syncroness
innovative product development

**NOVEMBER 10, 2013**

Syncroness, Inc.
Corporate Office
10875 Dover St., Unit 200
Westminster, CO 80021
Phone: (303) 429-5005
Fax: (303) 429-5025

# MODEL-BASED SYSTEMS ENGINEERING – AN ENABLER FOR REGULATORY DESIGN COMPLIANCE

**AUTHORED BY JERED DEAN, CODY HENDERSON & JOHN GARDNER**
**SYNCRONESS INC**

## THE PROBLEM

When working on new products, or even enhancements to existing products, it is easy to lose sight of the end application and user experience. This is particularly concerning when developing products on a schedule and budget and where investment has been made. Without model-based systems engineering, design teams will find it difficult to define concretely the performance and regulatory requirements of the devices they develop, adding additional risk and uncertainty to the effort.

## EXECUTIVE SUMMARY

Whether working in the FDA regulated medical field or explosive industrial environments, using model-based systems engineering provides the confidence, traceability and structure to meet regulatory requirements. In addition, we propose the methodology to be a means to optimize schedule predictability and development of the most appropriate products. In this paper, the systems engineering process being employed is briefly described and two case studies are explored that make the case for MBSE as a key enabler of efficient design to complex regulations.

## INTRODUCTION

Use of model-based systems engineering (MBSE) tools and systems engineering best practices provides the confidence, traceability and structure to efficiently develop products in regulated environments. This has been found to be true whether working in the United States Food and Drug Administration (FDA) regulated medical field or in explosive industrial environments. By capturing requirements, functional definition, architecture and verification activities in an integrated database model, the burden of proving compliance to an auditor is significantly simplified. In this paper, the systems engineering process being employed is briefly described and two case studies are explored that make the case for MBSE as a key enabler of efficient design to complex regulations.

Case study one describes Customer A who is designing a medical device compliant with FDA requirements while using less than 10% of the project man hours on systems engineering and regulatory compliance activities. It is demonstrated that the output of the system engineer meets the majority of the FDA design control requirements. Moreover, due to the systems engineering effort, the involvement of the quality/regulatory roles on the design phase has been minimized to final approval of the generated documents.

The second case study describes the use of the MBSE database as a requirements management tool for the analysis of complex standards. Customer B works in the heavily regulated explosive atmospheres industry and used MBSE to wade through complex, parallel requirements to establish a clear verification roadmap for future products. This effort has reduced the time required to develop a new product by over three months and has given the design team the confidence to innovate while minimizing the cost and schedule impact of third party certification.

## BACKGROUND

Syncroness is a product development company focused on contract engineering for a variety of industries, including medical devices, consumer products, test and measurement, oil and gas, and aerospace. Syncroness employs approximately 60 full time engineers with skills in project management, systems, mechanical, electrical, and software engineering. Much of the work is project based, and integrates the various engineering disciplines along with industrial design, human factors, and graphical user interfaces.

Due to the wide variety of industries in which Syncroness works, the systems engineering process has been developed with an emphasis on flexibility so that minimal changes are required to the baseline tools, regardless of the target industry. Customers in this business space often question the value of activities which fall outside the realm of traditional engineering efforts (design, CAD, analysis). By necessity , the system engineering process presented in this paper has been honed to create scalable tools to match the level of effort and realize the highest return on investment.

## THE APPROACH

### INTEGRATION OF THE PROJECT MANAGER & SYSTEMS ENGINEER

One theme that is referenced throughout this paper is the difference between a project manager and a system engineer. For the purpose of this paper, a project manager, (PM), is considered the individual who manages the project deliverables including scope, cost, and schedule. In addition, the PM often is responsible for project risk management, resource allocation and interaction with the customer. In contrast, a systems engineer, (SE), manages the system definition, technical risk, and implementation of the technical design. In general terms the PM manages the project while the SE manages the product design.

There has been a lot of work specifying and separating the domains of the Project Manager from that of the Systems Engineer (Kasse 2003, Haskins 2010); one of the more clear delineations is provided by the NASA Project Management and Systems Engineering Framework (NASA, 2011). It is the authors' experience that for small to medium projects, use of separate resources for project management and systems engineering tasks overly burdens the project. Alternatively, a combined resource that embodies both the project management and systems engineering tasks adds significant value (through synergy) and results in a lower overall project cost. This bias for a combined project manager/system engineer role is assumed throughout the paper and supported by the discussion on regulatory compliance provided below.

### PROPOSED SYSTEMS ENGINEERING PROCESS

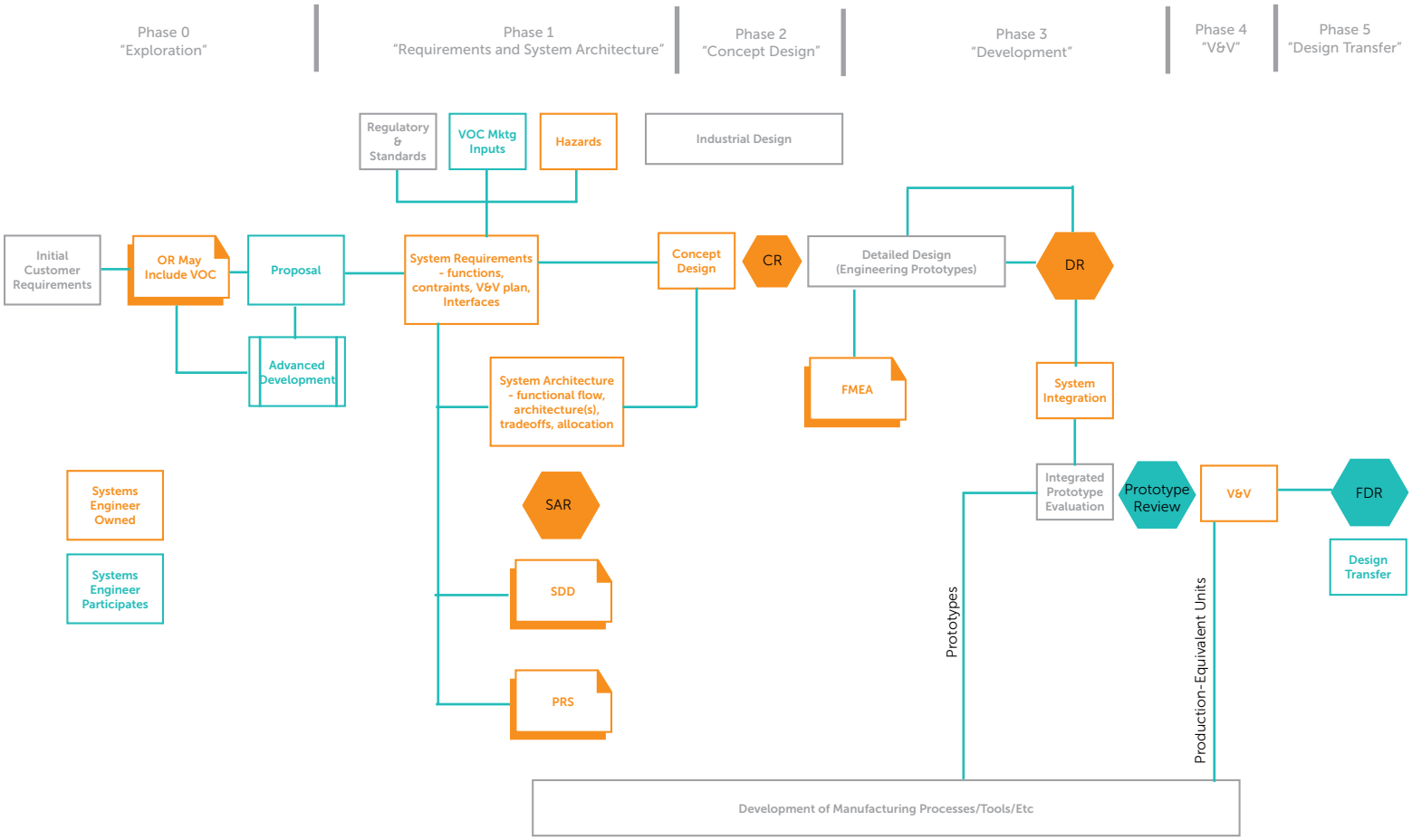
Key to a successful implementation of systems engineering for small to medium projects is the ability for the systems engineering process to be scaled. The following process in Figure 1 was developed to accommodate the wide variety of industries and size of projects seen in the engineering contract environment. One of the primary goals of this process is to allow flexibility through dictating high level activities, but not low level process.
The SE process is integrated into each of the product development phases. In Requirements and Systems Architecture, the systems engineer works on product definition including activities of Hazard Analysis, System Requirements, and Architecture. During Concept Design the systems engineer is engaged to ensure the product meets the system definition. During Development, the systems engineer works with the other technical resources to perform failure modes analysis and acts as a compliance check during the design review process. The systems engineer owns the integration effort and verification of subsystems and interfaces.

While none of the activities shown in Figure 1 are necessarily new or novel, there are several elements that have been tailored or added to the process. The first item of note is the integration of human factors, industrial design and use case definition (functional flow) into the very early stages of the requirements gathering and architecture definition process. The authors have found that early engagement of industrial design leads to more innovative designs faster, as opposed to waiting until the concept design stage. Similarly, involvement of fabrication, assembly, inspection and test (FAIT) representatives in the initial architecture discussion is critical to project success. The FAIT representatives are continually engaged throughout the process for early architecture brainstorming through to production release –this addition minimizes downstream manufacturability changes. This early and continuous involvement helps capture manufacturing issues early, typically achieves cost of goods sold savings, and gains buy-in from a key group of stakeholders. The goal is to synchronize and integrate the design transfer process (production) with the design process for seamless handoff at the end of the project.

Another key element of the process is the use of a risk driven prototyping strategy. Prototypes (also called mockups or engineering prototypes) are created throughout the design process for early, mid and end-game verification/validation of the design. These prototypes are planned as part of the risk mitigation activity and their testing is tied to the verification/validation requirements. If there are significant technical risks, some form of prototype will be built to address them or if a verification requirement looks difficult to achieve then a prototype will be created as early as possible for the specific purpose of that requirement. One example of a risk driven prototype would be a foam mockup to validate a soft requirement such as "the product should be ergonomic." Using the foam mockup to gain agreement on what "ergonomic" is early in the project avoids significant design modifications later. These engineering prototypes are created as early as the system architecture phase and continued through to the final, fully functional prototype validation activities. On projects with a high technical risk (low technology readiness level) the prototype early belief is pushed further and an "Advanced Development" stage is added prior to requirements definition to increase the technology readiness level (TRL) of the proposed system.

The following case studies used the process detailed in Figure 1, and one individual in the role of a PM/SE, in combination with the model-based systems engineering tool CORETM. Model-based systems engineering tools are typically used by Syncroness on large projects and projects that involve a regulated environment. For smaller projects in unregulated environments the level of effort and documentation applied to these activities is only as large as is needed. Details on this scalable approach for non-regulated, small to medium sized projects can be found in Kolozs et al (Kolozs et al. 2011).

## FIGURE 1: PROPOSED SYSTEMS ENGINEERING PROCESS



## THE CASE OF CUSTOMER "A"

Customer A specializes in the periodontal care market, making products for use by both dental professionals and consumers. Syncroness is currently engaged by the client to redesign an existing product for increased functionality as well as system cost reduction. MBSE enabled systems engineering is being used on this project due to the FDA regulated environment, although the project is considered medium sized (approximately 60 man weeks).

The periodontal device being designed has been categorized as a Class 1 medical device, which requires that FDA Design Controls apply (FDA 2010). An FDA investigator would examine the design history file (DHF) on a Class I device in the event of an audit and expect to see similar documentation to that contained in a Class II DHF.

The Food and Drug Administration details good manufacturing practice for medical devices in Title 21, Part 820, of the Code of Federal Regulations (CFR). While Part 820 has many sections on quality system requirements, purchasing, and other enterprise level concerns, Subpart C covers the requirements for Design Controls during product development (FDA 2010) which is the focus of this paper. The list of required design controls is surprisingly concise requiring only that design and development must be planned, design inputs tracked, design outputs clearly defined and quantified, the design be reviewed. In addition, Subpart C requires specific planning and documentation of verification, validation, design transfer and changes. Table 1 shows a listing of the FDA required design controls. Importantly, the FDA mandates that the listed design controls exist and are maintained, not how they are created or maintained. This flexibility in the CFR leaves the door open for the use of standard systems engineering outputs to meet the requirement.

Table 1 shows how the bulk of the FDA required design controls are satisfied by the output of the system engineer. The far left column contains the FDA required design control areas, the center column lists the proposed documents/deliverables that meet those design control requirements, and the far right column lists the proposed owner of those documents. Any documents which are not the responsibility of the systems engineer are primarily controlled by the project manager or are direct outputs of engineering required for any design effort.

Due to the overlap of the responsibilities of the project manager and the system engineer in regulated environments, it makes sense to have one person play both roles when possible. In the case of Customer A, the project manager and system engineer roles were filled by one person. In addition to aiding with regulatory compliance, combination of the two roles typically leads to a more cohesive combination of technical and business roadmaps and risk management activities. If the size of the project requires that the roles be split between two individuals, it is extremely helpful if they are each qualified for either role and/or have strong communication skills. The role of the system engineer, like the role of the project manager, requires strong leadership ability and emotional intelligence (Thomas 2011).

| FDA Requirement | Process Record/Deliverable | Record Owner |
|---|---|---|
| **Design & Development Planning** | Project Plan | Project Manager (PjM) |
| **Design Input** | Voice of Customer (VOC)<br>Risk Management (Hazard Analysis)<br>Regulations<br>Product Requirement Spec (PRS)<br>System Design Document (SDD) | Systems Engineer (SE) |
| **Design Output** | PRS Verification and Validation Plan<br>Project Plan<br>Technical Outputs (e.g. drawings, schematics, etc.) | Systems Engineer<br>Project Manager<br>Engineer |
| **Design Review** | System Requirements Review (SRR)<br>System Architecture Review (SAR) | Project Manager Coordinates<br>System Engineer Reviews |
| **Design Verification** | PRS Verification and Validation Plan<br>Detailed Verification Plan | System Engineer Creates<br>Project Manager Oversight |
| **Design Validation** | Validation Plan | Typically Performed by Customers |
| **Design Transfer** | Project Plan | Project Manager Plans/Coordinates |
| **Design Changes** | Engineering Change Order (ECO) | Engineering<br>Project Manager Approval |
| **Design History File** | Combination of the above items into cateloged directory | Project Manager and Systems Engineer Approval |

Table 1:
FDA Process Records

## CUSTOMER "A" CONTINUED...

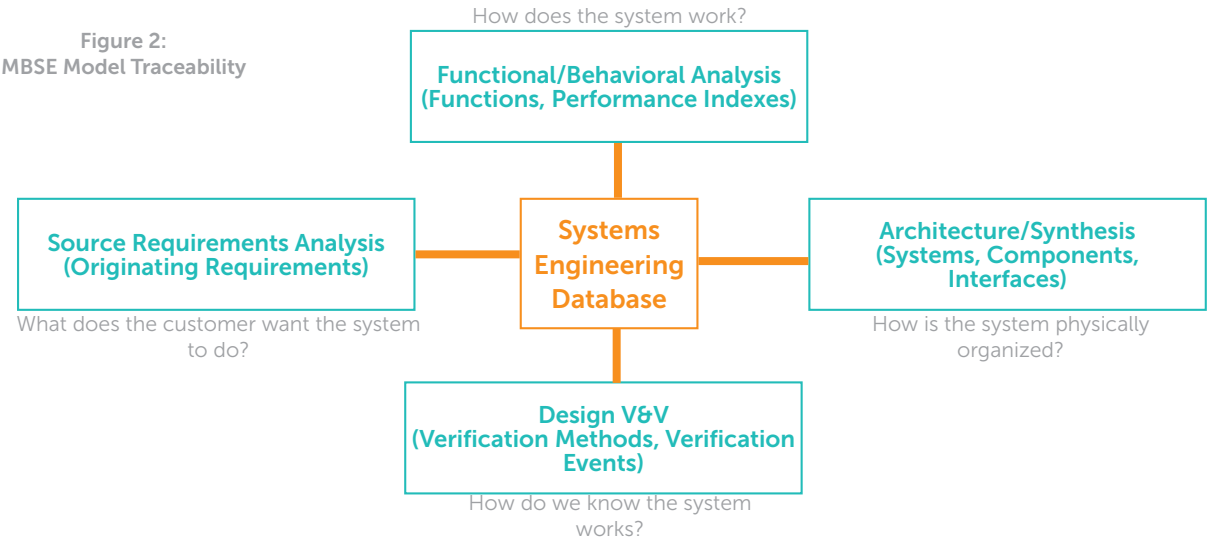
What model-based systems engineering (MBSE) adds to the regulatory compliance equation over non model-based approaches is reduced workload and full traceability. Two of the most complex documents listed in Table 1 (the PRS and SDD) are direct outputs of the MBSE database. Once the database has been established, updates can be quickly made and propagated throughout the design control documents without concern for conflicts. Moreover, the full traceability provided by MBSE tools, such as CORETM, between requirements, architecture, functions and verification activities means that the design team can be confident in compliance in the event of an audit. Figure 2 shows graphically this traceability framework.

While the focus of this paper and specifically this case study is based on regulatory compliance associated with development within the United States, it should be noted that the process may be extended easily to development in other countries. Most Asian/Asia-Pacific medical regulations are based on the FDA and follow the model directed in this paper. In Europe, due to efforts for communization of regulations there is an increasing correlation between FDA and ISO requirements. In some cases it has been found that there is a need to align the unique regulations (MDD, for example) with FDA. One method to address this is to capture all requirements from this additional regulatory agency into a model-based systems engineering repository (such as CORETM) and then correlate the requirements one for one with the previously captured and mapped FDA requirements. MBSE significantly eases this correlation burden by tying the requirements directly to the verification activities planned for development.

Alternative, non model-based systems engineering repository tools are also widely available and may be used on projects. It has been the authors' experience that these tools do not fully envelop the definition of the product and thus require additional systems engineering tools (in the form of paper-based tools) to complete system definition. The separation in tools thus results in a higher likelihood that changes which occur through the product development cycle do not get captured and propagated throughout all tools used on the project resulting in potential discrepancies in SE documentation.

Model-based systems engineering, as proposed in this paper, addresses the key FDA requirements for design controls during system development. Using the tools described above, customer A is in the process of taking a Class I medical device from concept through to full production in less than twelve calendar months. On this twelve month project only a fraction of the man hours (less than 10%) will be devoted to systems engineering. However, the design team is confident of FDA compliant design controls due to the traceability provided by MBSE, and that the appropriate-to-the market product will be designed, developed and producible at the desired cost. Moreover, the team is confident that the project will be completed on time and within budget. It is the authors' experience that a a medical device development effort using the SE/MBSE process may take 12 to 20 months as compared to the same effort taking 3 or 4 years without the SE discipline. This same trend has been documented by (Gardner 2001). Noted main efficiencies gained by implementing systems engineering, and especially model-based systems engineering, is the reduction in need for retest due to late entry requirements found during Verification tests or field Validation.

Figure 2:
MBSE Model Traceability



## THE CASE OF CUSTOMER "B"

Customer B produces measurement devices for use in explosive, industrial environments. Syncroness was engaged by the customer to lead a multidisciplinary team through the definition of a common platform which could be leveraged across the design of multiple products. In addition to definition of the common platform, the customer had not been introduced to systems engineering -- part of the project was introduction of best practices to the client team through leadership of the systems engineering process for one full development cycle.

Initial discussion with the client uncovered that they had not significantly changed their product base for many years. While there were many reasons for their reluctance to make changes, two of the major drivers were (1) concern of regulatory compliance and (2) concern with product field failures. Introduction and application of the systems engineering process focused the efforts of the team on tasks required for definition of this common platform. Model-based systems engineering provided the means to capturing in one cohesive model the regulatory requirements, system use cases, functional requirements, and verification activities.

Customer B was right to be concerned with regulatory compliance. Their products fall within the complex regulatory environment of explosive atmospheres (both gas and dust). These environments are regulated by multiple bodies dependent on the country, with the IEC 60079 series of standards representing the closest to a universal set of requirements available (IEC 2010). The common platform (and any design effort) needed to fully conform to eight complex IEC standards as well as multiple, parallel standards from other governing bodies such as the Canadian Standards Association (CSA) and the American National Standards Institute (ANSI).

Model-based systems engineering allowed the creation of a master verification plan that could be applied to all future products. This was very appealing to the customer who reported past experience with significant predictability issues due to V&V failures and/or post-launch re-design efforts. Using CORETM, each of the key specifications was decomposed section by section, requirement by requirement. This produced a fully traceable and searchable requirements "tree." Using the MBSE database as a regulatory management tool, parallel and/or repeat requirements could be mapped to a common, single requirement and standards could be filtered from the specification on demand. Once the specifications had been decomposed, verification activities were created and mapped to the requirements until all requirements were verified by some means. These verification activities ranged from keeping data sheets on file to design reviews to physical tests. In the end, MBSE allowed the creation of a list of verification activities, tied to specific standards, which fully test the product to the standards and can be leveraged on future projects. Moreover, the process gave the team the confidence needed that new designs could meet the complex regulations. Another benefit realized by this system model approach is the output of the model as a superb input to 3rd party certification groups.

Concerns with product field failures were alleviated during the project via similar means. Customer B had an existing list of verification activities that had historically been done on new products but which were not linked to functional requirements. As part of the project, enhanced functional flow block diagrams (eFFBD) were created to graphically document the intended functions of the product. These functions were then mapped to new, function-specific verification activities. Figure 3 shows an example of an eFFBD. This exercise highlighted many features and functions that had never been formally verified on the existing products; some of which had led to customer product returns in the past. The emphasis on functional flow also served as a key tool in development of the actual physical component common architecture. More information about using MBSE for common physical architecture development can be found in (Gardner 2001).

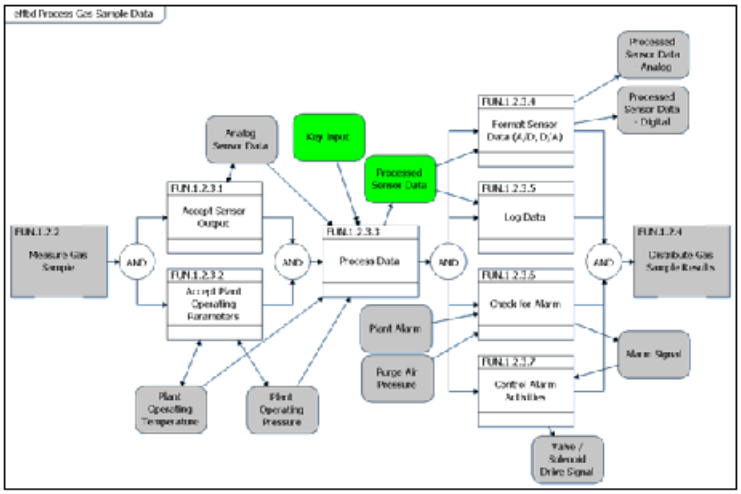



Figure 3:
Enhanced Functional Flow Block Diagram (EFFDB)

## CUSTOMER "B" CONTINUED...

When the functional and regulatory verification activities were combined into a cohesive verification plan, the customer had a clear picture of the steps required to confidently launch a new product. Due to the extensive regulations, over 300 individual requirements were identified for the common architecture. However, this list of requirements only lead to approximately 60 verification events which included everything from required design reviews and calculations to actual, physical testing. Even more importantly, thanks to the traceability provided by MBSE tools, when questions arose during certification Customer B could provide clear, logical explanations of the verification done and proven acceptability of the design. In addition, the customer has seen great value in now being able to accurately predict the time and cost of accomplishing V&V.

Initial preparation of the common architecture required approximately five calendar months and twelve man weeks to complete. It is anticipated that this common architecture effort will be used on more than six products and have a life of three to five years. The effort has led to a significant decrease in expected product development time with an estimated three month reduction in development time per product. Additionally, the common system model requires very little rework for each new project and can be updated in a matter of a few weeks for the development of a new product.

In comparison to alternative systems engineering tools such as non model-based parametric databases and traditional paper-based tools, MBSE has allowed for a single database to completely define the system. This allows for the changes made in one area of the model to propagate through all deliverables associated with the systems engineering process. This is of particular importance and benefit in the development of products for highly regulated environments as regulations often change and lead to updates required in system definitions. Use of non-linked and paper-based tools requires identified changes to be made uniquely and separately to the deliverables associated with sytem definition and does not give any feedback to whether the changes were implemented in a common manner or completed across all deliverables associated with definition of the product.

Looking forward approximately two years, Customer B will be selling into a new set of regulations that will make MBSE an even more vital part of their development efforts. In an effort to increase plant safety, IEC developed the 61508 family of standards which define the required design controls for functional safety of electrical, electronic, and programmable electronic safety related systems (IEC 2010). This set of standards is similar to the FDA requirements of Title 21 in that the standards do not focus on mandating design constraints but instead define a set of design controls and process that should be followed. In another parallel to FDA discussions, adherence to the IEC 61508 standards is proven primarily through a quality audit of the design documentation as well as the enterprise quality system. Table 2 shows the high level design control requirements of IEC 61508 and is based on (IEC 2010) and (Medoff et al. 2010).

As shown in Table 2 the majority of the required design controls are the responsibility of the system engineer. Similar to the discussion regarding Customer A, MBSE tools allow for concrete proof of compliance with the standard and, in the event of an audit, full traceability between the separate process records. The authors hope to work with one of the primary IEC 61508 certification labs and Customer B over the coming year to prove out that use of the process and MBSE will meet the requirements of the standard. Initial meetings have proved promising and appear to confirm the hypothesis presented in Table 2.

## CUSTOMER "B" CONTINUED...

| IEC 61508 Requirement | Description | Process Record/Deliverable | Record Owner |
|---|---|---|---|
| **Functional Safety Management Plan** | A documented plan for project management as it relates to safety functions | Project Plan | Project Manager (PjM) |
| **Product Safety Requirements** | Capture the safety integrity level requirements and list the safety functions of the device. | Originating Requirements<br>Risk Management (Hazard Analysis)<br>Regulations<br>Product Requirement Specification (PRS) | Systems Engineer (SE) |
| **Safety Validation Test Plan** | Create safety validation test plan mapped to safety requirements and functions. | PRS Verification and Validation Plan<br>Detailed V&V Plan | Systems Engineer |
| **System Architecture Design** | Clearly define architecture and identify interfaces; analyze failure modes | System Design Document (SDD)<br>Failure Modes and Effects Analysis (FMEA) | Systems Engineer |
| **Hardware Design and Implementation** | Some standard specific test such as component de-rating, ASIC requirements and fault injection are required | Drawings, schematics, reports<br>IEC specific fault injection/FMEDA analysis | System Engineer<br><br>Project Manager Oversight |
| **Software Design and Implementation** | Proper software engineering practice outlined | Software Architecture Specification (SAS)<br>Software Design Specification (SAS)<br>Software Analysis | Software Engineer |
| **Integration and Safety Validation Testing** | Execute integration testing of hardware and software components | Detailed Verification Plan and Report (DVP&R) | Systems Engineer |
| **Process Validation** | Final, internal audit of process records for completeness | Audit Report | Project Manager |

Table 2:
Functional Safety Design Records

## ABOUT SYNCRONESS

Syncroness is a contract engineering firm located in Westminster Colorado. With specialties in mechanical, electrical, software and firmware engineering, we focus on supporting our clients through our three core services: New Product Development, Sustaining Engineering and Production Equipment Design. Flanked by professional project managers and systems engineers, the company concentrates on serving highly-regulated industries including medical device, aerospace and defense.

## CONCLUSIONS

Whether working in the FDA regulated medical field or explosive industrial environments, using model-based systems engineering provides the confidence, traceability and structure to meet regulatory requirements, optimally develop the right product and cost-effectively prove compliance. In addition to the MBSE advantage, the authors propose that combining the roles of the project manager and system engineer cost-effectively aids regulatory compliance. Customer A is finding that use of model-based systems engineering as proposed in this paper efficiently produces the required design controls for FDA regulatory compliance. This has meant very low systems engineering cost (less than 10% of the project hours) without sacrificing confidence in the quality of their design history file. For Customer B, model-based systems engineering has provided a reusable roadmap for compliance with complex, parallel standards in the explosive atmospheres environment. Moreover, Customer B is using systems engineering to lay the foundation for compliance to new functional safety requirements entering the marketplace. Both customers gained a real, quantifiable advantage by using the proposed systems engineering approach to tackle design in regulated environments.

## REFERENCES

"Explosive atmospheres – Part 0: Equipment – General requirements," International Electrotechnical Commission (IEC), IEC 60079-0 ed6.0.

"Functional safety of electrical/electronic/programmable electronic safety-related systems," International Electrotechnical Commission (IEC), IEC 61508 ed2.0.

Gardner, J.R., 2001. "Design of a Common System Architecture for a Medical Device Application," Paper presented at the 11th Annual International Symposium of the International Council On Systems Engineering – INCOSE, Melbourne, Australia, Paper P104.

FDA, 2010. "Good Manufacturing Practice for the Medical Devices," Code of Federal Regulations 21, Part 820. Food and Drug Administration. Revised April 1, 2010.

Haskins, C., ed. 2010. Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities. Version 3.2. Revised by M. Krueger, D. Walden, and R. D. Hamelin. San Diego, CA (US): INCOSE.

Kasse, T. 2003. "The Differences Between the Project Manger and the System Engineer." Paper presented at the 6th Annual System Engineering Conference - NDIA, San Diego, CA (US), 10-23 October.

Kolozs, J., Henderson, C., Gardner, J., 2012. "Systems Engineering Lite." Paper presented at the 22nd Annual Internaltional Symposium of the International Council On Systems Engineering – INCOSE, Rome, Italy.

Medoff, M.D., Faller, R.I., 2010. Functional Safety – An IEC 61508 SIL 3 Compliant Development Process, Exida, Sellersville, PA.

NASA, 2011. NASA Project Management and Systems Engineering Competency Framework, www.nasa.gov/offices/oce/appel/pm-development/pm_se_competency_framework.html, Accessed on Nov. 4th 2011.

Thomas, J.A., 2011. "Wanted, System Engineers with Moxie," Presentation, http://community.vitechcorp.com/forum/default.aspx?g=posts&t=67, Accessed Nov. 4th, 2011.