Requirements Management with PLM: A Framework Supporting Consistency Across Product Development Groups

Michel Vrinat, Program Director, PLM, CAE/Europe; Don Brown, Chairman • November 12, 2009

Sizing Up the Challenge Confronting Product Development

The full range of requirements often swamps product development. An Aerospace program on average will start with several thousand customer requirements at the system level, hundreds of thousands at the lowest level of definition for the components, and an equivalent number of verification items and tests. In a perfect world, those cascading requirements and their related information would be defined once and used for the duration of the project.

In reality, requirements are always changing. Customers often shift their preferences, or reach a better understanding of their actual needs, after looking at an initial design. Market drivers may shift over the duration of the development project, or authorities may add new constraints driven by environmental and safety considerations. Sometimes the project encounters a roadblock that forces a revision of the initial targets. In effect, large projects face hundreds of changes every day, and new requirements baselines may be created every week.

Those demands raise the absolute need for requirements management tools to directly deal with change management, change impact analysis, revisions, options, baselines, and effectivities. A strong configuration management function needs to be provided.

As an example, a major U.S. DoD contractor relies directly on the up-front development of requirements to control the continuous changes encountered in product development. Without a baseline defined early in the effort, those in development all too often institute changes for refinements they think are simply “nice.” In addition, the customers generally think of extensions over time, many of which may well represent good ideas, but unfortunately destroy any chances of meeting the original time and budget targets. Historically, the dual tendencies of developers and customers to change and add to the scope of projects led to creeping requirements, with most projects falling behind and running over budget. An up-front early definition of requirements during the conceptual phase now avoids requirements creep, to keep the effort on schedule and on budget, with a formal change order process for official approvals structured with the initial definition of the baseline.

Requirements development as a recurrent activity involves multiple levels across many teams. The starting point is the expression of customer needs or market demand; it may be as short as one sentence or very detailed, with a profusion of texts, graphs, or even formulas.

Customer needs represent the first input in a long and complex process. Then program managers and product planners develop their own requirements based on this input to specify the functional needs. Those needs and sub-system requirements will be the input for the following phase of developing yet more detailed requirements, until the lowest level of components covering hardware, software, or electronics is fully specified and designed. All
of those steps in developing product requirements proceed in parallel with the definition of the verification and testing plan at each step.

The process of flowing down requirements finishes with a multi-disciplinary effort, with each area involved in detailing the definition of components across the hardware, software, and electronic domains. Issues that cut across disciplines are nearly impossible to catch with isolated requirements management tools that do not address the full range of development efforts. A change in the configuration of the physical product, or even the application of compatibility rules, may have an impact on a version to be used or a software release. That impact can only be detected and analyzed through a direct link from the requirements management system to the data and processes in place for the affected disciplines, supported by the full integration with a product lifecycle management (PLM) system.

**The Payoff and Reality of Integrating Requirements Management with PLM**

PLM provides a backbone or framework to support data and process consistency across each stage of product development, starting upfront with the early definition of a product or service based on customer needs. The impact of any proposed changes on requirements must be analyzed and traced across multiple and diverse disciplines. In the other direction, the impact of any changes in parts or systems can be directly traced and visualized back to requirements. Multiple disciplines, which may not be able to translate the effect of changes directly from one area to another, can then trace and reconcile the impact of any changes in their own specialty through the help of a requirements framework. The ability to trace and visualize the impact represents a major benefit of fully integrating requirements management with PLM.

Citing a substantial payoff from the full integration of major applications, the DoD contractor referenced above embraced a PLM strategy spanning the full product lifecycle and integrating three mainstream applications with the PLM system. Teamcenter’s enterprise knowledge-management capability creates the work breakdown structure and manages the flow of product development efforts. Teamcenter’s systems engineering and requirements management application maintains the traceability for tracking any changes back to requirements and the work breakdown structure. The requirements capabilities define the baseline from the early conceptual phase, creating the statement of work, defining the scope of various efforts, and scheduling the tasks. Teamcenter’s portfolio, program, and project management manages the cost and schedule, identifying critical deadlines for everyone. The full integration of project and requirements management with PLM enables the development groups to trace back the impact of any changes to requirements and tasks, to fully assess their effects.

In dramatic contrast, a top supplier of automatic test equipment focused solely on deploying Teamcenter’s systems engineering and requirements management to support integration across mechanical, software, and electronic development groups. The tight focus
cut the implementation time down to “several months,” which in turn drove down their change process to 12½ days from a lengthy 55-day cycle. Previously, they simply could not write good requirements, spending significant time and effort in the area with a third of the output not even needed. The systems engineering group and the program management office decided to focus on establishing a clear and concise requirements approach early in the development effort to support critical baselines. One tool and one process would be applied across disciplines, including hardware, electronics, and software. The effort has paid off most in supporting a requirements breakdown and traceability. It is not expected, however, to effectively contribute to the front end in translating qualitative customer needs, based on feedback and surveys, into requirements, which in their view will always necessitate human judgment.

With their new system, change management has become far more effective, with tighter controls based on the integration with requirements, and improved process support. Traceability enables all concerned to understand the impact of changes with the flowdown of requirements. Now, all involved can be comfortable that all requirements are indeed tested, and testing is no longer done for non-existent requirements – which had been the “ugly reality” previously.

To achieve the full payoff of an integrated enterprise system, requirements can never be considered in isolation, as they are strongly associated with the validation plan defined to ensure proper implementation. The combined information needs to be managed to support both the request and the proof that the request has been satisfied.

The result of these combined activities across multiple areas presents a rich and complex network of information. The management of that content must synchronize with all engineering activities taking place across product definition, development, manufacturing, delivery, and support. Indeed, the role and contribution of requirements management extends well beyond the initial input for customer needs to get a project started. Requirements serve as the common thread and reference line for all engineering work over the complete product life – a framework to promote consistency and reconcile changes or differences across groups.
Critical Priorities for a Requirements Management Framework with PLM

Requirements management requires a complete set of capabilities that establish a strong link across diverse engineering activities for product development:

- capture needs and search available requirements for re-use
- refine, flow down, and validate requirements
- allocate requirements
- ease of use for broad adoption
- verify the design and product against the requirements
- manage and trace the changes impacting requirements

Capturing needs is usually done with documents in textual form, which are easily parsed to highlight the significant requirements that can be extracted. Most of today’s products or systems contain many components or sub-systems that are the same or similar to those developed for previous projects. Rarely is more than 20% of the content of a product or system totally new. Indeed, that 20% proportion generally sustains good innovation to reinforce the appeal to customers. The introduction of radically new paradigms, such as the Apple iPhone, simply does not happen very often.

In general, one of the most important steps when starting a new project relates to the search of existing and established requirements that most closely match the actual new need targeted for the customer. Finding those requirements not only saves time in developing requirements, but it supports ready access to reusable information and deliverables. That may include a fully defined validation plan or reusable parts and software, allowing for huge savings. The highest level of reusability provides the highest payoff.

Indeed, a major supplier of heavy equipment gets its biggest payoff with requirements management from the commonality that supports reuse across more than three hundred different products. Common features can be used for the electronic systems to control hydraulics, transmissions, and engines. Instead of three hundred separate items, a series of product families with common features has been defined and maintained by carefully managing the underlying requirements. Teamcenter provides the database to organize a consistent product breakdown supporting both the definition of features and traceability. Designers can search the parts bin for appropriate components, and trace their links to check where and how the components are used.

In order to present the data to the requirements manager, the underlying tool must have ready access to established validation plans, the product definition for past designs, and product structures. In effect, the full and complete integration with a PLM system becomes a strategic capability.
The refinement, flowdown, and validation of requirements involve specific functions that must be fully supported by a requirements management tool. While the functions relate to operational steps taken internally within requirements management, they define and support strong links across diverse engineering activities such as product planning, product design, and simulation. They also provide the base to support interaction and collaboration with both the targeted customers and suppliers. As a result, they must be executed in a sufficiently open but secure environment to share the appropriate content. The related efforts involve the management of access rights, building baselines, illustrating as best as possible the significance of the requirement, and the identification of dependencies with their propagation down and across design levels and disciplines.

Allocating requirements to elements of the system architecture or product structure establishes and supports traceability between requirements and the specific details and approaches that have been identified to satisfy that requirement. The product planner, the system architect, the design engineer, the analyst, or indeed anyone responsible for ensuring that the requirement is correctly allocated to a technical solution, must have access to the requirements management tool. They must also be able to access any authoring or data management tool in the particular engineering discipline involved with that requirement. This critical need clearly presents one of the major benefits of having a PLM system deployed – linking together information from different sources in different contexts to create a consistent network of data and processes.

Ease of use for broad adoption: An easy-to-use interface for a broad range of occasional users, who are not experts with the specific software, is needed as well to achieve the full leveraged payoff of requirements management. Teamcenter’s user interface will prove critical for the supplier of automatic test equipment mentioned earlier, in the next planned effort to integrate verification and test. Previously, the scheduling and task assignment for testing was all completed manually, initiated by unstructured emails. Several days were needed to canvas all the groups involved to coordinate any changes in testing that might arise. Teamcenter’s live integration with Excel that supports easy access will document the requirements and support dynamic changes.

Usability represents a major plus for the DoD contractor as well, with the use of live interfaces with Microsoft Office applications at the front end for users of the Teamcenter suite. Roughly two hundred users currently rely on the requirements capabilities without involving a significant learning curve. Over six hundred are targeted, supported by easy-to-use interfaces and standard reports across multiple programs, with the full integration of Word and Excel.

The verification of a design or product against requirements can be done in multiple ways, according to the level of requirements flowdown involved in the validation, and the nature of the item in the system or product structure that is allocated to the requirement. Software involves testing functions; structural parts require simulation or physical tests;
Requirements Management with PLM: A Framework Supporting Consistency Across Product Development Groups

November 12, 2009

Manufacturing entails process simulation. All of these activities are executed with specific tools applying specific processes. All manage their own data sets. In order to verify that a requirement has been satisfied, a specific verification plan or test plan must be defined and associated to the requirement. Its execution must provide access to the means of simulating or testing the specific system or product item. That work becomes complicated and time consuming for complex products. Without the support of a PLM system to track all the data and to create the correct links between the disparate elements in the data, verification is too often only partially completed, or finalized too late, for corrective actions to avoid ballooning costs late in the product development cycle.

Managing requirements, or requirements administration, also represents a complex task. Many requirements management tools in the past neglected this aspect, mainly concentrating on text editing capabilities for requirements development. Requirements just represent another view of the system or product structure that must be managed with the same tools and capabilities. That includes the organization of requirements in structures with links and dependencies, versioning, baselines, change management, and configuration management. Like any other item in a product structure, the change process driving a project must cover requirements, and apply the configuration rules in effect. If design or engineering request a change, its impact on related requirements must be analyzed. The change process must include requirements managers in the change proposal and validation process. Similarly, if a change is requested on a requirement, the full impact on the current system or product design must be analyzed with design engineers actively involved in the change proposal and validation process. The fundamental need arises to unify the change process across all disciplines participating in a product development project, which in turn represents the core justification for a PLM system.