A Comparative Case Study of Architecting Practices in the Embedded Software Industry

Håkan Gustavsson
Mälardalen University
Västerås, Sweden
hakan.gustavsson@mdh.se

Jakob Axelsson
Mälardalen University
Västerås, Sweden
jakob.axelsson@mdh.se

***Abstract*— The goal of this study is to improve the understanding of how architecting is performed within the field of software-intensive systems. Architects at six different internationally well-known companies have been interviewed to understand their way of working. This paper presents the practices that are found most successful. The context of the different companies as well as the architecting practices are compared and analyzed. Many of the architecting practices found in the study can be explained by the context of the different companies. The study shows that architects at all companies mention a general lack of understanding of software-intensive systems within industries that used to be mechanical. The architects' view of their work is very similar independently of where they work. Also the way architecting is performed is very similar, but surprisingly only one company has a defined process for architecting.**

Keywords-component; Architecting, Embedded systems, Case study, Process

I. INTRODUCTION

Many traditionally mechanical companies in industries such as automotive, telecommunication, process automation, and defense are becoming more software intensive. The rapid increase of new functionality implemented through software enhances the burden of the system architecture to enable future growth of the system. The architecture of those software-intensive systems describes its building blocks and their relationships to each other and to the environment [10].

Architecting is defined by Maier and Rechtin [16] as the process of creating and building architectures. In our work, architecting is viewed as the process of shaping the architecture to meet customer demand by balancing requirements, guiding principles and product vision. As we see it, the architecting process is central to, and dependent on, many factors within the organization. The architects are constantly forced to make decisions on opposing factors such as continuous evolution versus product stability [20]. To stay competitive, companies need to adapt their processes to include the new discipline of software engineering.

In order to understand how different external factors affect the architecting process and to look for successful practices, the following research question is stated:

In what contexts are the methods used within the architecting process successful?

This paper presents a comparison of how architecting is performed at different companies. In the following section, related work is presented. System architecting is further defined in Section 3. In Section 4 the methodology of the case study is presented. The characteristics of the case companies are presented in Section 5. Analysis of each company in the study is then presented in Section 6 followed by general case study findings in Section 7. The results are discussed in Section 8, and the final section summarizes the conclusions and give some indications of future work.

II. RELATED WORK

There are many methods and tools available to aid the architects in their work. Examples of structured methods mentioned in industry surveys [1] are Pugh evaluation matrix [19] and the analytical hierarchy process (AHP) [22]. Dobrica and Niemela [4] make a comparison of eight different available software architecture analysis methods. The study found the Architecture Trade-off Analysis Method (ATAM) [12] to be the most suitable. The Cost Benefit Analysis Method (CBAM) [11] is an extension of ATAM and uses the quality attributes from ATAM but also considers cost when reasoning around the most suitable architecture. In a study of 46 companies in Finland [23] it was shown that the most common (76%) used concept selection method was concept review meetings, and similar results where shown in [8].

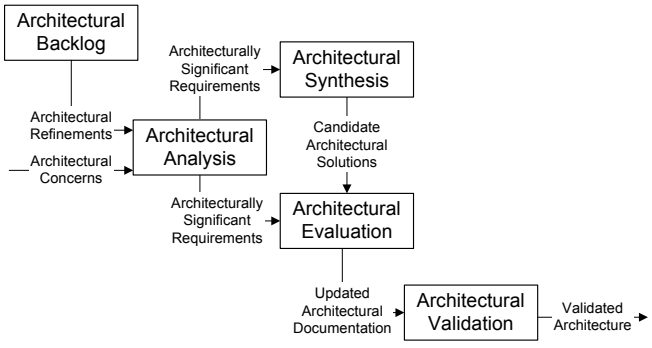

Figure 1. A generic process for creating and maintaining an architecture, adapted from [9].

There are very few publications on how architecting of software-intensive systems are done in practice. Decisions in the development process [8] and within the architecting

process [18] have been previously studied. Axelsson et al. [2] compare network architectures of three different automotive manufacturers and concludes that business and product characteristics have a large impact on the network architecture. Unphon and Dittrich [24] concludes that one must consider the organization and business domain when adopting a product line architecture. In a study of eight different software development organizations [25] it was found that the architecture is maintained and evolved through face-to-face communication rather than documents.

In a survey of 279 IT architects in the Netherlands, Farenhorst et al. [6] conclude that architects are lonesome decision makers, not very willing to share architectural knowledge, but eager to learn from others. A study made by Wallin and Axelsson [26] on architecture development at a car manufacturer presents a number of issues found within the process.

A generic process for creating and maintaining an architecture is presented by Hofmeister et al. [9]. That process is based on a comparison of five different software architecture design methods.

III. System Architecting

This paper will study how architecting is performed in different companies. The study was made on companies developing embedded systems including both hardware and software. These systems are mechatronic which adds complexity since many issues cross several engineering disciplines. The systems are resource constrained and trade-offs between the system behavior and the resources required are of great importance. Both hardware and software are mixtures of in-house development and deliverables from external suppliers. The systems are distributed on different hardware platforms and are sold in a large number of variants.

Architects will make different types of decisions depending on the companies' definition of their role. Decisions will range from choosing quality attributes to mapping communication [7]. The impact of the decision will also vary depending on how decoupled software is from hardware.

IV. Method

Different companies perform architecting in various ways and there are many different factors that influence. Many of those factors are thought to be soft factors [5] that are hard to find through, for example, a questionnaire. In order to understand the context in which different methods are being used, personal interviews was found to be the most appropriate method.

The case study was performed in seven steps:

1. The questions were developed and tested on people with similar roles, who were not included in the study.
2. Companies were chosen and a connection was established through a contact person. In collaboration with the contact person the architects were identified.
3. At least two interviews were held with architects at each company.
4. The current results of the study were presented to a broader audience at each company visited. During the presentation the situation at the visited company was also discussed.
5. Questions about the characteristics of each company were answered by the contact person.
6. The results were gathered in a database and analyzed.
7. The results were also reviewed by the contact person at each participating company.

The professional network of the authors was in many cases used to establish connections with the right persons and at one company the respondents were previously known to the interviewer.

The chosen format of the interview was semi-structured and the answers were audio recorded. A semi-structured interview has predetermined questions, but the order can be modified based upon the interviewer's perception of what seems most appropriate. Question wording can be changed and explanations given [21]. The interviews at all companies followed the same template and the answers given were then used to compare the companies.

To be able to compare the companies, a number of metrics were used that are presented in Table I. Every company and organization is different in many ways, and they may use different definitions of these metrics. We choose to use each company's own definition, rather than to enforce a common definition, since this increased the likelihood of getting good responses. The values have been given by asking, for instance, how many employees are working within the company's R&D organization. The answers will not be exactly comparable since R&D is not the same in all companies, e.g. supporting units are sometimes included or not. Even if the organizations would be the same, different companies count people differently, e.g. with or without consultants. The goal of the metrics is to give an overall picture of the different companies and that goal is thought to be fulfilled even if the definitions of the metrics are not exact.

V. Case Companies

The companies were chosen on three criteria:

- They do significant development of software-intensive systems.
- They are different in size and production volume.
- Together, they represent a mixture of different types of products and customers.

These criteria were chosen to give a broad spectrum of differences in business and organization, with the hypothesis that this should reflect differences in process and architecture [15, 27].

The studied companies are common in many ways. They are all financially successful and all have a very long Swedish history. They are also internationally well-known and considered premium brands within their business segments. The products are all software-intensive with a long

TABLE I. A COMPARISON OF THE CHARACTERISTICS OF THE STUDIED COMPANIES (ALL VALUES ARE APPROXIMATIONS).

Company / Context	Automotive 1	Automotive 2	Automotive 3	Defense 1	Industrial Automation 1	Industrial Automation 2
Size of R&D organization	Large	Very large	Large	Medium	Small	Small
Relative size of the embedded systems organization in comparison to total R&D	20%	13%	8%	18%	67%	24%
Number of architects	6	10	4	3+6	3+4	0-5
Management levels between architects and CEO	5	6	4	4	**2 and 4**	3
The power center of the organization	Line	Project	Project	Line	Project/Line	Project
Geographical locations of R&D organization	1	1	**~10**	1	2	3
Product variants	Very high	High	Very high	**Low**	Medium	Medium
In-house system development	50%	10%	80%	50%	**95%**	**90%**
Main customer	Business (small/large)	Private	Business (small/large)	Government	Business (large)	Business (small/large)
Magnitude of the investment for the customer	Medium/High	**Very high**	Medium/High	Small	Small	Medium/High

life-cycle (15-30 years) that may include multiple owners. In the following subsections, the characteristics of each company will be presented. The comparison is summarized in Table I and some clarifications of the measures are given below:

- The size of the R&D organizations and the number of product variants is relative in comparison to the other case companies.
- The relative size of the embedded systems organization is in comparison to the total number of employees within R&D.
- The measure "number of architects" shows how many architects that are working on a complete system level.
- The power centre of the organization describes if the architects consider the organizations to be project-oriented or line-oriented.
- The magnitude of the investment for the customer indicates the size of investment relative to the economy of the most common customer.

A. Automotive 1 (A-1)

This company produces commercial vehicles. The customers of the vehicles are both small and large companies. The product can be configured in a very high number of product variants. This is done using a common product line architecture that supports all different variants. The company has its R&D centralized to one location and has for a long time applied the thoughts of Lean [17] onto its development.

B. Automotive 2 (A-2)

This company is a car producer. The customers of the vehicles are mostly individuals and in some cases companies. This makes the magnitude of the investment for the customer often very high. The company has the largest R&D organization of the companies included in the study and its R&D centralized to one location. The relative size of the electronic and electric system development organization is 13 percent, which is explained by a low degree of in-house development. The architecting is divided into two groups responsible of traditional electrical systems and software-intensive systems.

C. Automotive 3 (A-3)

This is another producer of commercial vehicles. The company has R&D located at more than 10 different locations worldwide. As with A-1 the product can be configured in a very high number of product variants. The different product lines use the same software and hardware architecture on most in-house developed subsystems, but the interface between subsystems are not standardized between the different product lines.

D. Defense 1 (D-1)

As with most companies in the defense industry, the main customers are governments in different countries. The product variants are in comparison low. Customers usually purchase a unique variant of an existing product. The customer requirements are often detailed and may include demands on using a specific supplier of subsystems. The company has its R&D centralized to one location. There are three architects working on the complete system and six who work only with embedded systems.

E. Industrial Automation 1 (I-1)

The customer is mostly large companies. The development is mainly in Sweden, but some development is also done in Asia. The relative size of the electronic and electric system development organization is 67 percent, which is explained by a high degree of in-house development. The system is often integrated into a larger system. There are three architects working on the complete system and four who work only with embedded systems.

F. Industrial Automation 2 (I-2)

The customer of the systems is both small and large companies. As with I-1 the development is mainly in Sweden, but some development is also done in Asia and the US. The system is usually a major investment for the customer. The electronic and electric system development organization is the smallest of the companies included in the study.

VI. Analysis

The key architecting practices that differentiate from how work is done in the compared companies are presented below.

Company A-1 has a defined documented process for architecting. The progress of each task is visualized and controlled during a weekly follow-up meeting. Knowledge sharing is performed through lessons learned sessions after each large release. The high acceptance of processes makes architecting easier at A-1. The threats lie instead in the lack of tool support.

Company A-2 has separated the roles of modelling and architecting. The architects are not responsible for updating the architectural model. This is done by a group of people specialized in modelling. The architectural task is discussed at a weekly follow-up meeting. Company A-2 is also the only company in the study having a complete and updated model of the entire system. The division of the architecting into two groups does not seem to have any positive effects. Instead it causes friction and prevents the flow of information between the architects.

Company A-3 has been using a common software and hardware platform for a long time. This enables easy change of software components. The different product organizations are making decisions which affect the overall architecture without consulting the architects. The reason for this might be the relatively small amount of available architects.

The defense company D-1 was, not surprisingly, a master of requirement management. Requirement management is performed in the other companies, but not at the same detailed level. The requirement management system is also used to document reasoning of the design decisions. That knowledge is then used when changes are made to the design. The company manages to balance a strong system engineering practice with the agility of a medium size company. As with all the companies in the study this is historically a mechanical company, but the management's understanding of software-intensive system seems to be lower in D-1.

Company I-1 has two different types of architects: system architects and global architects. The global architect is the connection between strategy and business goals. The global architect has a budget and is thereby in a position to make larger architectural changes without being part of a project. Company I-1 uses roadmaps to communicate and create a common vision. This work is also a task performed by the global architect.

Company I-2 does not have the formal role of an architect, but is currently reviewing their way of working with electronic and electric system development. The need of some kind of coordinating role is very obvious and they are very aware of this fact. The different product lines have been developed more or less independently from each other and there has been little reuse of components. The company is very agile and innovative. In a future transformation those abilities must be given attention in order to keep that positive climate.

VII. Case Stud Findings

Architects at all companies mention a general lack of understanding of software-intensive systems within industries that used to be mechanical. The issue exists both at management level as found in [26] and with other stakeholders.

A. The Role of the Architect

The architects' view of their work is very similar independently of where they work. The architects primarily view themselves as facilitators, involving the right stakeholders in the architectural decisions or problem solving. They also consider themselves as coordinators and communicators of changes influencing the overall architecture.

B. Defining Architecture

When asked to explain what architecture means to them, most architects mention structure and form, some mention the building blocks and its interfaces. The user of the system is not often mentioned, only 40 percent. Only two architects mention business aspects and those two are both very senior:

The architecture is what connects the technology with the business model and culture of the company.

The architecture is the way we put the parts together to achieve our goal, but it also includes the organization and business.

C. Architectural Analysis and Synthesis

The most common methods used are design review meetings and safety analysis. Simulation of network utilization is also performed. One company has a predefined form for describing alternatives, but it is very rarely used. Alternative solutions are rarely documented or, as stated from one respondent:

Alternative solutions are often documented on a whiteboard or in some cases in an email.

D. Architectural Evaluation and Validation

There are no formal evaluation methods used as the ones mentioned in Section 2. Only one company mentioned feedback from test as a way of validating the architecture:

If it isn't a good solution we get to know there is a problem which we correct.

E. Process Improvement

The processes at all companies are very similar to the one described in Figure 1, with one big exception: there is no structured synthesis available at any company. It is also interesting that only company A-1 has a defined process for architecting. When asked what they would like to change in their way of working in order to improve, most mentioned how architectural knowledge [13] is managed.

The following answers to the question "How do you know if the architecting process is working well?" presents the architects' view of a healthy architecting process:

We do not really know, but the number of changes that are flowing the right way through the change review meeting is an indication.

When new functionality can be absorbed by the architecture without the need of large changes.

When the architecture is clearly communicated and there is no discussion about small issues.

F. Organization

As seen in Table I, the architectural teams are located on approximately the same hierarchical level relative to the size of the organization. The number of architects in A-1 and A-3 is significantly lower than A-2. This is mentioned as a problem by the architects at both companies. The two global architects at I-1 is the only case where architects have a clear responsibility for coordinating roadmaps.

A-3 is the only company with a large distributed development organization including sites worldwide. They experience difficulties in getting feedback on architectural changes. In the case of I-1 and I-2 the development made on other sites is very capsulated and they did not experience any large difficulties. I-2 had representatives from the other development sites on the main site. This made the cultural barrier less of a problem.

VIII. Discussion

The findings presented in the previous chapter are facts found analyzing the answers in the interviews. During the visits to the companies the authors have also built their own understanding of what the differences in how architecting is done depend upon. Those thoughts are presented below.

We see a clear correlation between the perceived maturity level [3] of the different organizations and how knowledge is shared. All companies have a very high degree of informal communication, but architects at the companies that have recurrent meetings are more pleased with the information available.

The different types of customer of the final products create different architectural concerns. The magnitude of the investment for the customer of products delivered by companies D-1 and I-1 are mostly small (Table I). This might be the reason why cost seems to be of lower priority at those companies. In contrast, at A-2 where the magnitude of the investment for the customer of the product is very high (Table I), cost is mentioned very often.

Kruchten [14] suggests that the productive time spent by architects can be classified into three categories of communication: internal (architecture design), inwards (input from outside world) and outwards (providing information). He argues that they should be roughly in the ratio 50% internal, 25% inwards, and 25% outwards. It is very hard to measure this in practice and we have not done so in this study, but communication patterns can still be observed. Even if no extreme variation can be seen, the understanding from this study is that there is a clear difference between the companies. The architects tend to be more satisfied when the inward and outward communication is distributed evenly and where the internal work is of significant size. Company A-3 and I-2 are examples of where the low number of architects supporting a large organization makes the time available for architecting too short. This results in architecting being performed by the developing groups without taking into account the overall system.

The power centers of an organization also affect how the work with the architecture is done. Nedstam [18] shows that there is a large difference in how work is done in an organization with strong line management and an organization with strong projects. This is found to be true also in this study. In the companies with a strong line organization, the line controls the architecting process, while in the companies with a strong project organization the process is controlled by the project. At company A-2 the power of development lies in the projects (Table I). The pressure from the projects might be the reason why the end customer is sometimes neglected. This could be the reason of the over-the-wall tendency, meaning that the deliveries of the documents are more important than the knowledge within.

IX. Conclusions & Future Work

This paper has presented the current state of architecting practices in three different industrial segments characterized by being software-intensive. For academia it presents a current view of how architecting is performed. The industrial reader is given a list of practices that can be used as an inspiration to improve the current architecting practice.

Many of the differentiating practices found in the study can be explained by the context of the different companies. The use of global architects with their own budget in I-1 is a solution to initiate long term architectural projects without having a customer order. The high degree of documented reasoning in D-1 is caused by the high degree of customer specific demands and large orders of very similar products. This forces the architects to make branches of the architecture to fulfill the customer demands and the reasoning is then used to ensure quality. The defined architecting process found at A-1 and the use of visualization

tools to track progress is explained by influences of Lean. Other practices such as the divided architectural teams in A-2 and the lack of formal architects in I-2 are more difficult to explain.

During the study it has been seen how the balance of power between line and project strongly affects how work is done. This relation would be of interest in a future study. The connection on how business strategy concerning Cost, Quality and Time-to-Market affects architecting could also be further analyzed.

The description of the architects as lonesome decision makers made by Farenhorst et al. [6] could not be seen in this study. One possible reason for this could be the cultural differences between Sweden and the Netherlands. Future work could therefore include studying companies in other countries. The methodology used was found to work very well. The presentation after the interviews at the visited company was found to be much appreciated. It was also an efficient way to validate the understanding given through the interviews.

ACKNOWLEDGMENT

We would like to thank all of the interviewees for contributing with all their experience and knowledge. Special thanks go to the contact persons at the different companies for coordinating the visits and reviewing the final result. This work has been financially supported by the Knowledge Foundation and the Swedish Agency for Innovation Systems (VINNOVA)..

REFERENCES

[1] C. S. Araujo, "The utilization of product development methods-A survey of UK industry," *Journal of Engineering Design,* vol. 7, pp. 265-277, 1996.

[2] J. Axelsson, J. Fröberg, H. Hansson, C. Norström, K. Sandström, and B. Villing, "A Comparative Case Study of Distributed Network Architectures for Different Automotive Applications," in *The Industrial Information Technology Handbook*, R. Zurawski, Ed. Boca Raton, USA: CRC Press, 2005, pp. 57-1 to 57-20.

[3] J. Axelsson, "Towards a process maturity model for evolutionary architecting of embedded system product lines," in *Proceedings of the Fourth European Conference on Software Architecture* Copenhagen: ACM, 2010, pp. 36-42.

[4] L. Dobrica and E. Niemela, "A Survey on Software Architecture Analysis Methods," *IEEE Transactions on software engineering,* vol. 28, pp. 638-653, 2002.

[5] R. Farenhorst and R. C. d. Boer, *Architectural knowledge management : supporting architects and auditors*. [S.l.: s.n.], 2009.

[6] R. Farenhorst, J. Hoorn, P. Lago, and H. v. Vliet, "The lonesome architect," in *Joint Working IEEE/IFIP Conference on Software Architecture (WICSA) & European Conference on Software Architecture (ECSA)*: IEEE, 2009, pp. 61-70.

[7] B. Florentz, *Software and System Architecture Evaluation and Analysis in the Automotive Domain*. Braunschweig: Technische Universität, 2008.

[8] H. Gustavsson and J. Sterner, "An Industrial Case Study of Design Methodology and Decision Making for Automotive Electronics," in *Proceedings of the ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference* New York, 2008.

[9] C. Hofmeister, P. Kruchten, R. L. Nord, H. Obbink, A. Ran, and P. America, "Generalizing a Model of Software Architecture Design from Five Industrial Approaches," in *Proceedings of the 5th Working IEEE/IFIP Conference on Software Architecture*: IEEE Computer Society, 2005, pp. 77-88.

[10] IEEE-1471, "IEEE Recommended practice for architectural description of software-intensive systems," *IEEE Std 1471-2000,* 2000.

[11] R. Kazman, J. Asundi, and M. Klein, *Making Architecture Design Decisions: An Economic Approach*: Carnegie Mellon Software Engineering Institute, 2002.

[12] R. Kazman, M. Klein, and P. Clements, "ATAM: Method for Architecture Evaluation," 2002.

[13] P. Kruchten, P. Lago, and H. van Vliet, "Building up and reasoning about architectural knowledge," *Quality of Software Architectures,* pp. 43-58, 2006.

[14] P. Kruchten, "What do software architects really do?," *Journal of Systems and Software,* vol. 81, pp. 2413-2416, 2008.

[15] S. Larsson, A. Wall, and P. Wallin, "Assessing the influence on processes when evolving the software architecture," in *Ninth international workshop on Principles of software evolution: in conjunction with the 6th ESEC/FSE joint meeting* Dubrovnik: ACM, 2007, pp. 59-66.

[16] M. W. Maier and E. Rechtin, *The art of systems architecting*. Boca Raton: CRC Press, 2002.

[17] J. M. Morgan and J. K. Liker, *The Toyota product development system : integrating people, process, and technology*. New York: Productivity Press, 2006.

[18] J. Nedstam, *Strategies for management of architectural change and evolution*. Lund: Lund University, Department of Communication Systems, Faculty of Engineering, 2005.

[19] S. Pugh, *Total design : integrated methods for successful product engineering*. Wokingham: Addison-Wesley, 1990.

[20] E. Rechtin, *Systems architecting : creating and building complex systems*. Englewood Cliffs, N.J.: Prentice Hall, 1991.

[21] C. Robson, *Real World Research-Second edition*: Blackwell Publishers Ltd., Oxford, UK, 2002.

[22] T. L. Saaty, "How to make decisions: The analytic Hierarchy Process," *Journal of Operational Research,* vol. 48, pp. 9-26, 1990.

[23] M. Salonen and M. Perttula, *Utilization of concept selection methods – a survey of finnish industry*. Helsinki, Finland: Helsinki University of Technology, 2005.

[24] H. Unphon and Y. Dittrich, "Organisation matters: how the organisation of software development influences the development of product line architecture," in *Proceedings of International Conference on Software Engineering*, Innsbruck, Austria, 2008, pp. 178-183.

[25] H. Unphon and Y. Dittrich, "Software architecture awareness in long-term software product evolution," *Journal of Systems and Software,* 2010.

[26] P. Wallin and J. Axelsson, "A case study of issues related to automotive E/E system architecture development," in *15th IEEE International Conference on Engineering of Computer Based Systems (ECBS)* Belfast: IEEE, 2008, pp. 87-95.

[27] F. van der Linden, J. Bosch, E. Kamsties, K. Känsälä, and H. Obbink, "Software Product Family Evaluation," in *Third International Software Product Lines Conference*. vol. Volume 3154 Boston: Springer Berlin, 2004, pp. 110-129.