

Knowledge Capture, Cross Boundary Communication and Early Validation with Dynamic A3 Architectures

Vickram Singh

Dresser-Rand AS

Kongsberg, Norway

vickram.singh@gmail.com

Tel (m): (0047) – 92482889

Gerrit Muller

Buskerud University College

Kongsberg, Norway

Gerrit.muller@gmail.com

Copyright © 2013 by Vickram Singh Permission granted to INCOSE to publish and use.

Abstract. Understanding and extracting systems information is a time consuming, demanding and expensive process. Complicating factors are cross-boundary communication methods and tools. We combine an informal and formal systems engineering method; Lean manufacturing principles and Model Based systems Engineering (MBSE) resulting in Dynamic A3 architecture. Dynamic A3 Architecture is a hierarchy of overviews from super-system to sub-system that the reader can navigate through active links. We applied the method to a lube oil system of a gas turbine package. We found that Dynamic A3 Architecture can ease internal and cross boundary communication, train new employees, facilitate knowledge capture, and share common understanding of the “system of interest”. A functional sequence diagram, which is a hybrid of a state and functional diagram, can assist in early validation of process applications.

Introduction

Dresser-Rand AS is located in Kongsberg (Norway), and has a long heritage of developing gas turbines from the early 1960's. Dresser-Rand proprietary KG2 gas turbine is available from 1530 kW to 2250 kW. It is a compact, proven heavy-duty industrial gas turbine. The company has delivered over 900 KG2 units, mainly generator sets, into 54 countries worldwide. The KG2 has earned a solid reputation as a reliable and easy to maintain unit. The recently developed KG2-3G engine achieves higher power output and lower emissions. A KG2-3G engine and generator are mounted on a steel baseplate that incorporates the oil reservoir and mounting arrangements for the generator, starting and fuel systems (Figure 1). The engine and generator are enclosed in a steel structure together with all the auxiliary systems and control systems. A gas turbine package is a complicated system with subsystems and components (Figure 1).

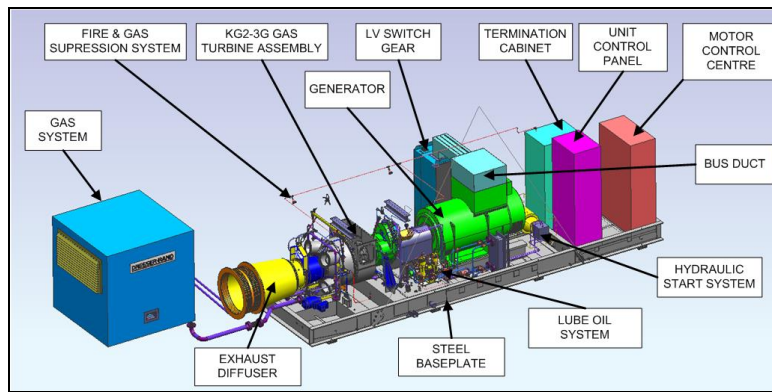


Figure 1: Main systems and components in a KG2-3G gas turbine package with the enclosure removed

Lube oil system: The research was performed on a lube oil system of a KG2-3G gas turbine package (Figure 2). Most typical representations used by engineers to show the lube oil system are 3D drawings, and Piping and Instrumentation Diagrams (P&ID) (see Figure 2). The lube oil system is designed to provide the KG2-3G gas turbine engine with clean and cooled lubrication oil, at correct modulated pressure. This includes pre-lubrication, prior to start-up and cooling-down sequence following a normal stop or shutdown. The main components are positioned in proximity of the reduction gearbox, with the exception of the lube oil cooler that is usually externally mounted. The main challenge of this subsystem is to ensure sufficient oil pressure to the gearbox and turbine at two critical states; start-up and shutdown.

Current way of working: In process applications, the most common communication tool is a P&ID (Figure 2 - Right). It gives an overview of the physical process layout and operator feedback loops. The major issue with this kind of a diagram is that it lacks the functional view and the sequence in which the desired system is supposed to operate.

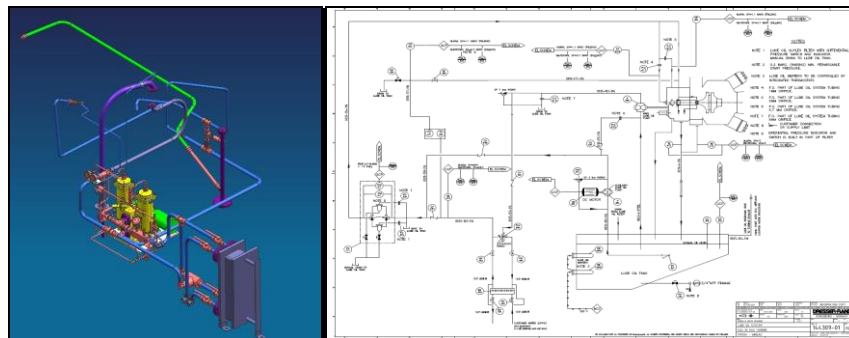


Figure 2: Lube oil system of a KG2-3G gas turbine package
Left: 3D model; Right: Piping and instrumentation diagram (P&ID)

Why trying a new approach? We analyzed symptoms of degraded performance, determined a problem statement, and transformed this into a goal as shown (Figure 3). We observed that main communication means in Dresser-Rand are 3D models, 2D drawings, and piping/instrumentations diagrams. These models and diagrams are not only common to Dresser-Rand in particular, but also to most of the engineering companies. They represent a good visualization of the physical view but not the functional part; how does the system work, and quantified relations. Symptoms that we observed are that less experienced engineers need a

lot of time to find information and they need experienced engineers to explain functionality and quantified performance. The functionality, that is the result of the dynamic cooperation of parts, is often ill understood, since it crosses physical boundaries. Engineers often poorly understand quantified system performance caused by the same cross boundary problem.

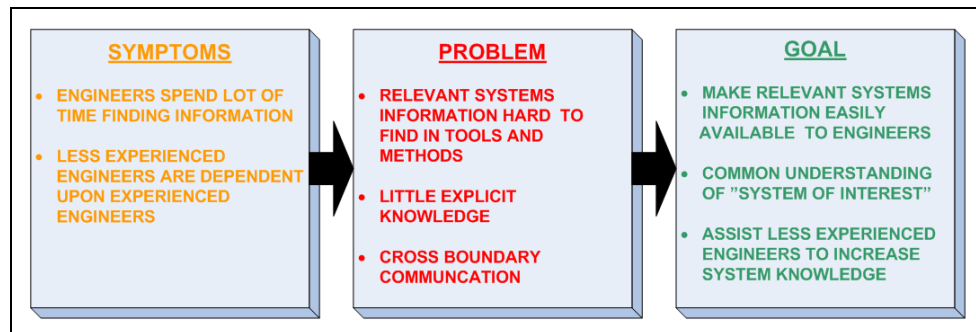


Figure 3: Overview of symptoms, problem and goal

Typical projects for gas turbine package engineering run for periods of 10-12 months. During the initial phase of a project, a team is established composing of a project manager and engineers (cybernetic, system mechanical, mechanical and electrical) where several of them are less experienced. The team works together in order to meet the customer needs, and try to deliver the project on schedule

Dynamic A3 architectures

The applied methodology is a combination of A3 and Model Based Systems Engineering (MBSE). A3 is an informal cross-boundary communication tool that emerges from LEAN (Kennedy 2010) manufacturing principles, which is a concept based upon Toyota's Production Systems (TPS) (Sobek 2008). On an A3, the information recorded is readable and digestible. A3 is a European standard size paper 297*420mm. , Project teams around the globe. use A3's as a source of cross boundary communication tool to solve or address various problems in their domain. Borches (Borches 2010) proposes A3's that are two sided; one is textual and the other composing of models and visualizations. Recommended information (Borches 2010) on the visual side is physical and functional models, quantifications, and specific design choices. On an A3, authors often use color notations to relate functions and quantified relations relative to each other. MBSE is a formalized application of modeling techniques to produce and control a coherent model of the system. It is used to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase, and continuing throughout development and later life cycle phases (Friedenthal 2008). Currently, MBSE is a field of promise with a proliferation of methods, techniques, and tools.

The combination of A3 and MBSE result in Dynamic A3 architecture (Figure 4). Dynamic A3 architecture is a top-level overview of any super system, in which the stakeholders can navigate both ways "top down" or "bottom up" and access relevant sub systems information at all times. It is composed of numerous A3's hyperlinked from a database or server, and made accessible through an internal or a remote website. Rationale behind this combination is that it is a relatively light-weight approach that fits in this project-driven organization.

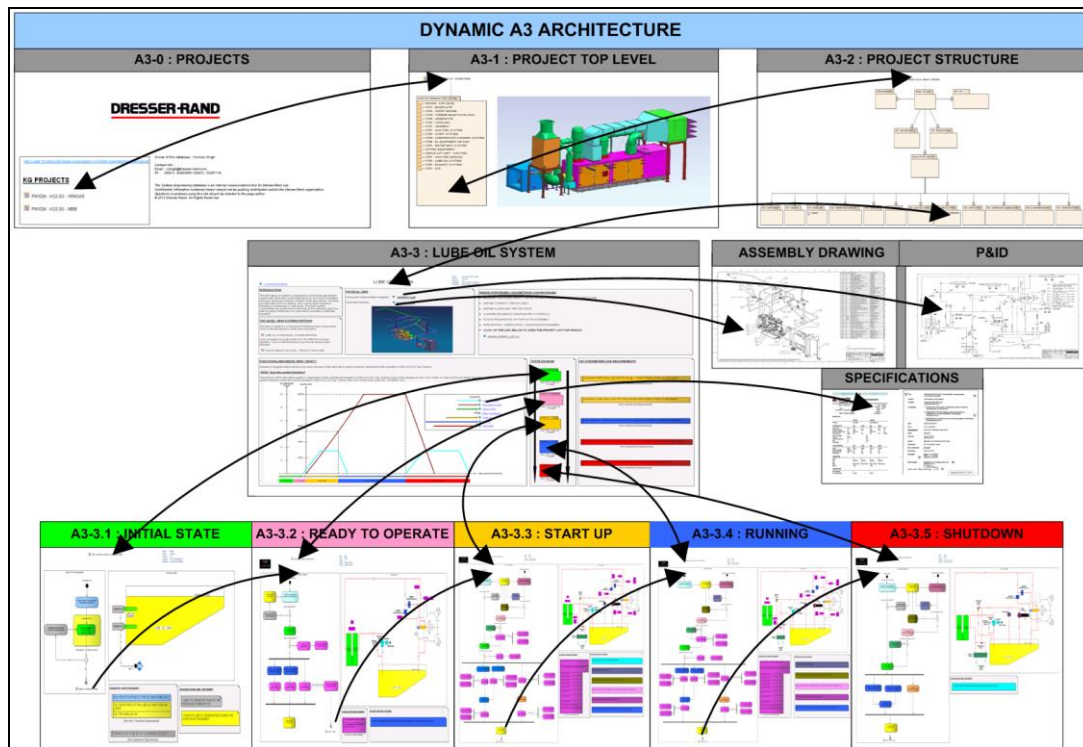


Figure 4: Dynamic A3 architecture applied to a lube oil system of a gas turbine package

Research Methodology

The primary author performed action research by applying the proposed technique as well as study its impact as researcher. The evaluations methods used were surveys, observations, and instant feedback from the internal stakeholders. The author did not set up any formal meetings to gather the feedback, but the approach was rather informal on regular basis (Table 1). He found that it was less time consuming, and much easier to gather feedback instead of organizing formal meetings or workshops. Main challenge during these informal sessions was to reveal what they want to see, like, or dislike. After the creation, 10 engineers with various engineering and/or technical background did final evaluation of the dynamic A3 architecture. This was a formal meeting, where we handed out surveys and the feedback was instantly processed.

Table 1: Evaluation methods; a combination of informal and formal methods

Evaluation Methods			
During Creation (informal)	Location	After creation (formal)	Location
Instant feedback	Personal offices/	Final thesis presentation	Meeting hall
Informal interviews	Coffee breaks	Workshop /Survey	
Observations			
Suggestions			
Likes			
Dislikes			

Current Effort Spent on Information Finding

We conducted a quantitative analysis to map the current situation at Dresser-Rand AS Kongsberg. We deployed a survey with 18 project participants who individually have various years of work experience in the R&D and the gas turbine packaging department. We picked all of the project participants randomly with various engineering background (cybernetic, electrical, system mechanical/process, and mechanical). The survey uses a five point Likert scale and is evaluated by using the Net Promoter Score, e.g. strongly agree are the promoters, agree are passive and no opinion, disagree and strongly disagree are detractors.

Hours absorbed associated with information finding: Only 10% of engineers absorb 30 minutes or less each week finding information. A normal engineer works 1650 hours/year (without overtime). *Approx. 14000 hours are absorbed (wasted) in the company each year because there is little explicit knowledge available.* According to the survey results, the majority of the engineers at Dresser-Rand Kongsberg absorb more than 2 working hours each week on finding relevant information in the current methods/tools and processes (Figure 5). Currently, there are around 200 employees in the company, where 132 of them are engineers. Assuming that the results below (Figure 5) hold for all the engineers in the company, we estimated hours absorbed per year for each group (Table 2).

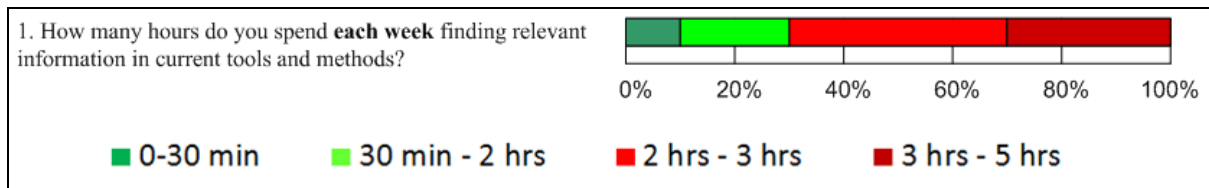


Figure 5: Survey feedback results on time absorption

Table 2: Hours/year absorbed by each group finding information

Group	1	2	3	4	Total
Engineers (%)	10	20	40	30	
Hours consumed per week (Median)	0,15	1,15	2,5	4	
Engineers (Number of employees)	13	26	53	40	
Working weeks (Per/year)	44	44	44	44	
Total hours/year	90	1300	5800	7000	14000

Dependency on experienced engineers: 80 % of the engineers in the company rely on experienced engineers to understand systems behavior and their functionality (Figure 6). *70% of those who disagree are engineers with more than 20 years with gas turbine packages.* There has been limited focus on information sharing in the company, and the systems information is captured in either the document management system or manual archives and folders. To find systems information is a time consuming process. That is why less experienced engineers have to rely on engineers who have been working with the systems for many years.

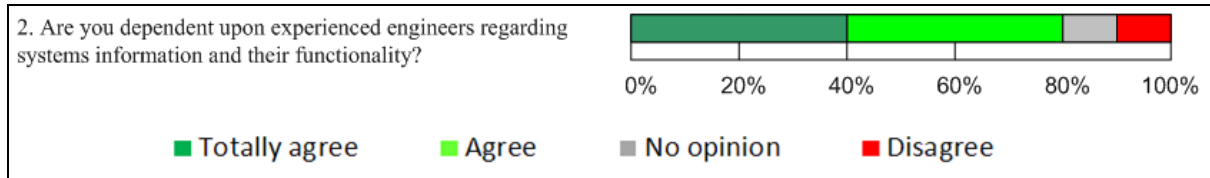


Figure 6: Survey feedback results on dependency

Systems knowledge: Only 24% of the engineers rate their system knowledge as “good” (Figure 7). *Around 80% of the engineers who rate their system knowledge as “good” have more than 20 years of experience with gas turbine packages.* Less experienced engineers do not know where to find information, and even if the information is present, it can be hard to extract and understand

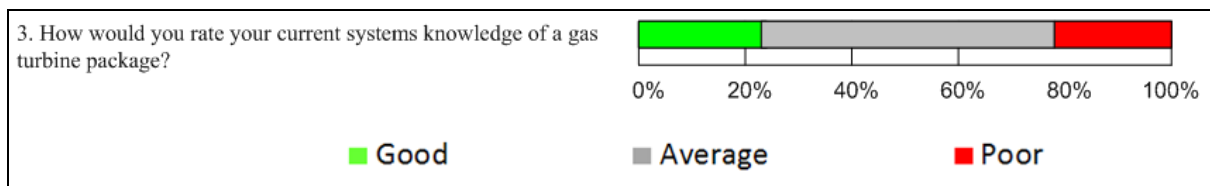


Figure 7: Survey feedback results on system knowledge

Research Approach

Developing Dynamic A3 architecture: We developed a list of objectives (Table 3) depending upon the feedback and the need from the stakeholders, gathered through informal methods (Table 1).

Table 3: List of objectives

Systems information	
Req. 1	The system should be easy to navigate in order to find relevant information
Req. 2	The system shall give a top level overview of the system addressed
Req. 3	The format of the system shall be easily readable
Req. 4	The system shall be easy to access and understand
Req. 5	The information provided in in the system shall be interactive navigation
Req.6	The system shall help to increase system knowledge of employees in the company
Req.7	The system shall help to understand systems behavior relative to the operation of the gas turbine
Req.8	The system shall ease internal and cross boundary communication
Req.9	The system shall help to reduce risks by early validation
Req.10	The system shall help to train new employees in Dresser-Rand

The challenge was to satisfy the listed requirements (Table 3). An internal system engineering

database was developed using Sparx Enterprise Architect, and we named the decompositions of the sub-systems according to a convention following the current project structure (Figure 4). Following the product/system structure in the documentation eases the access to the architecture. We developed a structure where it took only three steps to access any sub-system in a gas turbine package. We divided the levels of dynamic modeling into 9 models, representing their level of detail, structure, and sequences (Figure 4).

Developing Dynamic A3 architecture for lube oil system: During the creation of the lube-oil system architecture (Figure 8), the primary author extracted relevant systems information from the experienced engineers. He started developing functional state sequence diagrams, and initiated iteration processes with those who had the system knowledge. He found out that extracting systems information can be challenging, since experienced engineers were heavily involved in other projects. Time and resources were the main constraints during the extraction of system knowledge. The development of the lube-oil system architecture took approximately 4 months to complete.

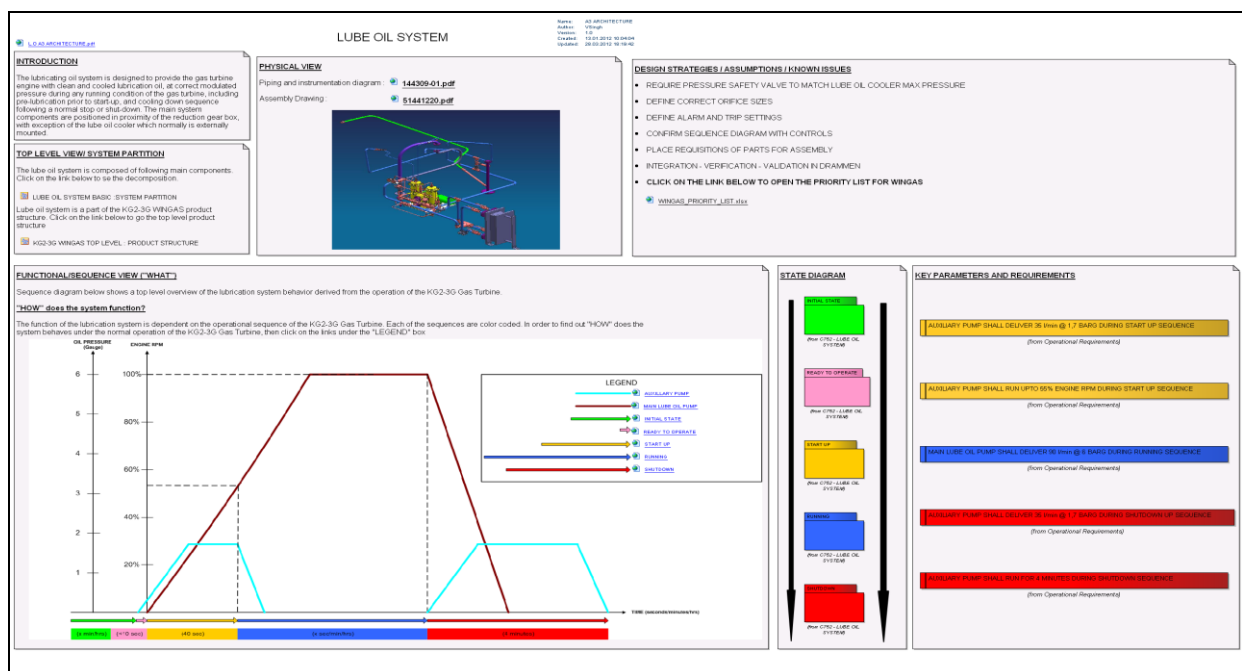


Figure 8: Lube oil system architecture - top level

Lube oil system architecture – top level (Figure 8): The top-level architecture of the lube oil system provides the stakeholder a physical as well as a functional model. It has some textual description together with hyperlinks to assembly drawings, specifications, piping/instrumentation diagrams, and functional sequence diagrams. The stakeholders can either chose to print the documents in pdf format or use it interactive providing flexibility to navigate within the model. On the top right hand side of the lube-oil system architecture (Figure 8), there is a textual description of the engineering work remaining to execute the project, as well as a link for follow-up engineering as advised by the project manager. The key parameters and requirements are color notated to relate the quantified relations. All the relevant information for the lube oil system is stored and documented in this view. A state sequence diagram shows the system behavior as interactions between system components (Friedenthal 2008). A function is a characteristic task, action, or activity to perform and achieve a desired

outcome. For any product, it is the desired system behavior (INCOSE 2004).

A **functional sequence diagram** is a hybrid of a sequence and a functional diagram. The combination of the two diagrams in a flow diagram, captured system behavior of the lube oil system (Figure 9). Further, system behavior related to the physical view and operational curve from the controls software (Figure 8). The internal stakeholders wanted to know the function of each component, interfaces, and their interactions from start to end at any operational state. No standard diagram is available that could capture their needs. After many iterations and feedback from the stakeholders, the researcher was able to combine a state sequence diagram and a functional diagram to a functional sequence diagram. These diagrams were able to capture:

- *Why: Why do we need these components?*
- *What: What is the desired function of the components?*
- *How: How will these components behave relative to the operation of the gas turbine?*

The lube oil system behaves differently in five various states. A functional sequence diagram for each state defines the system behavior. Each of the functional sequence diagrams is composed of a functional sequence diagram and a physical model. The elements in a functional sequence diagram provide a description of the active components, intended function, and relations using colors. This view also illustrates the requirements and constraints to the associated components.

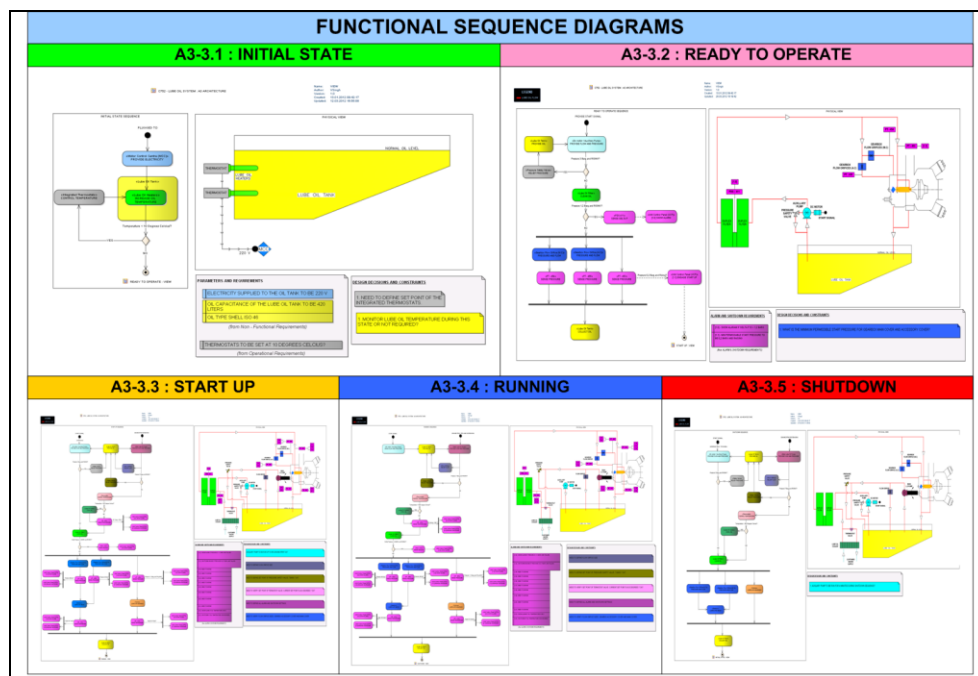


Figure 9: Functional state sequences of the lube oil system

Functional sequence diagrams capture functions of any given component or system, interfaces, and flow direction. The primary author developed these diagrams for each state and presented to the internal stakeholders. During the iteration process, he cross-referenced each of the states to the lube-oil system architecture (Figure 8). The purpose was to ensure that the system achieves desired behavior and the lube oil system meets its requirements. We used Model Based System Engineering techniques to hyperlink the functional sequence diagrams (Figure 9) to the lube-oil system architecture (Figure 8).

The integration of functional sequence diagrams, and lube-oil system architecture (Figure 8 and 9) in Dynamic A3 architecture (Figure 4) made it possible to navigate “top down” and

“bottom up”. The integration of the functional sequence diagrams makes it much easier to control and understand the coherent model of the system. The diagrams were easily cross-referenced to the P&ID (Figure 10 – Left) and top-level overview of the lube-oil system architecture (Figure 8) to detect any missing mechanicals components, control settings, assembly notes, or risks.

Early validation with functional sequence diagrams: stakeholders detected 8 out of 13 parameters in the current P&ID (Figure 10 – Left). We updated the P&ID with the new changes (Figure 10 – right)

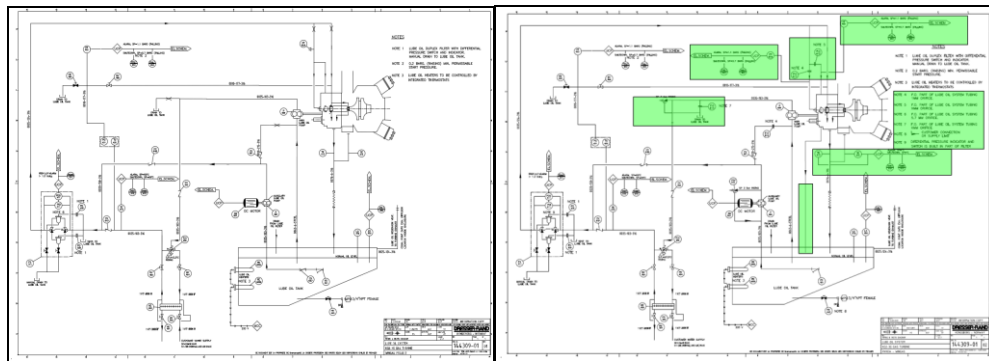


Figure 10: Lube oil system P&ID's

Left: Before functional sequence diagrams; Right: After functional sequence diagrams – changes highlighted in light green.

The functional sequence diagrams made the stakeholders aware of the functionality of the lube oil system (Figure 9). The increased system knowledge from the sessions resulted in the detection of the missing parameters. The stakeholders could focus on one state at one time rather than all the states combined.

Example: In the current design (Figure 10 – Left), there is a pressure safety valve missing on the downstream of the main lube oil pump. This system has been up and running for almost one year on the prototype engine without a pressure safety valve. Any blockage in the system could result in overpressure and damage the equipment. The finding was that functional sequence diagrams played a crucial role in the detection of such a high-risk element.

Final experiment and results

We conducted a formal meeting to evaluate the dynamic A3 architecture. The participants in this meeting were 10 engineers¹ with various engineering background, age, previous work experience, and position in Dresser-Rand. We handed out a survey to them to evaluate the architectures during this presentation.

We divided the structure of the meeting/workshop into 4 parts, and the participants were requested to fill out the survey after the end of each part.

Part 1: Current communication tools in Dresser-Rand: The primary author held a

¹ During this experiment, 90% of the participants were exposed to the dynamic A3 architecture (Figure 3) for the first time. Engineers/managers who had assisted in the creation of the dynamic A3 architecture were excluded from this experiment. This was done to get a better variation of the results from the meeting/workshop.

presentation regarding current communication tools and methods in the company. After the presentation, the participants had to fill out the first page of the survey regarding current communication tools in Dresser-Rand. Results from the part 1 of the survey are shown below (Figure 11).

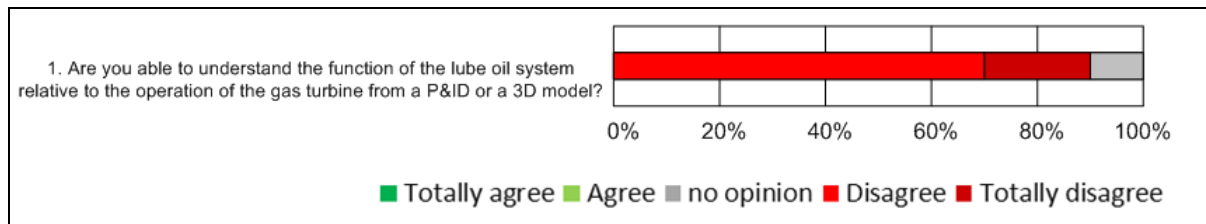


Figure 11: Survey showing stakeholders feedback on communication tools in Dresser-Rand

- With the use of current communication tools in the company, 90% of the engineers are unable to under the functionality of the systems (Figure 11).

Part 2: Lube oil system architecture: We exposed the participants to the lube-oil system architecture (Figure 8&9). The author of this paper gave a presentation regarding systems information, architectural navigation, and functional sequence diagrams. At this stage, they had all the prerequisites to evaluate the lube-oil system architecture (Figure 8 and 9).

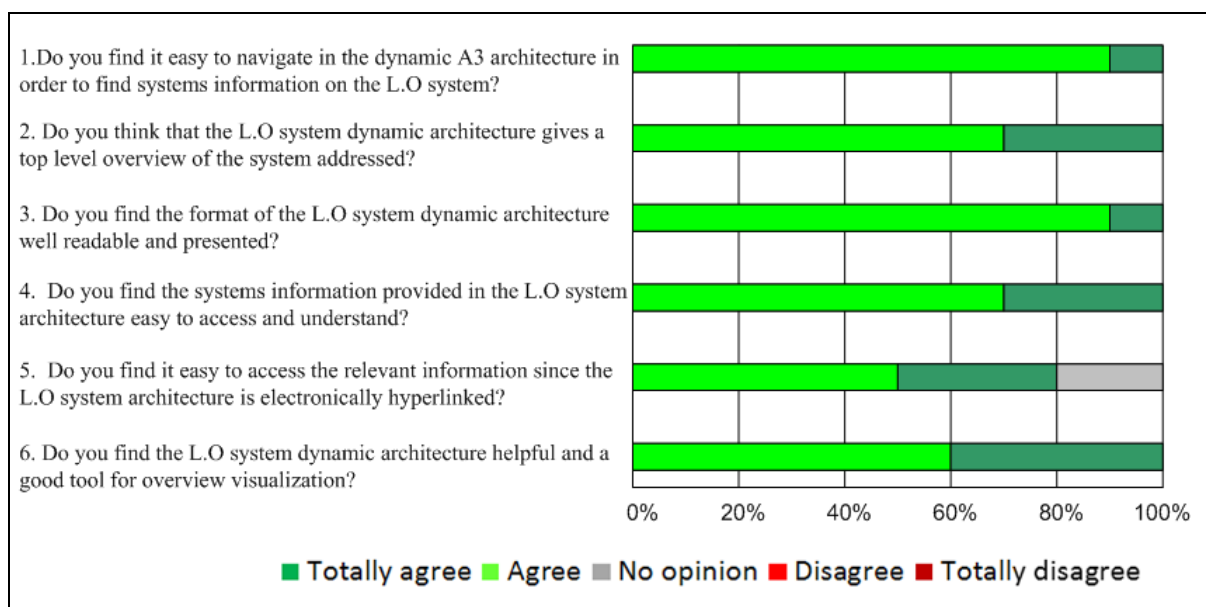


Figure 12: Survey showing stakeholders feedback on lube oil system architecture

The stakeholders had a positive interpretation of the layout and logic behind the system architecture. The participants agree that the lube-oil system architecture provides a top-level overview of the “system of interest”, and that the information captured is easy to access and understand. The evaluators were exposed to the lube oil system architecture for the first time. There are numerous hyperlinks in the architecture, which are not fully visible in the actual A3’s due to small font size and placement in the architecture. That is probably one reason why there is some “resistance” regarding electronic hyperlinks (Figure 12).

Part 3: Dynamic A3 architecture: The participants were required to evaluate the complete Dynamic A3 architecture from a objectives point of view (Table 3). During this stage, we presented core elements and potential of the Dynamic A3 architecture. The feedback was gathered and analyzed after the end of this session.

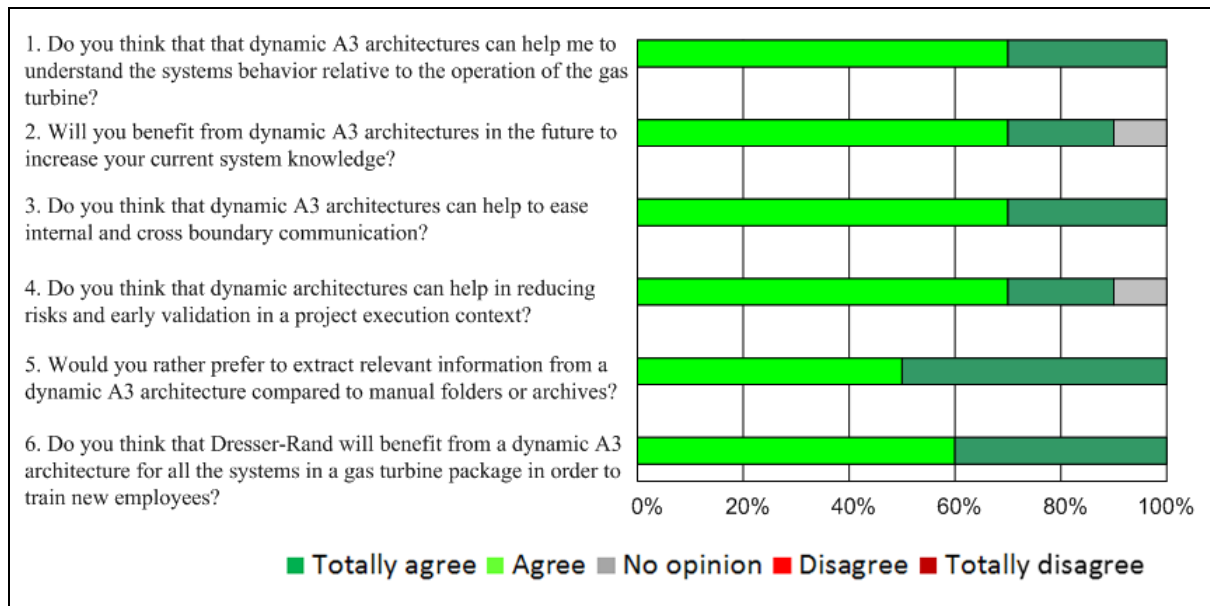


Figure 13: Survey showing stakeholders feedback on the dynamic A3 architecture

Engineers experience that Dynamic A3 architecture can help them to understand sub system behavior relative to super system behavior. This means that it will also increase the system knowledge in the company, and ease internal, and cross boundary communication. 50% totally agree that they would prefer Dynamic A3 architecture to extract relevant systems information. 40% totally agree that Dynamic A3 architecture can be used to train new employees in the company (Figure 13).

Net Promoter Score (NPS): A NPS calculation was performed to differentiate the outcome of the survey result from the formal meeting (Table 4). NPS can be as low as -100 (everybody is a detractor) or as high as +100 (everybody is a promoter). A NPS >0 is considered to be positive outcome. Dynamic A3 architecture has a NPS of 25, which means that this method has 25% more promoters than detractors. This is a clear positive finding, meaning that this particular audience accepts Dynamic A3 architecture.

Table 4: Dynamic A3 architecture has a NPS of 25

#Promoters	34	#Passives	82	#Detractors	4	Total	120
%Promoters	28	%Passives	68	%Detractors	3	NPS	25

Part 4: Dynamic A3 Advantages and concerns: After the end of the presentation, there was a discussion regarding advantages and concerns of the dynamic A3 architecture. During this workshop, we requested the participants to fill out the last page in the survey where advantages and concerns were recorded (Table 5).

Table 5: Feedback from stakeholders regarding advantages and concerns on the Dynamic A3 architecture

Advantages	Concerns
Easy to use and understand	Updating architecture
Good overall systems overview	Maintenance
All relevant systems information is present	Controlling documentation
Troubleshooting of systems during installation	
Accessibility through internal website	
Less time to find information	
Useful tool for service engineers	
All Engineers in a project work from a common base	
One point access	
Possible to access information remotely	
Copy architecture to new projects	

Conclusion

Dynamic A3 architectures can capture and present the functionality of the systems, which is a dynamic cooperation of parts. Using the A3 Dynamic Architecture stakeholders were able to focus during validation on one state at one time rather than all the states combined. That provides evidence for the consequences of having only a *physical* design view as opposed to incorporating not just a *functional* view but also a *behavioural* view. . A functional sequence diagram, which is a hybrid of a state and functional diagram, can assist in early validation of process applications.

Dynamic A3 architecture can help to improve internal and cross boundary communication among a diverse development team, manage project complexity, assist in early validation, facilitate knowledge capture, and design evolution. Engineers can work from a common database, access and extract relevant information for their tasks. Projects can use the common model created using Dynamic A3 architectures in design reviews to compare, evaluate designs, and detect possible areas for new upgrades. Based upon the results for this paper and a NPS of 25, engineers see the value of implementing Dynamic A3 architectures in the company

Future Research

We are moving from an era of text-based systems engineering to model-based systems engineering. The vision behind Dynamic A3 architectures is to provide a top-level overview of the “system of interest”, and connect engineers in a multi-discipline team to share a common understanding. The systems information that is integrated in Dynamic A3 architecture can be used to train new employees, and hence increase system knowledge in the companies. A complete dynamic A3 architecture of any super system can be transformed into applications in platforms like Android and IOS, and revolutionize the way we manage projects in the future. The authors of this paper expect that costs associated with finding information and dependency on experienced engineers can also be considerable reduced with Dynamic A3 architectures. More research is required to measure actual cost reduction. One possible concern with

Dynamic A3 architectures is live links; how users recognize them and how to ensure that users "know where they are and where they go". Further trade-off studies are required to reveal the hidden value and potential of Dynamic A3 architectures in the engineering world.

Acknowledgements

The development of Dynamic A3 architecture required involvement of engineers from a multi discipline team. The author of this paper is thankful for the essential feedback and support from KG2-3G development team and projects&engineering department in Dresser-Rand. Elisabet Syverud, Erik Hewett, and Arvid Stordalen served as an important contribution during the development and iteration process of the Dynamic A3 architecture. The valuable input from the evaluation team during the final presentation is also highly appreciated by the author.

References

Michael N Kennedy, 2010. Product Development for the Lean Enterprise: Why Toyota's System is Four Times More Productive and How You Can Implement It , Oaklea press, Richmond Virginia

Durward K. Sobek II, 2008. Understanding A3 Thinking: A Critical Component of Toyota's PDCA Management System , CRC Press, Boca Rotan, Florida

Borches, D. 2010. A3 Architecture Overviews, A tool for effective communication in product evolution, PhD thesis, University of Twente.

Sanford Friedenthal, Alan Moore and Rick Steiner, 2008. A Practical Guide to SysML: The Systems modeling Language, MK/OMG Press

INCOSE 2004 SE Handbook, Version 2a. A "what to" guide to all SE practitioners.

Biography



Vickram Singh.

Vickram Singh received his Bachelor's degree in mechanical engineering with specialization in aeronautical engineering from University of Agder in 2009. He has been working in the KG2-3G development team since 2009, and received his dual Master's degree in systems engineering from Stevens Institute of Technology and Buskerud University College in 2012.



Gerrit Muller.

Gerrit Muller, originally from the Netherlands, received his Master's degree in physics from the University of Amsterdam in 1979. He worked from 1980 until 1997 at Philips Medical Systems as a system architect, followed by two years at ASML as a manager of systems engineering, returning to Philips (Research) in 1999. Since 2003 he has worked as a senior research fellow at the Embedded Systems Institute in Eindhoven, focusing on developing system architecture methods and the education of new system architects, receiving his doctorate in 2004. In January 2008, he became a full professor of systems engineering at Buskerud University College in Kongsberg, Norway. He continues to work as a senior research fellow at the Embedded Systems Institute in Eindhoven in a part-time position.