AN IMPROVED FUNDAMENTAL DESIGN INFLUENCING METHODOLOGY IN A CONSTRAINED PROJECT MANAGEMENT ENVIRONMENT

A. Wessels* and L. Pretorius**

** Doctoral student, Department of Engineering and Technology Management, Graduate School of Technology Management, University of Pretoria, Lynnwood Road, Pretoria, 0002, South Africa, E-mail: ariewessels@gmail.com*
*** Professor, Department of Engineering and Technology Management, Graduate School of Technology Management, University of Pretoria, Lynnwood Road, Pretoria, 0002, South Africa, E-mail: leon.pretorius@up.ac.za*

Abstract: The aim of this paper is to develop conceptually from literature an improved design influencing concept in a constrained project management environment. The project management and systems engineering interface pertaining to project cost and schedule has been researched. Design as part of Systems Engineering (SE) and Project Management (PM) is an important process for product development. The interaction between SE and PM can influence the success of a product development project. A model has been developed to better understand why design iterations are fundamental to the design process. This model has been expanded into a constrained design influencing model that provides a better understanding of the influence of project management in the design process. This model shows that the project manager, particularly if he is under unrealistic constraints, can force a premature design release for integration to the next system level. This model helps to provide a fundamental understanding of the design process.

Key words: Design, project, management, system, engineering, iterations

1. INTRODUCTION

The aim of this paper is to develop conceptually from literature an improved design influencing concept in a constrained project management environment.

SE is an interdisciplinary approach encompassing the entire technical effort and processes of a project in order to develop a successful system that will satisfy a customer's needs [1, 2]. Design as part of the systems engineering process is an iterative and dynamic process and is a fundamental part of the systems engineering process for the development of the individual system components [3]. The individual system components when integrated will function as a whole to provide the required system functionality.

The systems engineering process is well documented in the systems engineering handbooks [1, 2]. Design iterations are fundamental to the systems engineering process, primarily due to the design influencing needed to drive the design to maturity. This will be further discussed in section 4. A design is successively refined until it is mature and acceptable for further integration into the system as shown in Figure 1.

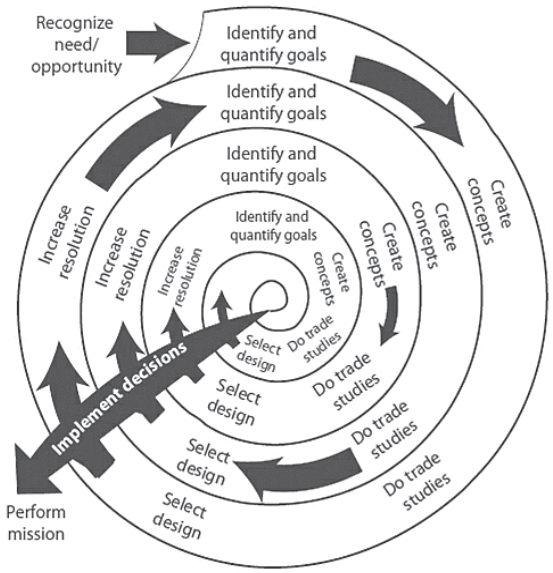

Figure 1: Successive design refinement [2].

According to NASA [2], successive refinement involves a recursive and iterative design loop driven by a set of stakeholder expectations where a draft architecture/design and the derived requirements are developed. Each step also involves an assessment of potential capabilities and potential pitfalls identified through experience-based review of the data associated with lessons learned from other projects [1, 2].

PMBOK [4] defines a project as a temporary endeavour in that it has a defined beginning and end in time, and

therefore has scope and resource constraints. Project management is the application of knowledge, skills and techniques to execute projects effectively and efficiently [4]. Project management is a structured milestone driven process.

The SE process is a "***static***" process in the sense that the processes have no schedule constraints. The SE process does not place any constraints on either the activity time, or a resource requirement on the individual process steps [1, 2].

A system cannot be developed using the systems engineering process by itself. A systems development project requires both the SE and PM processes. Figure 2 illustrates the two processes and areas of overlap. SE requires PM amongst others, to coordinate and manage the schedule as well as the consumption of resources to ensure ultimate project success, [5].

In order to develop an improved design influencing model, the effects of both the SE and PM processes should be taken into consideration, illustrated in Figure 2. This paper identifies the fundamental mechanisms that result in design iterations and the influence that management has on this process.

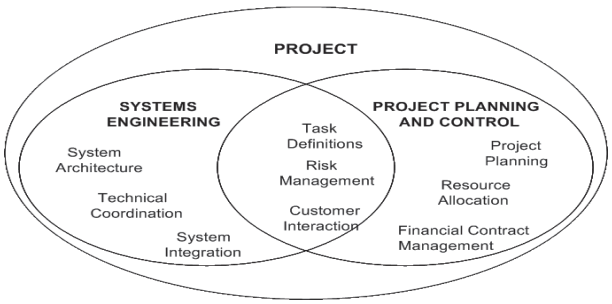

Figure 2: Systems Engineering environment [5].

2. PROBLEM STATEMENT

Since a process always functions within another process, the interface between the two processes can have a distinct influence on the project's performance.

A SE design team's primary objective is technical compliance with the design requirements. To achieve this, a design generally goes through a number of iterations until all the requirements have been achieved. The number of iterations required for a specific design cannot always be accurately determined at the start of the project. At best the number of iterations can be estimated from past experience depending amongst others on the maturity of the selected technology.

The project manager's primary focus is project cost and schedule. However, the project management process requires that all activities and resource expenditure must be accurately planned and managed at the start of a project. Iterations are not inherently supported by the PM process unless defined and planned [4].

The consequence is that indeterminate loops which are an essential part of the SE process for design optimisation are not supported by the PM process.

To further exacerbate the problem, teams behave in accordance to how they are measured [6]. The metric for design team success is primarily technical performance and compliance of a design to requirements. The metric for project management team success is primarily scope achievement, cost and schedule. Both criteria are necessary for the success of a project and customer satisfaction.

Checkland and Scholes [7], recommend that for the smooth and efficient running of a complex systems development project a soft systems methodology be used. Such a process cannot always be quantified and measured and depends entirely on the cooperation and team spirit of the individual members of the project team.

The consequence of the difference between the two processes for the development project is that the project manager endeavours to curtail the number of design iterations. This creates a conflict situation with the design team.

The negative impact of the conflict situation can be mitigated by optimising the effectiveness of the design team as will be discussed later.

3. RESEARCH OBJECTIVES

The aim of this paper is to determine the fundamental mechanisms that give rise to design iterations. Once these mechanisms are fully understood will it be possible to optimize design influencing and determine the effect that PM has on the system development process.

Design Science Research (DSR) methodology has been followed. DSR is to observe, analyse and understand the design process [8].

4. DESIGN INFLUENCING

Engineering design often involves a very complex set of relationships amongst a large number of coupled problems. The complex coupling leads to iteration among the various engineering tasks [9].

Buede [10] states that design influencing is a process to improve the future status of the product, and one that culminates in the allocation of resources to affect the chosen change. The objective of design influencing is to accelerate design optimisation with the aim of driving the design to maturity. The earlier this is addressed in the design process, the lower the cost impact of a design change will be [10, 11].

Before design influencing can be considered in detail, and a model developed, it is necessary to have a clear perspective of the most basic requirements of the design in question. A good design must, amongst other factors, function properly within its design parameters and environments and be cost effective. These environments are external influences and are at best predictions that cannot be controlled by the designer. To ensure that a design always behaves in a controlled and orderly fashion the designer must also consider the design's behaviour for out of specification conditions. A good example would be a software module processing the data from an external sensor. If the sensor provides data that is erratic and/or out of specification, the software must behave in an orderly manner and must not hang-up, but elevate the condition to the next system level [22].

Therefore, a good SE design team must not only focus on the technical requirements of a design but also on the constraints and external conditions which are inherently imposed on the design. This requires two different and almost opposing mindsets which are very difficult to vest in one design team alone.

Design influencing can be made more objective and repeatable by the application of influence diagrams and decision trees [10, 12]. Design influencing can be further refined by applying success frame and failure frame considerations to a design, [12].

The studies by Kim and Kang, [13] and Kuhn and Poole [14], found that teams that developed integrative conflict management styles made more effective decisions than teams that utilized confrontation and avoidance styles. They also found that teams that never developed a stable style were less effective than teams with integrative styles. Also Kim and Kang [13], found that cross-functional cooperation between teams in new product development had a positive impact on product development performance. A topic under review by different mind-set work groups will be looked at from all aspects and not just from one aspect as would be the case for a homogeneous mind-set group [13, 14], thus producing better results.

Applying this principle, the effectiveness of a design team can be improved by dividing the teams into two diverse groups addressing different aspects of the design process. One group to focus on the functional requirements and another group to focus on the non-functional requirements

To achieve the functional requirements, the design team must focus on design success. The design team must focus on all aspects to make the design work. The mindset of the team focussing on compliance with the functional requirements therefore works in the design success domain.

To address the non-functional requirements, the design team must focus on how the design can fail and how it must behave under those conditions to achieve the requirements. The mindset of the team focussing on the non-functional requirements can therefore be said to be working in the failure domain.

Such a division would lead to a ***Success Domain*** (SD) and ***Failure Domain*** (FD) team. The SD design team would then focus on the functional requirements whilst the FD design team would focus on the non-functional requirements. This will create a constructive conflict design environment. Now it is possible to develop a model to study the interaction between the two domains.

Applying these principles to improve team interaction and effectiveness, the two opposing but complementary design teams can be constituted by utilising the following two groups:

- A system/subsystem development team, referred to as the ***Success Domain*** (SD) team.
- A logistics engineering development team referred to as the ***Failure Domain*** (FD) team.

The Success/Failure domain concept is shown in the Figure 3:

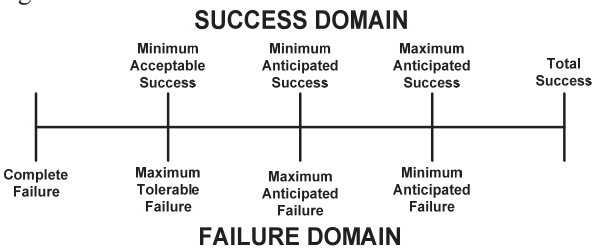

Figure 3: Success/Failure domain concept [12].

5. DEVELOPMENT OF AN IMPROVED DESIGN INFLUENCING MODEL

Dividing the design team into FD and SD groups will ensure that a design is thoroughly analysed and evaluated from all aspects before being released for integration at the next system hierarchy level. This will reduce the risk of an unexpected forced design chance during the system integration phase that will consume unplanned project resources.

5.1. Success domain team (SD)

The "Success Domain" design team must strive for design success. In other words the mindset of the SD team is: "*what is the minimum acceptable success?*" The mind-set of the SD team comprising systems engineering, subsystem development teams and design engineers are therefore set in the "***Success Domain***". This team's objective is to get the system, subsystems and associated software working in compliance with the requirements and development specifications.

5.2. *Failure domain team (FD)*

The "***Failure Domain***" team must identify design weaknesses. In other words the mindset of the FD team is: "*what is the maximum tolerable failure and what are the weaknesses in the design?*" The mind-set of the FD team is failure mitigation of the design. The whole objective is to analyse the system, subsystems architecture and designs to determine what makes them fail and what the maximum tolerable failures are.

5.3. *SD-FD team interaction*

A system can only be developed in a project management environment, since project management provides the time function (schedule) to the system development project, [5].

Placing a project management time function on the Success Domain (SD)-Failure Domain (FD) requirements and constraints, a dynamic design influencing model can now be developed shown in Figure 4. This model makes the static design influencing processes illustrated in Figures 3 dynamic. The model in Figure 4 shows the iterative design influencing process between the SD and FD teams. The objective of both teams is a successful compliant design.

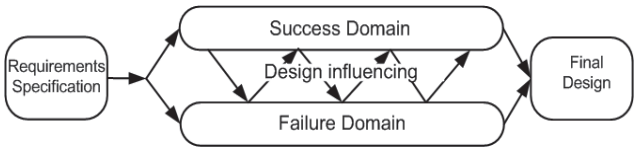

Figure 4: Interaction between the SD and FD teams

5.4. *Application of the SD-FD design influencing model*

One team is responsible for the development and architecture of the system whilst the other team is responsible for the design analysis.

Eisner, [15], states that if there is no coherent design, there is nothing to analyse. This implies that the SD team must first provide a concept design before it can be analysed by the FD team. Only when the Success Domain (SD) team makes a draft design available, can it be analysed by the Failure Domain (FD) team and feedback provided to the Design Review Board (DRB). In practice this is an informal iterative process between the SD and FD teams with short iterative cycles.

Expanding Figure 4 showing the interaction between the SD and FD teams, an unconstrained design influencing model can now be developed. Once the SD team has prepared a concept design, it can be analysed by the FD team and submitted to the DRB. The DRB will then order another design iteration if the concept design deviates from the design requirements. The design iterations will be repeated until all the design requirements have been satisfied. Once the design is acceptable, the design baseline is fixed and released for further integration into the system.

The DRB functions as a gate, similar to the Stage Gate model proposed by Markeset and Kumar [16]. This process effectively results in design iterations until the design is optimised and acceptable as illustrated in Figure 5.

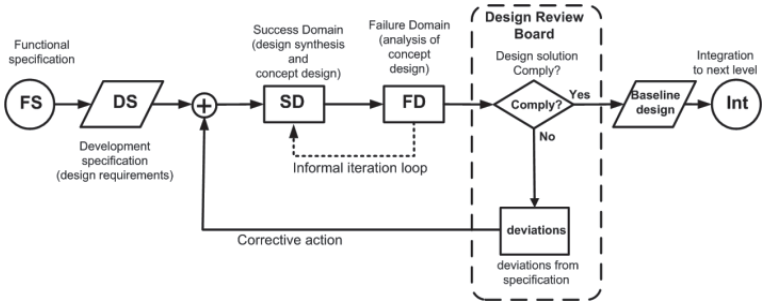

Figure 5: Unconstrained design influencing model

Expanding the SD block in Figure 5, the design engineer (as part of the SD team), produces a draft design by means of synthesis of the requirements and constraints.

Expanding the FD block in Figure 5, the logistic engineering analysts, as part of the FD team, analyse this draft design for the "***ility***"[1] performance requirements against the specification. The Design Review Board (DRB) refers any shortcomings or deviations from the requirements back to the SD team for another design iteration. This iterative design process continues until the design complies with all the requirements and the design configuration is frozen and placed under configuration control in preparation for the next level of system integration. The number of iterations required is generally determined by the maturity of the technology selected and the technical complexity of the design, [9]. The FD team can only perform the analysis after a concept design has been provided by the SD team. In other words design influencing is an "***effect-to-cause***" process.

This process, although at configuration item (CI) level, agrees with the successive refinement process in Figure 1. Again the question remains "*when is the design acceptable?*" This question is not trivial since a number of the design requirements such as reliability can only be verified after extensive qualification Test-Analyse-and-Fix (TAAF) testing [17]. Experienced design review teams normally take a calculated risk based on past experience with similar technologies and designs to expedite the release and baseline of a design.

5.5. *Real world design influencing model*

[1] Any of the engineering "ilities" (e.g., reliability, testability, producibility, supportability) [2].

The SE process by itself cannot bring a system into being. It requires the PM process to structure and manage the systems engineering activities and the consumption of resources, thereby ensuring the delivery of the system to the client on time and within budget. The two processes therefore cannot be separated and must function in an integrated harmonious manner.

5.5.1. Project management team (PM)

A project is "*a unique temporary endeavour, with a set beginning and end*" and "*the application of knowledge, skills, tools and techniques to a broad range of activities in order to meet the requirements of a particular project*",[4].

The project management must satisfy the requirements of the project stakeholders [4]. Therefore the development team objectives amongst others are:

- Successful project within cost and schedule
- Satisfied client
- Satisfied company management

The developed unconstrained design influencing model shown in Figure 5 can be expanded to incorporate the influence of project management.

As discussed in the previous section, PM objectives are different from those of SE and as such can place additional constraints, in particular those of cost and schedule on the design process.

5.6. Development of the real word design influencing model

Expanding Figure 5 and introducing the project management gate, a constrained design influencing model can be developed and is shown in Figure 6. The model adds project management to the design process. Project management is now formally represented on the DRB and can apply its influence to the design process.

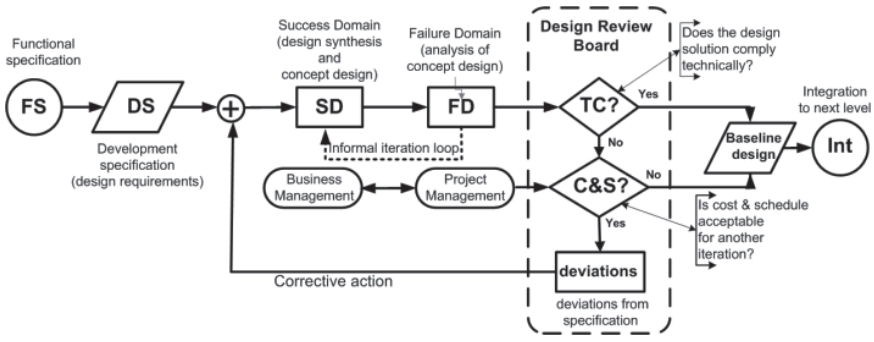

Figure 6: Constrained design influencing model

Thus in the real world, the design influencing model is constrained by project management as shown in Figure 6.

Whereas the systems team reviews a concept design from a pure requirements and technical perspective, the project management team reviews a proposed design from a project cost and schedule perspective as well.

Again in the constrained design influencing model, the SD team prepares a concept design, to be analysed by the FD team and submitted to the DRB. The DRB identifies any deviations of the concept design from the specification and if acceptable, the design baseline is fixed and released for further integration into the system, similar to the unconstrained design influencing model in Figure 5.

The iterative design for the constrained process design influencing model is identical to the unconstrained design process, but with the addition of a gate in the iterative design process controlled by the project manager. The project manager, depending on his constraints, generally cost and schedule, can allow design iteration or force a premature design release. The design may therefore not be fully optimised and mature to the satisfaction of the SD and FD teams.

If deviation of the concept design from the specification is identified by the DRB, project management has the final decision whether to allow another design iteration or to force a release of the design for the next level of integration. This increases the risk that problems may occur at the next level of system integration as a result of the prematurely released design.

5.7. Risk mitigation

A premature design release of a component due to PM constraints illustrated in Figure 6 can increase the risk that a latent design defect may surface later in the project. Components with inherent latent design defects very often only surface during the system integration and testing phase. This can be very detrimental to the project cost and schedule since other functionally coupled components may also be affected and forced to change as well.

Design review checklists can be used to mitigate these risks [1]. Design review checklists must be dynamic and must be regularly updated from company management information systems such as a Problem Reporting and Corrective Action System (PRACAS). The checklists must be universal and not project specific. The checklists must be developed to incorporate the lessons learned from not only the present system but also other systems under development as well as experience gained from field data.

The "***Stage Gate***" model was developed to reduce system development project risk, [10]. The gates ensure that the next phase of the program is not entered before the objectives of the first one have been achieved, confirming the validity of the developed models shown in Figures 5 and 6. The gate ensures that the next step is achievable and the risk of proceeding is acceptable, [1]. This also agrees with the findings by Sommerville, [18].

6. CONCLUSIONS

Underfunding and applying overly stringent and unrealistic schedules to a development project exacerbate the project risk. The general literature view is that apart from minimising the technical risks, ensuring that a project is not under budget and realistic timescales have been set, can reduce the risks of a system development project [2, 19, 20].

The literature also cautions against project underfunding [2, 19, 20]. The rationale for this caution can be deduced from the developed constrained design influencing model shown in Figure 6. A project manager under unrealistic cost and schedule pressure may be forced to take very high risks and release an otherwise unacceptable design.

In practice all that happens is that the problem is shifted to the next level of integration, where the resources required for corrective action become considerably more expensive, primarily due to the ripple effect of the corrective action throughout the system hierarchy.

Underfunding and unrealistic timescales can sometimes lead to the total failure of an otherwise promising system development project, [19].

A model has been developed to better understand why design iterations are fundamental to design. This model has been expanded to a constrained design influencing model that provides a better understanding of the influence of project management in the design process.

The model agrees with the discussed literature and also addresses some of the main shortcomings in the design process. The influence of PM on the design process is very often not taken into account. PM influence on the design process can have a distinct influence on design quality, [21].

This model shows that the project manager, particularly if he is under unrealistic constraints, can force a premature design release for integration to the next system level. The developed model provides a fundamental understanding of the design process.

7. RECOMMENDATIONS

The aim of this study was to develop improved design influencing conceptually from literature. Further evaluation is required with more case studies and cause effect analyses of the SD-FD design influencing concept in a constrained PM environment.

8. REFERENCES

[1] Incose: *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, INCOSE-TP-2003-002-03.2, Version 3.2, International Council on Systems Engineering (INCOSE), San Diego, CA, USA, January 2010.

[2] Nasa: *NASA Systems Engineering Handbook*, NSA/SP-2007-6105 Rev 1, National Aeronautics and Space Administration, NASA Headquarters, Washington D.C., USA, December 2007.

[3] C. Larman and V.R. Basili: "Iterative and Incremental Development: A Brief History", *IEEE Computer Magazine*, pp. 47-56, June 2003.

[4] PMBOK: *A Guide to the Project Management Body of Knowledge (PMBOK)*, Fourth Edition, PMI Standards Committee, PMI, USA, 2008.

[5] A. Kossiakoff and W.N. Sweet: *Systems Engineering Principles and Practice*, Wiley Inter-Science, NJ, USA, 2003.

[6] E.M. Goldratt: *Critical Chain*, The North River Press, Great Barrington, USA, 1997.

[7] P. Checkland and J. Scholes: *Soft Systems Methodology in Action*, John Wiley & Sons, Chichester, England, 2001.

[8] J.E. Van Aken: "Management Research Based on the Paradigm of the Design Sciences: The Quest for Field-Tested and Grounded Technological Rules", *Journal of Management Studies*, Vol. 41 No. 2, pp. 219-246, March 2004.

[9] R.P. Smith and S.D. Eppinger: "Identifying controlling features of Engineering Design Iterations", *Management Science*, Vol. 43 No. 3, pp. 276-293, March 1997.

[10] D.M. Buede; *The Engineering Design of Systems; Models and Methods*, John Wiley, NY, USA, 2000.

[11] L. van der Merwe: *A Product Development Process for a Photovoltaic Water Pump Installation in Small to Medium Enterprise*, Doctoral thesis, University of Johannesburg, South Africa, July 2002.

[12] Intellect: *Reliability: A Practitioner's Guide*, Relex Software Co., 2001 [Online]. Available at: https://www.intellectuk.org/index.php/latest-telecoms-newsviews/media-releases/component/searchadvanced/business-guidance/4393 [Accessed August 2013]

[13] B.Y. Kim and B. Kang: "Cross Functional Cooperation with Design Teams in New Product Development", *International Journal of Design*, Vol. 2 No. 3, pp. 43-54, 2008.

[14] T. Kuhn and S. Poole: "Do conflict management styles affect group decision-making? Evidence from a longitudinal field study", *Human Communication Research*, Vol. 26 No. 4, pp. 558-590, October 2000.

[15] H. Eisner: *Essentials of Project and Systems Engineering Management*, John Wiley & Son, NY, USA, 2nd Edition, 2002.

[16] T. Markeset and U. Kumar: "Integration of RAMS and Risk Analysis in product design and development work processes – A case study", *Journal of Quality in Maintenance Engineering*, Vol. 9 No. 4, pp 393-410, 2003.

[17] Military Handbook – 189: *Military Handbook: Reliability Growth Management*, US Army Communications Research and Development Command, DRDCO-PT, Fort Monmouth, NJ, 1981.

[18] I. Sommerville: *Software Engineering*, Addison Wesley Longman Limited, England, Fifth Edition, 1996.

[19] NAVSO P-3686: *Top Ten ways to Manage Technical Risk*, Office of the Assistant Secretary of the Navy (RD&A) Acquisition and Business Management, USA, October 1998.

[20] NAVSO P-6071: *Best Practices: How to avoid Surprises in the World's Most Complex Technical Process: The Transition from Development to Production*, Department of the Navy, USA, March 1986.

[21] S.L. Ahire and P. Dreyfus: "The impact of design management and process management on quality: an empirical investigation", *Journal of Operations Management*, Vol. 18, pp. 549-575, 2000.

[22] I. Alexander and L. Beus-Dukic: *Discovering Requirements: How to Specify Products and Services*, John Wiley & Sons Ltd, England, 2009.