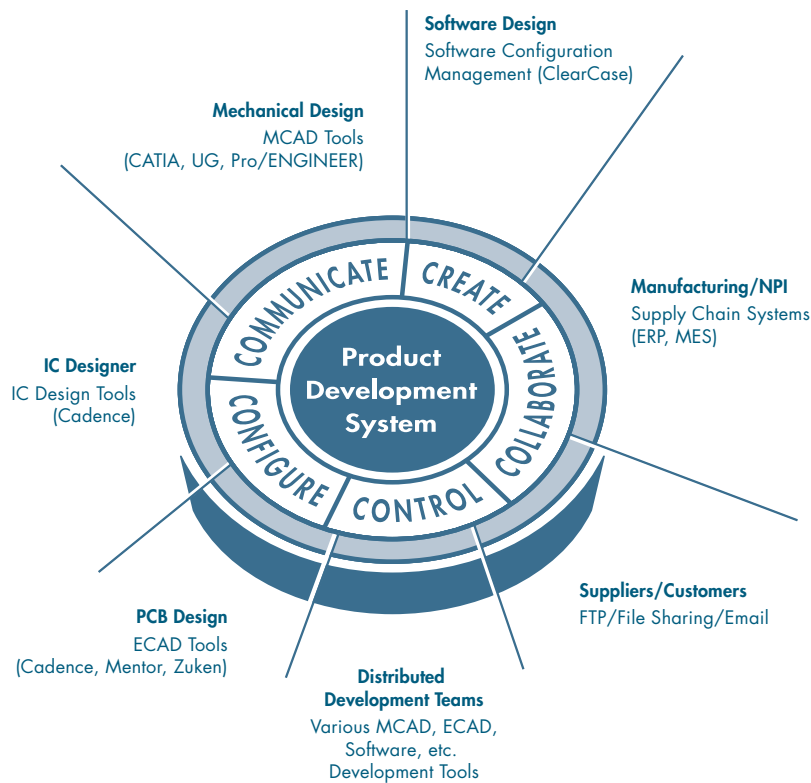


Figure 2: PTC’s Product Development System (PDS) synchronizes different engineering disciplines across different locations with the rest of the enterprise and value chain, improving efficiency, time-to-market and quality.

Windchill®, PTC’s production-proven content and process management software, is the heart of the Product Development System (Figure 3). Fast, secure, and requiring only a Web browser to access, this business collaboration software enables companies to streamline product development processes and deliver superior physical goods and information products. Windchill offers the benefits of a single

source of product information/content which enables development efficiencies and reduces errors and rework. In addition, Windchill provides complete product definition and collaboration capabilities to expertly drive cross-enterprise understanding of information—regardless of source.

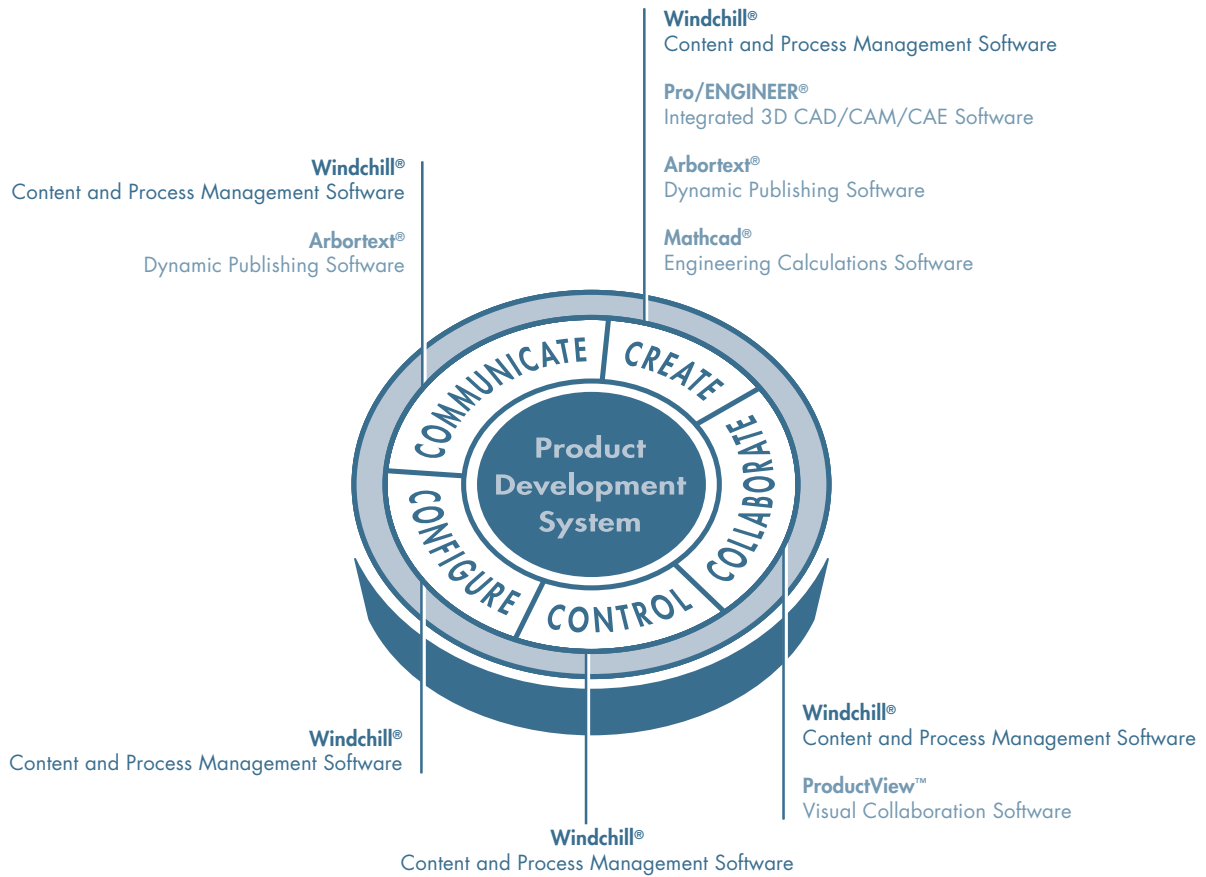


Figure 3: Windchill is the centerpiece of the Product Development System, enabling content and process management across the entire enterprise.

Software Engineering in the Product Development System

Windchill Integrations to Software Configuration Management (SCM) Systems

Windchill seamlessly integrates with leading SCM systems, such as IBM's Rational ClearCase, making software content an integral part of the overall product definition (Figure 4). Linking Rational ClearCase to Windchill synchronizes correct versions of hardware and software during design, release and product updates. Users can link components that are within PTC's system to executables or

source code files in IBM Rational ClearCase. This linkage extracts key attributes and configuration information from the SCM system. Integrating with the software configuration management system also streamlines software change management, automatically notifying cross-discipline users of software defects that will affect electrical or mechanical aspects of the design.

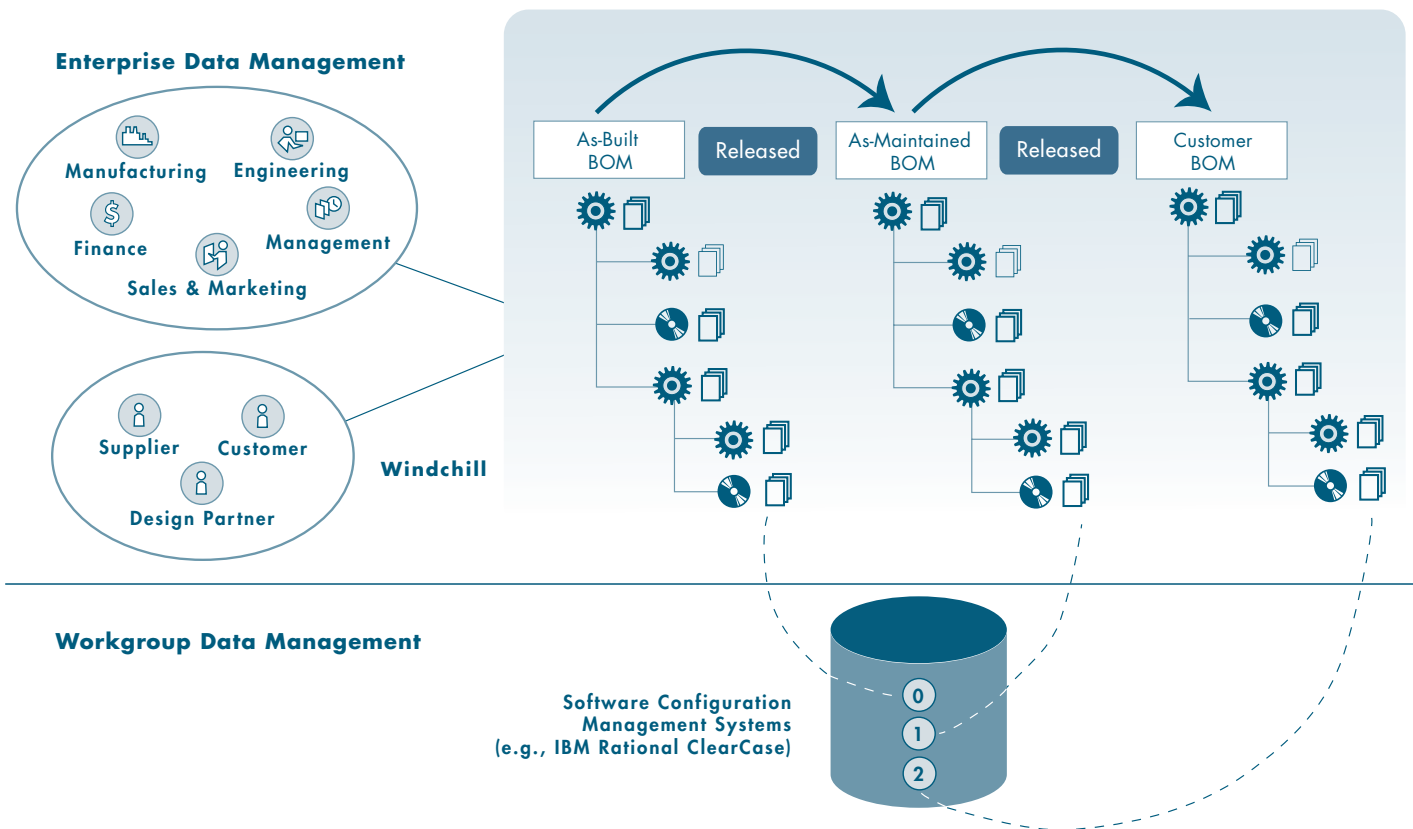


Figure 4: Windchill Integration to Software Configuration Management Systems enables tight coordination between hardware and software development teams.

Reducing Complexity: The Value of Synchronizing Embedded Software Development

The PTC Product Development System (PDS) is a key enabler of reduced complexity in the embedded software development process. The following capabilities, enabled by the PDS, can help companies improve software quality, time-to-market and product cost position.

Integrated BOM Management (Mechanical, Electrical, Software)

The PDS prevents the release of the wrong version of software by managing the entire product definition in a single system: Windchill. Each design discipline, whether it be Mechanical, Electrical or Software, plugs into Windchill to provide a direct relationship between the BOM and the design data that defines it. This makes sure that the correct version of the software executable or device driver is linked to the correct version of the overall product BOM, enabling streamlined release of the correct information to manufacturing.

Compal Electronics decided to implement PTC's Product Development System, based on PTC's Windchill® PLM (Product Lifecycle Management) solution, to modify the business process and enhance product lifecycles, quality standards, design capabilities, and to reduce costs. The results included shortened PART/BOM processes, engineering document transparency, shortened product design and modification, and integration of existing application systems, such as ERP and CAD. These improvements enabled Compal to use a single system, Windchill® from PTC, to manage a comprehensive product structure for their products that span MCAD, ECAD and Software components.

"Information transparency has successfully solved the difference in knowledge and information-sharing between R&D and manufacturing. Now, each operation's progress is transparent to the world, greatly reducing product development time and cost, while speeding the overall process."

–Wen-Kuang Chiu, Senior Manager, Information Systems,
Compal and PLM Project Manager

Improved Software Reuse

The Product Development System provides visibility into software developed across different products and divisions. With its high-speed classification search on top, the PDS makes sure that developers find reusable software quickly, so they don't have to start from scratch as often. The PDS enforces strict reuse policies, making sure that all software has gone through code reviews by architects or senior developers, documentation reviews, interface reviews, etc. before it is classified as truly reusable. Once the software is available for reuse, the PDS then tracks how many times it is reused so you can track the success of your reuse initiatives, improving both your cost and quality over time.

Improved Internal & External Software Development Coordination

The PDS's web-based collaborative project management capabilities dramatically improve coordination of software development teams both inside and outside the company. The PDS allows internal teams to work in sync with outside partners, such as offshore development teams, third-party design partners, external translation teams, customers and suppliers. Windchill supports this collaboration via virtual workspaces, giving team members—both inside and outside the firewall—access to key data and design information, as granted to them by the project manager. Project schedules, deliverables and milestones are visible to every team member, and personalized notifications can be sent to users so they stay updated on the latest product changes. This allows project teams to work more effectively with external software development teams, reducing integration challenges and improving quality.

Integrated Hardware/Software Change Management

The PTC Product Development System provides unprecedented visibility into the change process, allowing project managers to understand the actual status of changes and the dependencies that exist between hardware and software. Now, a single change process can link the hardware information to the related software information. The change process can ensure that the product definition is updated with the correct, changed version of the software before it is released. The ability of the PDS to manage the changing definition of a product over time as software bugs are fixed is represented in Figure 4.

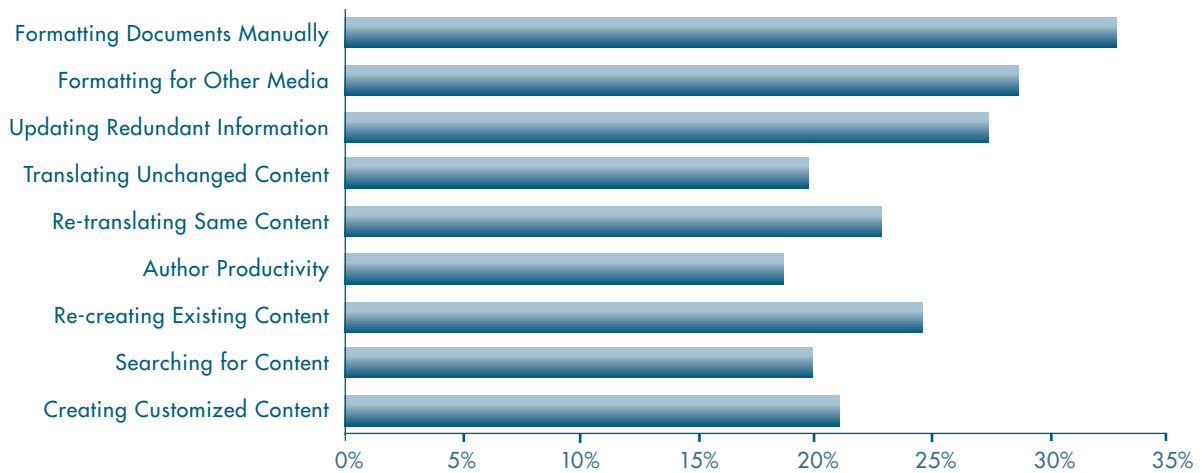
Improved Software Documentation for End-Users

Arbortext, PTC's dynamic publishing software, is an integral part of the PDS. Arbortext streamlines and automates the information publishing process and eliminates costly inefficiencies. Leveraging key Arbortext capabilities—including XML authoring, technical illustration creation, content management and dynamic publishing—companies can improve their publishing processes and, in turn, build significant competitive advantage and lasting differentiation. Via Arbortext and the rest of PTC's PDS, the entire publishing process is automated and streamlined, saving companies significant time and money. In fact, an independent study found that most Arbortext customers experience gains from 20% to over 50% in efficiency and productivity (Figure 5).

Real Results

“Most Arbortext customers see gains from 20% to over 50% in efficiency and productivity.”*

Q: Since using Arbortext products, what is the percentage improvement (e.g., time reduction) you’ve experienced in...



*Results of Independent IDC Survey of 430 Organizations

“It is clear that Arbortext’s customers are achieving a significant return on their investments.”*

Figure 5: Survey of improvements experienced by Arbortext dynamic technical publishing customers.

The Benefits of Synchronized Software Development

PTC’s Product Development System replaces makeshift design and project management processes with a coherent, integral system to manage hardware and software development. With teams and processes working alongside each other in the context of a single system, there are fewer coordination problems—improving cost, quality and time-to-market. Companies will no longer release products with the wrong version of software loaded; specific configurations of the software are now tightly linked to a specific configuration of the overall product. This eliminates wasted time spent on recalls and additional testing. Greater visibility is delivered at the complete product level, as well as at the software development level. Teams no longer work across disparate systems in isolation; everyone has the necessary and relevant information to efficiently make the right design decisions. Solving these challenges directly impacts the quality of software and the cost of developing it.

In addition, many customers have already benefited from the PDS in the Mechanical and Electrical domains. These companies can now apply the same benefits to the software side—leading to improved software development and streamlined new product introductions.

Conclusion

The role of software in product development will continue to grow, and the need to adopt a powerful, comprehensive solution is paramount. When companies fail to manage the communication gaps between hardware and software—and between internal and external teams—product quality declines and costs rise. Successful, forward-looking companies have realized that tolerating such gaps is no longer acceptable—they must completely bridge all development functions: Mechanical, Electrical, and Software. A comprehensive Product Development System that seamlessly integrates hardware and software management is proving to be an effective, cost-saving solution.

*“A Unified Framework,” Manufacturing Business Technology Magazine Whitepaper, 2008.

© 2008, Parametric Technology Corporation (PTC). All rights reserved. Information described herein is furnished for informational use only, is subject to change without notice, and should not be construed as a guarantee, commitment, condition or offer by PTC. PTC, the PTC logo, and all PTC product names and logos are trademarks or registered trademarks of PTC and/or its subsidiaries in the United States and in other countries. All other product or company names are property of their respective owners.