



CHALMERS
UNIVERSITY OF TECHNOLOGY



UNIVERSITY OF GOTHENBURG

Improving Collaboration between Mechanical, Electrical, and Software Engineers during the Early Stage Design Process in Embedded Systems Suppliers

A Study on Mechanical-Electrical-Software Integration at Volvo Penta

Master's Thesis in Computer science and engineering

Eugene Dvoryankov

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2025

MASTER'S THESIS 2025

Improving Collaboration between Mechanical, Electrical, and Software Engineers during the Early Stage Design Process in Embedded Systems Suppliers

A Study on Mechanical-Electrical-Software Integration at Volvo
Penta

Eugene Dvoryankov



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2025

Improving Collaboration between Mechanical, Electrical, and Software Engineers
during the Early Stage Design Process in Embedded Systems Suppliers
Eugene Dvoryankov

© Eugene Dvoryankov, 2025.

Supervisor: Helena Holmström Olsson, Department of Computer Science and Media
Technology, Malmö University

Supervisor: Jan Bosch, Department of Computer Science and Engineering,
Chalmers University of Technology

Examiner: Philipp Leitner, Department of Computer Science and Engineering,
Chalmers University of Technology

Master's Thesis 2025
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in L^AT_EX
Gothenburg, Sweden 2025

Eugene Dvoryankov
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

The increasing reliance on software in embedded systems has created new challenges for suppliers that traditionally focused on mechanical and electrical domains. In this thesis, collaboration between mechanical, electrical, and software engineers during the early-stage design process was analysed using a qualitative case study at Volvo Penta. 15 semi-structured interviews were conducted with engineers, architects, project managers, and leaders from five Volvo Group companies. Thematic analysis was applied to identify recurring patterns and challenges. 17 subthemes were generated and organised using BAPO model, under 4 main themes: Business, Architecture, Process, Organisation. Relations between these subthemes were analysed through BAPO framework. It was found that early alignment can be strengthened by new platform architecture, a new rule-based configuration tool that follows Model Based Systems Engineering practices and re-organisation to create a new unified role for a person to coordinate Mechanical, Electrical and Software disciplines. The results contribute to research on socio-technical alignment in embedded systems by combining perspectives from Model-Based Systems Engineering, Set-Based Concurrent Engineering, variability management, and the BAPO framework.

Keywords: Embedded systems, Cross-disciplinary collaboration, Early-stage design, Configuration management, Model-Based Systems Engineering (MBSE), Set-Based Concurrent Engineering (SBCE), Variability management, BAPO framework

Acknowledgements

I would like to show my gratitude towards my academic supervisor Jan Bosch who helped establish contact with Volvo Penta. I am grateful for the opportunity to do my thesis at Volvo Penta and pursue my interest towards hardware and software collaboration. At Volvo Penta, Anders Sandberg who was my industrial supervisor has been instrumental in helping me find people to interview and establishing important connections. Anders went above and beyond by mentoring me and providing important lessons during our weekly meetings. The random conversations outside of meetings were a great moral support during my time at Penta. I would like to acknowledge Torbjörn Holmström's contribution as he provided his guidance during weekly meetings and helped me to navigate the internal culture at Penta. Last but not least, I have to show my great appreciation to the amazing academic support I got from Helena Holmström Olsson. Her expert knowledge on the nuances of Master Thesis writing has greatly improved the scientific quality of this thesis report. She answered on an impressively short notice and all her feedback was top-notch accurate. I appreciate the help of people who agreed to be interviewed as without your help this thesis would not be possible. The weekly online meetings I had with Chalmers Software Center were amazing and is an approach I would definitely recommend to other researchers. I recommend all supervisors, both academic and industrial, to have weekly meetings with their PhD and/or Masters students to express how their week went, share their news and give the students a chance to get help.

Eugene Dvoryankov, Gothenburg, October 2025

Contents

List of Figures	xiii
List of Tables	1
1 Introduction	1
1.1 Problem Description	1
1.2 Purpose of the Study	2
1.3 Significance of the Study	2
1.4 Research questions	3
2 Background	5
2.1 What is an Embedded System?	5
2.2 Volvo Penta context	5
2.3 Electrification	6
2.4 Systems Engineering	6
2.5 Model-Based Systems Engineering	7
2.6 Set-Based Concurrent Engineering	7
2.7 Configuration rules for embedded systems	8
2.7.1 Rule-based validation methodology	8
2.7.2 Variability in Software Product Lines	8
3 Literature Review	11
3.1 Model-based Systems Engineering in Embedded Systems	11
3.2 Model-based Systems Engineering combined with Set-Based Concurrent Engineering	12
3.3 Business Architecture Process Organisation (BAPO)	13
3.4 Semi-structured interview papers	16
4 Research method	17
4.1 Data collection	17
4.1.1 Interview themes and questions	19
4.2 Data analysis	20
4.2.1 Themes and thematic codes	20
4.3 Open coding	21
4.4 Axial coding	22
4.5 Principles and guidelines followed	23

5	Results	25
5.1	Business	25
5.1.1	Subtheme B1: Reuse economics' benefits and limits	25
5.1.2	Subtheme B2: Sales acceleration without early validation	26
5.1.3	Subtheme B3: Customer offers vs. platform architecture	27
5.1.4	Subtheme B4: External system coupling raises internal costs	27
5.1.5	Subtheme B5: Need for predictable configuration in sales-as-a-service	27
5.2	Architecture	27
5.2.1	Subtheme A1: Internal vs. external development trend	28
5.2.2	Subtheme A2: Overlooking platform architecture	28
5.2.3	Subtheme A3: Lack of unified architecture across Volvo Group	29
5.2.4	Subtheme A4: Rising product complexity	29
5.3	Process	29
5.3.1	Subtheme P1: Disagreement on configuration processes (CTO vs ETO)	29
5.3.2	Subtheme P2: Configuration tool and configuration space	30
5.3.3	Subtheme P3: Early versus late customer choices	31
5.3.4	Subtheme P4: Testing and verification practices	31
5.3.5	Subtheme P5: Legacy and paper document processes	31
5.4	Organisation	31
5.4.1	Subtheme O1: Risk aversion and overlapping transformations	32
5.4.2	Subtheme O2: Responsibility for product variants	32
5.4.3	Subtheme O3: Lack of unified cross-disciplinary role	32
6	Discussion	33
6.1	Platform architecture	33
6.2	Configuration tool	34
6.3	Product variant growth	34
6.4	Late testing	35
6.5	Organisation	35
6.6	Contributions	35
6.7	Limitations and threats to validity	36
6.8	Future work	38
7	Conclusion	39
	Bibliography	41
A	Appendix	I
A.1	Person 1 - Volvo Penta	I
A.2	Person 2 - Volvo Penta	II
A.3	Person 3 - Volvo Penta	II
A.4	Person 4 - Volvo Penta	III
A.5	Person 5 - Volvo Penta	IV
A.6	Person 6 - Penta	V
A.7	Person 7 - Penta	V

A.8 Person 8 - Penta	VI
A.9 Person 9 - Group Trucks Technology	VII
A.10 Person 10 - Group Trucks Technology	VIII
A.11 Person 11 - Volvo Construction Equipment	IX
A.12 Person 12 - Volvo Construction Equipment	IX
A.13 Person 13 - Volvo Construction Equipment	X
A.14 Person 14 - Volvo Autonomous Solutions	XI
A.15 Person 15 - CPAC Systems	XI

List of Figures

3.1	The BAPO model	14
3.2	An example of how BAPO scores may look before and after	14
4.1	How open coding was conducted using a transcript	21
4.2	How a code was created using paragraphs	22
4.3	How multiple codes became a subtheme	22
5.1	Business subthemes and respondent recommendations	26
5.2	Architecture subthemes and respondent recommendations	28
5.3	Process subthemes and respondent recommendations	30
5.4	Organisation subthemes and respondent recommendations	32

1

Introduction

Developing modern engineering products is getting harder and harder as the number of features goes up and development cycles time goes down [1][2]. Embedded systems, that are found in cars, power systems, or even home appliances, are gradually getting more software-controlled features during the recent decades [3]. Features that used to rely purely on hardware (mechanical and electronics) now also depend on software to add more functionality, while improving energy efficiency and increase profitability [4].

This software driven shift is a source of modern innovation, and also a source of new challenges. Embedded system companies manage product development by dividing into three separate engineering disciplines: mechanical, electrical, and software [5]. Each of those disciplines have different development speeds, different standards, different tools and need to develop the same product in parallel to each other, while depending on the success of previous teams [6].

1.1 Problem Description

In this thesis, Volvo Penta has been chosen for a case study. This company is a subsidiary of Volvo Group that also owns "Volvo Construction Equipment" and "Volvo Trucks" among other companies.

At Volvo Penta, where this thesis is based, the previously mentioned challenges are very real. The company operates in two main segments: marine and industry. The marine segment is responsible for engines and drivelines for recreational boats and commercial vessels, as well as marine generator sets. The industrial segment creates engines for various machines, generator sets, and energy storage solutions [7]. Penta is an Original Equipment Supplier (OES), which means that they sell components that will be assembled into a complete product by their customers and sold further. One such example is marine engines that Penta sells to a shipbuilding company that sells a complete ship with Penta's engine inside it.

Volvo Penta's products are characterized by low production volumes and highly priced products. The company produces highly customizable components, focusing on meeting each of their customers specific needs, instead of an one-fits-all approach

which is unpractical for most their customers. In this company, misalignment can cause delays when testing how Mechanical-Electrical-Software sub-components interact. As more features in these products depend on software, it's getting harder to manage this kind of complexity.

1.2 Purpose of the Study

The reason why this problem is being addressed now, rather than 20 years ago, is due to Volvo Penta's recent transition toward electrification. This transition began approximately 5–10 years ago, depending on interpretation. The introduction of electric vehicles (EV) coincides with a significant increase in software content, as well as software complexity.

Historically, software development has been largely outsourced within the Volvo Group (Penta is one of many subsidiaries of Volvo Group), which may have made software related issues less visible than hardware ones. As new EV systems need tighter coupling between hardware and software, the resulting complexity reached a boiling point where software complexity became too expensive to manage through legacy processes or informal practices alone.

In short, electrification has made increased software dependency a higher priority, which prompts a need for more structured, cross-disciplinary collaboration between Mechanical, Electrical, and Software Engineers to address that.

The purpose of this thesis is to understand out how mechanical, electrical and software engineers can better align throughout the early-stage engineering design process. This thesis examines the workflows, communication methods, and tools Penta uses. The goal is to identify the root cause of the misalignment and to suggest specific ways to improve the collaboration between these disciplines.

By addressing this topic, this thesis hopes to provide engineers with better ways to work together, especially as products become more dependent on software. It is also a chance to make a theoretical contribution to research on how to handle these multidisciplinary challenges in a more structured way.

1.3 Significance of the Study

Improving collaboration of engineers from mechanical, electrical and software can carry measurable benefits, such as a theoretical 25 % cost saving during early stage design [8]. This progress is partly attributed to instant information access for all engineers. In addition, by combining new frameworks with established ones, researchers have manage to streamline early stage design [6]. This was achieved by reducing the number of possible system solutions from 200 initial configurations to 6 viable ones, and finally 1 optimal configuration [6]. This streamlining of early stage design process has resulted in time and cost savings during this stage.

1.4 Research questions

In this thesis, a few assumptions were made. The first assumption is that there should be more cooperation between Mechanical, Electrical, and Software engineers. The second assumption is that lack of alignment between engineers across different fields happens due to different workflows, priorities, and communication methods that are unique to each field. Building on these assumptions, two research questions were proposed:

- **Research Question 1:** How do engineers from Mechanical, Electrical, and Software disciplines align their workflows, priorities, and communication strategies to overcome challenges and enhance collaboration during the engineering early design process?
- **Research Question 2:** How can alignment between workflows, priorities, and communication strategies be improved during the early stage engineering design process?

2

Background

2.1 What is an Embedded System?

An embedded system is a system that has been programmed to control a specific function, is installed directly into the equipment and can not be re-programmed by the end user [9]. Embedded systems are used in consumer appliances, aviation, space industry, etc [3]. Heath et al. [9] explains how the user can choose how to use the embedded functionality but can not add or replace software to change the original functionality. The author explains that a Personal Computer, on the other hand, can be used as a word processor, gaming machine or other things, by changing the software of the PC. Contrast this with an embedded, that is designed to do only one specific task, including the options provided to control only that task, as Heath explains. This distinction is important to understand that an embedded system is different to a PC by the fact that a PC can be reprogrammed to add more tasks than originally designed by reprogramming the PC with new software, as one of Heath's key points.

2.2 Volvo Penta context

Volvo Penta is a global supplier of engines and complete power solutions for marine and industrial applications [7].

This company has been increasingly shifting towards software-defined features as part of its broader digital transformation and electrification efforts. This transition reflects a significant move from traditional hardware-centric designs to more integrated, software-driven solutions.

One important development is Penta's introduction of a connected platform aimed at enhancing off-highway vehicle productivity. This platform makes it possible for manufacturers to customize their machines with advanced features such as 360-degree camera views and sophisticated propulsion control systems. By using a high-performance computational node, the system connects to various communication mediums, sensors, and subsystems, helping the development of software-defined machines [10].

Additionally, Volvo Penta has unveiled the IPS Professional Platform, which uses efficiency-driven features like automatic engine start and stop based on power requirements. This innovation is an important move towards intelligent, software-controlled power management in marine applications. [11]

These initiatives are part of Volvo Group's overarching strategy to lead the digital transformation in the commercial vehicle sector. In collaboration with Daimler Truck, Volvo Group has announced plans to form a joint venture focused on developing a software-defined vehicle platform, underscoring the company's commitment to advancing software integration across its subsidiaries, including Volvo Penta [12].

2.3 Electrification

In the marine domain, Volvo Penta has introduced a fully integrated helm-to-propeller hybrid-electric package for commercial vessels. This system combines diesel engines with electric motors and battery banks, allowing seamless transitions between electric, hybrid, and hybrid boost modes [13].

In the industrial sector, Volvo Penta is developing modular and scalable Battery Energy Storage Systems (BESS) to support electrification in construction and mining applications. These systems are designed to be integrated into their customers' applications, providing efficient and reliable power solutions [14].

2.4 Systems Engineering

Systems engineering is an interdisciplinary field of engineering and management that focuses on how to design, integrate, and manage complex systems over their life cycles [1]. At its core, systems engineering evolves with time to address the ever changing demands enforced by harsh competition. Systems are no longer comprised of isolated files, but of interconnected systems of systems (SoS). As changes happen, the original requirements change in an unpredictable way. An example given by the authors is a mobile device that started with e-mail messages but with time added GPS, Internet, videos and social media [1].

Systems Engineering Body of Knowledge SEBoK [15] describes Systems Engineering as an interdisciplinary approach to solve problems in complicated engineering systems. According to Systems Engineering practices, mechanical, electrical and software systems (M-E-S), should be designed together concurrently [16]. More specifically, the authors recommend the M-E-S disciplines to collaborate during design and requirements process. This is done through M-E-S cross-disciplinary collaboration with design, verification, integration, and maintenance of complex systems.

2.5 Model-Based Systems Engineering

Model-Based Systems Engineering (MBSE) provides a structured approach for managing the design and integration of complex systems. Before the MBSE, to keep track of specifications and documents, an exchange between different stakeholders (developers, testers, users and customers) was popular [17]. The documentation-based approach was mainly based on paper copies, while MBSE is based on 3-dimensional computer-aided design models, among others. When you separate the features into distinct models and visualize them on a computer, you can add more features without losing understanding of the complete system [17].

MBSE helps Mechanical, Electrical, and Software (M-E-S) by providing tools such as SysML to define and visualize system architectures [17]:

- Mechanical engineers can model geometries, tolerances, and physical interfaces
- Electrical engineers can define signals, voltages, and power flow
- Software engineers model control logic, timing constraints, and communication protocols

2.6 Set-Based Concurrent Engineering

Set-Based Concurrent Engineering (SBCE) is an approach in product development where teams evaluate a set of possible solutions that meet requirements, rather than choosing a single concept and then tweaking and changing it so that it meets the requirements [18]. Over time multiple options are evaluated and the worst options removed as trade-off between different requirements are better understood. Important decisions are delayed until information is available to make rational system-level choices.

At Toyota where this methodologies was first discovered, multiple models and system prototypes are developed in parallel [18]. Teams use engineering checklists to define the range of possible solutions and communicate these to each other. Suppliers are also asked to come up with and prototype several alternatives, not just build a single assigned design. Among its benefits, SBCE reduce late rework and expensive changes by delaying decisions until late moments. It increases the likelihood of finding better solutions through exploration of many possible early solutions. SBCE improves system-level integration by promoting cross-functional collaboration. It also builds organizational knowledge by documenting alternatives, trade-offs, and outcomes.

2.7 Configuration rules for embedded systems

In this section 2 different approaches for setting up configuration rules will be presented.

2.7.1 Rule-based validation methodology

In their paper, Li et al. proposed a rule-based validation methodology to efficiently solve hardware-related configuration problems [19]. Their methodology rests on three steps:

- A formal configuration rule language, inspired by linear temporal logic (LTL), to define valid configurations
- A configuration rule data base that stores the valid configuration rules
- An analysis tool that verifies no configuration rule violation at runtime

Linear temporal logic is a first order logic [20]. First order logic enables the use of sentences that contain variables [21]. In this case, instead of the proposition: John is a man, one can state: "X is a man", where X is a variable [21].

2.7.2 Variability in Software Product Lines

In their paper, van Gorp et al. [22] describe variability management. Their methodology includes 4 steps:

- **Identification of variability:** In this step the variation points are created based on specific variability needs outlined by a so called "feature diagram".
- **Constraining variability:** To provide a cost-effective solution, the variant points need to be constrained with some flexibility in mind.
- **Implementing variability:** A few realization techniques can be chosen between so that variability can be implemented in the most cost-effective way.
- **Managing the variants:** To avoid losing cost-effectiveness, variants must be added and removed based on current and future needs.

The authors acknowledged the possibility for late configuration, which aligns with Penta customer's need as it was mentioned by Person 3 in their interview, see Appendix A.

... new software systems are required to employ various variability mechanisms that allow the software architects and engineers to delay the decisions concerning the variants to the point in the development cycle that optimizes overall business goals. For example, in some cases, this leads to the situation where the decision concerning some variation points is

delayed until runtime, resulting in customer- or user-performed configuration of the software system. In other cases, variability can be handled before compilation, thus removing complexity of the final product.“
(van Gorp et al. 2001)

This example provides justification for evaluating van Gorp et al. methodology in the case of Volvo Penta.

By implementing the Software Product Lines, a company aims to make developing and evolving products more cost-effective [22]. According to the authors, this is achieved by grouping similarities between products into a *product family*. Van Gorp et al [22], refer to a follow up paper by Rine et al. [23] who conducted an empirical study which shows that organisation get the best reuse outcomes during early stages of development. This aligns well with the scope of this thesis that is specifically concerned with early stages of development.

3

Literature Review

In a Systematic Literature Review (SLR) on the quality of embedded systems, 64 studies published between 1980 and 2024, were analysed by Şahin and Tarhan [24]. The authors have captured a number of trends using those studies in the embedded systems. One of trends is that the majority (51 studies) of analysed studies were solution proposals, which could signal a shift towards empirical assessment of proposed solutions. Another trend evolves around different domains of embedded systems are being evaluated, thus showing that more empirical studies with real-life validation needs to be made.

The authors have proposed three recommendations for future research [24]. These recommendations could provide some benefit to this thesis. The first recommendation is to focus on details on the product, code and component levels. The second recommendation is making an empirical study and to validating a proposed solution in industry setting. The third recommendation is creating tools and framework for evaluating and improving quality in embedded systems.

This thesis extends on the work of Şahin and Tarhan [24] by conducting an empirical study in a new industrial context with 16 interviews, thus directly answering the SLR's call for empirical evidence. A contribution to BAPO framework was made by applying in a previously untested context of Volvo Penta.

3.1 Model-based Systems Engineering in Embedded Systems

Akundi and Lopez wrote a paper about transition towards Model-Based systems engineering (MBSE), highlighting the tools and languages that were used during the transition process [25]. The authors found that Systems Modeling language (SysML) is the most widely adopted modeling language during the transition phase towards MBSE. If adopted correctly in the early-stage development, MBSE can improve information sharing by reducing the time needed to find relevant information and that would enhance integration across mechanical, electrical, and software disciplines. The review mentions that adopting MBSE can have a positive impact on Return on Investment (ROI) as well.

This thesis has found that Penta has not fully transitioned from paper copies and Excel documentation for product variant management, which confirms Akundi and Lopez [25] findings on document-centric systems engineering and the need to transition to MBSE in a new industrial setting at Volvo Penta.

Vogelsang et al. wrote a paper examining the organizational forces that drive or hinder the adoption of Model-Based Systems Engineering (MBSE) in the embedded systems industry [2]. Rather than focusing on specific tools or modelling languages, the authors conducted a qualitative study to understand the socio-technical and organizational conditions affecting MBSE adoption. The authors found that unrealistic or too optimistic expectations are among the biggest causes of negative experience with MBSE adoption. These expectations are driven by the need to manage increasing product complexity, improve product quality, and accelerate time-to-market.

However, significant problems to MBSE adoption also include factors such as resistance to change, inertia due to existing company culture, and misunderstanding about MBSE. Specific barriers to MBSE include the perceived incompatibility of MBSE tools with existing toolchains among the people that were interviewed. And also the need for engineers to adopt new ways of thinking—particularly in terms of abstraction.

This thesis addresses gaps from Vogelsang et al. [2] by proposing a rule-based configuration approach that would use MBSE logic with variation points and constraints tied to model elements.

3.2 Model-based Systems Engineering combined with Set-Based Concurrent Engineering

A 2019 study by Ammar et al. [6] in an automotive company on Mechanical-Electrical-Software discipline collaboration has shown to improve communication between participants by decreasing time and costs of a complex system development. The authors came up with a new method to connect Model-Based Systems Engineering (MBSE) with Set-Based Concurrent Engineering (SBCE) to reduce design alternatives and find an optimal configuration more efficiently. To visualise their activities, the researchers used SysML diagrams.

This approach was validated in an automotive setting, by developing a new concept solution for an automotive component, using the combined MBSE and SBCE method. The authors were able to generate 200 solutions and then by following the methodology, narrow down the solution space from 200 possible configurations down to 6, and eventually one final optimal solution.

However, the paper also has certain limitations. Their methodology is applied in a narrow case with a specific automotive component, and is not validated in a more complex case, such as a completed vehicle. Furthermore, the proposed method relies

on expert's knowledge to reduce number of solutions. This introduces a subjective element to the decision-making process.

During the semi-structured interviews, this thesis has found a new rule-based configuration tool was suggested during interviews. In theory, such tool could implement Ammar et [6] MBSE and SBCE method to generate many solutions for potential products and following their method, narrow down the solution to one product that can be offered to Penta's customers. This thesis shows that theoretically Ammar et [6] method can be used in an new industrial case at Volvo Penta.

Borchani et al. [26] present an alternative view on combining Model-Based System Engineering with Set-Based Concurrent engineering. The authors call their approach *Model based for reliability and manufacturability analysis (MBRMA)* which is a new way to reduce economic costs and development time of new products, while also improving communication, and filtering out suboptimal solutions, during early stage design phase.

Borchani et al. apply their MBRMA approach on an electrical component, where their approach reduced 15 potential solutions down to 3, with a final selection based on trade-off analysis with domain specific analyses. This analysis aims to select the last solution with the best score based on reliability, manufacturability, cost, and durability.

However, the paper has its limitations. Firstly, it relies on domain experts to define feasible design options and rank attributes like lead-time, production cost, and failure probabilities, noting that "experts intervene to rank attributes like lead-time or production cost in a scale of 1 to 5 which is not always accurate". This introduces subjectivity and variability into the design process. Secondly, while the authors proposed an algorithm to automate reliability and manufacturability analysis, they admitted that "due to the difficulty of implementing our approach in a SysML-based tool, we developed a proof of concept using Python language," thus revealing a gap in full toolchain integration.

This thesis has suggested that a new theoretical rule-based configuration tool could solve some of Volvo Penta's challenges. Such tool could potentially implement Borchani et [26] MBRMA method to generate many solutions for products and narrow down to one one product to be offered to Penta's customers. This thesis shows that theoretically Borchani et [26] method can be used in an new industrial case at Volvo Penta.

3.3 Business Architecture Process Organisation (BAPO)

A paper by van der Linden et al., introduced a framework for evaluating embedded systems that contain software [27]. The proposed framework for evaluation is based on the BAPO model. This is an acronym for Business, Architecture, Process and

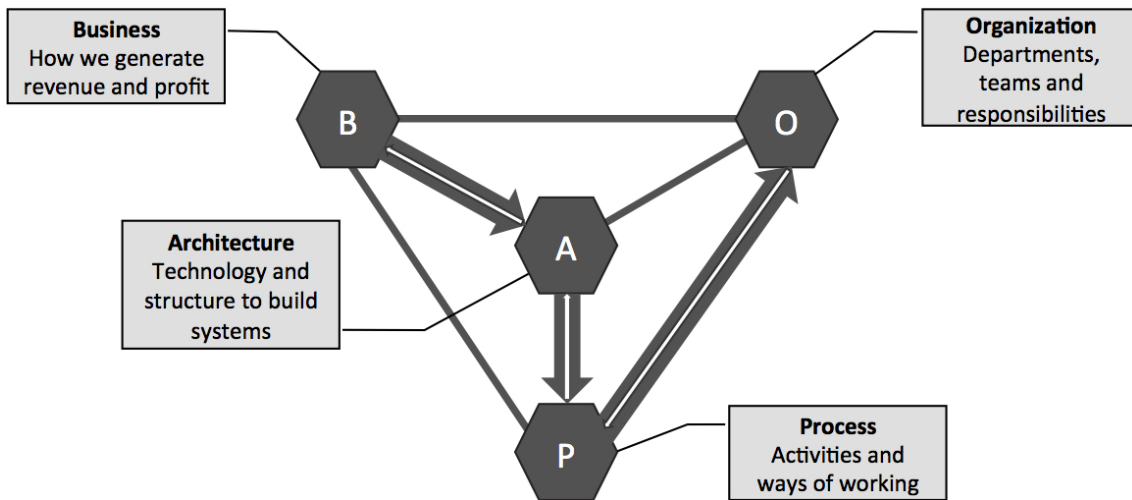


Figure 3.1: The BAPO model

Organisation model.

The interaction between Business, Architecture, Process and Organisation is such that a higher score in one dimension can, unless managed carefully, decrease the score of others, see Figure 3.2. An improvement should be evaluated through as many dimensions as possible so that an improvement in one dimension will not come at cost of other dimensions.

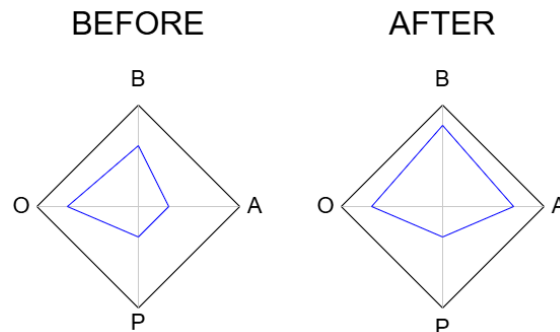


Figure 3.2: An example of how BAPO scores may look before and after

Business Dimension is about money-making and managing economic costs. One way to understand the business dimension is to look at the established business model in a company and ask some questions:

- *What costs do exist today?* If the company operate with a too complex **architecture** or the team's are not **organised** well enough, the costs will go up.
- *How satisfied are the customers?* By figuring out ways to build embedded products faster, cheaper and better than competitors, the company can expect higher profits and stronger market position.

Architecture dimension is concerned with how the software platform evolves with time. There is a lot change technologies which can be controlled by a coherent architecture strategy.

The following questions can be used to better understand architecture:

- What parts of the system are shared across all products, and how are they implemented?
- Is there a reusable software platform? If so, what does it include?
- Are the architecture decisions aligned with business goals?
- Is there a roadmap for architectural evolution as platforms change?

Process dimension explains the ways of working to make sure that products are made fast, cheap and with consistent quality. It includes planning, designing, coding, testing and maintaining.

The following questions can be used to better understand process:

- How much can be predicted before development is started?
- How much repetition there is in development?
- How easy it is to measure development?

Organisational Dimension is about how people, teams and entire departments are coordinated to ensure that people are working together without getting in each other's way.

These questions can be used to better understand organisation:

- How are responsibilities divided between people, teams and departments?
- How do different roles coordinate their tasks?
- Are teams located in the same building, or spread across cities or countries?
- Is there a clear structure or process for handling collaboration, communication, and decision-making?

This thesis has applied BAPO framework to organise findings from semi-structured interviews in the results section. Each of BAPO's four dimensions was applied to create a clear hierachy: theme -> subtheme -> code. By using BAPO as the top hiercrghy the interviews results could be used to confirm that BAPO framework is valid in a new industrial context at Volvo Penta.

3.4 Semi-structured interview papers

In their XP case study on integrating agile with stage-gate, Runeson and Höst chose a semi-structured approach to support their exploratory and explanatory goals [28]. The authors created an interview guide with topic blocks and rough time budgets. They selected relevant people with the companies, had two researchers present at all interviews, audio-recorded and transcribed all sessions, and also took notes during interviews. Later, the researchers coded transcripts (e.g., project model, communication, planning) and sorted people by their role and company to analyse patterns.

Runeson and Höst's [28] paper has been used as inspiration when conducting the 15 semi-structured interviews in this thesis. Specifically, the recommendations to audio-record interviews, while taking notes during interview session were helpful.

In their agile teamwork study, Strode et al. used semi-structured interviews in their case study about three co-located agile teams in a New Zealand bank [29]. Interviews were recorded, transcribed, and guided by a pre-defined schedule. The interviews they were complemented by observations and artefacts such as task boards, burn-downs, interface docs to ground accounts in “work as done”. The goal was to evaluate and adapt the “Big Five” teamwork model to agile contexts by mapping interview evidence to constructs such a shared mental models, trust, communication, leadership, peer feedback, redundancy, adaptability, and team orientation.

Strode's interviews shows how semi-structured interviews are applied to software engineering work such as role definitions and activities, how work is assigned or self-selected, who communicates with who and through which channels, any work-delegation patterns, required resources and tools (from configuration management to wikis), artefacts produced, and integration points. Beyond day-to-day work, their research explicitly targets coordination and dependency topics—big-picture architecture, testing flows and environments, parallel development and merge practices, change impact, expertise location, history of decisions, and integration cadence.

Methodologically, Strode demonstrates that semi-structured interviews work when you need context-rich explanations of socio-technical coordination—not just counts of practices. The same rationale underpins this thesis as early-stage alignment exists in interfaces, option decisions, gate evidence, and hand-offs, and those are best recovered by guided narrative plus artefact prompts, exactly as Strode operationalised.

Strode's [29] paper was used in this thesis to prepare before the interview studies. It was used to give a basic understanding of how semi-structured interviews were conducted in practice, as well as giving background on early-stage alignment.

4

Research method

This thesis applies a qualitative research method, in the shape of a semi-structured interview study to investigate the research questions found in Section 1.4. This methodology is chosen firstly because qualitative research is proven to be of great use for finding out why something is the way its is and common methods of data collection include semi-structured interviews [30]. The second reason for choosing qualitative research is to help the researcher discover new aspects of a field and form a deeper understanding of a subject [31]. The third reason for using qualitative research is to elaborate on theory. This means to create a concept and conduct empirical research and refine pre-existing ideas to develop new theory to improve accuracy of the phenomena studied [32].

Before a semi-structured interview, the questions are prepared in advance. During the interview, the questions are not asked in a predefined order, to allow interviewer to probe, improvise, and explore follow up questions as the conversation develops [28]. Semi-structured interviews are common in case studies because they balance comparability with openness. Interview sessions typically start with objectives and background, then move to the main questions (using funnel structure), and end with a brief summary for respondent feedback. Runeson and Höst recommend audio recording and full transcription so that transcripts can be reviewed by participants to go back to vague statements afterward [28].

Hennink and Kaiser have conducted a systematic review of empirical studies that assess saturation in qualitative research, and they have identified sample sizes for saturation given their limitations [33]. The authors suggest that 9-17 interviews or 4-8 focus group discussion are enough to reach saturation, if the study populations are homogenous, and the purpose is narrowly defined. Saturation in qualitative research is describing a situation where enough data has been collected, so that more data will not add more useful information to solve the problem [34].

4.1 Data collection

During a period of 6 months, between January and May 2025, 15 semi-structured interviews were conducted, see Table 4.1. The interviews lasted between 45-60 minutes. There were a set of predefined questions that have guided the interview

which are explained in Section 4.1.1. The candidates were asked to respond to a number of questions based on 3-4 themes that were chosen based on a person’s previous background and expertise. The follow-up questions were dependent on candidate’s previous answers.

Semi-structured interviews were used because they balance comparability with the ability to probe how alignment issues happen in their natural corporate environment. The interview respondents were recruited after careful consultation with a key employee at Penta who used his engineering expertise and connections to identify potential interview candidates.

	Role	Background	Organisation
P1	Architect	Hardware	Volvo Penta
P2	Architect	Software	Volvo Penta
P3	Systems Engineer	Software	Volvo Penta
P4	Systems Engineer	Software	Volvo Penta
P5	Project Manager	Hardware	Volvo Penta
P6	Architect	Software	Volvo Penta
P7	Project Manager	Hardware	Volvo Penta
P8	Architect	Hardware	Volvo Penta
P9	Systems Engineer	Software	Volvo Group Trucks Technology
P10	Systems Engineer	Hardware	Volvo Group Trucks Technology
P11	Architect	Software	Volvo Construction Equipment
P12	Architect	Hardware	Volvo Construction Equipment
P13	Architect	Hardware	Volvo Construction Equipment
P14	Architect	Hardware	Volvo Autonomous Solutions
P15	R&D Manager	Software	CPAC Systems

Table 4.1: Interviewee’s characteristics

4.1.1 Interview themes and questions

A set of interview questions were designed to answer the Research Questions, see Section 1.4, and to help learning about cross-disciplinary alignment, priorities, and improvements. Interview sessions were audio-recorded with consent and transcribed verbatim. All interview questions are attached to different dimensions of the research questions: exploring current alignment practices and challenges (RQ1), and looking at how those challenges might be resolved (RQ2). The interview questions in Table 4.2 are actual questions from the third interview, and are meant to demonstrate the interview process.

Main topic	Connected questions
Cross-disciplinary collaboration	How do engineers from M-E-S disciplines align their workflows?
Historical evolution of hardware and software usage	What has changed in the use of HW and SW in the past 20 years until now?
Alignment challenges and solutions	What challenges do you see to improve alignment in early stages?
Alignment challenges and solutions	What solutions do you see to improve alignment in early stages?
Timing and organizational perspective	What has changed in the use of HW and SW in the past 20 years until now?
Timing and organizational perspective	Why are you looking at this problem now in 2025 and not 5 or 10 years ago?

Table 4.2: Interview questions from the 3rd interviews

Research Question 1: How do engineers from Mechanical, Electrical, and Software disciplines align their workflows, priorities, and communication strategies to overcome challenges and enhance collaboration during the engineering early design process?

- *How do engineers from Mechanical, Electrical, and Software disciplines align their workflows?* → Directly explores alignment practices among different disciplines.
- *What challenges do you see to improve alignment in early stages?* → Addresses the barriers or difficulties in effective workflow integration and communication, which is central to RQ1.
- *What has changed in the use of HW and SW in the past 20 years until now?* → Provides historical context that can reveal long-term shifts influencing current collaboration practices.

Research Question 2: How can alignment between workflows, priorities, and communication strategies be improved during the early stage engineering design process?

- *What solutions do you see to improve alignment in early stages?* → Explicitly targets potential methods, practices, or tools to enhance alignment, answering the “how to improve” aspect.
- *Why are you looking at this problem in 2025?* → Examines the timing and urgency behind the push for improvement, helping to understand contemporary threats and motivations.

4.2 Data analysis

This thesis used reflexive thematic analysis to analyse interview data as formulated by Braun and Clarke [35], [36]. In their 2006 article, they define thematic analysis as "a method for identifying, analysing and reporting patterns (themes) within data" [35]. Reflexive thematic analysis extends this view by highlighting that the researcher should take an active role in the analysis so that is "a recursive process, where movement is back-and-forth as needed, not a linear model of progression" [36]. Working from a critical-realist stance and recognising that a real world exists even if we can perceive the reality only imperfectly through our own ideas [37], themes were treated as "active constructions of the researcher, not passive discoveries" [36].

4.2.1 Themes and thematic codes

A complete list of all codes generated during the 15 interviews can be found in Appendix A, at the end of this report.

In thematic coding, the interview transcript is broken into smaller, meaningful codes [35]. The reader finds codes, pieces of content that contain valuable data, and assigns labels to them. Purpose of thematic coding is to organise the data and lay the foundation for an analysis.

When generating themes, the focus is shifted from individual codes to find broader patterns [35]. The themes are created using similarities, connections, and patterns between the codes. According to Braun and Clarke [35], it could be helpful to visually represent themes in tables or mind-maps. Purpose of generating themes is to identify patterns to gain deeper understanding of the root causes of the problems.

During thesis the following assumption was made: only the codes that are *relevant to research questions* are considered and documented in the thesis paper.

During the semi-structured interviews, the respondents were given the opportunity to express their opinions about different subjects, with some alterations made to accommodate for people from different background and different expertise. One such accommodation was not to ask respondents with hardware background about the software-specific details and vice versa. Another included to ask people who are employed by other Volvo companies to mention the situation at their company and Penta to make comparisons, even though only Penta's situation is within the scope of this thesis.

To decide what a relevant code should include, the following criteria was generated:

- **The keyword criteria:** A code should be based on a piece of transcripts and contains words from the Research questions 1 and 2
- **The company relevance:** A code should included information about Volvo Penta

- **The sensitivity criteria:** A code shall not contain proprietary knowledge or anything that can violate the internal security policies.

4.3 Open coding

After collecting the transcripts, open coding was conducted [38]. Each transcript was read twice. The first reading session was done in a normal reading pace, to remember the interview. The second reading session was slow and deliberate, focusing on each paragraph of text. It should be noted, that one paragraph is not always one thematic code. A code can be explained in multiple paragraphs of text. For simplicity, the example below will have one paragraph be equal to one code. A visualisation of a transcript can be found in Figure 4.1.

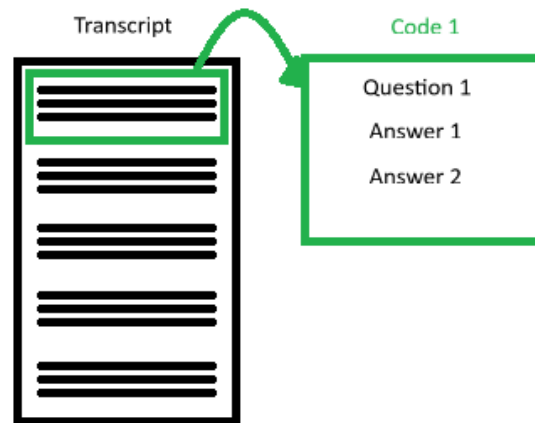


Figure 4.1: How open coding was conducted using a transcript

Each paragraph normally starts with a question that the interviewer asked. And then a paragraph ends with one or multiple answers for that question from an interview respondent. These answers contain ideas that are used to create a code. One or more paragraphs can contain ideas that will be translated into a thematic code. Those codes are then analysed to find patterns and some higher level logic between codes.

In Figure 4.2, presents an excerpt from the interview with Person 8. Here the question is concerned with any perceived change in collaboration between M-E-S teams at Penta. The respondent stated that they have not seen any improvements and that the people at their organisation work in SILOs, meaning everyone work in parallel and people do not collaborate. Based on the respondents answers, a thematic code was generated. The title for this code was chosen to be **Organizational silos**, which can be found in Appendix A, Person 8, Code 1. The answer was then rephrased to highlight collaboration challenges caused by Mechanical, Electrical, and Software teams working in silos.

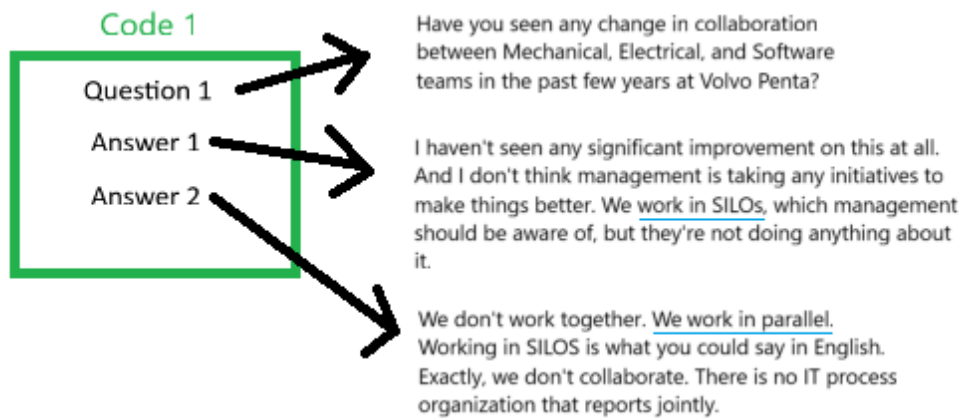


Figure 4.2: How a code was created using paragraphs

4.4 Axial coding

After the codes were generated, it was time to group them into subthemes following axial coding [39]. The thematic codes from previous step were analysed for patterns. For example, the code **Organizational silos**, see Appendix A, showed a lack of unified role for mechanical, electrical, and software disciplines in Penta's organisation. This code was assumed to be important enough to become a subtheme because it was directly connected to Research Question 1, as the code touch upon communication strategies between engineers from different disciplines. Following this logic, all subthemes were derived based any connection to at least one of the research questions. Some thematic codes were grouped together to form many to one relation with subtheme, while other codes had one to one relation with a subtheme. Not all codes were used to form subthemes as those were not deemed to be connected to the research questions. The subthemes were then put into an existing framework which is the BAPO model.

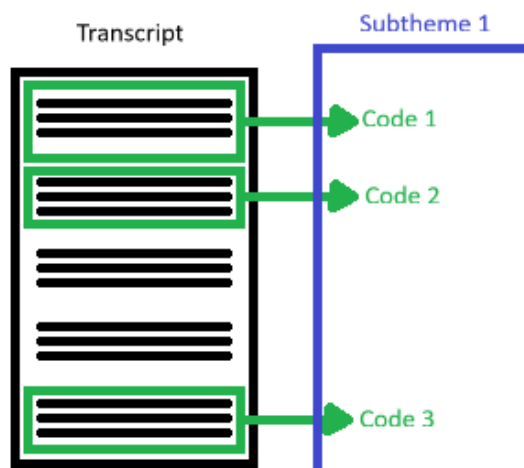


Figure 4.3: How multiple codes became a subtheme

4.5 Principles and guidelines followed

This thesis follows both the guidelines outlined by Hevner et al. [40] and the adaptation for the *Software Engineering and Technology* master's program at Chalmers provided by Knauss [41]:

- **Problem Relevance:** The misalignment of MES engineering practices is a significant industrial challenge that unless properly addressed will result in acceleration of diminishing returns for engineering.
- **Research Rigor:** Methods such as semi-structured interviews were applied consistently.
- **Semi-structured interview study:** The problem space was explored incrementally through interviews with people for various backgrounds, perspectives and organisations.
- **Research Contributions:** Contributions are made toward understanding of MES integration at new setting of Original Equipment Suppliers of Embedded Systems

5

Results

This chapter will contain findings from the semi-structured interview study, by presenting the themes identified during thematic analysis. A complete list of all thematic codes can be found in Appendix A. The interview material have been analysed according to Braun and Clarke's updated thematic analysis method [36], where the authors proposed a six-stage process for data preprocessing, thematic coding and theme creation.

The interview codes were organised using BAPO model. Each following section will explain the Business, Architecture, Process and Organisation aspects of Penta's current situation. Detailed analysis will be presented in Section 6, Discussion.

5.1 Business

The interviews describe two opposing views: advancing business deals with highly customised products (which drives up development costs) versus keeping those costs down. A strong customer centric sales strategy, without a strict platform architecture has resulted in more product variants than the organisation can track, leaving teams to rely on Excel and ineffective PDM tools. The subthemes below will explain the business challenges in more detail.

5.1.1 Subtheme B1: Reuse economics' benefits and limits

Recognised problem. Penta's business model encourages reusing hardware and software components across Volvo Group to spread development costs. However, adapting shared assets to Penta's unique applications introduces effort and cost. Some interviewee respondents expect "a breaking point" where component reuse becomes more expensive than focused in-house development for certain systems.

Interviewees suggested. There are no suggested solutions. **Mentioned by:** Person 1 (Code 1); Person 4 (Code 1, Code 2, Code 3).

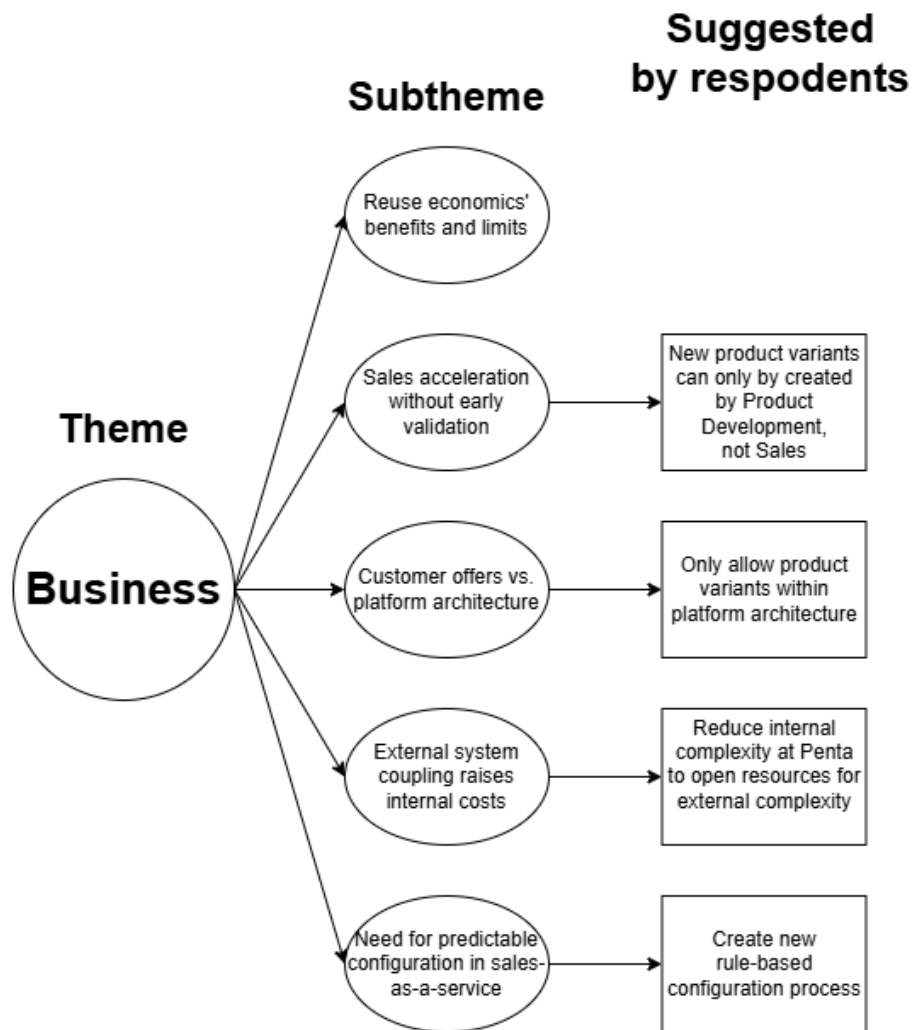


Figure 5.1: Business subthemes and respondent recommendations

5.1.2 Subtheme B2: Sales acceleration without early validation

Recognised problem. New products are suggested to customers before thorough technical validation, due to commercial pressure. Insufficient technical validation causes misunderstandings between sales and engineering and results in expensive fixes later in development. Interviewees described reliance on late physical prototype testing and integration bottlenecks, making problems harder and costlier to resolve. Product variants can enter the sales flow without sufficient review, which undermines platform rules and contributes to uncontrolled complexity. **Interviewees suggested.** To counterbalance businesses pressure, the authorisation for creating new product variants should be transferred from sales organisation to product development organisation. Sales should be able to propose new product variants but not be able to add those variants without explicit approval from product development, who should validate and authorise any suggested variant before it can be offered to customers. This suggestions could potentially slow down business deals alignment

with technical constraints and platform rules. **Mentioned by:** Person 3 (Code 1, Code 6, Code 7); Person 6 (Code 4); Person 7 (Code 3).

5.1.3 Subtheme B3: Customer offers vs. platform architecture

Recognised problem. Prioritising business deals at the expense of following a platform architecture has increased the number of product variants and increased product complexity. Sales engineers are not able to keep up with advancements and complexity of electrical vehicles and thus confuse EVs with simpler logic of diesel system which creates complexity when it previously did not exist. **Interviewees suggested.** By only allowing product variants within pre-defined platform architecture, there will be less product variants and less complexity. **Mentioned by:** Person 4 (Code 5, Code 6).

5.1.4 Subtheme B4: External system coupling raises internal costs

Recognised problem. Penta is a supplier that has customers that often combine Penta products with other suppliers' components, adding integration challenges that Penta cannot control. **Interviewees suggested.** It is implied that reducing internal complexity at Penta is necessary so that external complexity outside of Penta's control is easier to absorb. **Mentioned by:** Person 6 (Code 6).

5.1.5 Subtheme B5: Need for predictable configuration in sales-as-a-service

Recognised problem. As offerings become more service-like, customers need simple interfaces and predictable configuration pathways. **Interviewees suggested.** Interviewees expect codified rules to help customers and sales navigate legal and market constraints and platform limits. The mechanism of a rule-based configuration is presented in the Process section and is proposed, but not yet deployed. **Mentioned by:** Person 1 (Code 3).

5.2 Architecture

Interviews described how asset sharing across Volvo Group helps Penta to lower their development costs, but also creates complexity when Penta specific adaptations are required. Respondents also reflected that although local platform architectures exist at Volvo Trucks and Volvo Construction Equipment, there is the lack of a unified group-wide architecture. The subthemes below will explain the architectural challenges in more detail.

5.2.1 Subtheme A1: Internal vs. external development trend

Recognised problem. In earlier years Penta relied heavily on imported software due to limited in-house capacity. In the last five years, there has been a shift towards more internal development. **Interviewees suggested.** A respondent from Volvo Group Trucks Technology believes that if hardware and software complexity keeps rising, then more components will be developed in-house from scratch. This challenge is also present at Volvo Penta. Having more software development in-house to combat increasing software complexity, can be directly applied at Volvo Penta. **Mentioned by:** Person 4 (Code 1); Person 8 (Code 4); Person 9 (Code 6).

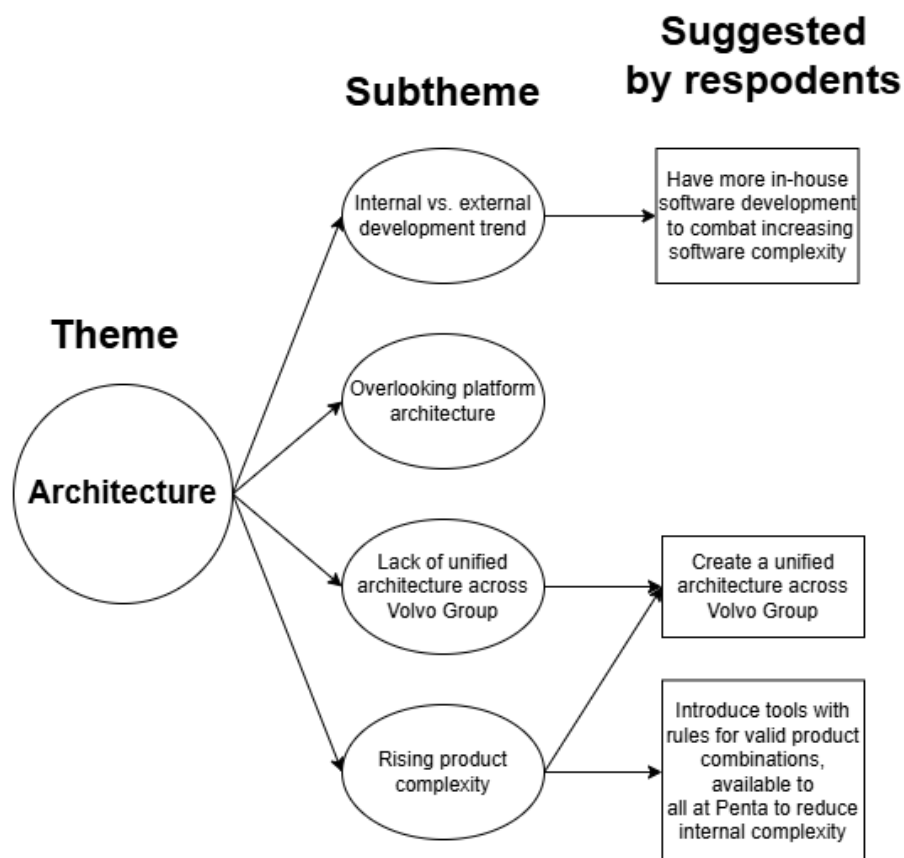


Figure 5.2: Architecture subthemes and respondent recommendations

5.2.2 Subtheme A2: Overlooking platform architecture

Recognised problem. Historically, Penta prioritised customer demands over platform architecture, contributing to a growing number of product variants and greater complexity. **Interviewees suggested.** There are no suggested solutions. **Mentioned by:** Person 4 (Code 5).

5.2.3 Subtheme A3: Lack of unified architecture across Volvo Group

Recognised problem. Other Volvo companies have their own architecture frameworks, but there is no single unified framework across the Group. Respondents from Penta and Volvo Construction Equipment have expressed a desire for unified architecture, and ongoing dialogue exists, though no concrete progress was reported. **Interviewees suggested.** Create a unified architecture framework across Volvo companies. **Mentioned by:** Person 7 (Code 6); Person 11 (Code 6); Person 12 (Code 4); Person 13 (Code 2).

5.2.4 Subtheme A4: Rising product complexity

Recognised problem. Interviewees mentioned that product complexity is growing faster than supporting models and tools can handle that complexity. **Interviewees suggested.** Introduce tools with rules for valid product combinations, available to all at Penta to reduce internal complexity. A respondent from Volvo Construction Equipment mentioned benefits of using Common Architecture Shared Technology (CAST) to address product complexity. **Mentioned by:** Person 1 (Code 2); Person 2 (Code 2); Person 4 (Code 5, Code 6); Person 6 (Code 5, Code 6); Person 9 (Code 2); Person 11 (Code 4); Person 12 (Code 5); Person 13 (Code 1).

5.3 Process

Interviewees described Penta's product configuration and development processes as fragmented, ad-hoc, and heavily reliant on tacit knowledge and manual documentation. Customers are often required to make configuration choices too early, inherited practices from other Volvo companies do not always fit Penta's context, and responsibilities for authorising new variants remain unclear. Testing and verification are primarily late-stage and field-based, with limited simulation capacity. The subthemes below will explain the process challenges in more detail.

5.3.1 Subtheme P1: Disagreement on configuration processes (CTO vs ETO)

Recognised problem. There are different perspectives differ on what process Penta follows, weather it is Configure-to-Order (CTO) or Engineering-to-Order (ETO) processes. ETO involves a close relationship with clients by a selling product that is highly customizable to customer needs. CTO involves creating product from a range of predefined options. **Interviewees suggested.** Some respondents recommend to Penta adopt a new official strategy to fully embrace CTO process in practice. This implies limiting the sales team to only suggest products from a limited configuration rule-set, so than the company can offer products that has been verified before suggested to customer. This can decrease the amount of maintenance needed after a product has been shipped to the customer. **Mentioned by:** Person 1 (Code 6); Person 7 (Code 1, Code 4); Person 8 (Code 6).

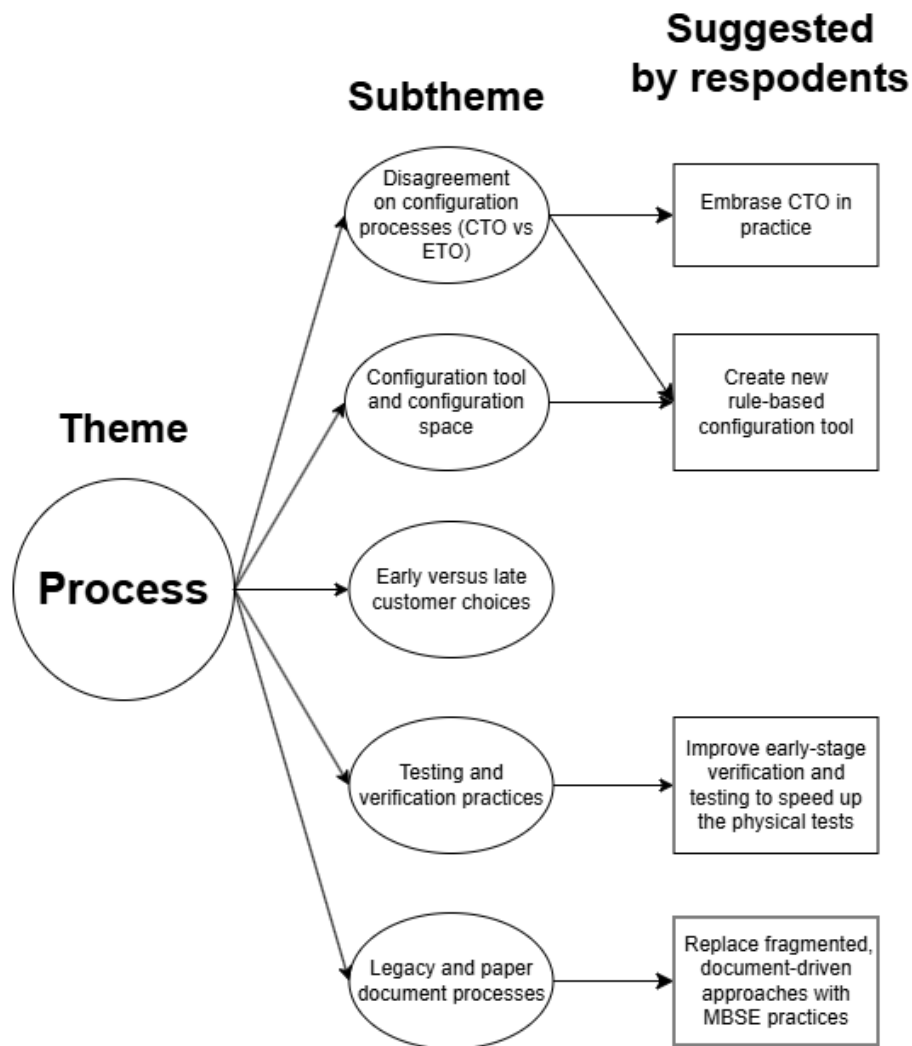


Figure 5.3: Process subthemes and respondent recommendations

5.3.2 Subtheme P2: Configuration tool and configuration space

Recognised problem. There is no unified process for how products can be configured. Product variants are created ad-hoc in response to new orders. There is a perceived resistance to adopt rule-based processes because they threaten established habits and informal knowledge. Customers sometimes want to delay choices, but the process requires early decisions. Invalid variants are sometimes proposed to close business deals, increasing late rework. **Interviewees suggested.** Develop a configuration tool containing a ruleset of all valid product combinations. Define the full configuration space of possible product variants to guide what can be offered. **Mentioned by** Person 1 (Code 4, Code 5); Person 2 (Code 1, Code 4, Code 6); Person 3 (Code 1); Person 6 (Code 4, Code 5); Person 7 (Code 2, Code 3); Person 13 (Code 4).

5.3.3 Subtheme P3: Early versus late customer choices

Recognised problem. Current processes force customers to make product choices very early. This causes discomfort for customers wanting to purchase in bulk and configure later. Existing flows are inherited from truck production and do not fit Penta's context. **Interviewees suggested.** There are no suggested solutions. **Mentioned by:** Person 2 (Code 1, Code 5).

5.3.4 Subtheme P4: Testing and verification practices

Recognised problem. Testing is heavily reliant on expensive field tests and late integration. Penta lacks the budget to extensively test with suppliers and instead imports tests with inherited components. Integration and validation are often delayed until late stages. **Interviewees suggested.** Increase simulation of component interactions to reduce dependence on physical test rigs. Improve early-stage verification and testing in the mechanical domain to reduce rework and service costs. Move toward continuous integration at the full product level as practiced at Group Truck Technologies. **Mentioned by:** Person 2 (Code 3); Person 3 (Code 2, Code 6, Code 7); Person 4 (Code 4); Person 9 (Code 3).

5.3.5 Subtheme P5: Legacy and paper document processes

Recognised problem. Penta remains reliant on Excel and paper document-oriented practices. There is no clear MBSE adoption strategy. **Interviewees suggested.** Replace fragmented, document-driven approaches with MBSE practices, to create shared and consistent artefacts across disciplines. **Mentioned by:** Person 7 (Code 5); Person 8 (Code 1, Code 2, Code 5).

5.4 Organisation

Interviewees described organisational challenges at Penta related to risk aversion, overlapping transformations, unclear responsibilities across disciplines, and confusion about process ownership. Respondents highlighted that changes are often seen as risky with uncertain rewards, and large-scale organisational fixes lack clear leadership. At the same time, several enablers for cross-disciplinary collaboration were identified—such as early working sessions, plain-language communication, collocation, and project-paired teams. Suggestions focused on moving variant authorisation responsibilities to Product Development, clarifying ownership across Mechanical, Electrical, and Software domains, and adopting enablers that directly address barriers in collaboration. The subthemes below will explain the organisational challenges in more detail.

5.4.1 Subtheme O1: Risk aversion and overlapping transformations

Recognised problem. Large-scale organisational changes are perceived as too risky with uncertain rewards. Penta's organisation is undergoing multiple simultaneous transformations (e.g., shift to systems rather than components, electrification, new workflows), requiring cultural change and alignment. **Interviewees suggested.** There are no suggested solutions. **Mentioned by:** Person 2 (Code 4); Person 3 (Code 8).

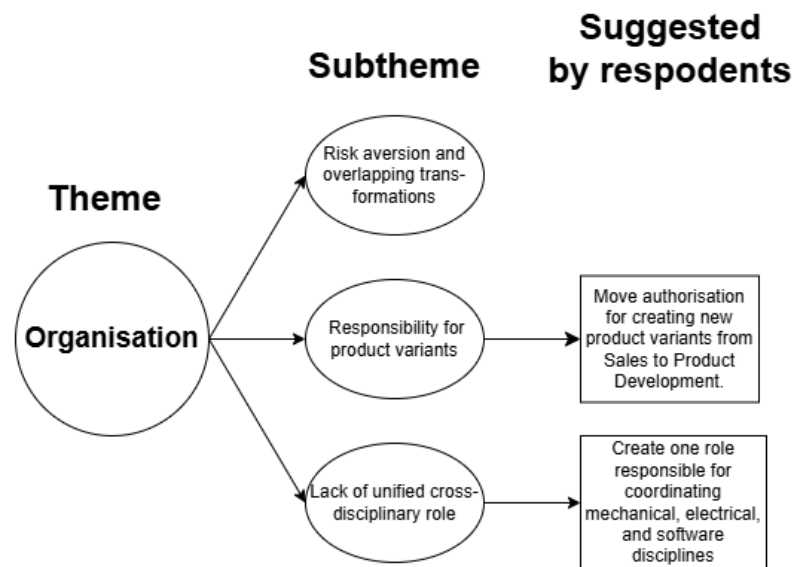


Figure 5.4: Organisation subthemes and respondent recommendations

5.4.2 Subtheme O2: Responsibility for product variants

Recognised problem. Current organisational setup allows Sales to create and push through new product variants, creating complexity. Responsibility boundaries between Sales and Product Development are blurred. **Interviewees suggested.** Move authorisation for creating new product variants from Sales to Product Development. **Mentioned by:** Person 6 (Code 4); Person 7 (Code 4).

5.4.3 Subtheme O3: Lack of unified cross-disciplinary role

Recognised problem. There is no single organisational role covering Mechanical, Electrical, and Software. Although some roles partially unite electrical and software, no individual has a centralised oversight or accountability over all three disciplines. This gap causes integration delays and coordination difficulties for cross-disciplinary teams. Major issues are resolved within projects, but no one is responsible for improving processes across the line organisation. **Interviewees suggested.** Penta would benefit from reordering internal organisation so that one person will have responsibility for all three of mechanical, electrical, and software disciplines. **Mentioned by:** Person 8 (Code 1, Code 2).

6

Discussion

This thesis aim to understand how mechanical, electrical and software engineers can better align workflows, tools and communication during the early-stage engineering design process. This thesis examined 15 interviews, and organised the results based on Business, Architecture. Process and Organisation themes of BAPO model [27].

6.1 Platform architecture

There is a desire at Penta to stay responsive to customers preferences and allow any modifications that satisfy customers needs. That only works if business options, in this case new product variants, are defined in a platform architecture, which is currently not the case at Penta. There is a platform architecture called "Common Architecture & Shared Technology" (abbreviated CAST), and both Volvo Trucks and Volvo Construction Equipment have their own versions of it. Penta has no CAST, which means business has no proper architecture that can manage product variability. As long as business is without a platform architecture, complexity costs will appear later as higher maintenance costs for all products that Penta sells.

A key difference between Volvo Trucks and Volvo Construction Equipment, on one hand, and Volvo Penta, on the other hand, is volume and mass-production capabilities. Volvo Trucks and Construction Equipment can design a product and then manufacture thousands of identical units. Volvo Penta, by contrast, builds one-offs or very small volumes of certain products. This small volume approach complicates efforts to introduce an architecture platform, because there are so much customer-specific work versus what could go into platform. This could explain why it has been difficult for Penta to adopt a platform based approach and why the Volvo Trucks and VCE platforms platforms do not translate to Penta: they are optimized for high volume products. For Penta, this mismatch can potentially be resolved by having an discussion about their business strategy before moving forward with a platform architecture. For future research, it would be interesting to dive deeper into platform architectures that fits small volume, unique customers order models, such as those at Penta.

6.2 Configuration tool

There have been suggestions fix the growing number of product variants by creating a configuration tool with codified rules. Those rules are meant to limit what kind of product variants can be suggested by Penta sales to prevent a situation where product variants with non-existing module combinations are proposed to customers. The main issues with product variants with non-existing module combinations is expensive engineering effort needed to verify the variant. A more cost-effective solution would be to strongly encourage sales to only use tested and existed module combinations. A such potential configuration is supported by rule-based validation methodology [19] (see section 2.7.1), specifically a formal configuration rule language to define valid configurations, and tool that verifies no configuration rule violation. The configuration tool can be created in way that can enforce 4 steps (proposed by van Gurp et al. [22]) for variability management, namely: identification of variability, constraining variability, implementing variability, and managing the variants.

This theoretical, but currently non-existing tool, can not replace a platform architecture, instead a configuration should be used as a complementary tool that will improve Penta's architecture and by doing that also become a part of Penta's processes. Penta currently relies on engineers know-how on what module combinations in product variants can be used. For Penta, this configuration tool could potentially automate this process by showing valid combinations for the tool users. If the configuration rules are made explicit in the tool, invalid combinations can be stopped in early stage instead of discovering them in late tests or in the field. At the same time, product variability management gives controls over variation points which is in line with Model-Based Systems Engineering [25]. For researchers, this configuration tool could be evaluated in a new pilot study to measure real effects of such tool in industry settings.

6.3 Product variant growth

Across interviews the same problem was repeated multiple times: product variants keep growing, while Penta's current tools and processes can not keep up. From a product-line perspective, this is under-constrained variability. The fix is better structure: declare variation points, constrain them, and manage families intentionally rather than cloning one-offs. BAPO can help explain that if business flexibility is pushed without adding architecture and process safeguards, then initial product cost will move downstream and increase integration, testing, and maintenance costs instead.

Penta currently relies on tacit know-how and ad-hoc flow around new variants. A more deliberate process would make the product variation points visible, debugging errors possible, and the responsibilities unambiguous. BAPO's process questions are a good checklist: what can be predicted early, where is there repetition, and how do we measure progress. A set-based rhythm helps here—start wider, use criteria to narrow, and commit only when we have enough information. If we make MBSE

artefacts (models, interface checks, simulation results) part of the gate deliverables, verification moves earlier without losing control. Interviewees ask for exactly that: clearer gates, earlier evidence, fewer exceptions. The goal is not bureaucracy; it is to make the cheapest checks happen first and to push only feasible variants forward.

6.4 Late testing

Respondents describe a reliance on late field tests and rigs. That is expensive and slow, and it tends to hide integration issues until the last moment. The alternative, that was identified during interviews, was earlier simulation and more system level integration testing. This is consistent with the MBSE idea of moving verification and validation earlier by use shared system models, making interfaces explicit, and attaching checks to models instead of to the final build only. Combining MBSE with SBCE can help exploring a many credible product options, test them quickly in models, and save weak solutions, as options for later use. A problem that was consistent throughout the interviews was that mechanical, electrical, and software engineers need a shared picture of the system earlier, and that picture needs to carry the right details for each discipline. Systems Engineering and MBSE provide practical solutions such as modelling a product structure, the interfaces, and the behaviour centrally, which should make the transfer from one discipline to another explicit and check the changes as the design evolves.

6.5 Organisation

Finally, none of this holds without organisational support. Challenges such as unclear ownership over Mechanical, Electrical and Software disciplines and resistance to change are socio-technical. It was suggested to create a new unified role for a person to coordinate Mechanical, Electrical and Software disciplines. To set up new platform architecture, and complement it with configuration rules, there need to be new roles and routines that back them up. That means clarifying who owns which interface, who approves variation at which level, and how line organisations maintain and improve the process over time.

6.6 Contributions

This thesis was designed to explain why configuration and variant complexity accumulates and how it propagates across Business, Architecture, Process, and Organisation dimensions. To support this aim, 15 semi-structured interviews were used to gather data about Volvo Penta's current situation.

New dataset A qualitative dataset was created from sixteen anonymized interviews across multiple Volvo companies, spanning business, architecture, process, and organizational dimensions. The scope was designed to capture pain points of Penta. This dataset constitutes a reusable baseline for future evaluation and interventions

which could enable future work whether the same challenge persist or whether future interventions have altered them.

Use case of BAPO model A thematic coding approach was shown where interviews were analysed based to Business, Architecture, Process, or Organisation. By structuring the analysis with BAPO, cause-and-effect chains could be traced, for example, optimistic sales promises lack of architecture, late verification and ownership ambiguity. And as the result, this thesis recommended Penta to have a discussion about business strategy before moving forward with a platform architecture. Then create a new configuration tool to be used to together with new platform architecture, while that tool would support MBSE and SBCE practices and change Penta's organisation to support the new tool.

Participants and coverage. 15 interviews were conducted across roles such as architects, project managers, systems engineers, technology leads and R&D managers. Participants were selected from different cross disciplinary backgrounds within mechanical, electrical and software disciplines, across different Volvo units to gain a variety of expert opinions. In each interview session, questions prompts were tailored to the respondent's expertise to get the most useful information for this thesis.

Interview protocol and accommodations. An interview guide was prepared and then adapted depending on different interview situations. When interviewees represented other Volvo companies, questions were changed to transfer their practices and unique conditions to Volvo Penta. In this way, examples grounded in the other Volvo companies were encouraged to gain a different and relevant point of view to that of Penta.

Data handling and coding. All interview transcripts were anonymized and then coded using a six-stage thematic process: familiarization, initial coding, theme generation, review, definition, and reporting. Coding was constrained to statements relevant to the research questions and organizational context, while proprietary details were excluded by design. Codes were subsequently aggregated into themes anchored to the BAPO dimensions so that each finding was located within one of the four dimensions.

6.7 Limitations and threats to validity

Internal validity This thesis relied on observations of Volvo Penta's workflows, more specifically opinions from interviewed employees. This introduced a potential for confirmation bias among the respondents to view the situation only through Penta's perspective. To address this threat, candidates were selected among 5 different Volvo companies (including Volvo Penta) to get viewpoints of companies with similar issues and presumably existing solutions. When conducting comparisons with other organizations' practices, there is a risk that external actors could hold back vital information due to competitiveness and integrity considerations, thus limiting the applicability of that knowledge. To address this threat, interview partici-

pants from within Volvo Group companies were chosen. This has encouraged trust and openness in sharing important details regarding questions that were asked during interviews. To mitigate role bias, interview candidates from different roles and organisations were selected. The roles included architects, systems engineers, project manager and leaders from five organisations. This role diversification reduced bias of one role influencing the results, and made the discussion more nuanced.

External validity This thesis aimed to understand root cause of the challenges explored and identify explanations that can be generalised to larger number of cases. Given the nature of the interview process, the conclusions that were generated can be generalised to other organisations within Volvo Group or outside actors that specialize in embedded systems domain not excluding original equipment suppliers within that domain. A sample of companies with expertise in both software and hardware was used; although some organisations such as Volvo Penta have a larger focus towards internal hardware development than software one. This can arguably be a positive aspect for Penta to learn practices from similar companies with more internal software development. Thus, this report contains a more detailed and nuanced picture than if cases with less software development were included. The selected companies are of diverse size; with 2 out of 5 companies reported having between 150 and 250 employees [42, 43]; while the rest have at least 1000 employed in Sweden alone [43, 44, 45]. Only medium or large sized companies can maintain organisations large enough that cross-disciplinary collaboration would become too complex and too expensive to be dealt with ad-hoc. Smaller companies, on the other hand, might not face these issues with the same severity, but some could face the challenges on a smaller scale.

Delimitations This thesis has been set up to contain the scope within a time-frame of a Master Thesis. The study itself was limited to include Gothenburg offices of participating Volvo companies, without any supplier, customer or any organisation outside of Volvo Group. It should be noted that CPAC Systems is part of Volvo Group, although not containing the brand of Volvo in its name. The data collection was restricted to semi-structured interviews, without any quantitative research methods, such as surveys as those could yield the desired goal of understanding, not quantifying, the challenges at hand. The roles of interviewee were chosen to reflect their knowledge an involved in the company.

During planning of this thesis, there was an idea of interviewing people with different levels of experience and at different types of hierarchies, more specifically interviewing junior engineers. This idea was initially dropped as it became clear that lack of experience and lack of influence on other's work would of juniors would not those employees relevant. This claim is supported by the likes of Klug and Bagrow who suggest that higher levels of experience and influence are factors to consider in group dynamics to ensure that people can cooperate in the best way possible [46].

6.8 Future work

To advance knowledge creation, there are three tracks that can build on this study and turn the themes into something that be can tested, compared, and improved over time.

Pilot the configuration tool in a new project. The immediate next step is a constrained pilot where researcher can encode a minimal ruleset (option constraints + interface checks) and bind it to the system model. Scope it to a few product with real variation (e.g., a handful of hardware–software options). The goal is to prove that “rule before build” reduces invalid configurations reaching rigs and cuts the time spent on hand-offs. Define a clear owning role for rules, a change path, and an approval step, so organisation can be prepared.

Bind rules to MBSE artefacts so violations show up where they are created. In a future project with configuration, every rule should reference a concrete model element (interface, allocation, requirement) so that a broken rule points to a location in the model, not an abstract checklist. Designers will be able to see violations as they work, and system leads review evidence without spending time debugging integration issues and rule violations manually.

Reuse platform and interfaces across different organisations. Several interviewees mentioned cross-unit reuse as both an opportunity and a source of problems. In future a small cross-unit working session can suggest “do not fork” interfaces and define their variation points explicitly. This could test if there will be faster integration if options change.

7

Conclusion

This thesis aims to understand how mechanical, electrical and software engineers can better align workflows, tools and communication during the early-stage engineering design process and how it propagates across Business, Architecture, Process, and Organisation (BAPO) layers within Volvo Penta. A qualitative design based on sixteen semi-structured interviews was employed, and findings were synthesized through a BAPO-lense.

Across the Business layer, it was found that new product variants were frequently accepted before early technical validation, which expanded a potential configuration space in ways that were not explicitly bounded. Exceptional product cases were allowed to be added to enable short-term deal closure and are currently a norm. Without visible safeguards, expectations in product variants set in ways that later collided with technical feasibility. Penta lacks a platform architecture, while variation points, dependencies, and constraints were not represented in a uniform manner. Traceability from customer-facing features to hardware/software variants was incomplete, which introduced friction in reuse and integration. On the Process side, requirements and configuration rules were scattered across tools and repositories, verification was concentrated late, and feedback loops back to product definition were slow and manual. At the Organisation level, decision rights for introducing or modifying variants were fragmented, and overlapping transformations diluted accountability for configuration quality. Taken together, these findings depicted not a single root cause but a set of reinforcing loops through which local optimizations produced global instability.

Anchored in these findings, a configuration-tool concept was specified for the local context. The proposed tool is presented as a complement to establishing a new architecture at Penta. The concept emphasizes (i) transparency, with rules and constraints recorded with change history, (ii) guidance, with early detection and prevention of invalid combinations, and (iii) governance with role-appropriate workflows that bind product decisions to architectural variability and release processes.

Bibliography

- [1] S. Friedenthal, A. Moore, and R. Steiner, “Chapter 1 - systems engineering overview,” in *A Practical Guide to SysML (Third Edition)* (S. Friedenthal, A. Moore, and R. Steiner, eds.), The MK/OMG Press, pp. 3–14, Boston: Morgan Kaufmann, third edition ed., 2015.
- [2] A. Vogelsang, T. Amorim, F. Pudlitz, P. Gersing, and J. Philipps, “Should I stay or should I go? on forces that drive and prevent MBSE adoption in the embedded systems industry,” *CoRR*, vol. abs/1709.00266, 2017.
- [3] M. Malewski, D. M. J. Cowell, and S. F. and, “Review of battery powered embedded systems design for mission-critical low-power applications,” *International Journal of Electronics*, vol. 105, no. 6, pp. 893–909, 2017.
- [4] J. Bosch, *Speed, Data, and Ecosystems*. CRC Press, 2017.
- [5] L. Bretz, L. Kaiser, and R. Dumitrescu, “An analysis of barriers for the introduction of systems engineering,” in *Procedia CIRP Volume 84*, p. 783–789, 2019.
- [6] R. Ammar, M. Hammadi, J.-Y. Choley, M. Barkallah, and J. Louati, “Integrating the sbce principles with mbse methodology for the mechatronic system’s design,” 2019. Cited by: 1.
- [7] “Ab volvo penta - årsredovisning för räkenskapsåret 2023.” Accessed: 2025-02-21.
- [8] D. Fielder, C. C. Gattaz, U. J. Tanik, M. Tanik, and F. Sobrinho, “Transdisciplinary convergence - a vital consideration in engineering solutions,” vol. 2016-July, 2016. Cited by: 2.
- [9] S. Heath, “1 - what is an embedded system?,” in *Embedded Systems Design (Second Edition)* (S. Heath, ed.), pp. 1–14, Oxford: Newnes, second edition ed., 2002.
- [10] iVT International, “Volvo penta showcases connected platform to enhance off-highway vehicle productivity,” April 2025. Accessed: 2025-04-21.

- [11] V. Penta, “From power to superpower: Volvo penta introduces all-new ips professional platform,” May 2023. Accessed: 2025-04-21.
- [12] P. Newswire, “Volvo group and daimler truck intend to form a joint venture for a software-defined vehicle platform to amplify digital transformation,” May 2024. Accessed: 2025-04-21.
- [13] Workboat365, “Volvo penta unveils hybrid-electric package for commercial marine vessels,” August 2024. Accessed: 2025-04-21.
- [14] P. International, “Volvo penta at bauma 2025: next steps in electrification,” February 2025. Accessed: 2025-04-21.
- [15] S. E. B. of Knowledge (SEBoK), “Guide to the systems engineering body of knowledge,” June 2015.
- [16] C. Zheng, J. Le Duigou, M. Bricogne, and B. Eynard, “Multidisciplinary interface model for design of mechatronic systems,” *Computers in Industry*, vol. 76, pp. 24–37, 2016.
- [17] O. M. Group, “Model lifecycle management for mbse,” n.d. Accessed: 2025-04-21.
- [18] D. K. I. Sobek, A. C. Ward, and J. K. Liker, “Toyota’s principles of set-based concurrent engineering,” *Sloan Management Review*, vol. 40, no. 2, pp. 67–83, 1999. Winter.
- [19] L. Li, P. Wagner, R. Ramaswamy, A. Mayer, T. Wild, and A. Herkersdorf, “A rule-based methodology for hardware configuration validation in embedded systems,” in *Proceedings of the 19th International Workshop on Software and Compilers for Embedded Systems*, SCOPES ’16, (New York, NY, USA), p. 180–189, Association for Computing Machinery, 2016.
- [20] V. Diekert and P. Gastin, “First-order definable languages,” in *Logic and Automata: History and Perspectives* (J. Flum, E. Grädel, and T. Wilke, eds.), vol. 2 of *Texts in Logic and Games*, pp. 261–306, Amsterdam University Press, 2008.
- [21] J. P. E. Hodgson, “First order logic,” 1995. Archived from the original at Saint Joseph’s University.
- [22] J. van Gorp, J. Bosch, and M. Svahnberg, “On the notion of variability in software product lines,” pp. 45 – 54, 02 2001.
- [23] D. C. Rine and N. Nada, “An empirical study of a software reuse reference model,” *Information and Software Technology*, vol. 42, no. 1, pp. 47–65, 2000.
- [24] M. C. Şahin and A. K. Tarhan, “A multi-perspective review on embedded systems quality: State of the field, challenges, and research directions,” *Journal*

of Software: Evolution and Process, vol. 37, no. 3, p. e70007, 2025.

- [25] A. Akundi and V. Lopez, “A review on application of model based systems engineering to manufacturing and production engineering systems,” *Procedia Computer Science*, vol. 185, pp. 101–108, 2021. Big Data, IoT, and AI for a Smarter Future.
- [26] M. Borchani, M. Hammadi, N. Yahia, and J.-Y. Choley, “Integrating model-based system engineering with set-based concurrent engineering principles for reliability and manufacturability analysis of mechatronic products,” *Concurrent Engineering*, vol. 27, p. 1063293X1881674, 12 2018.
- [27] F. van der Linden, J. Bosch, E. Kamsties, K. Känsälä, and H. Obbink, “Software product family evaluation,” in *Software Product Lines* (R. L. Nord, ed.), pp. 110–129, Springer Berlin Heidelberg, 2004.
- [28] P. Runeson and M. Höst, “Guidelines for conducting and reporting case study research in software engineering,” *Empirical Software Engineering*, vol. 14, no. 2, pp. 131–164, 2009.
- [29] D. Strode, T. Dingsøy, and Y. Lindsjörn, “A teamwork effectiveness model for agile software development,” *Empirical Software Engineering*, vol. 27, no. 56, pp. 1–50, 2022.
- [30] L. Busetto, W. Wick, and C. Gumbinger, “How to use and assess qualitative research methods,” *Neurological Research and Practice*, vol. 2, no. 1, p. 14, 2020.
- [31] J. Corbin and A. Strauss, *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. Thousand Oaks, CA: SAGE Publications, Inc., 3rd ed., 2012.
- [32] G. Fisher and H. Aguinis, “Using theory elaboration to make theoretical advancements,” *Organizational Research Methods*, vol. 20, no. 3, pp. 438–464, 2017.
- [33] M. Hennink and B. N. Kaiser, “Sample sizes for saturation in qualitative research: A systematic review of empirical tests,” *Social Science Medicine*, vol. 292, p. 114523, 2022.
- [34] M. Naeem, W. Ozuem, K. Howell, and S. Ranfagni, “Demystification and actualisation of data saturation in qualitative research through thematic analysis,” *International Journal of Qualitative Methods*, vol. 23, p. 16094069241229777, 2024.
- [35] V. Braun and V. Clarke, “Using thematic analysis in psychology,” *Qualitative Research in Psychology*, vol. 3, no. 2, pp. 77–101, 2006.

- [36] V. Braun and V. C. and, “One size fits all? what counts as quality practice in (reflexive) thematic analysis?,” *Qualitative Research in Psychology*, vol. 18, no. 3, pp. 328–352, 2021.
- [37] R. Bhaskar, *A Realist Theory of Science*. New York: Routledge, 1975.
- [38] S. McLeod, “Open coding in qualitative research,” 2024.
- [39] S. McLeod, “Axial coding in qualitative research,” 2024.
- [40] A. Hevner, A. R. S. March, S. T. Park, J. Park, Ram, and Sudha, “Design science in information systems research,” *Management Information Systems Quarterly*, vol. 28, pp. 75–, 03 2004.
- [41] E. Knauss, “Constructive master’s thesis work in industry: Guidelines for applying design science research,” in *2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, 2021.
- [42] Volvo Autonomous Solutions AB, “Volvo Autonomous Solutions AB (559285-4219) – Bolagsfakta.” https://www.bolagsfakta.se/5592854219-Volvo_Autonomous_Solutions_AB, 2025. accessed: 2025-09-16.
- [43] Aktiebolaget Volvo Penta, “Aktiebolaget Volvo Penta (556034-1330) – Bolagsfakta.” https://www.bolagsfakta.se/5560341330-Aktiebolaget_Volvo_Penta, 2025. accessed: 2025-09-16.
- [44] Volvo Construction Equipment AB, “Volvo Construction Equipment AB (556021-9338) – Bolagsfakta.” https://www.bolagsfakta.se/5560219338-Volvo_Construction_Equipment_AB, 2025. accessed: 2025-09-16.
- [45] Volvo Lastvagnar Aktiebolag, “Volvo Lastvagnar Aktiebolag (556013-9700) – Bolagsfakta.” https://www.bolagsfakta.se/5560139700-Volvo_Lastvagnar_Aktiebolag, 2025. accessed: 2025-09-16.
- [46] M. Klug and J. P. Bagrow, “Understanding the group dynamics and success of teams,” *Royal Society Open Science*, vol. 3, no. 4, p. 160007, 2016.

A

Appendix

A.1 Person 1 - Volvo Penta

- **Code 1: Component reuse within Volvo Group.** As part of their business model, there is internal collaboration where Penta chooses and modifies components from other Volvo Group companies to meet Penta's needs. This aims to maximise total profitability by sharing development costs among all projects.
- **Code 2: Increasing product complexity.** The complexity has increased exponentially faster than the supporting information models and tools.
- **Code 3: Shifting customer expectations.** There is a market shift toward sales-as-a-service. Penta can not expect external parties to be experts on Penta's product, so there is a demand to simplify product interfaces and automate new product creation with "codified rules".
- **Code 4: Defining configuration space.** There are no established rules or standards on how a new product can be configured. The current way-of-working is experimental progression where a product variant is created ad-hoc, based on new customer orders. This has led to inconsistencies in how product offerings are understood and documented internally in Penta.
- **Code 5: Internal resistance to structured configuration tools.** There is cultural resistance to deploy new rule-based systems. This is partly due to old habits but also subjective trust in informal knowledge built by years of experience.
- **Code 6: Different opinions on product strategy.** There are different perspectives on how the current processes should work. The opinions differ on whether the company should design products from pre-defined options (Configure-to-order), or design products based on unique customer requirement specifications (Engineering-to-order).

A.2 Person 2 - Volvo Penta

- **Code 1: Early vs. late configuration.** The respondent explains that Volvo Penta's current processes needs the customer to make product choices very early. Some customers want to buy Penta's products in bulk and on the contrary finalise the configuration later.
- **Code 2: Increasing complexity.** Each addition of hardware and software combinations increases the number of potential product variants. The current tools can not to handle this growing complexity.
- **Code 3: Limited testing capacity.** There is no working system for early validation or simulation of all possible variants, before the product is assembled. This forces the company to use test rigs and perform "tests in the field" to integrate the components late in the development process. This practice is expensive and time ineffective. A early stage detection is preferred by the interviewee.
- **Code 4: Organisational caution towards large changes.** Large-scale fixes are perceived as being to risky with uncertain rewards. Without a compelling case major process or IT infrastructure may be de-prioritised.
- **Code 5: Inherited processes unsuitable for Penta.** The configuration flow is inherited from other Volvo companies and is optimised for truck production. The respondent notes that this process may not fully align with Penta's unique operations and product offering.
- **Code 6: Reliance on experts & informal knowledge.** Correctly configuring a product requires deep tacit knowledge held by very few people, while the current tools can not prevent technically invalid configurations.
- **Code 7: Cross-disciplinary communication barriers.** The respondent observes that different terminology and technical jargon creates communication barriers between engineers and decision-makers. New ideas need to be communicated in a language that is understood by all parties to create real impact and a sense of urgency.

A.3 Person 3 - Volvo Penta

- **Code 1: Alignment between business and technical teams.** The respondent has mentioned challenges in communication between technology and business teams. Misunderstandings and expensive fixes can arise when product variants are proposed by the sales organisation, without thorough technical validation, in order to accelerate business deals.
- **Code 2: Evolving customer needs during development.** New customer requirements and commercial opportunities can emerge mid-project. These

changes can alter the project scope and require additions that were not accounted in initial planning, which increases the workload for engineers.

- **Code 3: Strategic move towards in-house software development.** Penta is increasingly taking more responsibility for development of software, to meet their specific needs. This is a long-term shift from the current ways-of-working where Penta inherits, modifies and integrates complete software from other Volvo companies. This strategic choice is due to rising software complexity and need for tailored solutions at Penta.
- **Code 4: Integration challenges across engineering teams.** Mechanical, electrical, and software parts are developed at different timelines. While mechanical components are long-lived and stable, electronics and software evolve quickly. This leads to integration challenges at later development stages.
- **Code 5: System-level requirement and alignment challenges.** System-level requirements are sometimes unclear, which leads to confusion about what is needed versus what is available. This places pressure on system engineers to add new system requirements ad-hoc.
- **Code 6: Enhancing early testing and simulation practices.** There is a demand to improve early-stage verification and testing in the mechanical domain. This could speed up validation and decrease reliance on physical tests of prototypes in the later stages.
- **Code 7: Integration bottlenecks and late testing.** Integration and testing effort are sometimes delayed until later stages on the final product. Setting up early validation could improve the product quality and reduce the service costs of urgent repairs on products in use.
- **Code 8: Organisation scaling and change management.** The organisation is undergoing multiple and overlapping transformations. These include shifting from component sales to system offerings, electrification, and integrating new workflows. All of these require internal cultural change and alignment.

A.4 Person 4 - Volvo Penta

- **Code 1: Limited in-house development.** Penta chooses to outsource software development to reduce of the cost of internal development. Efforts are focused on reusing software within Volvo Group, this includes connecting existing software subcomponents and adding minimal modifications, rather than coding from scratch.
- **Code 2: Strategic reuse vs. customization trade-offs.** While software and hardware reuse is encouraged to reduce cost and maximize return on investment, Pentas's unique applications require additional fixes that increase complexity and cost. The company frequently faces decisions on when it's cost-

efficient to adapt shared components vs build new components from scratch.

- **Code 3: Long-term limits of reuse economics.** The respondent believes that there will be a breaking point when reuse is no longer cost-effective. Penta sells many different systems, some of which are produced in low volumes, which means that fully internally-made components are too expensive per unit. Yet outsourcing still requires costly integration, maintenance and testing efforts.
- **Code 4: Shared testing capabilities.** Other Volvo companies have larger budgets and a capacity to extensively test components with their suppliers. This helps Penta to keep a tighter budget by importing tests when the components are inherited from other organisations.
- **Code 5: Growth of product variants due to customer demand.** A strong customer centric sales strategy at Penta has led to prioritization of customer demands, at the expense of following a platform architecture. This has contributed to growing number of product variants and more product complexity.
- **Code 6: Changing complexity and compatibility between different components.** Electric vehicles are more complex than traditional diesel or gasoline systems. In the past it was often possible to mix and match components freely, but this no longer works for electrical vehicles. EV systems have unique requirements, so combining their parts without careful planning can lead to new problems. This is challenge for sales who have not been keeping up-to-date with new technical advancements and complexities of EV's and confuse EVs with simpler logic of diesel systems.

A.5 Person 5 - Volvo Penta

- **Code 1: Importance of sync meetings for early-phase alignment.** Early project meetings are important to align teams and resolve questions in requirements, for example hardware and signal handling.
- **Code 2: Efficient sync meeting strategies.** Meetings are more effective when treated as working sessions with small focused teams. The respondent believes in effectiveness of workshops, where the discussions are separated by engineering function.
- **Code 3: Face-to-face interaction vs. written communication.** The respondent believes in the value of in-person collaboration, especially in the early phases. In-person sessions help align team members and establish shared understanding through active participation, while also providing a boost to the team morale.

A.6 Person 6 - Penta

- **Code 1: Late adoption of systems engineering.** System Engineering is a relatively new and immature process at Penta. The respondent mentions that after system-level requirements are created, the software is integrated with hardware after-the-fact, instead of having software and hardware developed currently.
- **Code 2: More testing & simulation is needed.** To better test software and hardware integration, more simulation of components interaction is needed. The current ways-of-working with physical test rigs can not handle all possible combinations of software and hardware components.
- **Code 3: Late stage hardware-software shared development.** The respondent suggests that development should be shared among software and hardware teams from the get go. To minimise costs and increase development speed, specific details about implementation should be delayed until the system architecture is defined and properly understood.
- **Code 4: Authorisation and responsibility restructuring.** The authorisation for creating new products variants (configurations) should be moved from Sales to Product Development. Sales can suggest a new configuration, but they will need to send their proposal to the Product Development who will test, validate & authorize the new configuration so that it can be added to the configuration space for Sales to use.
- **Code 5: Decrease complexity through configuration tool.** By creating a tool that contains coded rules for all technically valid product combination, known as the configuration space, and reduce complexity.
- **Code 6: Customer driven complexity.** Customers often combine Penta's products with other supplier components, in marine and industrial machines. This creates an external complexity, outside of Penta's control, which creates a higher demand to decrease complexity at Penta.

A.7 Person 7 - Penta

- **Code 1: Expectation and reality of manufacturing processes.** There is a perception that Penta operates in Configuration-to-Order, although the company does not have mature tools for that. Instead the company operates as Engineering-to-order where each new product order is catered to the customer demands.
- **Code 2: Lack of configuration options.** A configuration/combination set of all possible product variants is needed so that people can better understand what products Penta can offer. There needs to be a governing model for how a product is defined across Penta.

- **Code 3: Constraining configuration space.** The respondent believes that a digital tool can contain the ruleset of all combinations. Each combination in the ruleset needs to be verified and given a technical authorization by engineering. Then a combination can be sent to Sales team that can offer that as product that meets customer needs.
- **Code 4: Adopting configure-to-order in practice.** The company needs to adopt configure-to-order process in practice, to decrease product complexity. If the Sales team can only suggest products from a limited configuration rule-set, than the company can offer products that has been verified before suggested to customer. This can decrease the amount of maintenance needed after a product has been shipped to the customer.
- **Code 5: Document and process fragmentation.** There is a difficulty to manage the product complexity due to fragmented documentation and manual way-of-working with product variants, such as having variants in local Excel files. The respondent believes that current way-of-working is outdated and should be replaced with Model-based System Engineering.
- **Code 6: Lack of shared architecture across Volvo Group.** Volvo Group has no unified architecture framework, although most Volvo companies has their own architecture framework. Penta expressed a desire for an unified architecture across all Volvo companies.

A.8 Person 8 - Penta

- **Code 1: Organizational silos.** There are collaboration challenges caused by Mechanical, Electrical, and Software teams working in silos. Although some roles partially unite electrical and software, no individual has a centralised oversight or accountability over all three disciplines. This organisational fragmentation leads to integration difficulties and delays, which necessitates creation of a new unified role.
- **Code 2: Project-based internal communication.** The cross-disciplinary communication is handled inside the project groups. When large issues are identified, new project groups are created. Those groups are not responsible for improving internal processes connected to the issues, instead the process are supposed handled by line organisations which creates confusion.
- **Code 3: Project vs product development confusion.** Some concept products are marketed as complete solutions, without being extensively verified in early stages. In some cases, the software may not perform as intended, and these projects are sold as application projects. At that point, Penta sends people to fix the software so that it works locally for the customer. When an update comes out, it overwrites what Penta had previously fixed at the customer.

- **Code 4: Outsourcing of software development.** Penta has historically had no internal software development, instead their product development imported all software from other Volvo companies and Penta's subsidiary. In the last 5 years, Penta has slowly started development more own software, although most software is still imported from elsewhere.
- **Code 5: Insufficient adoption of Model-Based Systems Engineering.** Penta still remains dependent on outdated document-oriented practices and lacks a clear strategy for adopting modern Model Based Systems Engineering practices for everyone involved in development.
- **Code 6: Misalignment between official strategy and practice.** There appears to be an opinion difference in how Penta processes are perceived. In practice, the organisation is doing Engineering-to-Order, which involves a close relationship with clients and a product that is highly customizable to their needs. However, there is belief that the company is doing Configure-to-Order, which involves creating product from a range of predefined options. The respondent highlights that Penta currently does not have the methods, processes and tools to support Configure-to-Order.

A.9 Person 9 - Group Trucks Technology

- **Code 1: System complexity management.** The interviewee mentions that a modern truck is a system-of-systems made up of many electronic control units (ECU). Keeping all those ECUs working together, while also measuring their inter-dependencies, has become a primary engineering challenge in the last decade.
- **Code 2: Product complexity control.** A strong customer centric sales strategy that prioritises selling the exact product the customer wants, creates exponentially more hardware-and-software combinations. The company controls complexity by limiting the options their customer can choose from when configuring their product.
- **Code 3: Insufficient continuous integration & complete-product testing.** The respondent firmly believes that true quality, comes only by integrating the complete product as often as possible. The current component-only continuous integration is insufficient and will become a challenge in the long-term.
- **Code 4: Iterative evolution over big-bang rewrites.** Previous attempts to replace legacy code with a brand-new platform have not succeed because those attempts discarded the internal know-how during the transition process. The respondent points out that sustainable change must come through small, understandable steps that always leave the whole product working during all stages of change.

- **Code 5: Hardware centric mindset needs to change.** The interviewee suggests that some business leaders with hardware-centred expertise may not fully recognise the unique long-term risks that are caused by rising software complexity. This mindset slows down the necessary change to tackle software complexity effectively.
- **Code 6: The need for more in-house software development.** The respondent believes that software & hardware sharing within Volvo Group is profitable while the complexity is low. If the complexity continues to rise, each product line will need to develop its own system from scratch.

A.10 Person 10 - Group Trucks Technology

- **Code 1: Strengthen business-engineering communication.** The respondent points out that improved knowledge sharing and clearer communication of technical jargon can offer opportunities to strengthen communication between business and engineering units.
- **Code 2: Organisation scaling plans.** The respondent's organisation has grown from 70 to 800 people. The processes and structures evolve to become more mature so the larger organisation can work efficiently at this size.
- **Code 3: Truck's software development.** Group Trucks Technology makes most of their software in-house. And the software can be shared to other organisations such as Volvo Penta.

Code 4: Shared processes & component reuse across Volvo Group. Using common tools, engines and production processes across Volvo companies is described as a collective strength by avoiding each organisation "developing its own products" and thus leveraging advantages of being part of a larger group of companies.

- **Code 5: Limitations of current tools.** The home-made PDM tool is currently used among all Volvo companies and is well perceived among engineers. As the company aims to become more software-driven, this tool does not provide not enough support for system-of-systems workflows.

Code 6: Business influence on technical decisions. The respondent mentions that business units may push for specific technologies, such as new battery cells that can clash with system constraints. (voltage, packaging, cooling) This creates a need to negotiate to compare business value and engineering feasibility side by side.

A.11 Person 11 - Volvo Construction Equipment

- **Code 1: Platform modernisation.** The respondent explained that in the last 15 years, his company created an abstract software platform where software modules can nowadays be auto-generated based on feature flags. The decision to create a shared software platform was a previous agreement to create an electronics architecture so creating a shared architecture for both electronics and software felt "natural".
- **Code 2: Physical co-location for efficiency.** There are both formal and informal meetings between software, electronics and mechanical teams. The more serious problems are solved during formal meetings, while small changes are handled informally. The software, electronics and mechanical teams seat together on the same floor in the open-space office, so they can meet each other in person and solve small issues faster.
- **Code 3: Management-Engineering gap.** The management usually has a background in mechanics and electronics but not in software. So they do not possess an intuitive understanding for growing software complexity, which poses a challenge when trying to explain the severity of software complexity.
- **Code 4: Software complexity growth.** With time the codebase doubled in size so the developers can't keep all the code in their head any more. Instead they use tools for drawing models to understand how one change in code, can create unexpected effects down the line.
- **Code 5: Geographical location and development speed.** The software development is split between the Swedish and the international software teams. The respondent expressed an opinion that having more software developed in Sweden makes the development go faster and easier.
- **Code 6: Need for shared platform across all Volvo brands.** The respondent expressed a wish a shared electronics and software platform across all Volvo brands.

A.12 Person 12 - Volvo Construction Equipment

- **Code 1: Alignment starts in requirement.** The respondent believes that to make development more cost-effective, the requirements should be as general as possible, without specifying any of the mechanical, electrical and software discipline at the start. Then after evaluation, the details will be assigned to the most cost-effective option, whether its mechanical, electrical or software or a combination of them.
- **Code 2: Cross-disciplinary requirements tool.** The requirements are not documented in one place, rather with different tools, including Excel. To solve this, a new user-friendly tool for cross-disciplinary requirements docu-

mentation is being slowly introduced.

- **Code 3: Information sharing.** The respondent expressed a desire for having hardware, software and manufacturing information shared and accessible to everyone.
- **Code 4: Modular CAST Architecture.** Common Architecture Shared Technology is a reusable, cross-product electronics and software platform. It's used used to share design unit variants to form modules and modules are combined to form machines.
- **Code 5: Increasing Software Complexity.** More ECUs, wiring and controllers means more design options and the respondent expressed the need for integrated methods.

A.13 Person 13 - Volvo Construction Equipment

- **Code 1: Solving product complexity using architecture framework.** The software complexity has increased because of digitalisation, electrification, automation and increasing service offering. To solve this, Volvo Construction Equipment started using Common Architecture Shared Technology (CAST).
- **Code 2: Component sharing among Volvo companies.** There is an ongoing dialogue between Volvo Construction Equipment and Volvo Penta about adopting the same framework. Once they introduce a common framework, the two organisations easier can reuse each other's modules.
- **Code 3: Integrated hardware and software modules.** There is a long-term ambition to create complete modules that contain both hardware, software and all ECU connections. This will reduce re-integration work as today hardware and software are still joined together late product development workflow.
- **Code 4: Configurator tool limits product variation points.** Volvo Construction Equipment's sales configurator limits the customer to a few dozen key choices. The tool auto-selects the mandatory market- and legislation-enforced options, and also connects each choice to the correct hardware and software variants for each unique build.

Code 5: Variability and complexity. There is a trade-off between design variability and time (when each solution enters or leaves production across multiple plants), and it creates an "explosion" of product variants, all of which must be maintained for at least 15 years.

Code 6: Limited reuse from other Volvo companies. Only about a 20 percent of all components from other Volvo companies are currently reused by Volvo Construction Equipment. It's mostly engines that still require specific

makeover for construction machines and unique documentation rules.

A.14 Person 14 - Volvo Autonomous Solutions

- **Code 1: Business model and transport-as-a-service.** Their business model revolves around providing transport-as-a service. The company rebuilds trucks into autonomous vehicles, removing the driver's cabin, and offers these modified trucks as a service to their customers. This shift towards a service means that hardware-software integration is more important than ever, because the services needs continuous operational updates to fix errors and maintain high-quality service to their customers.
- **Code 2: Software driven innovation.** VAS has adapted software-driven innovation as their software adds new functionalities faster than hardware can. The attention to software development helps them to quicker adapt to market demands and customer needs. As software evolves faster, this shift also fits well with their service-oriented business, which enables VAS to remain competitive and responsive to emerging trends.
- **Code 3: Cross-disciplinary communication challenges.** Different terminologies between different teams can cause misunderstandings and delays. For example, terms such as "deploy" for software can mean the word "release" for hardware. Different terminologies needs to be clearly documented to prevent misunderstandings.
- **Code 4: Resource bottlenecks for releases.** Mechanical parts often have long lead times, which creates a bottleneck in overall project schedules. A new mechanical part can delay a release by 6 months, if a press mold for a factory needs to be ordered. Bottlenecks make cross-disciplinary collaboration difficult in practice.

A.15 Person 15 - CPAC Systems

- **Code 1: Project-based communication.** Communication between CPAC and Penta is handled through regular meetings with dedicated teams. For example, the leadership team at CPAC can have meetings with leadership at Penta. Development of a new component triggers a new project at CPAC and a new corresponding project at Penta, which kickstarts the communication channel at both parts.
- **Code 2: Improvements in system integration.** CPAC deliver both hardware and software subcomponents to Penta, as a part of a complete marine system. While CPAC can manage a portion of the system, they do not always have oversight of key components such as the engine or transmission. To improve full system integration and compatibility, CPAC wishes for end-to-end responsibility for the entire system they deliver to Penta.

- **Code 3: Improvement of requirement version control.** There is an interest for improving the current requirement version control. The current system create new versions when a requirements is changed, despite of what the change is. Small changes, such as title change or typos in requirements text are seen new version, which is something CPAC wishes to change as it increases their administrative burden.
- **Code 4: Desire for a unified information model.** There is an interest in improving information sharing with a single system for hardware and software sharing. However concerns exist over how a new tool could satisfied specific needs for both hardware and software.