Requirements-Driven Product Development

Case Studies in A&D, Medical Electronics, and Automotive

May 2009

A White Paper Prepared by

Collaborative Product Development Associates for

IBM

TABLE OF CONTENTS

I: The Rapid Evolution of Requirements Management and DOORS

INTRODUCTION

Over the last decade, requirements management has emerged as a primary driver for quality improvement across product development, by closely coordinating efforts across all disciplines. This has led to the emergence of many requirement management tools, with IBM Rational DOORS – previously Telelogic DOORS – representing the market leader. This document discusses the challenges in implementing DOORS in a rapidly evolving environment and the clear payoffs from using DOORS to integrate across diverse disciplines and to boost the collaborative effort. Three leading users of the software in very different businesses contributed to the effort – Sharon Crossby from the Astrium division of EADS, Susan Hacker from the Clinical and Medical Products Group of Cardinal Health, and Ed Griffor of Chrysler. Ron Zorn, a member of the IBM Telelogic Leadership Council and the Telelogic Leadership Council and the Executive Advisory Board, also contributed significantly to the discussion.[i]

CRITICAL ASPECTS FOR MANAGING REQUIREMENTS

Product development groups rely on DOORS to gather, manage, and allocate requirements. Given the rapid rate of innovation for requirements management across the full development cycle, all current offerings on the market face major challenges in fully serving the widening breadth of functionality and compounding detail demanded by leading users.

A requirements management tool can be evaluated from at least three aspects of managing requirements:

- Gathering, developing and sharing requirements across multiple disciplines, as well as defining the requirements database structure
- Change traceability
- Requirements reuse

Each of these aspects, as well as the challenges and payoffs of implementing DOORS, was discussed by the three users interviewed.

II: Managing Requirements with DOORS Across the Full Range of Product Development at EADS

As part of its PLM harmonization program called PHENIX, EADS has defined a mandatory policy for requirements management to be applied across all divisions, including Airbus; Eurocopter; Astrium Services, Satellites, and Space Transportation; and Defense and Security.

That policy clearly defines a fundamental role for requirements management at all stages of product development, from concept definition, through design and development, production, support, and disposal. It also identifies the core functions that good requirements management practice must support:

- Capture the needs and develop the requirements
- Allocate and flow down the requirements
- Validate the requirements
- Verify the design against the requirements
- Verify the product against the requirements
- Manage the requirement changes

Validation and verification must be applied at all levels of the requirements breakdown, from the system-level decomposition, through sub-systems, to the functional breakdown and physical component specifications. This practice provides the primary means for supporting consistency across engineering disciplines. Too many times, the requirements management function is applied at a high level of design involving system design or conceptual design, and then stops, without establishing any formal link with actual engineering activities, such as detailed design, simulation, or manufacturing.

EADS standardized on DOORS as their requirements-management tool, and entered into a world-wide contractual agreement with IBM Rational – originally Telelogic – to acquire, distribute, and support the product.

DOORS supports all functions at the operational level needed to support the EADS requirements management policy as outlined above. Indeed, it implements the traceability required during product development to ensure that the product will match customer expectations in all aspects of performance.

EADS ASTRIUM

EADS Astrium is the number one firm in Europe, and the number three worldwide, in space transportation, satellite systems, and services, including Ariane, the International Space Station, Envisat, and Mars Express. Astrium employs 12,000 people in five countries: France, Germany, the UK, Spain, and the Netherlands.

Astrium represents a key contributor to the definition of the EADS requirements-management policy and its deployment across the group. A long-time user of DOORS, Sharon Crossby runs the requirements management effort in that division. In her view, the main challenge in the space industry derives from too many requirements, and the contradictions that arise between them. The volume of requirements presents the biggest issue as customers tend to create huge requirement specifications, which in turn requires Astrium to go back with queries for clarification on any contradictions. Sometimes they have to accept requirements as they are and interpret their significance internally. The designs are extremely complex, and the internal business requirements may be so constraining that an up-front discussion with customers to establish an agreement on the right set of requirements becomes critical for the success of a program.

WELL-DEFINED PROCESSES ARE CRITICAL

As the software tool of reference at Astrium, DOORS provides great help in assessing requirements, in looking for similarities, and in keeping track of all changes and relationships. But the software is not enough on its own. A process with a methodical review of requirements is necessary, based on strong standard practices, as well as training and competency development for the engineers. The tool supports the process well, and without the tool the process would not work. However, the process must be well defined as the foundation.

Astrium has over two thousand engineers registered as DOORS users, with a tenth of those involved in editing requirements, and all others reviewing and exploring the database, to understand relationships and dependencies of requirements at all levels. With more than six thousand engineers overall, there is clearly room for increased participation in requirements management covering the complete development process. Today, the traceability of requirements goes down four levels, from system to sub-systems definition. Requirements management involves multiple engineering disciplines including system design, architecture, software, mechanical, and simulation – though not yet manufacturing and operations. It does not yet fully address the lower level of software engineering, nor mechanical engineering at the component level. Transverse functions such as safety and security, as well as marketing and sales, also need access to requirements management, and efforts to cover those areas have already been initiated.

To address the issues of complexity, critical areas must be identified and prioritized. Astrium applies three levels of prioritization within DOORS: high, medium, or low. There is also a need for a graphical view of the complete requirements structure, provided by external software – netViz from CA – to understand how requirements should be related to each other and which documents need to be related to each other.

MANAGING THE APPROVAL PROCESS FOR CHANGES

Another key aspect of requirements management is how cross-functional teams resolve conflicting and proliferating changes across disciplines such as the electrical and mechanical areas and across the supply chain with multiple suppliers. There is a strong need for a common approval process to coordinate multiple changes done by each team, both on the requirements and on the verification methods. A change may extend beyond requirements to involve system design, verification engineering, and so forth, and a good interface with each team is required. This is very important to manage correctly. From the Astrium point of view, while DOORS manages the requirements, it does not support a sufficient change process on its own that coordinates both project changes and configuration management.

Within the DOORS environment, however, a requirement cannot be modified without documenting the details of what was changed, when, and by whom; it then saves the history of the requirement content. The change can be initiated inside DOORS and exported to a change management system for approval. Or a change may come from an external source that is implemented upon approval. With a required change, an analysis of the impact must be performed to identify all linked items, including the lower levels of requirements decomposition, and across verification methods, before the change process can be launched. The management of dependencies across multiple disciplines is vital. Currently, EADS Astrium is investigating the potential integration with tools such as IBM Rational ClearQuest and ClearCase for change management, while the EADS PHENIX program already supports a standard process for change and configuration management implemented with a common repository for a master product definition.

Astrium, like any other EADS division, has multiple sites in Europe performing engineering on the same projects. They all share the databases for DOORS, which supports concurrent requirements engineering.

Astrium is prototyping the integration with system modeling tools like Rational Rhapsody, Rational Tau, or Sparx Enterprise Architect, but has not yet implemented any system in production. To make a full deployment of the integration, it has to be inexpensive and easy to use. The benefits of integration must outweigh the cost and additional time and complexity involved with the process. Even though integration may be available, it can still represent a sub-optimal alternative.

INTEGRATING TEST

On the testing side, the integration with requirements is done by writing the test specifications in DOORS and linking them to requirements from the top level down. The next level of test procedures can also be done in DOORS. For some areas there is feedback specifying certain conditions such as go or no-go, the test number, and pass or fail results. There is a full integration on what will be tested against requirements, which avoids specifying unnecessary tests. The full

integration also ensures that the test verifies the latest version of requirements. Test case documents are written in parallel by another person at the same time as requirements, stored in DOORS, and fully traced to requirements. There may be a large number of test cases for the same requirement, at times distributed to different suppliers.

PROJECT MANAGEMENT WITH DOORS

In the support of project management, DOORS can provide metrics to show how well the project is progressing against milestones. For example, it may verify that all the customer requirements have been traced to design/lower level requirements by a Critical Design Review, or CDR. If the project/product is well managed, these metrics stimulate the project manager to act upon the results to be sure to meet the milestones. As part of the project baseline, DOORS supports configuration control by providing read-only data to export into the configuration management or PLM tool.

THE PAYOFF WITH DOORS AT EADS

According to Sharon, without DOORS, quality and cost would be at risk. DOORS provides essential support for a good process. But, if it's a bad process, DOORS will not make it any better. Changes made by email, as an example, cannot be tracked. In that case, product development simply does not consistently follow the process. The level of adoption by users is mixed. Reasonable support may come from areas that derive value from its adoption, while other users demand a solution handed to them without need for a learning curve. They may or may not follow the process depending upon their own perceptions of the results. In approaching a new process, targeting the potential leaders can present a major payoff, as they may become advocates, telling everyone else how much the solution helps.

The adoption level by projects/programs varies significantly, from programs with active experts in requirements management using DOORS and other programs ignoring it. This strongly depends upon the personal involvement of local "champions," as well as the cultural background of the different groups. Top-down recommendations from management play a role, but adoption directly by the team represents the key. Sharon's team spends a significant amount of time in awareness training and motivation sessions with targeted users. Customer mandates requiring DOORS as the requirements management tool, such as those coming from ESA, the European Space Agency, definitely boost acceptance, and adoption.

DOORS' biggest win relates to its support of traceability in tracking customer needs through fulfillment and across the full development cycle. That traceability can demonstrate to a customer that the proposed solution meets their needs. In terms of development efforts in the continuing evolution of DOORS, Astrium would most like to have web access that is more intuitive, and direct workflow integration.

III: Meeting Regulatory Requirements and Driving Reuse at Cardinal Health

Cardinal Health is an $87 billion global distributor and manufacturer of medical and surgical supplies and technologies. Susan Hacker, Senior Software Technical Editor, works directly in supporting requirements management as the DOORS database administrator for the clinical and medical products group, which represents approximately $1.8 billion in revenues with 14,200 employees worldwide. The products supported cover a broad range of needs including infusion pumps for intravenous medication, automated medication and supply management systems, thermometry and respiratory products, and wireless, barcode-enabled patient identification systems.

MANAGING CHANGE TO MEET REGULATIONS WITH DOORS

The most important application of DOORS relates to infusion pumps. Because all changes to requirements must be tracked and validated to meet regulatory needs, DOORS was readily justified. Cardinal Health relies on the built-in change-proposal system in DOORS, rather than the new module directly targeting change management. Introducing the new module would potentially impact many development processes worldwide – a risk no one wants. In addition to the customary organizational resistance to changes, patient risk issues demand proven approaches and established processes for verification and validation, while regulatory concerns introduce significant hurdles in adjusting procedures and processes. One of the important challenges is that the FDA, the Food and Drug Administration, directly audits compliance with a lengthy list of rules.

Every update involving a feature upgrade, a change in response to a problem, or simply a bug fix, requires a documented analysis. Before DOORS, all this analysis was done manually. Moreover, with modular systems, a new module with new features complementing the original involves an evaluation of all the original requirements for applicability and retest, with a complex tracing back to user needs and certifications. With a new module for the infusion pump, developers review every requirement to see if it applies for the new module. Four or five hundred changes could be involved to produce a new module, impacting a whole list of pre-existing modules. If a requirement applies to the new module, then a change proposal must be processed.

DOORS addresses the growing complexity of managing feature upgrades with dramatic and major assistance in the reuse of requirements. User-defined needs migrate from one release to the next, and new ones may be added. Typically, the addition of a new module might involve two hundred and fifty requirements, whereas a new feature may need considerably less. Ninety to ninety-five percent of the existing requirements will migrate over, two to five per cent may be new, and some of the existing requirements may be obsoleted. With DOORS, the upgrade inherits the established structure and links with user needs, requirements,

and test. The links are maintained across all projects and versions. The test management system manages the test cases, and interfaces directly with DOORS. Roughly twenty-five floating licenses serve over three hundred users with access control to log in to the system. Five super users rely on the system most of the time, while roughly thirty intermediate users access the system two or three times a week. The rest consist of casual users, and as many as seventy-five with access never use it.

THE MECHATRONIC CHALLENGE WITH INFUSION PUMPS

The most complicated systems relate to Alaris infusion pumps, which involve electrical, mechanical, and software development efforts. By common agreement, requirements are generally maintained at a high level, with the design group breaking down requirements all the way to specifications that cover parts and features. For software, the breakdown does not extend to the code itself. Indeed, it is extremely difficult to keep specifications current that link into software code, and to provide traceability. For the electrical and mechanical groups, many of the specifications are effectively laid out in drawings that are handled by Siemens Teamcenter. Developers can place a reference on the drawing or document that identifies the related requirements, but there is no direct link to DOORS.

PRODUCT DEVELOPMENT AND REQUIREMENTS MANAGEMENT

Product development breaks down into six phases:

1. Define the idea
2. Define the project in terms of users' needs; establish feasibility and assess profitability; search patents
3. Feasibility and planning; derive requirements from user needs
4. Design and verification
5. Limited release and validation
6. Post release and follow up

Requirements management with DOORS begins with phase three, and extends to the next phase of design and verification. To transition from phase four to phase five all tests must be completed. Simulations have been completed, all requirements met, and in-house clinicians have conducted hands-on evaluations of the system as well. Risks cannot be taken, and any offering must be fully tested before use with a patient. If there is feedback on requirements in the latter two phases, that information would roll back to define a variant of the product, an upgrade, or a new module.

The software development group also employs Enterprise Architect from Sparx Systems to model and simulate system design. Adoption of model-driven development approaches, recognized as a potentially powerful alternative by several in the company, has been limited in part by the inertia of change, related in part to the overhead of regulatory FDA audits. The Enterprise Architect system is partially integrated with DOORS.

> Without DOORS, major problems would arise. For example, one major product has evolved over fifteen years, with roughly two thousand requirements tracing to volumes of documentation. Every requirement has a test case. Starting over from scratch would involve the work of writing all new test cases, followed by an FDA audit for approval. "We could not develop the infusion pump without DOORS," states Susan. "Even two or three hundred requirements would involve a huge job of manually tracking user needs to hazards, through requirements, specifications, and test. The linking in DOORS makes it much easier to do. Eighty to eighty-five percent of the traceable items are linked, and we can handle rest manually."

IV: Fully Coordinating People, Process, and Tools with Requirements Management at Chrysler

At Chrysler, Powertrain Product Engineering (PTPE) together with Electrical Engineering Core (E/E Core) spearheaded requirements management using DOORS across the engineering process. While the challenges of dealing with far more frequent changes in software development represented one of the driving factors, the major motivation related to the pairing of hardware and software that substantially increased complexity. Moreover, the electrical engineering group fully understood the thinking of the designer and engineer who would be using the solution, and who would benefit from an effective approach addressing the escalating complexity of mechatronics designs. Addressing cultural issues proved to be a higher priority than the technical solution itself. By comparison, years earlier the vice president in charge of components and processes had tried to drive the first and early efforts targeting requirements management. That early program stalled because the managers did not directly and effectively target the actual designers producing the parts. In a sense, management's top-down thinking simply did not connect with the part-centric culture of the designers and engineers in the automotive sector.

Today, over five hundred users rely on DOORS for requirements management in specifying parts, systems, and processes. The documentation and management of dependencies, however, will steadily broaden the use of DOORS to include teams in purchasing and quality control as well, ultimately involving thousands of users. The area up and running today focuses on part and system specifications, but as the feature documentation and the specification process itself extends across the organization, usage will increase dramatically.

A FULL SYSTEM PERSPECTIVE FOR REQUIREMENTS MANAGEMENT

The approach starts at a very high level targeting the customer experience, and drills down to the functional decompositions that are collected to define the system. On the mechanical side, at the beginning of the effort requirements management focused on the part level and lacked a system perspective. The mechanical group continues to lag in fully understanding the need for a system view linking the part to customer needs. Similarly, early in the program embedded software focused on detailed algorithms. Both groups lacked a system perspective at that stage. Whenever software developers changed their code, they did not think of the task as assembling pieces of code. They rewrote the whole code, and reuse of any sections of code was haphazard. They lacked a systematic way to approach changes, and there was no systematic reuse. The advent of embedded C code, however, with object-oriented capabilities, modularized many of the algorithms, and enabled reuse of the modules. It also forced the software developer to think at a higher conceptual level about reuse. Complementing the discipline of modularization that aided change management, requirements

management enabled the programmer to track his work on the objects back to the original customer needs. That in turn stimulated the management of the complex inter-relationships between design decisions and customer needs.

Chrysler relies on the combination of IBM Rational DOORS, Synergy, and Change, to manage the full range of requirements from fully automated software testing to physical vehicle validation. In addition, HP Quality Center manages all tests directly, from component to system to full vehicle, and provides maturity reports, while integrating with both DOORS and Synergy. Quality Center manages the tests by mapping the results to requirements in full support of real-time verification and validation.

Overall, people and process must be fully coordinated from the conceptual definition, through design, development, testing, deployment, and support, using the requirements management tools. Indeed, the people and process issues represent even higher priorities than the technical capabilities of the tools. Team members and stakeholders must be continuously connected to establish and maintain traceability across the full development cycle.

THE CHALLENGE OF CONFIGURATION MANAGEMENT IN AUTOMOTIVE

At Chrysler, IBM Rational Synergy manages the configuration of software in advanced hybrid systems and the configuration of combined software and hardware in electrical systems. It also supports the change and release process. Configuration management in the automotive sector presents particular and major challenges with so many parts involved combined with many models and variants. All of these factors contribute to a major combinatorial problem. There are parts, systems, features, sales codes, vehicles – many different items and combinations, all of which cannot possibly be tested. Indeed, how many different items and configurations does management need to have tested for compliance in meeting customer requirements? How many potential combinations need to be reviewed? Synergy's role is to support whatever choice is made. Once it is decided what will be tested and how, Synergy tracks the choices, and manages the integration of higher level systems into larger systems. It will not, however, provide the judgment for choosing which combinations to test and review. It will only track those chosen, and the work done. It is a dynamic approach to reduce the need to retest. By testing a combination of objects, in effect a higher level object is tested. It may then be reused as long as the customer needs do not change. Synergy helps to build on those choices by enabling higher order reuse.

IBM Rational Change manages all changes to configurations of product data, including baseline requirements. The documentation and discipline promotes a repeatable and reliable process for processing and capturing defect data and change requests.

Requirements-driven development at Chrysler traces implementation requests to development tasks and objects. DOORS' functions in particular include strong reporting capability.

ADDRESSING THE FULL CHALLENGE OF PEOPLE, PROCESS, AND TOOLS

In automotive, the challenges associated with requirements-driven development relate more directly to the evolution of the development culture than to the current capability of the software tools. For example, automotive lacks any tradition of creating and managing requirements, as in the past the skill of producing cars relied on the people themselves. Physical test results dominated, without any requirements directly reflected in the analysis. Given that history, requirements evolved only later after the development process had matured as a set of multiple and isolated models, reflecting different targets such as cost, quality, and customer needs. To reconcile those differences, the specifications were largely left incomplete to facilitate direct tradeoff analyses.

Today, the requirements management process at Chrysler is analyzed with a direct audit of the data, followed by the use of CMMI (Capability Maturity Model Integration) to assess the level of maturity of the effort.[ii] Industry, government, and later the Software Engineering Institute (SEI) of Carnegie Mellon University jointly developed the CMMI Framework, a set of integrated CMMI models, a CMMI appraisal method, and supporting products. Powertrain at Chrysler began using the CMMI model three years ago to assess the steps needed to reach defined levels of maturity and to design its own processes. The application of the CMMI framework to the requirements management process provides valuable information on the current state of the process, and lays the foundation for making the right choices about the changes that should be made.

In general, the software tool filters the requirements database based upon attributes laid out by specific templates covering inputs and outputs to fulfill multiple goals, in order to generate different views of the data. At Chrysler, those views are then referenced by the users to help understand the meaning of requirements, and to identify any inconsistencies in project plans. Freezing a baseline, and then applying version control, provides the basis to first obtain a commitment from the individuals involved in development, and then to coordinate any changes.

Edward Griffor, a Technical Fellow at Chrysler, clearly recognizes and respects the vision of reconciling the efforts of the many disciplines involved in product development through requirements management technology. That vision, however, must be tempered by the reality that no system today directly addresses the conflicting language and terminology of the physics and logic domains. The success of the release process itself depends upon a review by those involved in the design directly contributing their expertise. Indeed, formal design reviews target the objective of having the people and experts involved recognize and address any outstanding problems. While repetitive tasks justify the effort to codify a solution for automation, that codification itself requires human interaction to define a process and the pre-requisites for automation. In effect, continuous improvement can be achieved by automating various rules in

the evolution of the design process, but that evolution will regularly result in incremental improvements rather than a grand-slam, total solution. Indeed, the release process may never be fully automated, and must address the combined needs of people, process, and tools.

GATHERING, DEVELOPING, AND SHARING REQUIREMENTS ACROSS MULTIPLE DISCIPLINES

Through the body of the report, all three of our leading users have commented on each of the three critical aspects for managing requirements – their early development, change traceability, and reuse.

Considering the first aspect, DOORS supports documents in the form of a set of objects that can be viewed through different filters, and provides varying security levels. The capability supports one central repository of all requirements, which helps maintain consistency across the multiple disciplines and phases of product development. Multiple views of the data serve different groups of people in an organization, as well as different use cases for the data, and can extend outward to be shared with suppliers.

Given the evolving corporate structure and profile for distributing work across organizations, multiple teams are far more likely today to be working on the same project from different locations. Synchronization of the distributed database becomes the biggest challenge given the likelihood of changes across the teams and geographies. The synchronization must support real-time access, as all changes must be viewed or approved by stakeholders, including subcontractors. Compare this to approaches performed only a decade ago, when few people would see and manage the real requirements in an organization. Today, at least half of the developers read the requirements. This sharing has been made easier by DOORS. "The idea that a programmer can get in and look at a low level requirement saves a lot of time later on. This is a huge help," states Ron Zorn of Zorn Industries. On this account, a web-based user interface becomes mandatory in distributing a requirements document by simply sharing the database over the web and supporting secure access control. The web access provides a real-time interface with the repository. Today, DOORS supports remote access via web browser and data exchange between remote databases. It does not, however, support and synchronize distributed databases.

Another payoff from DOORS consistently referenced by leading users relates to the ability to freeze requirements and create a baseline for a release. Organizations then rely on the baseline to manage requirements for various system releases, and teams may work in parallel on multiple releases at the same time. With the emergence of multidisciplinary engineering efforts in mechatronics and avionics, however, further extensions are needed to fully integrate with the product lifecycle management (PLM) software and to fully reconcile data across multiple engineering domains. Third parties such as Stoneworks provide software that links DOORS to PDM solutions.

Ambiguous and incomplete specifications present a crucial and challenging area for effectively managing requirements, especially since difficulties may not be discovered until relatively late in the development cycle. Moreover, the rising

complexity of the systems being designed and the increasing likelihood that multiple teams will be involved compound the complexity confronted for effective requirements management. All too often, these specifications are to be gathered from people's memory and knowledge. These are creative moments for the people who are trying to specify the requirements. Support from a requirement-gathering tool can help in structuring these creative moments, and can stimulate constructive analysis to accelerate an effective resolution. In addition, the SEI (Software Engineering Institute) has strong recommendations relating to the data gathering and review processes covered in their guide for process improvement, CMMI, or Capability Maturity Model Integration. The emphasis focuses on discovering and removing ambiguities and addressing incomplete requirements as early as possible. The Institute identifies this issue as the biggest cause of rework in the development and testing process. An emerging development methodology, Agile, also identifies these issues as central to system development.

Upfront system modeling and more detailed discussions with customers represent activities that may help identify ambiguous or contradictory requirements early in the design cycle. DOORS' support of structured and prioritized requirements fulfills these much needed capabilities. DOORS also provides a description of the functional structure and links between functions and requirements that map the two. With that foundation, Chrysler links DOORS with CMMI to audit and verify the development processes. "Bring in CMMI, or a tailored version of CMMI as an auditing tool, and it helps identify any lack of user input, incomplete requirements specifications, and the volatility of many changes with requirements." states Dr. Ed Griffor of Chrysler.

DOORS, with requirement document traceability, directly supports the implementation of quality standards, and system development methodologies like Agile, by providing unified document templates, by supporting consistent review processes, and by automatically generating audit reports. For Dr. Griffor of Chrysler, DOORS has been a valuable help: "DOORS supports a consistent approach for documenting your requirements. You create a template, and list the line items for the basic information expected, which clarifies what data to put in and how."

These leading users expect IBM Rational to extend its support of standards at higher levels as the development approaches continue to evolve, to automate more processes, and to improve the user interface. Any changes in basic templates involving values or attributes must be distributed to all affected DOORS modules. A wizard capability would be helpful that clarifies questions about quality standards or the development methodology to be used. Once the quality standard and development methodology is defined and implemented, DOORS can then improve on the automation of the processes and templates the organization and project team follows. Currently, most organizations generate hordes of documents and artifacts just to fulfill the requirements for quality without experiencing many of the benefits of standard and tailored processes.

Even after the standardization and automation of the processes supported by graphical representations is provided, the incomplete or ambiguous requirements may still be a concern. System simulation, or prototyping, then represents the most likely approach for dealing with problems of scope and ambiguity with requirements by identifying those requirements that have not been met. Hence, integration with simulation or modeling tools becomes a top priority. Existing interfaces with MATLAB/SimuLink do provide an integration of the requirements management tool with the simulation tools.

REQUIREMENTS STRUCTURE AND ROUNDTRIP TRACEABILITY

Traceability represents another fundamental need for requirements management. Originally it involved the tracing of requirements at various specification levels, such as system specifications that must be clearly linked to each subsystem specification, and vice versa. Organizations like Chrysler and EADS dramatically extend the approach to clearly understand when, why, and by whom a change has been implemented. The structure for requirements, with links between system-level requirements, sub-systems, and component requirements, provides the ability to analyze the impact of changes at the different levels and across disciplines. The top-down view helps with coordination and planning, monitoring of progress, and confirmation of compliance to requirements.

To fully capitalize on the potential improvements available from the emerging trends, requirements management tools will need to fully integrate with other product lifecycle management software covering simulation and development, as well as testing. For example, each requirement object will need to be linked directly to a simulation model, a piece of code, or a test case. While the complexity and challenges in managing the effort increases as the effort drives deeper into finer levels of granularity, the payoff presents the potential of fully coordinating efforts across all disciplines in real time. Another major and direct benefit of building a requirements structure that dramatically improves the visibility of a project's status is the identification of a missing simulation model, a fragment of source code, a test case, or even a test result.

An important aspect of requirements management relates to the reviews and sign-offs. Every responsible party with components affected by a particular change must review and approve the proposal. Whole teams may need to collaborate and agree on specific methods and approaches in implementing a change. For example, if the hardware has already been fully specified and validated, a change in software may be needed. An approval process supporting conditional approvals also helps avoid a multiplicity of reviews. The conditions identify those special states that must be met, and focus the attention of all those involved to more quickly respond to any proposed change in requirements.

Chrysler relies on a test data management tool linked with DOORS. Although they expect to benefit from improvements in the integration, the current links ensure that every requirement is supported by at least one testing document. This is an excellent method to verify that the requirement is not ambiguous and is testable.

A graphical representation of the requirements structure, traceability matrices, and other data may significantly ease the management of complexity. System requirements are most often broken down into subsystem and component requirements. A clear graphical view laying out the connectivity specifications helps to visualize the system. The approach considers the whole system as black box, and then breaks it down into subsystems, and then into components at increasing levels of detail. The visualization of the whole structure is supported by the layout of interconnectivities as attributes with bi-directional traceability. Each black box also links to a requirement document. In effect, each requirement then expands to reference a model, and vice versa, the reference model refers to specific requirements as well. As a document-oriented tool, however, DOORS needs to integrate with visual software tools to support these needs. Indeed, an IBM partner, Integrate Systems Engineering Ltd., supports graphical traceability with their add-on product called TraceLine.

Bottom-up traceability enables designers and analysts to track all areas impacted by a change to verify that the expected impact does address the targeted requirements, defects, and/or requests for change. The development effort must trace back to the original requirements. Otherwise, developers may work on the wrong version of the specification or not fully understand the context and business value. Roundtrip traceability supporting both top-down and bottom-up tracking helps prevent unnecessary, unfocused development, and costly rework.

REQUIREMENTS REUSE

The volume of requirements that must be addressed continues to increase significantly. As Sharon Crossby of EADS Astrium concludes, "With too many requirements, and internal business processes and constraints, the volume itself and the associated issues of potential contradictions becomes a much bigger challenge than any missing requirements." As a result, the reuse of established requirements becomes paramount, involving the ability to focus on the differences between the requirements for the existing and established designs, with those for the new version, to meet emerging needs. Moreover, since there are far more sets of test plans than requirements themselves, the focus on these differences necessarily extends to simulation and test. Indeed, it extends to the consideration of the system architecture, and source code as well. In terms of the full integration with simulation, as an example, each step, each state, and each transition references its counterpart requirements. Any changes immediately flag the associated requirements that are affected, which in turn notifies all other functions impacted. Clearly, the challenge of integrating the complex data produced by a multitude of simulation and modeling tools raises challenges in tracing and linking the changes, given their specific and narrow focus.

CONCLUSION

The three leading firms across automotive, health care, and aerospace in this review face a diverse set of challenges in their product development processes. The pairing of hardware and software, as well as the frequency of changes in software development, drove requirements management at Chrysler. The complex tracking of all changes to requirements, partly driven by regulatory needs, fully justified DOORS at Cardinal Health. At EADS Astrium, both quality and cost would be at risk without DOORS. Despite their dramatic differences, requirements management has proven to be a critical initiative that will witness far broader participation across all disciplines in the full product development cycle. The approach defines a framework for collaboration and integration serving remarkably diverse areas of expertise.

The tool itself has gone a long way in improving support, with enhancements that have evolved and changed rapidly over the last three years. The payoff has grown in the process. Indeed, these organizations appear to be highly dependent on the tool. As Susan Hacker of Cardinal Health states explicitly, "We could not manage the full development process without DOORS."

All three users suggest that a successful program in requirements management must extend well beyond the software tools to directly address people and process. Indeed, the challenges related to requirements-driven development relate more directly to the evolution of the development culture itself than the current capability of the software tools. The tools support the people and the process well, but the people must have the understanding needed to fully embrace the approach, and the process must be well defined as the foundation.

---

i Sharon Crossby has been working in the space industry for fifteen years and in the requirements management domain for the last twelve. After running her own business as a prototype wireman, she earned a BEng with honors in Engineering and Engineering Systems from Portsmouth Polytechnic in 1991. While working at Matra Marconi Space as a verification engineer on the ESA Envisat ASAR project, she became involved in closing out retrospective customer requirements. This long and complex activity started Sharon on the course of developing and implementing effective requirements management at Astrium. Sharon is now heavily involved in the coordination effort of improving requirements management across all of Astrium. She has the role of Deputy Chair of the EADS PHENIX Requirements Management Task Force, with a special interest in the implementation of EADS policies and the efficient management of EADS licenses for DOORS. EADS is a global leader in aerospace, defense, and related services with revenues of 43.3 billion Euros in 2008, and a work force of 118,000. The group includes Airbus, Eurocopter, and EADS Astrium.

As the Senior Software Technical Editor for Clinical Technologies and Medical Products at Cardinal Health, Susan Hacker directly supports the DOORS database and process development for requirements management. With over eleven years of experience in the effort with Cardinal Health, Susan began her career in the health sciences with Pyxis Corporation in 1994, which was later acquired by Cardinal Health. Earlier, she served as an advanced electronics instructor at the ITT Technical Institute in San Diego, and as an electronics technician for the Navy.

Dr. Edward Griffor is one of the two existing Walter P. Chrysler Technical Fellows, one of the highest technical specialist positions in the sector covering automotive and other industries. He is Chairman of The MIT Alliance, a professional association of scientists, engineers, and business experts trained at the Massachusetts Institute of Technology. Ed completed his doctoral studies in EE and Mathematics at MIT in 1980 and was NSF/NATO Postdoctoral Fellow in Science from 1980-83 in Norway. He also taught in the U.S. and abroad, and is regarded as a world expert in the use of mathematical methods to manage the design of electronic controls and algorithm/embedded SW design. In addition to his

work at Chrysler LLC, Dr. Griffor is an expert in systems biology and Adjunct Professor at the Center for Molecular Medicine and Genetics of Wayne State University in Detroit.

Ron Zorn from Zorn Industries benefits from over fifteen years experience in software configuration management. He has extensive skills from performing on software technology projects in all phases of the software lifecycle from concept through implementation and maintenance. Ron has worked with Telelogic Synergy since it was Continuous 4.4. He has been a member of the Telelogic Leadership Council since 2005 and on the Executive Advisory Board since 2006.

ii CMM and CMMI are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.