An approach to the lean development of complex products

CONFERENCE PAPER | Complexity, Quality Management | 19 July 2006

Pessôa, Marcus Vinicius Pereira | Loureiro, Geilson

How to cite this article:

Pessôa, M. V. P. & Loureiro, G. (2006). An approach to the lean development of complex products. Paper presented at PMI® Research Conference: New Directions in Project Management, Montréal, Québec, Canada. Newtown Square, PA: Project Management Institute.

Marcus Vinicius Pereira Pessôa

Geilson Loureiro

New product development can be understood as some kind of information-based factory (Bauch, 2004): Its typical phases (Figure 1) include creating, gathering, and evaluating information and reducing risk and uncertainty at the same time with the target to gradually develop a new and error-free product which can then be realized by manufacture, sale and delivery to the customer. Through these value identification, value implementation, and value delivery cycles, the integration between product, process and organization creates a complex system that daunts engineers and managers; where complex system formally refers to a system of many parts that are coupled in a nonlinear fashion, and given the properties of the parts and the laws of their interactions, it is not a trivial matter to infer the properties of the whole (Simon, 1969).

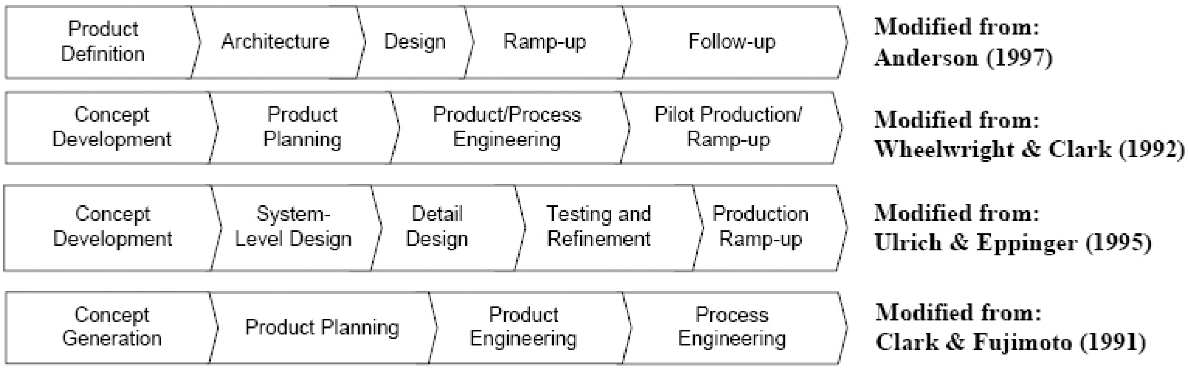

Figure 1: Typical product development phases.

Furthermore, in complex engineering products or systems (such as cars, aerospace systems, software, or industrial equipment) the coupling of individual components or modules may turn engineering changes in a component into “snowballs,” in some cases causing long rework cycles and turning virtually impossible to anticipate the final outcome (Mihm, Loch, & Huchzermeier, 2002).

This work proposes an approach to the lean development of complex products. Lean development aims to create value, with its two elements of doing the right job and doing the job right (Murman et al., 2002); where the former maximizes the value creation, while the latter minimizes the waste in the process.

The approach considers the value delivery complexity as including product, processes, and organizational elements (Loureiro, 1999). It aims to derive a project plan that is based on a network of value confirmation events that pulls only the necessary and sufficient information from product development activities. Such a scheduling method is being applied to aerospace systems projects and has a high potential to be spread over other complex engineering projects.

This paper is organized as follows: First, the concepts of value creation and waste reduction are presented in the context of product development. Next, the approach and an implementation alternative are presented. Finally, preliminary results are discussed and the information is summarized in the concluding remarks.

Value Creation

Value is the basis for lean thinking. The value, as defined by the stakeholders, pulls the product development, which itself pulls the project planning. A plausible definition of value in product development is “the right information products delivered at the right time, to downstream processes/customers, where it is quantified by form, fit, function, and timeliness of information products” (Walton, 1999, p.17).

Every project task should be directed toward creating deliverables, which themselves materialize the identified value. Thus, an activity on a project is value-added if it transforms the deliverables of the project in such a way that the customer recognizes the transformation and is willing to pay for it (Mascitelli 2002). The value life cycle through product development can be represented by the value creation framework (Figure 2).

Value Identification: Involves identifying the stakeholders and their value needs or requirements

Value Proposition: Where the needs of key stakeholders come together

Value Delivery: Creation of the interconnected chain of activities (“value stream”) to convey the benefits required to the stakeholders (not only the client). Value delivery depends on adding value at every step along the value stream.

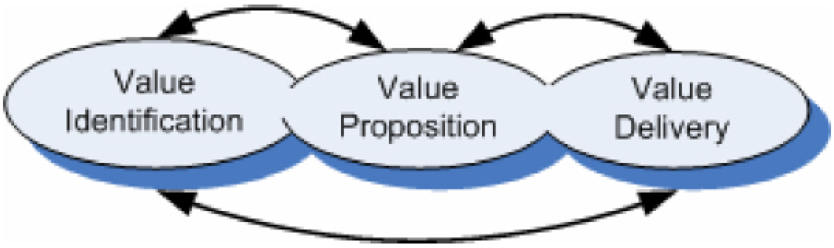

Figure 2: The value creation framework (Murman et al., 2002, p. 11).

Systems Engineering and Value Identification and Proposition

The inherent complexity in the development of complex engineering products is a serious obstacle to value creation. By concurrent engineering principles, when a product is conceived it already constrains its life-cycle processes and the organizations that perform those processes. In order to manage the inherent complexity of the product development activity, one must consider from the outset product, processes, and organization in a concurrent and integrated manner; otherwise complexity will escalate. Failure to encapsulate the problem to be solved promotes complexity escalation, according to Warfield (1994).

By the characteristics presented, the product plays only a part of the whole complexity. The total view framework (Figure 3), proposed by Loureiro (1999), represents the complexity associated not only with a product that is itself complex but also with the interactions among the product, its life-cycle processes and organizational attributes: It “is a modeling framework that integrates the product, its life-cycle processes, and the performing organization throughout the requirements, functional, and physical analysis processes, at all levels of the product hierarchy, deriving attributes as emergent properties of a whole integrated system” (Loureiro, 1999, p. 152).

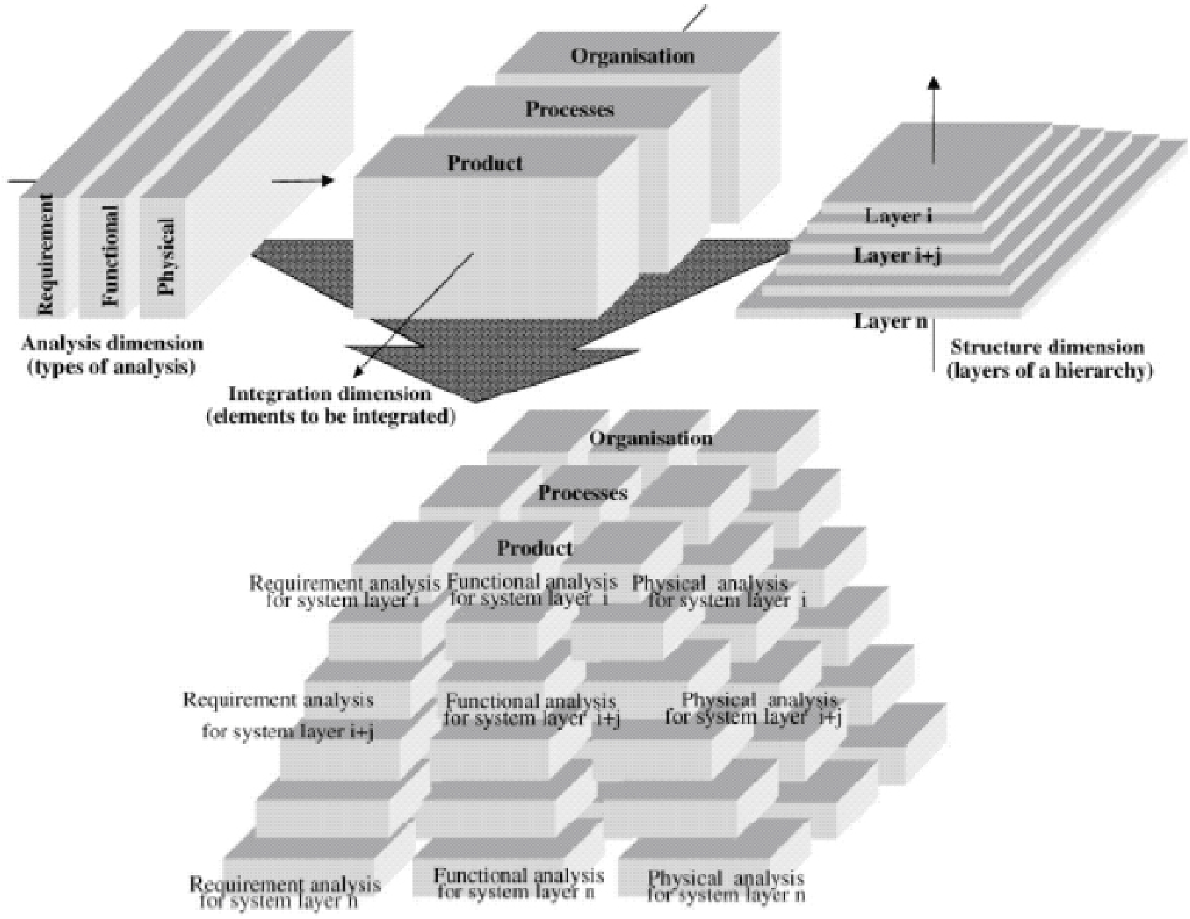

Figure 3: The total view framework (Loureiro, 1999, p. 152).

This work advocates a total view while applying systems engineering to value identification and proposition in the development of complex engineering products, where, in each part of the system, product, process, and organization are considered.

Waste Reduction

On value delivery, the lean thinking main goal is to eliminate all the waste from the system. Waste is any activity that consumes resources but does not create value. Womack and Jones (1998) noted that an ideal design process would operate much like a single-piece flow in a manufacturing system. Such an analogy suggests that in a lean product development, a new product design would move continuously from concept to production, without stopping due to bureaucratic needs and without backflow to correct mistakes (Walton, 1999).

In manufacturing, for instance, Toyota has established a set of seven waste categories that ease the task of waste finding and elimination. Figure 4 shows these seven wastes from the product development perspective.

	Waste	Product development perspective
1	Overproduction	Too much detail, unnecessary information, redundant development, over-dissemination, pushing rather than pulling data, and lack of synchronization of processes/tasks not only in terms of time but also regarding content, quantity and capacity.
2	Transportation	Inefficient transmittal of information, unnecessary movement of information (data transfer), excessive handoffs, task switching, information incompatibility, communication failure, multiple sources, security issues, and not standards based.
3	Waiting	Idle time due to unavailable information, manpower or computing resources, information created too early, late delivery, suspect quality.
4	Processing	Inherent to a non-optimized process (with nonvalue-added activities or functions), unnecessary product features, inappropriate use of competency, the use of inappropriate tools, excessive approvals, unnecessary serial effort, too many iterations/cycles, unnecessary data conversions, excessive verification, unclear criteria, not processed per process, no transformation instructions, propagation of bad decisions, processing of defective information and multiple tasking when not required.
5	Inventory	Too much information, poor configuration management, complicated retrieval, and stocks or rather data storage in product development and testing equipment and prototypes, which often fix a lot of resources and are under-utilized or unnecessary because they do not provide really new information to the development process.
6	Unnecessary Movement	Required manual intervention, lack of direct access, information pushed to wrong sources, reformatting.
7	Defective Product	Lacking quality, conversion errors, poor configuration management and incomplete, ambiguous, or inaccurate information, lacking required tests/verification; tolerance exceeded.

Figure 4: Seven wastes on product development
(Adapted from: Bauch, 2004; Millard, 2001; Walton, 1999).

The Approach

The approach described in this section applies the lean principles, based on value creation and waste reduction, to derive a project plan that is based on a network of value confirmation events that pulls only the necessary and sufficient information from product development activities. It considers the following project plan requirements to allow the lean development of complex products (Pessôa & Alves, 2005):

Value-based: The product scope must incorporate all and only the customer value.

Deliverable-oriented: Every project task should be directed toward creating deliverables, which themselves must materialize the value.

Lean itself: The planning process and the plan itself must promote the waste drivers avoidance during execution.

Synchronization: The scope and activities definition and sequencing changes must keep the bottom-level dependencies, where the project result is achieved through a stepwise value aggregation effort.

Allow parallel development: Technical risk avoidance can be achieved through parallel development of alternatives.

Knowledge capture: The improvement program must systemically take advantage of every opportunity of learning.

Figure 5 briefly presents the approach where (1) value as stated by the stakeholders is analyzed to determine which system architecture best delivers them; (2) a parallel development strategy is defined; (3) a value confirmation network is created; (4) the project activities are defined; (5) the schedule is consolidated.

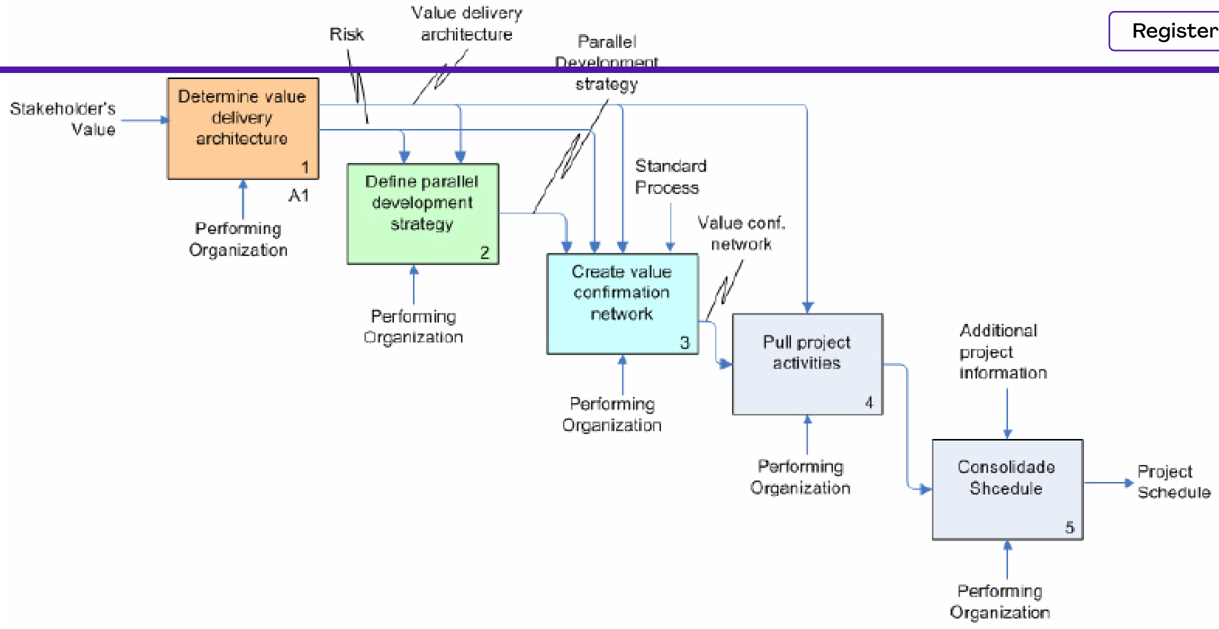

Figure 5: The approach on an IDEF0 notation.

System Value Delivery Architecture Determination

This step includes both value identification and proposition. The value as stated by the stakeholders is translated into systems engineering requirements through requirements and functional analysis. These requirements are themselves deployed on the value delivery architecture. The subsystems that compose the architecture consider from the outset product, processes, and organization.

Parallel Development Strategy Definition

The strategy determines whether a particular subsystem will be developed through a set of alternatives: instead of relying on just one development alternative (frequently the low-risk one), a set of alternatives is carried in parallel. Once the use of parallel development on all subsystem may not be necessary, useful, or possible, a strategy must be defined. This approach proposes that the subsystems that deliver more value and/or have higher risks on delivering this very value (the higher f [value, risk]) are the best candidates to perform parallel development.

Value Confirmation Network Creation

Knowing the value, how it is delivered by each subsystem and the risks on doing that, a set of value confirmation events (checks, tests, experiments, etc.) can be listed. If the company has a standard methodology, as suggested by maturity models such as Capability Maturity Model Integrated (CMMI®) (Software Engineering Institute, 2002) and Organizational Project Management Maturity Model (OPM3®) (Project Management Institute, 2003), the value confirmation network creation will have the predefined tests and evaluations as a starting point. The event dependencies are then determined and a value confirmation network is created.

Project Activities Definition

The activities are pulled from each subsystem (product + process + organization) by the confirmation events from the network. Each subsystem must provide all and only the required information, parts, prototypes, etc., to enable each confirmation event. During execution (Figure 6), the confirmation events signal the related subsystems to start their work, avoiding the early definition/creation of unnecessary information/parts. Thus, a pulled flow of information is created, instead of the pushed flow that is a characteristic of traditional schedules.

Schedule Consolidation

The final schedule is consolidated by considering resource availability, schedule compression techniques, etc.

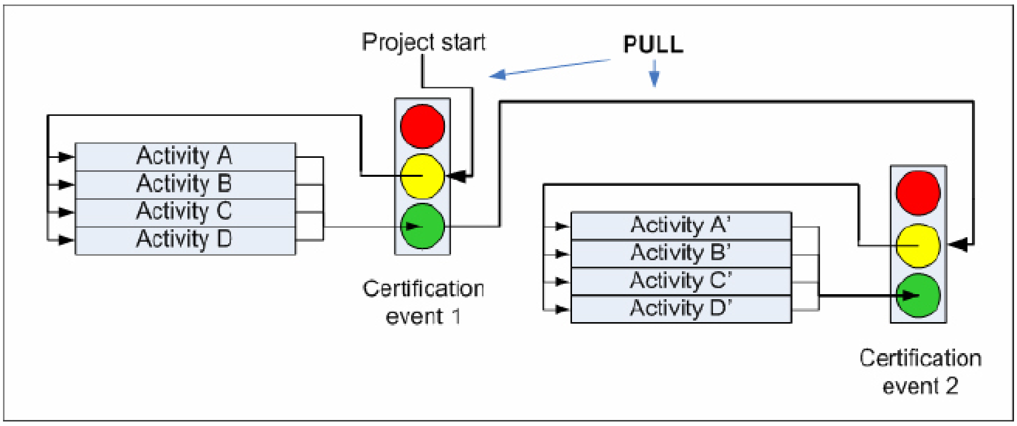

Figure 6: Pulling activities during execution.

An Example of the Approach Implementation

This approach is still under development and is being studied in the context of aerospace projects. One implementation has been made with the adaptations of known tools and techniques, such as quality function deployment (QFD), set-based concurrent engineering (SBCE), design structure matrix (DSM), and precedence diagramming method (PDM). Figure 7 presents the tools used for steps 1 to 3, presented in Figure 5. Steps 4 and 5 implementation is presented in Figure 8. Note that the colors shades of the steps in Figure 5 are the same as the corresponding tools in Figure 7. The authors acknowledge that the method presented in this section is one of many alternatives and that these tools and techniques were chosen based on general knowledge and documentation availability.

To determine the system architecture and define a parallel development strategy, a QFD matrix (Figure 7) is used, where the following occurs:

- The value defined by systems engineering requirements and functional analysis is correlated to the subsystems that will deliver this value.
- The coupling between subsystems is documented in order to later help on activities sequencing.
- Each value is graded according to the stakeholders' perceived importance.
- The total value a subsystem should deliver and the perceived risk to successfully deliver it create a relative rank of best choices to parallel development. A Pareto chart helps with visualization and comparison.

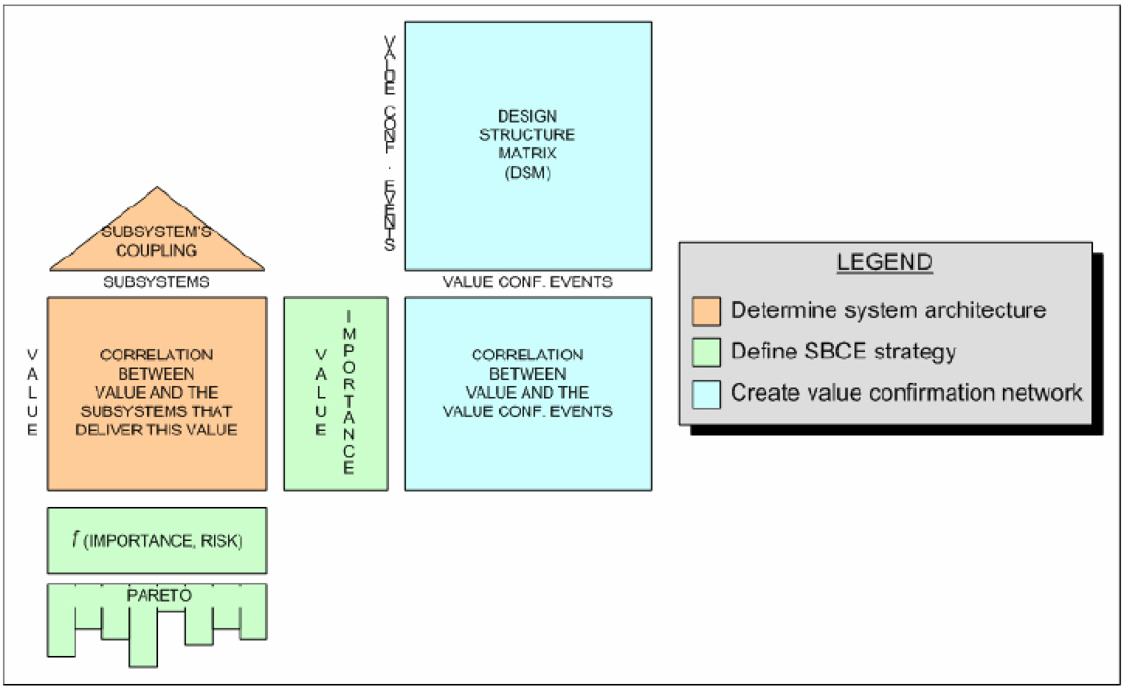

Figure 7:– From value to value confirmation.

A correlation matrix (Figure 7) is then used to create the best set of value confirmation events. These events not only include tests and evaluation, to determine whether the value has been incorporated to the system, but also check if risks have been eliminated in order to allow narrowing during set-based concurrent engineering (SBCE).

SBCE is an upgrade of the concept of concurrent engineering used by Toyota. SBCE allows decisions to be delayed and design options to remain open until it is absolutely necessary to select a point solution. SBCE is a set of simple and repetitive development cycles that achieve high innovation in products and manufacturing systems, avoiding risk through redundancy, robustness, and knowledge capture (Kennedy, 2003).

During SBCE, the development team does not establish an early system level design, but instead establishes sets of possibilities for each subsystem, many of which are carried far into the design process (Figure 9). These sets consider all functional and manufacturing perspectives, building redundancy to risk while maintaining design flexibility. The final system design is developed through systematic combining and narrowing of these sets, when alternatives are eliminated according to the growth of knowledge and confidence (Kennedy, 2003).

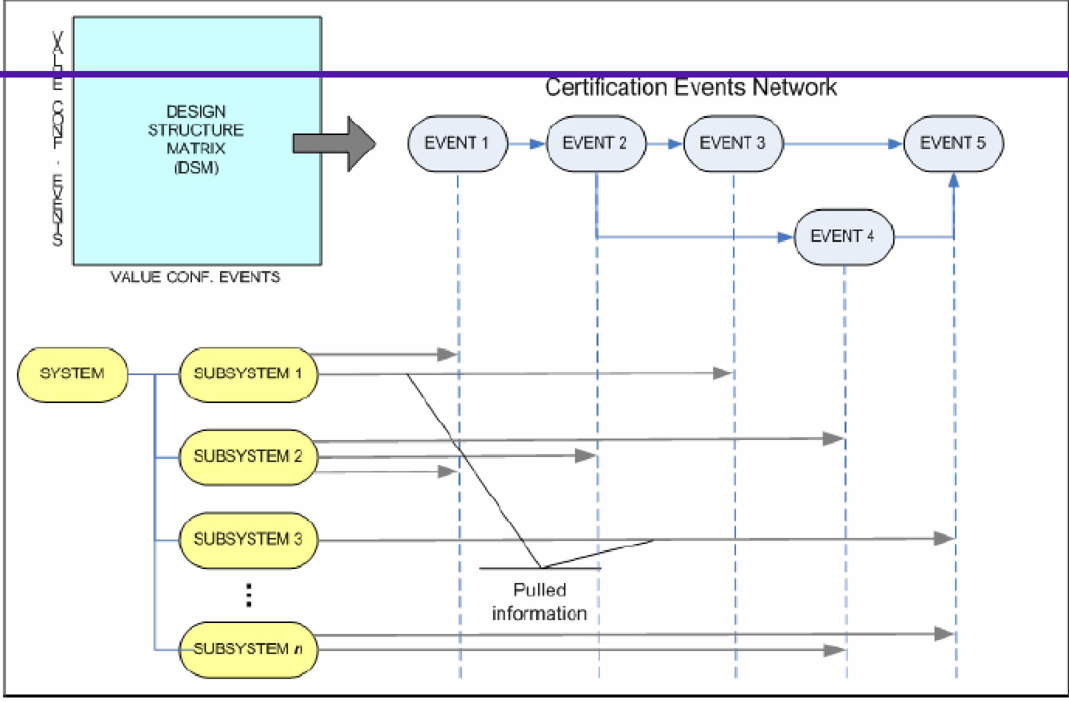

Figure 8: Confirmation events pulling activities from the subsystems.

This set of confirmation events is ordered by a task-based Design Structure Matrix – DSM (a good introduction to DSM is presented by Yassine, 2004), whose output is a confirmation events network (Figure 9). To define the activities, each event "asks" the related subsystems the necessary information, parts, prototypes, etc., that it needs to be performed. Depending on the subsystems' coupling, these activities are themselves sequenced. During execution the confirmation events act as signals to pull the activities.

Finally, the schedule is created, considering other planning issues, such as resource availability, compression techniques, etc.

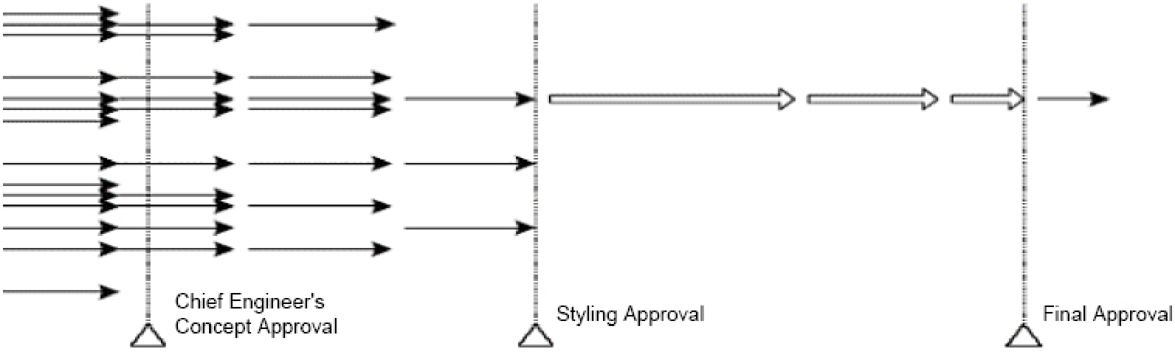

Figure 9: Set-Based Concurrent Engineering (Modified from: Ford & Sobek, 2004).

Discussion of Preliminary Results

Even though still under development, the approach presented was applied to two aerospace projects. The first is an ongoing project where the method was used to determine the value tree, the SBCE strategy, and the homologation plan development. The development company's decision, though, was to use traditional management. Being so, the authors decided to compare the decisions and the work performed with the ones that would be obtained by the application of the method. The second is an already finished and well-documented project where a what-if analysis is being done according to the method.

The strategy chosen to compare the traditional and the new way is as follows:

> On value identification and proposition: Analysis of contractual and requirements documentation used and the results achieved by the use of systems engineering (what-if).
>
> Value delivery: Check the evaluation plan completeness on guaranteeing all the stakeholders' value, and determine the rework cycles due to early definition and to not using SBCE.

The first preliminary result was the development recognition of the method's usefulness on value identification and proposition. Now the decisions and work of the projects are being analyzed.

Conclusion and Future Work

The problem studied was to find a way to deal with the product development inherent complexity. To solve this, a method was designed that applies lean principles to the project planning phase in order to reduce waste and create a schedule that represents the information flow, where the project is planned by using a pulled information progression network that represents the value flow.

The method consists of five steps: (1) value as stated by the stakeholders is analyzed to determine which system architecture best delivers them; (2) a parallel development strategy is defined; (3) a value confirmation network is created; (4) the project activities are defined; (5) the schedule is consolidated.

The main tools used to implement the method are QFD and DSM. The former correlates the value and the value delivery architecture, and helps the SBCE strategy determination. The second defines the sequence of the value confirmation events that themselves pull the necessary information from the subsystems.

An application to the development of two aerospace projects demonstrated that the method improves value identification and proposition, and now the analysis of its impact on value delivery is being carried out.

Anderson, D. M. (1997). *Agile product development for mass customization.* Chicago: Irwin Professional Publishing.

Bauch, C. (2004). Lean product development: Making waste transparent. Diploma Thesis. Cambridge, MA: Massachusetts Institute of Technology.

Clark. K., & Fujimoto, T. (1991). *Product development performance: Strategy, organization, and management in the world auto industry.* Boston: Harvard Business School Press.

Ford, D. N., & Sobek, D. K. (2004). *Explaining the second Toyota paradox by modeling real options in product development.* Manuscript submitted for publication.

Kennedy, M. N. (2003). *Product development for the lean enterprise.* Richmond, VA: Oaklea Press.

Loureiro, G. (1999). A systems engineering and concurrent engineering framework for the integrated development of complex products. Thesis (PhD), Loughborough, UK: Department of Manufacturing Engineering, Loughborough University.

Mascitelli, R. (2002). *Building a project-driven enterprise.* Northridge, CA: Technology Perspectives.

Mihm, J., Loch, C. H., & Huchzermeier, A. (2002, March). Modeling the problem solving dynamics in complex engineering projects. *INSEAD Working Paper.* Fontainebleau, France: INSEAD.

Millard, R. L. (2001) *Value stream analysis and mapping for product development.* Thesis (S. M.). Cambridge, MA: Aeronautics and Astronautics, Massachusetts Institute of Technology.

Murman, E., Allen, T., Bozdogan, K., Cutcher-Gershenfeld, J., McManus, H., Nightingale, D. et al. (2002). *Lean enterprise value: Insights from MIT's lean aerospace initiative.* New York, NY: Polgrave.

Pessôa, M. V. P., & Alves, J. M. (2005). Planning processes and lean philosophy: Can they live together? 2005 *PMI Global Congress Proceedings* – Toronto, Canada, Newtown Square, PA: Project Management Institute.

Project Management Institute (2003). *Organizational project management maturity model–(OPM3®).* Newtown Square, PA: Author.

Simon, H. A. (1969). *The sciences of the artificial.* Cambridge: MA: MIT Press.

Software Engineering Institute. (2002). CMMI™ for systems engineering/software engineering/integrated product and process development/supplier sourcing, Version 1.1, Staged Representation. *Technical Report CMU/SEI-2002-TR-012.* Retrieved October, 18, 2005, from http://www.sei.cmu.edu/pub/documents/02.reports/pdf/02tr012.pdf

Ulrich, K., & Eppinger S. (1995). *Product design and development.* New York: McGraw-Hill Inc.

Walton, M. (1999). Strategies for lean product development: A compilation of lean aerospace initiative research. *Research Paper 99-02.* Cambridge, MA: Massachusetts Institute of Technology.

Warfield, J. N. (1994). *A science of generic design: Managing complexity through systems design* (2nd ed.). Ames, IA: Iowa State University Press.

Wheelright, S., & Clark, K. (1992). *Revolutionizing product development.* New York: The Free Press.

Womack, J. P., & Jones, D. T. (1998). *A mentalidade enxuta nas empresas.* São Paulo, SP, Brazil: Editora Campus.

Yassine, A. A. (2004). An introduction to modeling and analyzing complex product development processes using the design structure matrix (DSM) method. Milan, Italy: *Quaderni di Management*, 9.

Related Content

Agile, Quality Management, Change Management, Complexity, PMO, Decision Making, Business Analysis, Governance, Leadership, Organizational Project Management, Estimating, New Practitioners, Procurement Management | 25 October 2019

PM Network

Using the project management process to improve lessons learned resolution results

By Anderson, Peter | The U.S. Department of Defense (DoD) has an established lessons learned program that documents both shortfalls for corrective action and best practices for wider dissemination. Most phases are…

ARTICLE | Quality Management, PMO, Strategy, Complexity, Procurement Management | 1 April 2018

Project Management Journal
Implications of Network Relations for the Governance of Complex Projects

By Sebben Adami, Vivian | Verschoore, Jorge Renato I Our article aims to answer the call for studies on new perspectives of complex projects and their governance. We adopt the social network approach to investigate the implications of network…

ARTICLE I Quality Management, Risk Management, Complexity I 1 December 2017

Project Management Journal
The Three Secrets of Megaproject Success

By Shenhar, Aaron | Holzmann, Vered I Past studies have often voiced concern that important megaprojects have repeatedly failed due to extensive overruns, misunderstanding of expectations, or both. In this article, we contend that this…

ARTICLE I Quality Management, Risk Management, Complexity I 1 November 2017

PM Network
Tapping Out

By Greengard, Samuel I Offshore oil rigs are going offline. Depressed prices, rising operating costs, aging equipment and, most importantly, tapped out wells have all contributed to a perfect storm: the need to…

ARTICLE I Quality Management, Risk Management, Complexity I 1 October 2017

Project Management Journal
The Successful Delivery of Megaprojects

By Locatelli, Giorgio | Mikic, Miljan | Kovacevic, Milos | Brookes, Naomi J. | Ivanisevic, Nenad I Megaprojects are often associated with poor delivery performance and poor benefits realization. This article provides a method of identifying, in a quantitative and rigorous manner, the…