28th annual INCOSE international symposium
Washington, DC, USA
July 7 - 12, 2018

Creating and Applying A3 Architecture Overviews: A Case Study in Software Development

Anders Viken
Norway
anders.viken@hotmail.com

Gerrit Muller
HSN-TMN-NISE
Kongsberg, Norway
gerrit.muller@gmail.com

Abstract. We observed in our company that the commissioning phases of projects were inefficient when it came to effort, measured in hours. In order to meet this challenge, the management team decided to create a software program that would enable shipyard engineers to perform software changes by themselves directly in the automation systems. By enabling shipyard engineers, the goal is to save valuable person-hours on each commissioning phase.

The A3 report is a tool that can be used in most problem-solving situations. It can be used both as a way of structuring thinking and as a tool for communication and learning. After Toyota's success of using it as a tool for problem solving and report writing, many other domains have adopted the A3 report as a structured problem-solving approach.

In this study, the author explains how A3 reports can facilitate validation and communication during system design of a software program. During software development, the project team used the A3 reports when discussing the software internally and with stakeholders. Observations recorded during this study indicate that the A3 report is a valuable tool for sharing ideas and capturing feedback, both in planned and spontaneous sessions. This study also explains how the format of the A3 report can encourage reading and writing of documentation in hectic industrial settings.

Introduction

Company. The company delivers systems for dynamic positioning and navigation, marine automation, safety management, cargo handling, subsea survey and construction, maritime simulation and training, and satellite positioning. This study has its origin from the Offshore, Oil & Gas segment, targeted at vessels and installations using the marine automation system. For confidentiality reasons, the A3s are presented with limited resolution.

Case. The case is a software tool designed to enable shipyard personnel to carry out loop checking of a vessels' automation system without personnel on-board. The department in the company responsible for delivering automation systems on carriers initiated a project with the goal of delivering this software tool. This paper refers to the development state of the tool during the first five months of the development period.

Problem statements. A challenge when working in the competitive market today is that software development has to be done fast, while research and development budgets are cut. Naturally, that leads to down prioritization of some tasks. In our company, we observed that the software development department does not prioritize to write high-quality software documentation. A consequence of this is that the engineers in our company do not find the existing documentation valuable. We also observed that our engineers are unable to work independently with these programs, and use valuable time every week on troubleshooting. In order to deal with these problems, The Company needs to find a way to encourage documentation, while supporting communication and validation of software programs, without down-prioritizing software quality in the development process.

A3 reports in early software development. To support the development process, and to encourage validation and communication between stakeholders during the project, the author introduced A3 reports to the tool project. During the development phase of the project, the author created, used, and evaluated the use of several A3s. The author created the interconnected A3s with different levels of abstraction, in order to capture both top-level considerations and aspects of each sub-system.

Research questions. In order to assess whether the A3 creation effort fulfils the expected benefits, the author assess these research questions:

- How do A3s facilitate stakeholder communication?
- How can A3s be used to capture feedback in different meeting locations and situations?
- How well does the A3 format encourage reading and writing of documentation?
- What factors can prevent the success of implementing A3s in an organizational context?

State of Art

The A3 report originates from Toyota Motor Corporation. Toyota uses A3 reports to give status reports on ongoing projects, to propose changes to practices and processes, and to solve problems. Toyota introduced A3 reports as a way of recording their Plan-Do-Check-Act (PDCA) cycle. PDCA is a high-level methodology that aims to improve long-term system performance, not just take care of a localized problem (Sobek & Smalley, 2008). By using A3 reports as a tool for following up problems and improvement opportunities, Toyota has continuously improved their working processes, which has resulted in outstanding organizational performance.

Daniel Borches further developed these ideas, and proposed to use A3 Architecture Overviews (A3AO) as a tool for knowledge sharing and effective communication of architecture knowledge (Borches, 2010). Figure 1 illustrates an A3AO that uses both sides of a standard A3[1] paper size. One side (A3 Summary) displays textual information regarding a certain system aspect, while the other side (A3 Model) displays models and visualizations clarifying that aspect.

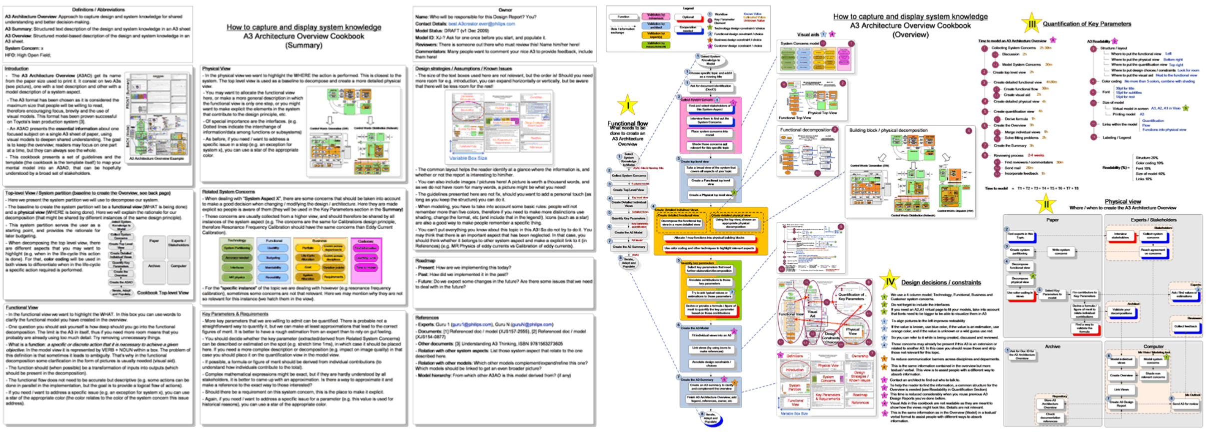


Figure 1. Borches A3 Architecture Overview Example. Left: A3 Summary; Right: A3 Model.

By using the structure suggested by Borches, the creator of the A3AO is limited to include a "human friendly" amount of information. The A3AO format permits the author to include information about many system aspects. However, the paper size forces the author to select and process this information

[1] A3 is an international paper size standard of 297 x 420 mm (American metric equivalent of 11 x 17 inches)

wisely. The models on the A3AO contribute to visualizing the system of interest in a practical way, close to the experience of stakeholders (Muller, 2016).

In an industrial case study, Borches and Bonnema (2010) aided Philips Healthcare in using A3 Architecture Overviews to collect, abstract, and present the architectural knowledge of a Magnetic Resonance Imaging (MRI) scanner. This study showed that A3AOs "provide an effective framework to support decision making when evolving complex systems." In addition, they found that the A3AO is "an effective tool for communicating among diverse disciplines and departments, and enables gaining deeper insight into a system."

Inspired by the initial work by Borches, further studies have taken place at the University of Twente and the University College of Southeast Norway. The studies experimented and developed Borches' theories further in different industrial settings and domains. Wiulsrød and Muller (Wiulsrød, Muller, & Pennoti, 2011) reviewed A3AOs by performing reverse architecting of a diesel control system. Wiulsrød and Muller concluded that the structure and format of the A3AOs facilitated discussions and enabled knowledge sharing. Polanscak (Polanscak, 2011) applied A3s during the development of a computer tool for a manufacturing cell. Polanscak found the A3 approach to be a useful tool to both share and capture architectural insight, as well as helping stakeholders stay focused when discussing complex issues. Kooistra (Kooistra, Bonnema, & Skowronek, 2012) created an A3AO to describe the Thales Tacticos combat management system, which is a complex 'System of Systems' (SoS). At the beginning of the study, Kooistra found that the many stakeholders had no consensus regarding the most important aspects of the system. The A3AO raised stakeholder awareness by providing overview and defining a clear scope of the SoS. Singh (Singh & Muller, 2013) used the lube oil system of a gas turbine package as research case. Singh combined A3s and Model Based Systems Engineering, resulting in a hierarchy of overviews from super-system to sub-system that the stakeholders could navigate through active links. Singh named this hierarchy 'Dynamic A3s architecture'. Singh found that the A3s eased internal and cross boundary communication, helped manage project complexity, assisted in early validation, facilitated knowledge capture and helped train new employees.

Based on the above state of the art, the author concluded that the industry can apply A3AOs in many different contexts, and that creation and use of A3AOs can be a valuable way of making important information available and visible. The main reason for implementing A3s in this project was to use A3s as a tool for validation and communication **during** the development of the tool. According to Borches, the philosophy and intention of the A3 is more important than following his suggestions (Borches, 2010). With that in mind, the author followed Borches' approach where applicable, but did not follow all of Borches' suggestions. The author created his own template that encouraged both visual models and textual elaborations. Further, the author did not follow the suggestion of creating two-sided A3s, but created only one-sided A3s during the project. Sobek and Smalley seem to support this approach: "The thinking and processes behind the A3 reports are more important than the actual A3s**"** (Sobek & Smalley, 2008). During this study, the author has been observing, analyzing, and evaluating the thinking, processes, and communication regarding several interconnected A3s.

Research Methodology

This study uses action research (O'Brien, 1998) as research method. Riel (2010-2017) defines action research as "a systematic, reflective study of one's actions, and the effects of these actions, in a workplace or organizational context." The author chose action research as approach for this study, as Riel considers it an appropriate approach for doing research on a method applied in an industrial context (Riel, 2010-2017).

The author worked on a software development project, and introduced the A3 reports to the project team with a learning-by-doing approach. The author developed the A3 reports during the whole development period, and applied them in discussions with the development team and meetings with stakeholders. At the same time, the author worked as a software developer on the project and could observe and reflect upon how the development team used the A3 reports.

A homework project for the SEMA 2016 course[2] at USN, Norway triggered the creation of the first A3s. The homework project modeled several aspects of the tool, including the initial revisions of two A3 architecture overviews. The project group presented the A3s for stakeholders in Norway. Two of the three participants in this group were part of the project team developing the tool.

After presenting the A3 architecture overviews for stakeholders in Norway, the feedback indicated a need for better documentation of similar software programs. This feedback triggered a verbal discussion where the stakeholders explained areas of improvement when it comes to documentation and knowledge about these software programs. To understand the stakeholders' needs better, the author created a survey. The survey topic was the documentation and knowledge about internal software tools. The author sent the survey out to software engineers who use these tools regularly.

The author collected observations and feedback from A3 sessions by using an observation template. The author made the observation template with inspiration from the "In Session Observation Template" suggested by Polanscak (2011). This template provides a set of guiding questions and attributes that helped in recording observations after A3 sessions and discussions. The table attributes (Table 1) capture the kind of session, information about the A3 report and questions about the use of the A3 report. In addition, the attributes capture questions generated, any new questions, concerns or requirements generated, and changes to models after the sessions.

Table 1: Session attributes

Session questions/attributes
Kind of session
Physical location of session
Planned/unplanned session
A3 purpose
A3 usage
Number of session participants
Supported in answering questions
Supported in generating new questions/concerns/requirements
Models changed/added after session
General observations

Figure 2 illustrates the research approach during this study. The survey about internal software programs indicated a need for better documentation. Based on the analysis of the survey, the author created several A3s during the development period. In addition, the author cooperated with other engineers on the software development. When the author identified a need for validation or design decision, the author made an A3 report for that purpose. The stakeholders in the project had discussions around the A3 reports in planned and unplanned sessions and meetings. If the stakeholders identified any generated or changed requirements during the sessions, the development team adjusted the system requirements and the software. After the A3 sessions, the author

[2] Course description at www.gaudisite.nl/SEMA.html, course material at www.gaudisite.nl/SEMAallSlides.pdf.

consolidated and analyzed the observations done by using the observation template. After adjusting software and analyzing the observations, the author updated the relevant A3 reports.

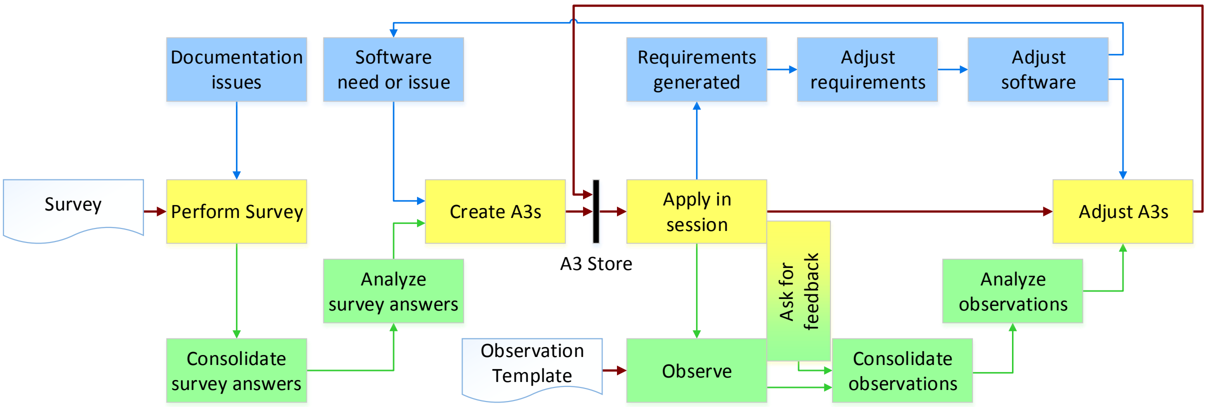


Figure 2. Research approach. Figure inspired by (Polanscak, 2011).

Based on the analysis of the observation templates recorded during the research project, the author was able to identify benefits, challenges, and blockers that either contribute to or prevent a successful implementation of A3s. The observation templates provide several logical segments, in order to divide and structure information. This gave the author the opportunity to easily create statistics and analyze the observation templates as a whole. The analysis of the observation templates gave an indication of what factors were most relevant for this study.

Based on the analysis of the observation templates, and the author's personal experiences during this study, the author initiated discussions with each of the project participants. These discussions served as debrief after the software development phase, and clarified further the stakeholders' opinions of how the A3s worked **as a tool for validation and communication.**

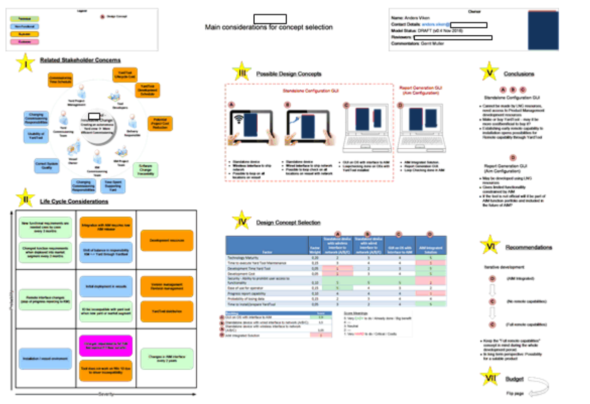
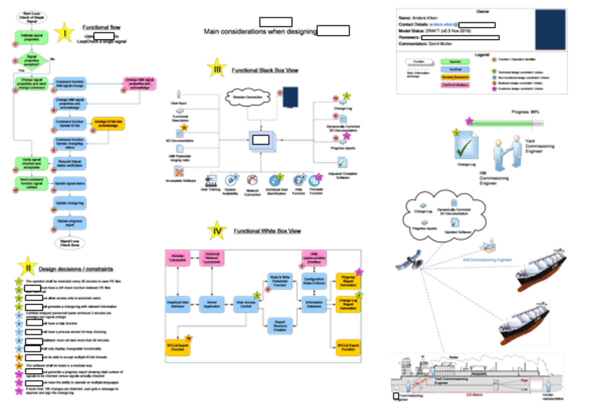
Case Study

The author based the research presented in this paper on an internal software development project in the company. The department delivering automation systems on carriers initiated the project. The project goal was to reduce 500 hours on each commissioning. Experienced commissioning engineers in Korea created the user requirements for the software system. They identified that the engineers use unnecessary time on following up and performing repeating changes in the automation system. This happens mainly during early commissioning tasks such as loop checking. By empowering shipyard engineers to perform a set of identified changes to the company's automation systems, management believed that the project could reach its goal.

The author brought this user requirement as input to the SEMA[3] course in August 2016. During this course, the coursework modelled several aspects of the tool: stakeholder concerns, physical models, functional models, design strategies, and quantifications. These aspects coincide with elements proposed by Borches (2010) as standard content on A3 architecture overviews. As part of the course homework, three students developed two top-level A3s (Table 2), based on the A3 Architecture Overviews Cookbook developed by the University of Twente, Embedded Systems Innovation and Philips (A3 architecture overviews, 2015).

[3] Course description at www.gaudisite.nl/SEMA.html, course material at www.gaudisite.nl/SEMAallSlides.pdf.

Table 2: Top-level A3 examples

Top-level A3s	
A3-L0: Concept Selection	**A3-L0: Design Considerations**
	

The three students presented the top-level A3s for the management team and relevant stakeholders in a dedicated session. The feedback from the session participants indicated an interest for the modeling approach to documentation. The participants also discussed a need for better documentation of similar internal tools. Based on this feedback, the author initiated an internal survey with software engineers in the department delivering automation systems on carriers. The goal was to get opinions from the software engineers regarding documentation of internal software programs similar to the tool. A sub-goal of the survey was to investigate the level of internal knowledge regarding these internal software programs.

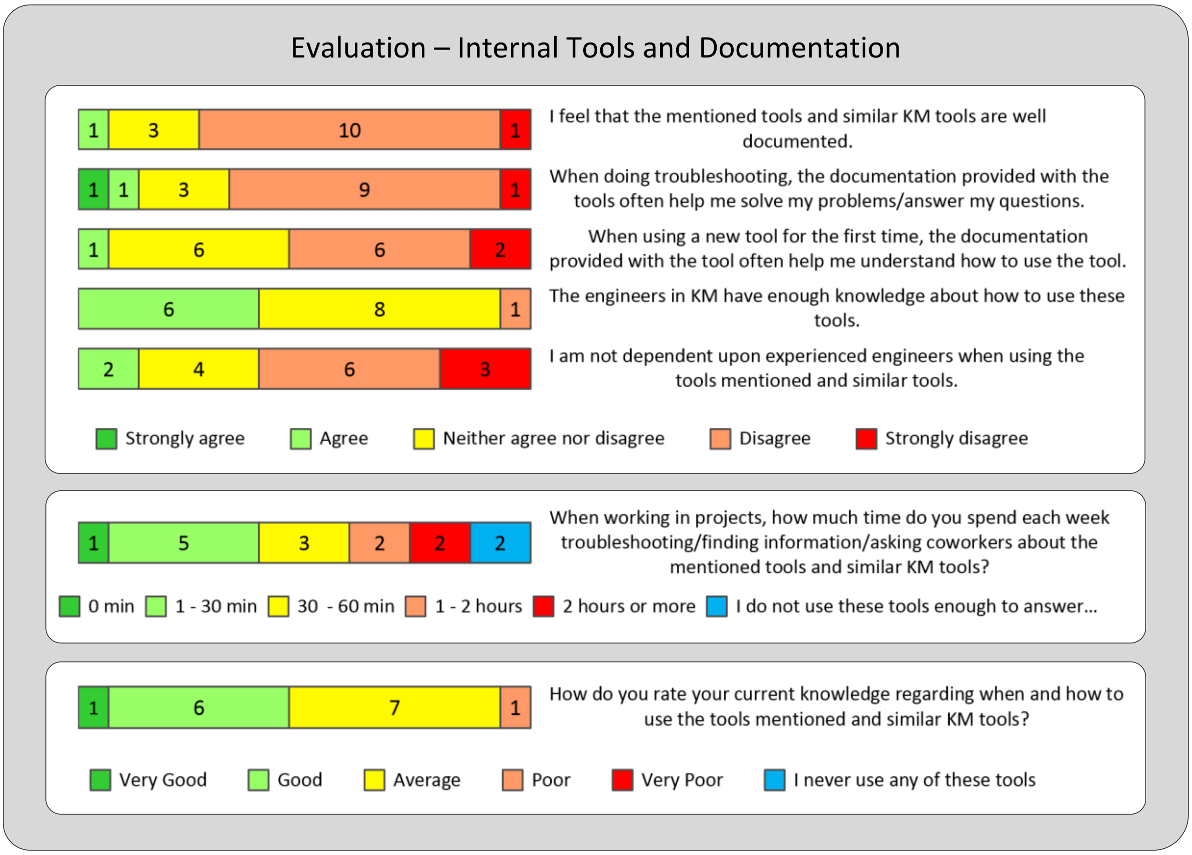


Figure 3. Survey feedback from software engineers regarding knowledge and documentation of internal tools.

The 15 software engineers who participated in this survey (Figure 3) rated their knowledge on internal software tools as average or better. According to the survey participants, the documentation of internal software tools can be improved:

- The documentation does not help solve problems and does not give the users answers to questions during troubleshooting.
- When using a new software tool, the documentation does not help the users in understanding the tool.
- The software engineers are dependent upon experienced engineers when working with the tools.

As an observation, six out of the 15 survey participants felt that engineers in The Company have enough knowledge about these tools. However, only two out of 15 participants are able to work independently with these tools, which may be a cause to wasting valuable time on investigating correct usage.

Based on the feedback from the presentation of top-level A3s, and the evaluation of the survey, the author decided to investigate further how to use A3s for describing the software program to be developed. Both (Borches, 2010) and (Van de Laar & Punter, 2010, p. 125) recommend to use multiple levels of abstraction for conceptual modeling. For creation of A3s, (Muller, Wee, & Moberg, 2015) observed that creators of A3s tend to discover that they need multiple levels of A3s at various abstraction levels. Following these recommendations, the author developed lower-level A3s, with the goal of capturing the aspects of each sub-system of the tool.

Before creating the lower-level A3s, the author created a template. This template functioned as a starting point for each of the lower-level A3s. The template contains five main sections and one top bar. The sections contain information about system-of-interest, operation flow, design strategy, assumptions, known issues, and a roadmap for future implementation. In addition, the template contains a big section for the main message of the A3. The author divided this section into several small sections for some of the A3s. The top bar contains smaller sections with information about the model and its corresponding subsystem. Each section is resizable, but its position is always consistent from one A3 to another. Figure 4 illustrates the lower-level A3 template. Table 3 contains two examples of lower-level A3s.

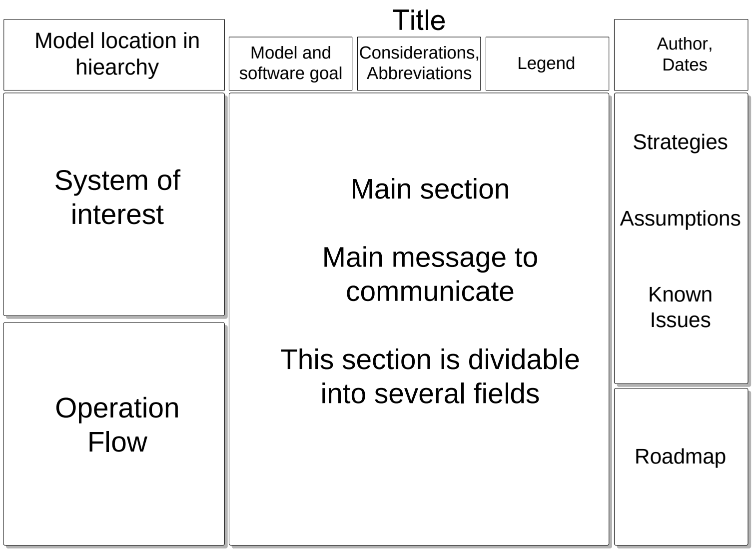
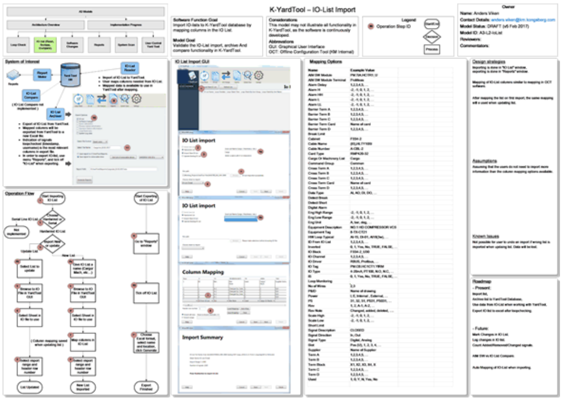
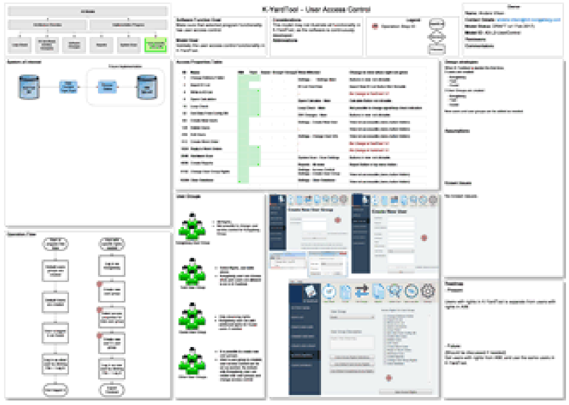


Figure 4. Lower-level A3 template.

Table 3: Lower-level A3 examples.

Lower-level A3 examples	
A3-L2: IO List Import	**A3-L2: User Access Control**
	

During the development phase, the author held dedicated A3 sessions with the lower-level A3s. The author recorded in total 12 A3 sessions in the observation templates. The sessions were both planned and unplanned. The location of the sessions varied; three of the sessions were planned Skype meetings with stakeholders from Korea, while the stakeholders had five unplanned sessions in the coffee break location or the nearest office. The number of session participants varied from two to 11, with an average of four participants. In all the sessions recorded, the participants answered stakeholder questions, generated new questions or concerns, and/or discovered new requirements.

The author expects that the A3s will be valuable also after the development period of the software program. The users of the program can use the A3s as user manuals, software developers can use the A3s for further development, managers can use the A3s for quick introduction to the broad aspects of the software, and new employees can use the A3s as a starting point for training. By distributing the same documents to all stakeholders, a common platform for communication is established. The software developers can get feedback on the documents from stakeholders across disciplines and departments. Receiving comments from stakeholders with different technical experience and different ways of thinking, provides the opportunity of comparing comments from different stakeholders and detecting inconsistencies.

Evaluation / Research findings

In this chapter, the author discusses several benefits, challenges, and blockers. These factors either contribute to or prevent a successful implementation of A3 architecture overviews **for validation and communication purposes**. To structure the findings, the author presents the benefits and concerns during creation and use. The author also discusses benefits and concerns regarding the use of A3 architecture overviews **as part of the final documentation of a software program.**

During Creation

Tools. The author created all the A3s in Microsoft Visio. In addition, the stakeholders used pen, paper, and whiteboards when modeling and commenting on the different aspects of the system. By avoiding the use of custom-made tools, the development team did not need any introduction or education to create or comment on the A3s.

Communication with stakeholders. As observed by Meskestad (2014), discussions about architecture and design without the use of models could easily lead to misunderstandings. The author experienced this while discussing a subsystem with relevant stakeholders. When talking about the system without any model as reference, the conversation quickly became unproductive, as the author did not have a complete understanding of all system aspects. However, when the author had modelled that subsystem in a draft A3, the stakeholders could quickly point out mistakes in the A3. After the stakeholders had given their comments, the author could use the A3 as a reference for communication with other stakeholders. In this way, the A3 served as a conversation platform that helped the stakeholders discuss the same things. These findings coincide with (Engebakken, Muller, & Pennoti, 2010).

The author noted that the colleagues on the team were less willing to prioritize communication around A3s in stressed periods. They prioritized instead finishing the development of the software. The author found this understandable, as the project manager measured the software developers on finished functionality according to the work breakdown structure (WBS). A suggestion for future projects would be to include early validation and creation of A3s as part of the WBS. In that way, the colleagues would see the advantage of using time on A3 sessions for early validation.

Novelty of method. A concern when considering using A3 architecture overviews is the time needed to plan and execute the creation of the A3s. The first A3 took much more time than the later ones, as the creator quickly got the modelling experience needed. The author suggests that people responsible for creating A3s should have some experience in modelling functional views, physical views, design strategies, and quantifications. The use of a template mitigated the risk of wasting valuable time on the A3s, as the creator knew what to include in the A3. The use of a template also ensured a consistency between the different A3s, as they had the same structure. This coincide with (Meskestad, 2014), who also found that the use of a template "decreases the threshold to first-use, and prevent potential noise created by several different styles."

During Use

Meeting location. The author felt that it was natural to choose the location of the sessions based on the number of session participants. For sessions with two to three participants, the author chose in six out of six sessions to have the session in the office of one of the participants, or the coffee break location. The small tables and limited space forced the participants to focus on one A3 at once, and therefore increased the focus on the relevant sub-systems in the discussions.

For sessions with more than three people, the author chose in five out of six sessions a meeting room as location for the discussion. The author found that the meeting rooms did not fit well for discussing a single set of A3s with more than three people. This was due to too wide tables in the meeting rooms, making it hard to see content of an A3 from both sides of the table. For these sessions, the author printed several copies of the A3s. Polanscak (2011) also suggests printing several A3s for sessions with more than three meeting participants, in order to increase the attention given to the A3s on the table.

The average time spent on unplanned sessions in the coffee break location or colleague office was 15 minutes. However, when the author booked a meeting room or a skype meeting beforehand, the sessions lasted on average 66 minutes. The planned A3 sessions lasted on average 4 times longer than the unplanned A3 sessions. The author found that planned meetings always lasted as long as the allocated time, and that the conversations often lost focus at the end of long meetings.

Capturing feedback. Having an A3 that presented relevant information, allowed the author to get feedback from several stakeholders in different locations within short time frames. The quick feedback from different stakeholders made it possible for the author to detect inconsistencies or design faults.

In meetings where the author had printed out several copies of the A3s, the meeting participants took notes on the A3s. In two of the sessions, the allocated meeting time ran out before the author had collected all comments and feedback. After these sessions, the participants brought their copies of the A3s with them. Later, the author visited the participants' offices in order to collect all the feedback necessary.

Skype meetings. Before using the A3s to communicate with stakeholders in Korea, the author suspected limited interaction with the A3s during Skype meetings. However, during the meetings, the author noted that the meeting participants interacted actively with the A3s. Two workdays before these A3 sessions, the author sent the A3s to all meeting participants. The participants used the A3s to prepare before the meetings. During the skype meetings, the participants in both locations printed the A3s and put them on the meeting tables.

In two skype meetings, the software was running on a shared screen during the presentation. The author noted that even though one of the meeting participants presented and explained the software, the stakeholders in Korea used the A3s when discussing the software functionality. They talked in their local language while pointing on the A3s, and followed up by asking questions in English to the meeting participants in Norway. The author wrote any unanswered questions and proposed changes on the printed A3s with a red pen.

In one skype meeting, the meeting participants did not use the live software. Instead, the participants chose to use the A3s to discuss the software functionality. The author opened the A3s on a shared screen. Both participants in Korea and Norway could control the mouse and keyboard during the discussions. All participants could write comments and proposed changes directly in the A3s during the meeting. After the meeting, all meeting participants agreed that this was a successful way of discussing the software development, especially for unfinished software. One of the meeting participants noted the advantage of sending the A3s to all meeting participants beforehand.

A3s as part of system documentation

Reading documentation. The author observed in five out of five planned meetings that all the meeting participants read the A3s as preparation for the meetings. During the meetings, the participants commented that the novelty of the A3 format was interesting. This interest triggered the stakeholders to find time in their busy schedules to prepare for the meetings. On several previous occasions, the author has discovered that managers and engineers with tight schedules do not have the time to prepare for all meetings. The meeting participants commented that an advantage of the A3 format is that it consists of both visual models and text, making it more digestible than traditional text documents. One of the meeting participants commented that he did not have the time to study all the details on the A3s, but could use his available time on studying the visual models on the A3s.

Writing documentation. In the survey regarding internal tools and documentation (Figure 3), 11 out of 15 participants disagreed or strongly disagreed that "the internal tools are well documented." After investigating this question further with colleagues, the author found that some of the internal tools did not have any user manuals at all. This motivated the author to ask the software development department about the blockers that prevent system documentation. A finding was that the software developers often write the documentation after creating the software code. A risk when documenting software after finishing the code is that the developers start coding a new software program before the documentation is finished. This could especially happen in periods were several software programs have deadlines at the same time. Another finding was that software documentation is not a prioritized task in the software development department, as their focus is to deliver software with good quality. The author would suggest for this department to create A3s by using the same approach as in this study. The developers should create the A3s in parallel with the software development, to mitigate the risk of ending up with no documentation. The possibility of using the A3s as a communication platform would also be an advantage. As the developers could distribute the A3s to relevant stakeholders before releasing the software, the possibilities for early validation (Frøvold, Muller, & Pennoti, 2010) could motivate the department to increase their focus on documentation.

Learning new topics. One of the problems detected during the survey regarding internal tools and documentation (Figure 3) was that the current documentation does not help the users in understanding new software tools. When working on new software tools, it seems to be a good choice to use A3s with multiple levels of abstraction, both as introduction to the tools and as user manuals. According to Borches (2010), 73% of practitioners stated that for learning new topics, A3 Architecture Overviews (A3AOs) is better than documents. In this project, the top-level A3s introduced the overall system, while the lower-level A3s introduced each of the sub-systems. The author noted that stakeholders always understood the A3 documents, and could quickly comment on the content in the A3s. Stakeholders in Korea commented during the Skype meetings that the A3s gave a good illustration of the dynamics of the software, and that the A3s were easier to understand than traditional text documents. This feedback coincides with (Borches, 2010), who found that 80% of practitioners would rather read an A3AO than a document.

Reflection

The research methodology used in this study gave a thorough understanding of the stakeholders' involvement with the A3s. This understanding is mostly qualitative, as the number of actively involved stakeholders is relatively small. Therefore, the author has not attempted to quantify the success of introducing the A3s in this project. To validate further the benefits of A3s in software development projects, more people should be actively involved throughout an entire development project.

During the pre-development phase of the project, the author created and presented A3s with high abstraction levels for stakeholders in Norway. Later, when the author presented the A3s for stakeholders in Korea, the feedback during the presentation provided valuable input to the development team. To ensure active stakeholder involvement, and increase the benefits of early validation in the project, the author would suggest involving all relevant stakeholders in early A3 sessions.

A goal for this study was to ensure stakeholder communication and software validation by using A3 architecture overviews. The systems engineering handbook provides many ways of improving communication and performing validation of a system. This paper does not compare the A3 method to other engineering tools. Further, the author has not measured the results of the effort, as the company does not have a relevant baseline for comparing the results. Nevertheless, the observations and feedback during the study indicates that the implementation of A3s in this case study was successful.

Conclusion

Previous studies have stated that A3 Architecture Overviews can improve the communication in a project. This study confirms this during a software development project for The Company. In this study, the project participants used A3s with multiple levels of abstraction to validate and document various aspects of the system-of-interest.

The project team had good interaction with the A3s, both in spontaneous sessions, regular meetings and in Skype meetings. Participants could use the A3s to study the system-of-interest before meetings, and the A3 responsible and system responsible could use the A3s to capture feedback during and after meetings. Findings suggest that the A3 format could encourage reading and writing of documentation in hectic industrial settings. In addition, engineers could use the A3s as a tool for quick introduction to the system. However, the results suggest that encouragement from management with respect to early validation and A3 development plays an important role. This applied especially when developers had to prioritize between early validation and early software development.

The collection of A3s gave a good illustration of the software program in the early development phase, and became a natural part of the system documentation.

Future Research

In this study, the author adapted the cookbook presented by Daniel Borches (2010), in order to model the system-of-interest with multiple levels of abstraction. The author spent time early in the project on evaluating the applicability of the cookbook, and on creating a template for the lower-level A3s. For further study, the author suggests to create a cookbook or a set of templates customized for software development. Polanscak (2011) also propose developing, applying, and further analyzing cookbooks related to domains where the most important knowledge is not architectural based.

In a Skype meeting between Norway and Korea, the meeting participants cooperated on adjusting and improving the A3s in Microsoft Visio. The meeting participants found this interactive way of working with the A3s successful. A research question that needs further study is how increased interactivity of the A3s can improve the advantage of using A3s during meetings. The work done by Brussel and Bonnema (2015) deserves follow-up since it looks at interactive possibilities and how new technologies can make A3 Architecture Overviews a more useful tool.

Acknowledgements

The company supported the master project and writing of this paper.

References

A3 architecture overviews. (2015). Retrieved from Embedded Systems Innovation by TNO: http://www.esi.nl/solutions/future-proof-systems/a3-architecture-overviews.dot

Borches, D. (2010). A3 architecture overviews : a tool for effective communication in product evolution. *PhD thesis, University of Twente*.

Borches, D., & Bonnema, G. (2010). A3 Architecture Overviews - Focusing architectural knowledge to support evolution of complex systems.

Brussel, F. F., & Bonnema, G. M. (2015). Interactive A3 Architecture Overviews. Intuitive Functionalities For Effective Communication. *2015 Conference on Systems Engineering Research* (pp. 204-213). Procedia Computer Science.

Engebakken, E., Muller, G., & Pennoti, M. (2010). Supporting The System Architect: Model-Assisted Communication. *World Scientific Publishing Company. Systems Research Forum. Vol. 4, No. 2*, 173-188.

Frøvold, K., Muller, G., & Pennoti, M. (2010). Applying A3 reports for early validation and optimization of stakeholder communication in development projects. *Unpublished Master Paper*.

Kooistra, R. L., Bonnema, G. M., & Skowronek, J. (2012). A3 Architecture Overviews for Systems-of-Systems. *Complex Systems Design & Management*.

Meskestad, L. (2014). Test-focused Knowledge Sharing using A3-assisted Communication. *Unpublished master project, University College of Southeast Norway*.

Muller, G. (2016). LEAN and A3 Approach to Supporting Processes.

Muller, G., Wee, D., & Moberg, M. (2015). Creating an A3 Architecture Overview; a Case Study in SubSea Systems. *INCOSE International Symposium*, 448-462.

O'Brien, R. (1998). *An Overview of the Methodological Approach of Action Research.* Toronto: Faculty of Information Studies, University of Toronto. Retrieved from http://www.web.ca/~robrien/papers/arfinal.html

Polanscak, E. (2011). Supporting Product Development: A3-assisted Communication and Documentation. *Unpublished master project, University College of Southeast Norway*.

Riel, M. (2010-2017). *Understanding Action Research.* Retrieved from Center For Collaborative Action Research: http://cadres.pepperdine.edu/ccar/define.html

Singh, V., & Muller, G. (2013). Knowledge Capture, Cross Boundary Communication and Early Validation with Dynamic A3 Architectures.

Sobek, D., & Smalley, A. (2008). *Understanding A3 Thinking: A Critical Component of Toyota's PDCA Management System.* New York: Taylor & Francis Group.

Van de Laar, P., & Punter, T. (2010). *Views on Evolvability of Embedded Systems.* Springer Science & Business Media.

Wiulsrød, B., Muller, G., & Pennoti, M. (2011). Architecting Diesel Engine Control System using A3 Architectue Overview.

Author Biography

Anders Viken received his Master's degree in systems engineering from the University College of Southeast Norway in 2017. He worked as a system engineering student in a company working on maritime systems from 2014 until 2017. In 2014, he received his Bachelor's degree in computer science. Since August 2017, he has worked as a developer in the research and development department in Jotron AS in Larvik, Norway.

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 system architect, followed by two years at ASML as manager systems engineering, returning to Philips (Research) in 1999. Since 2003, he has worked as 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 full professor of systems engineering at University College of Southeast Norway in Kongsberg, Norway. He continues to work as senior research fellow at the Embedded Systems Innovations by TNO in Eindhoven in a part-time position.

All information (System Architecture articles, course material, curriculum vitae) can be found at Gaudí systems architecting http://www.gaudisite.nl/