Adding Extra Dimensions to Message Sequence Charts

Visualizing Component and Date Variability

Ignatius Aries Kurniawan
Abstract

Adding Extra Dimensions to Message Sequence Charts: Visualizing Component and Date Variability

Ignatius Aries Kurniawan

Message Sequence Charts (MSC) are now widely used to describe communication between system components. The diagrams have been formalized, recommended, and even extended. The extensions of the MSCs typically handle additional complexities, e.g. expressing multiplicities. Certain MSC instances are substitutable or omissible with no or minimal effect in the drawings. Additionally, system changes over time would be more easily observed, compared, and maintained if they are visualized in one drawing.

This thesis introduces a concept of Master Chart to extend the MSCs. It is done as a case study in Scania, a heavy truck and bus manufacturer. Contextual inquiry and PACT framework are employed to generate design ideas. Then a prototype .NET application is developed and compared with an existing MSC drawing tool. All test subjects successfully completed the drawing tasks in the prototype tool albeit more slowly than in the old tool. Usability issues are discussed for future full-scale implementation. Based on this study, having the Master Chart as an extension to traditional MSCs seems to create more meaningful and maintainable drawings especially on systems with a lot of variability.
This work owes its existence to the grace of the Lord, the love of my parents,
and the support of the Swedish Institute Scholarship.
# Table of Contents

1. Introduction .......................................................................................................................... 9
   1.1. Background .................................................................................................................. 9
   1.2. Research Question .................................................................................................... 10
   1.3. Goals ............................................................................................................................ 10
   1.4. Delimitations ............................................................................................................. 10

2. Theory ..................................................................................................................................... 11
   2.1. Interactive Visualization ............................................................................................ 11
   2.2. Existing Standard ........................................................................................................ 13
       2.2.1. Message Sequence Chart .................................................................................. 14
       2.2.2. Symbolic MSCs and Live Sequence Charts ...................................................... 15
   2.3. Contextual Inquiry ....................................................................................................... 16
       2.3.1. Contextual Interview ......................................................................................... 17
       2.3.2. Affinity Diagram ............................................................................................... 17
       2.3.3. Persona ............................................................................................................... 18
   2.4. PACT Framework ....................................................................................................... 18
       2.4.1. People .................................................................................................................. 18
       2.4.2. Activity ............................................................................................................... 18
       2.4.3. Context ................................................................................................................ 19
       2.4.4. Technology ......................................................................................................... 19
   2.5. Prototyping ................................................................................................................... 19
   2.6. Usability Testing .......................................................................................................... 19

3. General Method ................................................................................................................... 21

4. Case Study: Scania’s Use of MSCs .................................................................................... 22
   4.1. Investigation ............................................................................................................... 23
       4.1.1. Method ............................................................................................................... 23
       4.1.2. Result .................................................................................................................. 24
       4.1.3. Analysis .............................................................................................................. 28
   4.2. Prototyping ................................................................................................................... 39
       4.2.1. Main User Interface ............................................................................................ 39
<table>
<thead>
<tr>
<th>4.2.2.</th>
<th>Selecting a Use Case .................................................................</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.3.</td>
<td>Master Chart .............................................................................</td>
<td>40</td>
</tr>
<tr>
<td>4.2.4.</td>
<td>Signal Library ...........................................................................</td>
<td>47</td>
</tr>
<tr>
<td>4.2.5.</td>
<td>Object Property ..........................................................................</td>
<td>48</td>
</tr>
<tr>
<td>4.2.6.</td>
<td>Object Validity ...........................................................................</td>
<td>48</td>
</tr>
<tr>
<td>4.3.</td>
<td>Usability Testing ........................................................................</td>
<td>49</td>
</tr>
<tr>
<td>4.3.1.</td>
<td>Method .........................................................................................</td>
<td>49</td>
</tr>
<tr>
<td>4.3.2.</td>
<td>Analysis .......................................................................................</td>
<td>55</td>
</tr>
<tr>
<td>5.</td>
<td>Conclusion ..................................................................................</td>
<td>58</td>
</tr>
<tr>
<td>6.</td>
<td>Future Work ................................................................................</td>
<td>59</td>
</tr>
<tr>
<td>7.</td>
<td>References ..................................................................................</td>
<td>60</td>
</tr>
<tr>
<td>8.</td>
<td>Table of Figures .........................................................................</td>
<td>62</td>
</tr>
<tr>
<td>9.</td>
<td>Appendices ...................................................................................</td>
<td>64</td>
</tr>
<tr>
<td>9.1.</td>
<td>Interview questions ....................................................................</td>
<td>64</td>
</tr>
<tr>
<td>9.2.</td>
<td>Formative test task ....................................................................</td>
<td>65</td>
</tr>
<tr>
<td>9.3.</td>
<td>Comparison test task ..................................................................</td>
<td>65</td>
</tr>
<tr>
<td>9.3.1.</td>
<td>MSC Drawer Training ...................................................................</td>
<td>65</td>
</tr>
<tr>
<td>9.3.2.</td>
<td>MSC Drawer to draw 2 MSC .........................................................</td>
<td>66</td>
</tr>
<tr>
<td>9.3.3.</td>
<td>Master Chart tutorial ...................................................................</td>
<td>66</td>
</tr>
<tr>
<td>9.3.4.</td>
<td>Master Chart to draw 2 MSCs ....................................................</td>
<td>66</td>
</tr>
<tr>
<td>9.4.</td>
<td>Formative test transcription .......................................................</td>
<td>69</td>
</tr>
<tr>
<td>9.4.1.</td>
<td>Formative Test P01 Transcript ...................................................</td>
<td>70</td>
</tr>
<tr>
<td>9.4.2.</td>
<td>Formative Test P02 Transcript ...................................................</td>
<td>74</td>
</tr>
<tr>
<td>9.4.3.</td>
<td>Formative Test P03 Transcript ...................................................</td>
<td>77</td>
</tr>
<tr>
<td>9.5.</td>
<td>Comparison test task log .............................................................</td>
<td>79</td>
</tr>
</tbody>
</table>
1. Introduction

1.1. Background

Message Sequence Charts (MSC) have been widely used in various stages of Software Engineering process[1]. The drawing has been formalized[2] and recommended[3] by the International Telecommunication Union as a language to describe communication behavior of system components. Because it is easily understandable by both specialists and laymen, the MSC format is often used as a medium to communicate between groups of different backgrounds[2]. However, despite their intuitive nature, MSCs still have certain limitations. Several studies have attempted to introduce extensions to the traditional MSC format. One particular limitation that this study acknowledges is its limitation in visualizing complex behavior[1]. For instance, MSC standard does not specify a notation to specify multiplicities[3]. This means if the same message is sent to a group of components, the message has to be drawn for each member of the components, either in the same or in a separate MSC. Thus, to allow high level overview of the behavior, symbolic instances[4] and additional notations[5] have been proposed. On top of this, the developers could also update the system behavior as changes are introduced over time. When some messages are updated with new ones, creating a whole new MSC not only consumes extra storage but also rules out an overview of how the system behavior has changed over time.

Modern visualization can be employed to allow humans and computers to cooperate and combine their capabilities[6], even to extend MSCs. It could enable MSCs’ visual elements to adapt to multiplicities and system behavior changes, as they can be considered extra dimensions in a two dimensional drawing. For example, depth cues, such as shadows, perspectives, and occlusions can be used to suggest a third dimension on the user’s screen.

Visualizing multidimensional data in a two-dimensional screen is not a trivial task[7]. Three dimensional representations are not necessarily better as we perceive the towards-away dimension less well than the other two[8]. Therefore additional techniques can be used, e.g. interactivity. With the possibilities that modern computer systems offer, a good visualization is no longer just a static picture[9] as computers support certain visual query with little cost[8]. In fact, some interaction designs have also been proposed to UML Sequence Diagram [10], [11], which is a format similar to MSCs for modeling system behavior[12]. Adopting such measures, visualizing extra dimensions in MSCs might be possible.
1.2. Research Question

- Certain components behave similarly and therefore their presences in the system are omissible or substitutable without changing the MSC’s content. Other components, when present, can change the system behavior to varying degrees. How can one MSC incorporate and visualize this component dimension?
- After one point in time, MSC’s content could be modified due to system updates. Instead of being redrawn from scratch, how can an MSC visualize this "date" dimension, i.e. the changes on its content after certain date?
- How easily can the visualization be understood?

1.3. Goals

- The main goal of the study is to suggest a design concept for visualizing MSCs with two extra dimensions, namely component and date dimensions
- The sub goal of the study is to evaluate the usability of the design by implementing and testing a demo application

1.4. Delimitations

- This is a case study conducted at Scania, a heavy truck and bus manufacturer. The allocation and ECU family variants of the vehicle functions would become the component dimension in the MSCs. The production date variants of the vehicle functions would become the time dimensions in the MSCs. The users and source data used in this study are specific to the company.
2. Theory

2.1. Interactive Visualization

The use of interaction is so profound that information visualization now refers to “the use of computer-supported, interactive, visual representations of abstract data to amplify cognition.” [13]. This visualization basically comes in four stages (see Figure 1): collection and storage of data, data transformation, visual representation, and human perceptual and cognitive system[9].

![Figure 1 - Four stages of visualization][9]

After understanding the source data, the critical question is how to transform it to support optimal decision making[9]. It has been almost two decades since Shneiderman coined the visual-information-seeking mantra: “Overview first, zoom and filter, then details on demand”[14]. Indeed, in a cognitive loop involving a computer screen, the users should see information they need exactly when they need it[8].

The growing trend in interactive and dynamic visual displays supports exactly this principle[7]. By allowing drilling up and down between overview and details, the screen can show only information relevant to the current cognitive task. For instance, tooltips has become a common way to drill for more information in desktop applications[8].

“Overview first, detail on demand” means providing data on different scale with a way to navigate between them. The ability to modify the scale of aggregation facilitates empirical analysis and verification of results[7].

Interestingly, the complexity of scaling and navigation in MSCs’ component and date dimensions is analogous to navigation in traditional spatio-temporal structure. In an interactive map, cities are aggregated into provinces, countries, continents, and so on. Viewing on continent level will usually hide the cities. Similarly in a diagram, the components can be aggregated into different levels.
To illustrate this idea, let GA be a group of component A1, A2, and A3. Let GB be a group of component B1 and B2. If all members of GA send the same message to all members of GB, one message would be visible from GA to GB at group level and the six individual messages are hidden. This case is pictured in Figure 2 below:

![Figure 2 - GA to GB overview and its detail view equivalent](image)

The date dimension is by nature a temporal dimension. Andrienko argues that it is impossible to represent variations of continuous spatial phenomena in one image[6]. To get around this problem, designers commonly used techniques such as multiple maps, animation, and interactive tools[6].

Multiple maps break the visual into a series of single-date maps. It enables perception of change by letting viewers jump from map to map[6]. In simple MSC terms, this can be as simple as opening multiple windows, each showing an MSC for a particular period of time. See Figure 3 for an example:

![Figure 3 - "Multimaps" for showing diagrams in different date dimension](image)

Animation maps the temporal dimension in the data to our physical time. Because the previous states disappear quickly, it is not suitable for comparison task. Adding interactive functions to manipulate the speed of time could be a way to solve it[7]. Figure 4 shows an interactive drawing that has a slider.
The temporal dimension could be treated either as time points or time intervals[6]. Naturally, values in a time point is only valid at that moment in time, while values defined for an interval would be valid anywhere between two time points. Generally, breaking the temporal dimension into multiple maps helps people focus on spatial patterns, while animation and interactive display helps them focus more on changes and events[7].

In any case, the time scale does not have to be uniform[7]. Depending on the density of the data, the time scale could be denser in some intervals and less so in the others.

Analytical tools must support analysis at multiple scales both in space and time dimensions[6]. This does not concern only the display of the data. The data source could also come in different scales, in which case they must then be consolidated. For instance, the users may want to add a new signal from all GA members to B2 only. Lastly, the data from different scales may interact and influence each other[6]. For instance, if A3 to B2 has a different type of message than the rest of the components, how does the overview message between GA to GB look like?

### 2.2. Existing Standard

Admittedly, many kinds of concepts are not inherently spatial, nor is all spatial information three dimensional[8]. Particularly for abstract non-spatial data, two dimensional views with efficient navigation system (e.g. zooming, hyperlinks) have fared better than their three dimensional counterparts[8].

Moreover, abstract domains in particular have no natural visual cues. Assigning visual encoding to abstract data becomes crucial for robust and reliable visualizations[7]. Although visual thinking is based on pattern perception[8], conventions have a greater role in diagrams and non-pictorial visualizations[9]. Visual designs are known to be generally hybrids; exploiting both pattern finding and conventions[9]. These conventions, in the form of semantics, may transform intuitively explained meanings into formally defined meanings[15]. In the realm system behavior visualization, one of the most widely used formats is Message Sequence Chart (MSC).
2.2.1. Message Sequence Chart

The use of MSCs started in telecommunication industry, where it benefits much from MSC’s scenario based description[1]. International Telecommunication Standard (ITU) issued recommendation in 1992[1], [3]. Since then, the language has been formalized[2], maintained actively, and available in some commercial design tools[1].

According to the ITU standard[3], a Message Sequence Chart (MSC) is a “trace language for the specification and description of the communication behavior of system components and their environment by means of message interchange.” Not only is it ideal for visualizing communication between components in a system, it is also meant to be intuitive and transparent, therefore easy to learn, use, and interpret[1]. Since MSCs are not specific to any application domains, they have been used in wide range of applications, such as requirement specification, interface specification, simulation and validation, test case specification and documentation[1], [3], [5].

Generally, an MSC (see Figure 5) describes communication between system components, and also between these components and the outside world, called environment[3]. On a most basic level, it consists of system components, instance axis, and messages.

![Figure 5 - Basic MSC][1]

Although basic MSCs are able to describe simple scenarios clearly, structuring mechanisms are needed to depict more complex behavior[1]. One way to do this is by using inline expressions[3] (Figure 6). It allows alternative, parallel and sequential composition, iteration, exception and optional regions in the MSC[3].

![Figure 6 - Inline expression symbol][2]

The operator keyword such as seq, alt, par, loop, opt, and exc are written in the top left corner to indicate sequential composition, alternative composition, parallel composition, iteration,
optional region, and exception[3]. The alt operator in particular defines alternative executions of MSC sections and only one of them will be executed[3].

Nonetheless, a normal MSC expresses only one execution trace[16][15][14], not a complete behavior of a system[16]. Standard MSCs do not have notations to specify multiplicities[5]. When the system owns several behaviorally similar objects, this leads to massive specifications[17]. One suggestion to handle multiplicities is by writing allowable ranges on the top of the instances[5]. As shown in Figure 7 this MSC can then be mapped to an identical drawing with inline expressions[5].

2.2.2. Symbolic MSCs and Live Sequence Charts

The limitation of MSCs in expressing multiple objects has given birth to a number of extensions. A symbolic MSC [17], [18] depicts message exchanges between a set of objects. Therefore unlike a lifeline in an MSC, a lifeline in a Symbolic MSC may be either concrete or symbolic[17].

Similar extension has been suggested to Live Sequence Charts (LSC), an extension to MSC format itself. An LSC allows MSC to distinguish mandatory and optional behavior. Yet it is still limited to singular objects, which lead to a large number of scenarios[4]. The LSC format is then extended further with symbolic instances, or classes of instances, that binds to multiple objects[4].
Furthermore, symbolic instances and multiplicities attribute also exist in UML Sequence Diagrams (SD) which are specified in the UML Metamodel (UMM) [12]. Unlike MSCs, an SD does not have inline expressions, although adoption has been suggested [19]. Figure 8 below illustrated this idea.

![Figure 8 - An MSC and its symbolic equivalent][17]

However, even without multiplicities, Sequence Diagrams can be very large and therefore confusing to the viewers. IBM Toronto Software Laboratory embedded a small navigation map in the drawing software to navigate around the large diagram [11]. Sharp evaluates interactions in sequence diagram to manage the complexity of large, reverse engineered diagrams [10]. Some of the interactions include zooming, filtering, collapsing, and selecting for details [10].

In practice, MSCs support cognition well through pattern finding [8], especially in tracing messages between different components. Yet they come with shortcomings too. Because in abstract domains graphical proximity is perceived as conceptual proximity, closer elements are perceived to have a shared property [7]. Longer lines in MSCs are seen as larger or farther away [10]. In spite of any patterns formed by the objects’ proximity [8], in an MSC the distance between components does not really mean anything.

### 2.3. Contextual Inquiry

Contextual Inquiry has been an industry standard field data gathering technique [20]. It reveals the tacit aspect of users’ activities, based on the premise that people are not completely aware of their work until they do it for real [20]. The inquiry captures real business practices and activities instead of self-reported practice or official policies [21]. Thus it is commonly used as a first step in Contextual Design, which is a user-centered design method to do in-depth field research and design technical products [20]. Contextual Design utilizes Contextual Inquiry to create affinity diagram for “quick hit” of the primary issues [22]. On top of that, consolidated work models can be used for a more complete picture. The method takes advantages of a cross functional team to generate insights [20], with lighter versions available for smaller teams [21].
In general, Contextual Inquiry observes four principles: context, partnership, interpretation, and focus[23].

1. *Context* means doing the inquiry in the real settings. It allows capturing ongoing experience rather than summary experience and concrete data rather than abstract data. All contextual interviews are done in FO’s own workplace.

2. *Partnership* improves ordinary apprenticeship attitude, since the goal is not to learn the work but to build a system to support the work. It allows the interviewer to interrupt users and engage them in conversation about the reasoning behind their actions.

3. *Interpretation* is performed by actively forming hypothesis and validate with the users. While doing the actual work, the user will be least biased and able to correct wrong interpretation.

4. *Focus* allows the interviewer to stay on topic without taking the control completely from the users. For example, a user’s mental model and system’s usability issues are more important that some technical bugs in the current system that may appear in the study.

One way to do Contextual Inquiry is by doing Contextual Interview. Contextual Interview is a one on one interaction between the interviewer and the user while the user is doing the work in their workplaces[21].

### 2.3.1. Contextual Interview

Contextual Interviews, as a form of Contextual Inquiry, follow the same principles of context, partnership, interpretation, and focus. In the beginning a Contextual Interview looks very much like a traditional interview, with introduction and general overview of the situation[20], [23]. Then there is a transition to user’s activity where the user does the actual work. Typical contextual interviews takes 1½ - 2 hours[20], [21], [23]. A master/apprenticeship attitude is often adopted during the interviews[21]

### 2.3.2. Affinity Diagram

Affinity diagrams[23] organize individual ideas and insights to show common structures and themes. The contextual interviews are interpreted into small groups, typically using sticky notes on a wall, based on issues they describe. The groups of the notes emerge from the data and are not predefined. The affinity diagram is built from bottom-up and groups of notes are grouped again into a bigger theme. In the end of the process, the affinity diagram becomes a single hierarchical structure that reflects issues across the user’s population. This way the complexity of the qualitative data becomes manageable and consumable.


2.3.3. Persona

Personas are archetypes of users describing their goals and behavior[24], [25]. Unlike user roles, it focuses more on behavior rather than job description or authority[25], [26]. Persona description typically contains a name, a photo, and set of goals [25] and narrative that covers skills, frustration, attitude, tasks, and environment [25], [26]. Persona is good as a medium to stand on users’ shoes [24] and iterate the design until it feels right for the personas [25].

2.4. PACT Framework

PACT (People, Activity, Context, Technology) framework[24] is a framework to analyze information gathered during the data collection process. It is not only useful for analysis but also for design. Essentially, it scope the collected data into variety of Ps, As, Cs, and Ts. By framing it into those four aspects, the framework separates gives better understanding about people who will use the systems, their activities, the context of the activities, and the interactive technologies available for design.

2.4.1. People

It is important to be aware of people’s physical, psychological, social, and mental model differences[24]. Psychological differences, for instance, could be difference in spatial skills, attention, memory, etc. Social differences are the different goals, motivations, and level of knowledge that users possess.

Mental models are users’ own model of the system, which can come from interacting with the system, reading manuals, etc.[27] They also often differ from the designers’ conceptual models[24]. Thus, designers must help users to form accurate mental model, e.g. by providing a clear, logical, and consistent conceptual design for the system[24] and its visible parts[27].

2.4.2. Activity

Some characteristics of the activities are important to consider too[24]. Temporal aspects, such as how frequent the activities are, whether they happen under peak times, or if they are often interrupted. Other aspects to investigate are whether the activities are carried out alone or in a team, if they are well defined or vague, and their criticality. Data requirements to support the activities should also be studied.
2.4.3. Context

The next aspect to keep in mind is the context in which the activities take place[24]. Context can be organizational context, social context, or physical circumstances. For example, the design may be influenced by the location of the activities, the availability of any manuals or support, and other norms in the organization.

2.4.4. Technology

Finally, interaction designers work and shape technology. Typically, interactive systems consist of hardware and software, which could also be divided as input, output, communication, and content[24].

2.5. Prototyping

A prototype is both a tool to explore design[26], [28] and evaluate it [24], [26]. In reality, it may emphasize a set of characteristics and deemphasize others [24], [26]. Prototypes come in different forms, such as papers, cardboards, even in real software [22], [25]. In general, they can be categorized into low or high fidelity prototypes.

Low fidelity prototyping is useful in the early stage of design because it is simple, cheap, and quick to produce or modify [26]. It investigates broad underlying ideas such as content, form and structure, design tone, and navigational [24]. Hence, in the beginning of the project, paper sketches are used to explore how an MSC could be visualized and to check it with the users.

High fidelity prototyping, on the other hand, depicts the final product to a greater extent [24], [26]. Therefore it is suitable to communicate ideas in a detailed way to stakeholders or for testing out technical issues [26]. However, it has limitations too. High fidelity prototypes take longer to create, may distract discussion from the design essence to the superficial, build resistance to change at the developer side, and are more vulnerable to bugs [26]. The users may believe the prototypes to be real, so using real or detailed data in these prototypes can avoid confusion [24].

2.6. Usability Testing

It is quite common to see the term Usability Testing used to describe any test to evaluate a system. Ruben and Chisnell[29] define usability as:

"process that employs people as testing participants who are representative of the target audience to evaluate the degree to which a product meets specific usability criteria."
According to ISO 9241 [ISO9241] the main components of usability are effectiveness, efficiency, and satisfaction. Usability testing uncovers these aspects to inform design[29]. Specifically, there are four main types of usability testing: Exploratory or Formative Study, Assessment or Summative Test, Validation or Verification Test, and Comparison Test. They are used for different purposes, as described in the table below:

<table>
<thead>
<tr>
<th></th>
<th>Exploratory / Formative</th>
<th>Assessment / Summative</th>
<th>Validation / Verification</th>
<th>Comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Objective</strong></td>
<td>Examine preliminary design concepts</td>
<td>Evaluate lower level components</td>
<td>Evaluate how the product compares to usability benchmarks</td>
<td>Compare two designs</td>
</tr>
<tr>
<td><strong>When</strong></td>
<td>Early</td>
<td>Early or Midway</td>
<td>Late</td>
<td>Any</td>
</tr>
</tbody>
</table>
| **Method Overview**  | • Extensive interaction between participant and moderator
• Prototyping
• Informal
• Realistic task or walkthrough | • Less interaction between participant and moderator
• Introduce more test control
• Realistic task
• Quantitative measures | • Little to no interaction with moderator
• Standards are defined prior to the test
• Quantitative data is essential | • Can be used as conjunction with other tests
• Can be either loosely or tightly controlled
• Performance and preference data |

*Figure 9 - Comparison between different types of usability testing*[29]*
3. General Method

The general method in this project is a case study, which is a study of a case within a real-life, contemporary context, or setting[30]. It focuses on behavior and settings instead of the total context or natural flow of the behavior[31]. Though these imposed constraints narrow the focus, it retains the essential interest in participants’ natural behavior[31]. It is a good approach when the case has clearly identified boundaries and the researcher seeks to provide an in-depth understanding of it[32].

The data collection in case study research is typically extensive, drawing on multiple sources of in-depth information [32]. Yin [30] recommends six types of information to collect: documents, archival records, interviews, direct observations, participant observations, and physical artifacts. Relying on one source of data is typically not enough to develop this in-depth understanding [32].

Case study research begins with the identifying a specific case, either on an individual, a small group, or an organization[32].
4. Case Study: Scania’s Use of MSCs

This study specifically takes the use of MSCs in Scania as a single instrumental case study[33]. Scania uses MSCs regularly to document vehicle functions and communicate them across groups[34]. Documenting every component and time variants is difficult[34] and testing everything impractical[35]. As the company produces new MSC for every component and date variants[34], there are challenges on ensuring data integrity, accuracy, completeness, and process efficiency.

The ability to visualize such variant information, which now lies in separate drawings, into one MSC would potentially benefit the company. Hence the setting is deemed suitable for the case study. Twenty weeks of thesis work is performed during Jan 2015 – June 2015 at Scania under System Integration (REST) department, which use MSCs extensively to test functionality at complete vehicle level.

This study follows the recommendation of vast data collection. It looks at internal documents and past thesis projects as a starting point, while doing interviews and observations in the form of contextual interviews. The contextual interviews produce affinity diagrams and persona, which are then framed and analyzed using the PACT framework. The PACT analysis in turn leads to design ideas that manifest in prototypes. Finally, a usability study is conducted to evaluate a high fidelity prototype.
4.1. Investigation

A full Contextual Design process is not adopted because of its nature as a team-based method. Even on its simplest form, a lightning fast Rapid Contextual Design requires a team of two and an additional helper. Because the study has one researcher and very limited time, only Contextual Inquiry is adopted as a tool to investigate the case. For the same reasons, affinity diagram and personas are created but consolidated work models are not built.

Contextual interviews are conducted to the main users. On top of that, traditional interviews to other users and investigation on internal documentations are performed. The result of all this investigation process, including the affinity diagram and personas from the contextual interviews, are framed in a PACT Framework for further analysis.

4.1.1. Method

4.1.1.1. Contextual Inquiry

4.1.1.1.1. Contextual Interview

To really understand how Function Owners (FO) produce MSCs, contextual interviews are conducted to 4 of them. Function Owners draw MSCs to document User Functions (UFs, which are the vehicle functions. More on User Functions later) they are responsible for, using Scania’s specific processes and tools.

An FO’s work on MSCs is by nature intermittent. The diagrams are updated only as a response to a change request (either because of functionality update or correction). Such infrequent activities further motivate the use of contextual interviews in the data gathering process. The FOs are likely to have forgotten the exact steps of creating an MSC, so seeing the work in action would reveal more useful information than asking abstract interview questions. During the period of the study, only few FOs happened to pending MSCs work for normal contextual interviews. The others had to demonstrate their work by retrospective account.

To complement this contextual interviews, traditional interviews are also conducted to 4 System Architects (SA) and 2 System Integration Testers (ST). They are involved in the MSC’s creation process rather indirectly. SAs are responsible for standardizing objects in MSCs[36] and generally knowledgeable about the database, so they are questioned about the availability of data and its structure. On the other hand STs are responsible for creating MSC guidelines and using MSCs to test vehicles[34]. They are questioned about MSC guidelines, their views on variants, and how they use the MSCs. Complete list of interview questions are available in Appendix 9.1.
4.1.1.2. Affinity Diagram

The affinity diagram is built to consolidate data gathered during the contextual interviews with Function Owners. No predefined themes are assumed as they appear based on the notes.

4.1.1.3. Persona

The rich contextual data about FOs could be used to generate some personas. Unlike SAs and STs who have their own groups and are relatively homogenous, the FOs are spread across different working groups. Multiple personas are created to bring FOs archetypes into life and represent their unique goals and behaviors.

4.1.1.2. PACT Framework

Data from contextual inquiry and the investigation of internal documents are organized for deeper analysis in the PACT framework. For instance, the affinity diagram’s major themes are spread into all aspect of People, Activity, Context, and Technology; the personas provide a clear description for the People aspect of PACT; the company’s standards and conventions are put together as Context; and so on.

4.1.2. Result

4.1.2.1. Affinity Diagram

The contextual interviews conducted to 7 FOs reveals 4 major groups in the Affinity Diagram:

1. The knowledge about vehicle functions, MSCs, and variants spreads and varies across users
   - FOs, SAs, and STs collaborate and share knowledge

Figure 10 - Clustering the affinity diagram from interviews with FO, SA, and ST
- FOs relies on historical data and ST to determine variants
- FOs know less of legacy and stable vehicle functions
- Some FOs only design and test vehicle functions, others also implement
- The boundary of one vehicle function documentation is not always clear

2. Consistency among document is crucial
- FOs believe MSCs are good documentation if they are up to date
- FOs are afraid of making mistakes (e.g. typo in free text) in the MSCs
- The Allocation Element Diagrams are often more updated than the MSCs

3. Current MSC drawing process is not efficient
- FOs create many redundant information among MSCs or between MSCs and other documents
- FOs use workaround to make current tool more efficient

4. Function Owners draw MSC very irregularly
- Drawing MSCs is seen as a small part of a FO’s job
- FOs do MSCs only 2-3 times per year
- FOs takes time to remember how to draw MSCs

### 4.1.2.2. Persona

**Tobias**

Tobias is a 30 years old development engineer at Scania. He is from Sundsvall, has engineering degree and now lives in Södertälje. He joined Scania four years ago. Then two years ago he was moved to RBE, where he inherited some old User Functions on top of some newer User Functions that he also owns. He has updated some functions which were implemented by his colleagues. However there are few old, stable functions that he is not very familiar with.

“Thankfully my functions have only a few variants”, he said. He usually gets away with documenting them by mimicking previous work and making no more than necessary additions. He is always very careful with his documentations. He hates making mistakes, partly because he will have to revise it again. With the current MSC Drawer software (more detail about the software later) he uses the auto-populate often and almost never uses the free text.

In general he understands the usefulness of MSCs but thinks it is too cumbersome to do with the current tools.

**Goals**

1. To have his MSCs approved with minimum error and revision
2. Get the job done without too much learning or hassle
Joakim

Joakim is a relatively young engineer from Jönköping. He worked in another truck manufacturer before he joined Scania 2 years ago. So far he likes the challenge of his job at REV group in Scania.

That is particularly because he is responsible in developing many different functions. He knows about User Functions that he owns but he feels documenting them is difficult because his functions has many variants. He sometimes asks testers for advice about what Scenarios there are to cover. He likes it when he finds out about new Scenarios to cover while doing an MSC.

When he does his MSCs, he sits for 2 to 3 weeks then probably will not update them again in more than 6 months. Creating MSCs is highly repetitive task for him, as his MSCs are very similar to each other. To get around this he regularly hacks the source MSC files (.mpr) to copy paste their contents. Another problem for him is how his MSCs can quickly become outdated, rendering the diagrams inaccurate.

He sometimes wonders if he has provided enough information in his MSCs for his colleagues, particularly the workshops.

Goals

1. His documentations to stay up to date
2. To minimize repetitive work in many MSCs
3. Easy tool to pick up after a long time

Roland

Roland is a senior engineer who has been long working at NEC group in Scania. He was born 37 years ago and raised in Stockholm. Programming has always been both hobby and occupation for him, so he likes the fact that he can implement his own functions at Scania. Combined with the fact that he is responsible only for a few User Functions that are not very variant dependent, he knows all his own User Functions extremely well. He knows about his variants and always in discussion with the integration testers. However, he does not particularly like doing all the different requirement documentations. Coding is always more fun. "Here at NEC we are all programmers", he said proudly. He thinks there is too much redundant documentation for his small UF. He does MSC when he feels like it or when it is urgent, usually about 2x a year for a week each time. He thinks the current MSC drawer is very buggy for his standard.

Goals

1. Spend as little as possible on documentations, including MSCs
2. Reliable tool to document his functions because he thinks a FO should not be a tool expert
3. Minimum redundancy in documentation
## Persona summary

### Function Owner

<table>
<thead>
<tr>
<th>Tobias</th>
<th>Joakim</th>
<th>Roland</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>20 years old</strong></td>
<td><strong>28 years old</strong></td>
<td><strong>26 years old</strong></td>
</tr>
<tr>
<td>From: Sundsvall</td>
<td>From: Jönköping, lives in Södertälje</td>
<td>From: Stockholm, lives in Stockholm</td>
</tr>
<tr>
<td>Worked at RIE</td>
<td>Worked at RIE for 2 years ago with prev work experience</td>
<td>Worked at NEC for 6+ years ago</td>
</tr>
<tr>
<td>Characteristics</td>
<td>Characteristics</td>
<td>Characteristics</td>
</tr>
<tr>
<td>Designs User Functions</td>
<td>Designs User Functions</td>
<td>Designs and implements User Functions</td>
</tr>
<tr>
<td>Spend most of his time developing and testing User Functions</td>
<td>Responsible for many User Functions</td>
<td>Likes to code, spend a lot of time programming</td>
</tr>
<tr>
<td>Responsible for 6 User Functions</td>
<td>Knows User Functions very well</td>
<td>Knows own User Functions very well</td>
</tr>
<tr>
<td>Knows only parts that he has updated</td>
<td>Many Scenarios to cover</td>
<td>Responsible for few User Functions with almost no variants</td>
</tr>
<tr>
<td>Knows little about User Functions that he inherited or are very stable</td>
<td>Little knowledge about variants, usually work with testers</td>
<td>Proposes some Scenarios and confirm to testers</td>
</tr>
<tr>
<td>Knows some variants but not too many</td>
<td>Does MSCs 2x / year for 3 weeks</td>
<td>Proposes some Scenarios and confirm to testers</td>
</tr>
<tr>
<td>Scenarios loosely defined, mostly follow previous work</td>
<td>Thinks creating MSC is repetitive yet quickly becomes outdated</td>
<td>Proposes some Scenarios and confirm to testers</td>
</tr>
<tr>
<td>Does MSCs very carefully and afraid of mistakes</td>
<td>Some curiosity about tools and willing to explore functionality to certain degree</td>
<td>Proposes some Scenarios and confirm to testers</td>
</tr>
<tr>
<td>Uses populates and copy paste from other MSCs</td>
<td>Regularly hacks MSCs to speed up his task</td>
<td>Proposes some Scenarios and confirm to testers</td>
</tr>
<tr>
<td>Understands that MSCs is useful but takes too much time with the current tools</td>
<td>Becomes aware of new MSCs to create while working on an MSCs sometimes</td>
<td>Proposes some Scenarios and confirm to testers</td>
</tr>
<tr>
<td><strong>Goal</strong></td>
<td><strong>Characteristics</strong></td>
<td><strong>Goals</strong></td>
</tr>
<tr>
<td>MSCs is approved with minimum errors and revision</td>
<td>Designs User Functions</td>
<td>To spend as little time as possible on MSCs</td>
</tr>
<tr>
<td>To get the job done without too much learning for a tool, especially after not using it</td>
<td>Responsible for many User Functions</td>
<td>Easy and reliable tool to get the job done because he thinks Function Owners should not be a tool expert</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Minimum redundancy in documentation</td>
</tr>
</tbody>
</table>

*Figure 11 - Summary of the three Function Owners’ personas*
4.1.3. Analysis

4.1.3.1. PACT Framework

4.1.3.1.1. People

Function Owners

Tobias, Joakim, and Roland represent more or less 100 Function Owners in Scania. The actual distribution of the Function Owners within Scania’s Research and Development corporate unit is illustrated below:

![Function Owner distribution inside R&D’s groups](image)

Function Owners are typically engineers although their level of experience varies. Some have worked for 2 years or less while it is not uncommon to see ones with 4+ years of experience or even 10+.

It is important to note that being a Function Owner is not the main role of the employees in Scania. Ownership of a function is given to someone who happens to work closely with the function or its main component. For instance, a development engineer working in Engine Control Software section could be responsible for Engine Protection function. The ownership of a function can also be passed to other employees if the original FO moves away from the group or even the company. Some functions are more stable than others and have been passed down for years without significant modifications.

Their motivations and goals in creating MSCs vary even more. The Function Owners who develop their own functions might be inclined to use their time to code rather than drawing the sequence charts. Others with vast number of variants want to draw their MSCs as efficiently as possible. Some FOs have fewer number of documentation because their functions work the same way regardless of the vehicle configurations. In that case they can concentrate on the accuracy of the drawings and might just want to get them approved by the System Testers. The top 10 groups who own the most variants (indicated by the number of Scenario, more on Scenario later) are shown below:
To help FOs to use the MSC Drawer software properly, there are wiki page about MSC workflow[37], its step by step[37], and general guidelines[34]. Although the tool itself is not very complex, FOs who have not been working on MSCs for a long time follow the manuals very carefully. On the other hand, Function Owners who have done MSCs recently are not as reliant on the manuals. It appears that the user’s mental model is only formed after many consecutive use of the current tool.
4.1.3.1.2. Activity

MSC workflow is very much CR driven[37]. The complete workflow from a Change Request until the completion of an MSC is shown on the diagram below:

![Diagram of MSC workflow](image)

**Figure 14 - Official workflow for creating MSCs[37]**

After a CR for a User Function is registered, related FO, SA, and ST for that function sit together in a systemization meeting. In this meeting they decide, among others, what signals and components to use in a function and what Scenarios to produce. As revealed in the interviews, the knowledge to make such decision is currently shared among the three roles. At the systemization meeting, SAs registered the components and signals related to the function as an Allocation Element Diagram (AED) in the Sesamm database. The FO then uses it to design, implement, and document their function. The documentation comes in various formats, including
in MSC format. The STs, from their experience using MSCs to test different vehicle variants, often knows better what variants are there and how the MSCs should look like.

Currently, the process of drawing MSCs relies on the completeness of component and signal registration in AEDs. MSC Drawer, the tool the FOs use to draw MSCs, retrieves signals and components from AEDs. If a FO wants to use an unregistered component and signal, she has to wait until one SA adds it to the AED. However, there are some signals that are impractical to register in the AEDs (e.g. signal to/from the outside Environment). In that case the FO would have to add it as a free text. This has a risk of human errors (e.g. typo or incorrect copy pasting).

The current MSC Drawer is only used in the activity number 3 (Draw MSC) and 11 (Edit MSC). Other related activities in handled with different applications. For example, the registration of the components and signals is done in Sesamm Tool by the System Architects. The submission and feedback for MSCs are done via emails.

In reality, MSC updates does not happen more often than 3 times per year for any FO.

4.1.3.1.3. Context

Vehicle Function

To design the system, it is important to understand the nature of the vehicle function that FOs, SAs, and STs are dealing with. Scania’s vehicle system is a platform for hundreds of functions (“User Function” / UF) which are divided into sub-functions (“Use Case” / UC). A User Function could be seen as a package of coherent functionality. User Functions are described in several documents, such as User Function Requirement (UFR) and Function Allocation Description (FAD)[37]. An example of User Function is UF129 – ESP Control.

Typical User Function consists of several Use Cases. In Scania a Use Case is defined as “A functionality of a User Function that can be triggered or used by the driver or vehicle entity”[38] or simply as a function with activity and response. For instance, a UF Auxiliary brake may have two UCs: Activate auxiliary brake using lever and Deactivate auxiliary brake using lever. This can then visualized as a tree, as shown in the Figure 15 below:

![Figure 15 - User Function may have several Use Cases](image)

Variants

Furthermore, each UC can be implemented in many different vehicle configurations. Different component and parameter configuration possible to realize a UC is called Variant. Major difference in signal interface is called Allocation variant and recorded in the system as Scenario (SCN).
A SCN is internally defined as "An implementation of the Use Case"[38] or, in a more elaborate version, "A configuration of a Use Case with unique signal flow". For each unique vehicle build, only one SCN in a UC must be valid[38]. For instance, the UC Activate auxiliary brake using lever may have two SCN: Activate auxiliary brake using lever in a truck with Tachograph and Activate auxiliary brake using lever in a truck without Tachograph. Another example is shown in Figure 16 below. Today, all Scenarios are created through collaboration among FOs, SAs, and STs.

Furthermore, within one Allocation variant there are further, less-significant variants (see Figure 17). Different ECUs from the same family makes ECU Family variants. Different installation of switches and actuators makes Component variants. Different ECU software parameter makes Parameter variants. All these possibilities result in enormous number of vehicle variants.

![Figure 16 - Use Case may have several Scenarios](image)

![Figure 17 - Variant structure at Scania](image)
Documentation: AEDs and MSCs

The architecture of each UF is documented in an Allocation Element Diagram (AED). Most User Functions have one AED [Figure 18]. An AED is an architecture diagram of Allocation Elements (AEs) and Function Variables (FVs) that they exchange between each other [Figure 19 is an example]. AEs are logical components which are implemented in physical components[39]. For instance, AE462 - Engine Information Provider is implemented in S6 ECU. FVs contains standardized packages of signal and messages. For instance, FV396 - EngineSpeed wraps the actual signal EEC1.EngineSpeed.

Due to its nature as an architecture diagram, AEDs by default do not hold variant information. If one component is substitutable with another, both of them are present in the drawing. Similarly, optional components are drawn the same way as mandatory components. Therefore AEDs merely shows the union or gross list of all possible components and signals under a User Function. Not every UC and SCN under that UF will utilize all the objects, but nothing outside the AED may be used. Although exceptions exist, this is generally the norm.

Currently there is a strict one-two-one relationship between MSCs and SCNs[38], resulting in a one-to-many relationship between AEDs and MSCs (see Figure 20). An MSC describes signal flow between the environment, external sensors, the driver, ECUs, and actuators relevant to a function on a timeline[38]. Simply put, an MSC is a drawing of a SCN. The focus of the document is the CAN signal[37]. The Function Owners are the main responsible party in the drawing process. The
System Architects are involved in the registration of the objects to the database, and the System Integration Testers in reviewing the diagrams.

**Start of Production (SOP) date**

Every year Scania has defined several Start of Production (SOP) dates when Change Requests (CRs) are implemented. CRs are written for all changes affecting the IT systems in Scania’s products[40]. It drives release process deliverables[41]. Some triggers to write a CR includes: introduction of a new function, changes in existing functions, changes in signal flow between ECUs, introduction or revision of a hardware, and introduction of a new ECU system[40]. Every object in Scania system (UFs, UCs, SCNs, components, signals, documentations, etc.) has a start and end SOP date, scoped a period of time during which the object is “active”.

**Component and time dimensions in MSCs**

Those two aspects, the variant and SOP date, make component and time dimensions in the MSCs. First of all, there are enormous combinations of Allocation and ECU family variants alone. The company’s principle of modularity means most of the components are substitutable or even omissible. Even on Allocation variant and ECU family variant level alone, possible vehicle builds are too many. Yet each vehicle build performs function in a different way and needs to be documented in a separate MSC. This is the component dimension in the MSCs. Additionally, change requests that happen on a certain SOP dates may also change how an MSC looks. This SOP dates become the time dimension in the MSC. A different variant or a different SOP date might produce a new MSC, as illustrated below:
In a summary, Scania faces 3 main challenges in its MSC drawing process:

1. AED-MSC integrity: Huge number of MSCs are difficult to maintain up to date and in sync with other documentations, particularly with AEDs. Currently there is no straightforward way to see if the two documents are consistent with each other.

2. MSC completeness: Currently what MSCs to create is determined by discussion between the ST and the FO on need base. There is no systematic way to identify what variants are technically possible and compare them with available MSCs.

3. Inefficient drawing: Much effort and time to manually draw all MSCs for all variants and CR, despite most of them being similar. Moreover, as the current MSC Drawer does not have copy paste feature, many FOs open the drawings with a text editor to copy-paste the contents. This workaround is not only inefficient but may also contribute to MSCs’ inaccuracy.

4.1.3.1.4. Technology

All Function Owners use standard desktop computer with windows 7 operating system for their everyday work. They have two or three monitors with varying size. In MSC context, some relevant applications within Scania are Sesamm database, SesammTool, and MSC Drawer. The relationship between them is shown in Figure 22:

![Figure 22 - Relationship between Sesamm database, Sesamm Tool, and MSC Drawer](image)

There are obviously other relevant systems for the production of an MSC, for example a tool to show CAN messages and its arguments. Production database also holds variant information in its Functional Product Characteristic (FPC) codes and some overlapping data with SesammDB. However, SesammDB, SesammTool, and MSC Drawer are the major part in the development of MSCs and have been used by all Function Owners. Thus, those three are chosen as the focus technology to analyze.
Sesamm Database

The Sesamm database (SesammDB) is a highly normalized database with anchors, attributes, tie, and knots. Its model is illustrated in Figure 23. In the context of this project, SesammDB is the source of UF tree data, including their AEDs and MSCs.

![Figure 23 - Part of the SesammDB data model]

The database has, among other things, list of all UF, UCs, SCN, components, messages, signals, etc. It is used as a source to create AEDs and MSCs. Every UF has one or more associated AED id, which lead to list of components and signals used in the UF. On top of the list, the graphical AED drawings are also saved in xml format. The MSCs’ metadata are saved in the SesammDB but their contents are saved as text files (.mpr) in the network drive.

SesammTool

![Figure 24 - SesammTool]
SesammTool is a desktop application that reads and writes to SesammDB. It is widely used across different groups in Scania. When drawing MSCs, Function Owners sometime use SesammTool to look up extra information. General UF structure, AEDs, past MSCs, signal arguments, or component details are some information that could be useful during MSC drawing process. System Architects use SesammTool to draw and update AEDs. System Integration Testers might use it to open AEDs and MSCs, although they can also use other tools to open MSCs or mostly just print them.

**MSC Drawer**

![Figure 25 - MSC Drawer](image1.png)

MSC Drawer is, as it name suggests, the desktop application Function Owners use to draw MSCs. The FOs begin by selecting a SOP date and navigating the UF tree to find their UCs (see Figure 26). They can then open an existing MSC or create a new one under the UC.

![Figure 26 - MSC Drawer: Open MSC](image2.png)
In case of starting a new MSC, an Auto-populate feature could be used to lay all the registered components and signals into the blank chart. The signals would of course have to be reordered and even duplicated if they appear in more than one place. Sometimes it brings more trouble than benefits so not all FOs use this feature.

Every MSC signal is drawn by following these steps (see Figure 27 for illustration):

1. Clicking the instance lifeline of the source component
2. Dragging to the instance lifeline of the destination component (an arrow appears)
3. Releasing the left click
4. Selecting the message, signal, as well as inputting additional information in the pop up box that appears.

![Figure 27 - Drawing a signal](image)

The signals in the pop up box are retrieved from the SesammDB. Some signals, such as signals to or from Environment, do not exist in the database and have to be added as free text. Additional MSC objects, e.g. Alternate boxes and Actions, can also be added using the MSC Drawer.

Deleting a signal is possible by right clicking the signal and clicking Delete Signal. Repositioning a signal that is already in place is currently impossible. Copy and pasting signals are not supported either. Some Function Owners choose to open the MSCs with a text editor and edit or copy its content more flexibly.
4.2. Prototyping

The prototyping in this project comes in three forms: paper sketches, interactive prototypes (with Axure, a prototyping software), and a demo WPF application (with C#, .NET framework). Each fidelity level is regularly checked with the users, since most of them are reachable for weekly feedback. Paper sketches and interactive prototype are employed to portray UI, visualization, and interaction design concepts. The demo application implements core functionality and is used in the usability testing.

The main user interface owns some new interface components that are not present in the current MSC Drawer.

4.2.1. Main User Interface

![Main User Interface in wireframe](image)

The main user interface looks quite different from the current MSC Drawer. Instead of showing the MSC in the whole application space, it gives room to a Signal Library, an Object Property box, and an Object Validity box. Those three interface components bring the data forward to support basic actions in MSC drawing, e.g. viewing what signals to add and what their properties are.

The major design change is the introduction of the Master Chart in place of Message Sequence Chart. Master Chart offer Function Owners the ability to drill up and down on the component dimension, thus drawing the signals on any component variant.

On top of the window, basic information about the User Function tree is displayed. Master Chart represents how a function works across component and date dimensions. In other word, Master Chart could be considered as MSC at Use Case level, consolidating the component and SOP date variants under it.
4.2.2. Selecting a Use Case

Going to a Use Case to open its Master Chart is possible with the User Function search box. Some Function Owners are familiar with their function names or codes, especially if the function changes quite often. Using the search box, which gives suggestions on search hits, would be a quicker way to navigate through hundreds of User Functions compared to using traditional tree structure. For Function Owners who cannot remember their function names and codes, the full tree of User Functions is available one click away through the Browse button.

After a User Function is selected, the Use Case list box is populated with the Use Case under the function. The user can select the use case by clicking it and then clicking Continue.

4.2.3. Master Chart

The Master Chart is a proposed new concept to visualize and draw MSC across multiple dimensions. The notations are very similar to traditional MSCs. However, in addition to standard MSC conventions, it is equipped with the ability to traverse across the component and date dimensions.
Navigating through component dimensions in the Master Chart

Each of the MSC’s instances in the Master Chart is configurable to different abstraction level. This abstraction follows the ECU family tree defined within Scania. For instance, an Electronic Brake System (EBS) ECU instance is changeable to an Automatic Braking System (ABS) ECU since they are part of the same family of Braking Management System (BMS) ECU. Additionally, switching to an abstract instance of BMS ECU is also possible, as illustrated below:

With the abstract BMS ECU instance, it becomes possible to draw a signal that applies to both ABS and EBS. Some ECU families demand more complex tree. Therefore the tree should be configurable by the Function Owners. For instance, under the Gearbox Management System (GMS) ECU family, the components could be grouped into custom subgroups of Automatic Gearboxes and Manual Gearboxes, and so on. See the illustration below:
The ability to manipulate the MSC instances is simple yet very powerful. The messages between the ECUs are sometimes different on a very specific configuration only. With the flexibility to change the instances at will, even to a different abstraction level, any signal can be drawn at any level.

Figure 33 - The three top signals are drawn for all GMS. The last signal is drawn only for A5
Drilling down to a specific component configuration, drawing a signal, and then seeing the result is quite straightforward. What if the user drills up back to a more abstract level? The signals applicable to all configurations under the current active configuration are shown as clear black arrows. The signals applicable to only some configurations under the current active configuration are shown as grey arrows:

![Diagram of signal configurations](image)

*Figure 34 - On a higher level, variant-specific signals are greyed*

It is difficult to show to which particular configuration a grey signal applies to without cluttering the chart. The configuration information is available one step away in a tooltip when users hover over any signal:

![Diagram with tooltip](image)

*Figure 35 - Hovering over a signal reveals its configuration information*

Furthermore, it is possible to bring the context menu by right clicking a signal and then going to its specific configuration:
Additionally, basic MSC drawing operations are supported. Deleting a signal is possible by right clicking the signal and selecting Delete. Repositioning signals is possible by dragging an arrow and dropping it on its new position.

With this capability, the signals can be drawn on the exact configuration they apply to. Not only is this way of drawing MSCs more efficient, it also allows the MSCs to cater for future changes. For example, the user may want to draw a signal that is applicable to all GMS. If there is a new GMS ECU in the future, the system could detect it and suggest that the signal should be applied to the new GMS ECU as well.

Date dimensions in the Master Chart

Handling the date dimension is less straightforward than managing the component dimension. Scania’s current work practice attaches start and end SOP date to each Scenario (Figure 37, Figure 38). The Scenarios indicate when they are active and also serve as concrete objects to test.

![Figure 36 - Going to a signal’s configuration from the context menu](image1)

![Figure 37 - The start and end SOP date is written in the SCN info detail tab](image2)
If similar process is kept, it means the system should assign date dimensions to the Scenarios. The Scenario becomes a set of component configuration with start and end SOP date. For instance, a Master Chart with four different Scenarios may look like this:
A slider can then be used to change SOP date and inactive Scenarios are greyed:

![Master Chart Editor](image)

Figure 41 - The tab headers respond to the SOP date slider

As an alternative to drawing in the Master Chart, the users can also draw signals in the Scenario tabs. The signals drawn in a Scenario tab are directly assigned to its specific point in component and date dimensions. The results can still be observed in the Master Chart. Such approach fits Scania’s current practices well because the users are not forced to learn how to use Master Charts straightaway. They can instead use the predefined Scenarios tabs and work there. It is also fairly easy to graphically see what Scenarios are active on a particular SOP date.

However, this mental model does not suit the Master Chart’s system model and could be misleading. Just like component dimension is a property of each signal, the start and end date are, strictly speaking, also properties of each signal. In the Master Chart, it is the individual signals that get activated and deactivated overtime, not the whole Scenario.

Removing the date dimension from Scenario and attaching it directly to the signals have certain advantages. With this model, it can take any vehicle specification (consisting of its component configuration and date of production) to retrieve the signals and generate an MSC on demand. This is illustrated below:

![Master Chart](image)

Figure 42 - Illustration of selecting a SOP date and component configuration
In other words, a Scenario becomes merely a saved set of component configurations. The Function Owners may eventually be fluent enough drawing MSCs directly in the Master Chart without Scenario tabs at all. Whether or not they can do it successfully is one important question to answer in the testing later.

4.2.4. Signal Library

In the MSC Drawer, the signals are drawn by clicking the source instance lifeline and dragging the cursor to the destination instance lifeline. An arrow is then created, followed by a pop up box. The pop up box has drop down lists of registered signals and input fields for custom texts. Although this mechanism works, the pop up box hides important data one step further from the users, which is what the signals are to use. The signal library brings the data forward. It retrieves the list of source components, destination components, messages, and signals via stored procedures in the SesammDB. Then it shows them to the users in three list boxes.

The current use of the current auto-populate feature supports this idea. Auto-populating the MSC with all registered components and objects is not very useful because the signals are not ordered. Yet some FOs use the feature to get an overview of what signals and components to use. Soon afterwards, they delete everything again. The signal library (Figure 43) eliminates the need to do that.

The interaction mode of dragging mouse to create signal arrow from one component line to another is also prone to slips. During observation, Function Owners every now and then drew arrows from or to the wrong component. They realized their mistakes after seeing that the signal drop down lists do not have the signals they are looking for. The signal library list boxes allows user to select source and destination component from more accurately. Moreover, the list boxes also serve as filters: after source component is selected, only relevant signals and destination components are shown in the other list boxes, and the other way around.
Another important improvement is the list boxes can mark which signals have or have not been added to the MSCs. FOs can easily see the registered objects in the AEDs that they have not used in the MSC. Assuming AEDs and MSCs are drawn at the same functionality level, which is on Use Case level, this would ensure consistency between the two documents.

Lastly, the ability to directly draw in the chart is kept for its intuitiveness. One potential improvement to minimize the slips, for instance, is to add a snap to line feature.

### 4.2.5. Object Property

The Object Property box shows important information about the signals currently being drawn. It responds to the Signal Library to show ECU family, ECU code, signal’s FV ID, etc. of the selected components and signals. Some FOs could be unfamiliar some components and signals in their MSCs. This information box serves as the first step before the FOs decide whether to leave the application to look for more information or not.

### 4.2.6. Object Validity

The object validity box is a one way to input the start and end date of each MSC signal. To give the user a context, the start and end date of the component, message, and signals are shown. The user is then presented with a slider to adjust the MSC signal’s period of validity.
4.3. Usability Testing

In this study, a formative test and a comparison test are conducted to address two questions, respectively:

1. What are the usability problems in the new MSC visualization?
2. How is the new MSC visualization’s performance compared to the old format?

The participants are Function Owners who, unlike System Architects or System Integration Testers, are directly involved in the MSC drawing process. Function Owners with variants in their functions are particularly relevant.

4.3.1. Method

4.3.1.1. Formative Test

It is important to explore early on what problems the users may have in the new concept. To identify usability problems, the test takes form of a formative test[29] with has low degree of control, moderator’s intervention, and Think Aloud method.

The audio is transcribed and coded in categories / themes. The common themes are not pre-made and arise from the data itself[26], [30].

4.3.1.1.1. Participant

Three Function Owners from different departments are recruited.

4.3.1.1.2. Location and setup

The test takes place in a 6m² room with desktop PC and a 24 inch monitor. The two MSCs are printed side by side on a piece of paper, which is handed to the participant. The moderator sits behind the participant.

4.3.1.1.3. Task

To use the new tool to redraw two printed MSCs, which are variants of the same Use Case. Further detail of the task is available in Appendix 9.2.

4.3.1.1.4. Measure

Some interesting qualitative measures:

- How do they interpret different UI elements in the tool?
- Which part of the new tool can they grasp immediately, and which one need further instructions?
- What are the steps they take to draw an MSC?
- How easily can they navigate (drill up, down) across variants?
What results do users expect behind every action?
How well does the system match users expectations?
What kind of information are users interested in and how accessible are they in the tool?
What aspects are particularly confusing in this new approach of drawing?
What are the best and worst features?

4.3.1.1.5. Result

From the screen and audio recordings, common usability issues are identified, grouped, and sorted by their severity. The severity level is described below:

<table>
<thead>
<tr>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>The work cannot continue</td>
</tr>
<tr>
<td>High</td>
<td>The work is wrong</td>
</tr>
<tr>
<td>Medium</td>
<td>The work pauses</td>
</tr>
<tr>
<td>Low</td>
<td>The work isn’t affected but may improve if it’s fixed</td>
</tr>
</tbody>
</table>

**Figure 46 - Severity level**

Complete audio transcription is available in the Appendix 9.4. In a summary, one critical, two high, four medium, and four low issues are identified, as described below:

<table>
<thead>
<tr>
<th>ID</th>
<th>Category</th>
<th>Issue</th>
<th>Severity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.11</td>
<td>Signal library</td>
<td>The program crashes if the user does not deselect existing filters before adding a new signal</td>
<td>Critical</td>
</tr>
<tr>
<td>1.21</td>
<td>Global</td>
<td>The components in the signal library and in the chart are not in sync</td>
<td>High</td>
</tr>
<tr>
<td>1.22</td>
<td>Signal library</td>
<td>No possibility to register a new signal or add it as a free text</td>
<td>High</td>
</tr>
<tr>
<td>1.31</td>
<td>Signal library</td>
<td>The user forgets to deselect existing filters when trying to add a new signal</td>
<td>Medium</td>
</tr>
<tr>
<td>1.32</td>
<td>Master Chart</td>
<td>It is not obvious that the master chart’s top components are substitutable</td>
<td>Medium</td>
</tr>
<tr>
<td>1.33</td>
<td>Master Chart</td>
<td>The effect of deleting a global signal in the Scenario tab is not clear</td>
<td>Medium</td>
</tr>
<tr>
<td>1.34</td>
<td>Signal library</td>
<td>In case of gateways, it is not clear if the destination component is the gateway or the final component destination</td>
<td>Medium</td>
</tr>
<tr>
<td>1.41</td>
<td>Master Chart</td>
<td>The signal arrow shape is too thin to click or right click, e.g. when deleting</td>
<td>Low</td>
</tr>
<tr>
<td>1.42</td>
<td>Master Chart</td>
<td>Direct drawing in the chart is still expected</td>
<td>Low</td>
</tr>
<tr>
<td>1.43</td>
<td>Signal library</td>
<td>The user expects to be able to drag the signal from the signal library</td>
<td>Low</td>
</tr>
<tr>
<td>1.44</td>
<td>Global</td>
<td>Signal argument suggestion is desired</td>
<td>Low</td>
</tr>
</tbody>
</table>

**Figure 47 - Identified usability issues**

After all the tests were over there were few days to fix some of the identified issues above. However the time was not enough to fix everything before the comparison test began.
4.3.1.2. Comparison Test

The second test is a controlled comparison test between the old and the new design. Thus, unlike in the first test, in this test the Scenario tabs are not shown. The users must draw using only the Master Chart when drawing in the new tool.

This test uses within-subject design, in which the same group of participants use both the old MSC Drawer[42] and the new prototype to draw similar MSCs. Within-subjects design is chosen for its main advantages over between-subjects, which is eliminating variances that occur due to individual differences or selection effect[31]. It also has some practical benefits, such as requiring fewer participants[31] and efficiency when giving instructions only once for both conditions[31].

Although all participants draw using both tools, the order is counterbalanced [29] to minimize Sequence effect[31], which could, for instance, give the second condition a positive practice effect [31] or a negative frustration effect.

Contrary to the first test, in this test the moderator makes little to no intervention. A pre-test questionnaire is given to gather participant’s background information. A non-structured post-test interview is conducted to discuss about what happens during the test, their impressions, etc.

4.3.1.2.1. Participant

Six Function Owners are recruited from different work groups to make sure that their functions have different complexity level and to ensure representativeness.

To eliminate confounding effect due to Testing [31], where participants become proficient in using the tool due to previous exposure, no participant in this test has seen how the Master Chart tool work before.

4.3.1.2.2. Location and setup

The test takes place in a quiet, 6m² room with desktop PC and a 24 inch monitor. All instructions for training, tutorial, and tasks are printed on a set of papers and handed to the participants. The moderator sits behind the participant.

4.3.1.2.3. Task

A simple “training” about MSC is performed prior to the first task. The participants are presented with a simple MSC to modify with the old tool. This training minimizes the confounding effect of History[31] as some participants may have done MSC recently while the others probably haven’t done it in months. It improves control in the experiment[31] by taking the dependent variable (i.e. performance in drawing MSC) to a more equal level before the experiment begins. The first task is to use the old MSC Drawer to replicate a set of two MSCs, which are variants of a function. The MSCs are taken from real MSCs but they are simplified.
A simple tutorial is given prior to the second task. The tutorial introduces basic design elements of the Master Chart, for example how to add a signal. The second task is to use the Master Chart to replicate another set of two MSCs, which are also variants of a function, but different from the first task. The complexity of this set of MSCs is similar to the first set.

Three participants do the tasks in reverse order, i.e. drawing using the new prototype before the old MSC Drawer. In a summary, the order looks like:

Three participants
1. MSC Drawer Training
2. MSC Drawer to draw 2 MSCs
3. Master Chart tutorial
4. Master Chart to draw 2 MSCs

Other three participants
1. Master Chart tutorial
2. Master Chart to draw 2 MSCs
3. MSC Drawer Training
4. MSC Drawer to draw 2 MSCs

Further detail of the task is available in Appendix 9.3.

4.3.1.2.4. Measure

The main measure is number of completed task out of the two tasks presented. A task is considered incomplete if the participant cannot finish it within 15 minutes. The time it takes to complete each task is also measured although it is not directly comparable to each other.

4.3.1.2.5. Result

Pre-test questionnaire

<table>
<thead>
<tr>
<th>Question</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
</tr>
</thead>
<tbody>
<tr>
<td>How long have you worked in Scania?</td>
<td>5 years</td>
<td>4 1/2 years</td>
<td>3 years</td>
<td>4 years 3 months</td>
<td>16 years</td>
<td>1 year</td>
</tr>
<tr>
<td>How long have you been a Function Owner?</td>
<td>5 years</td>
<td>2 1/2 years</td>
<td>3 years</td>
<td>3 1/2 years</td>
<td>9 years</td>
<td>1 year</td>
</tr>
<tr>
<td>How many User Functions are you responsible for?</td>
<td>4 UFs</td>
<td>10 UFs</td>
<td>1 UF</td>
<td>7 UFs</td>
<td>2 UFs</td>
<td>2 UFs</td>
</tr>
<tr>
<td>How frequent do you work on MSC?</td>
<td>1x / year</td>
<td>1x / month</td>
<td>N/A</td>
<td>1x / 5 years</td>
<td>1x / 2 months</td>
<td></td>
</tr>
<tr>
<td>How long does it take to update an MSC?</td>
<td>2 hours</td>
<td>1/2 hour</td>
<td>2 hours</td>
<td>1/2 hour</td>
<td>N/A</td>
<td>2 hours</td>
</tr>
<tr>
<td>How long ago was the last time you did an MSC?</td>
<td>1 year</td>
<td>3 weeks</td>
<td>3 months</td>
<td>6 months</td>
<td>7 years</td>
<td>2 months</td>
</tr>
</tbody>
</table>

*Figure 48 - Pre-test questionnaire result*
**Task completion**

All six participants completed both tasks successfully within 15’ time limit. Five of them spend longer time drawing the MSCs in the Master Chart than in the MSC Drawer.

![Figure 49 - Task completion time with MSC Drawer and Master Chart](image)

**Post-test Interview**

In the post-test interview the users are free to express their impressions, confusions, etc. Complete test log is available in Appendix 9.5. Some issues from the first formative test resurfaced during this test. Additionally, there were three additional issues identified during the post-test interview, as described below:

<table>
<thead>
<tr>
<th>ID</th>
<th>Category</th>
<th>Issue</th>
<th>Severity</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.11</td>
<td>Signal library</td>
<td>The program crashes if the user does not deselect existing filters before adding a new signal</td>
<td>Critical</td>
<td>Resolved</td>
</tr>
<tr>
<td>1.21</td>
<td>Global</td>
<td>The components in the signal library and in the chart are not in sync</td>
<td>High</td>
<td>Resurfaced</td>
</tr>
<tr>
<td>1.22</td>
<td>Signal library</td>
<td>No possibility to register a new signal or add it as a free text</td>
<td>High</td>
<td>N/A</td>
</tr>
<tr>
<td>1.31</td>
<td>Signal library</td>
<td>The user forgets to deselect existing filters when trying to add a new signal</td>
<td>Medium</td>
<td>Resurfaced</td>
</tr>
<tr>
<td>1.32</td>
<td>Master Chart</td>
<td>It is not obvious that the master chart’s top components are substitutable</td>
<td>Medium</td>
<td>Solved</td>
</tr>
<tr>
<td>1.33</td>
<td>Master Chart</td>
<td>The effect of deleting a global signal in the Scenario tab is not clear</td>
<td>Medium</td>
<td>N/A</td>
</tr>
<tr>
<td>1.34</td>
<td>Signal library</td>
<td>In case of gateways, it is not clear if the destination component is the gateway or the final component destination</td>
<td>Medium</td>
<td>N/A</td>
</tr>
<tr>
<td>2.35</td>
<td>Signal library</td>
<td>It is not clear what filters are active in the Signal Library</td>
<td>Medium</td>
<td>New</td>
</tr>
<tr>
<td>1.41</td>
<td>Master Chart</td>
<td>The signal arrow shape is too thin to click or right click, e.g. when deleting</td>
<td>Low</td>
<td>Resurfaced</td>
</tr>
<tr>
<td>1.42</td>
<td>Master Chart</td>
<td>Direct drawing in the chart is still expected</td>
<td>Low</td>
<td>Resurfaced</td>
</tr>
<tr>
<td>1.43</td>
<td>Signal library</td>
<td>The user expects to be able to drag the signal from the signal library</td>
<td>Low</td>
<td>Solved</td>
</tr>
<tr>
<td>1.44</td>
<td>Global</td>
<td>Signal argument suggestion is desired</td>
<td>Low</td>
<td>Resurfaced</td>
</tr>
<tr>
<td>2.45</td>
<td>Signal library</td>
<td>There is no way to see what signals have not been added</td>
<td>Low</td>
<td>New</td>
</tr>
<tr>
<td>2.46</td>
<td>Master Chart</td>
<td>The grey signal names are not very readable</td>
<td>Low</td>
<td>New</td>
</tr>
</tbody>
</table>

*Figure 50 - Three new issues appeared. Some issues from the formative test have been solved or are not applicable in this test*
4.3.2. Analysis

User's Mental model of the Master Chart

The formative test does not offer any training about the Master Chart. Instead, it offers predefined Scenario tabs next to its tab. Naturally, as it resembles today’s drawing process, all participants used the Scenario tabs when drawing MSCs. Although they understood that the Master Chart is a consolidated view of the Scenarios, none of them actually used it actively drill up and down to a different abstraction level (Issue 1.32). This shows that despite their general understanding of what the Master Chart is, the ability to traverse across component dimension is still something foreign.

Furthermore, if a signal could be created and deleted on any level, what happens if it is created on a higher level and then deleted on a lower level (Issue 1.33)? This question is less critical if there is only one Master Chart view. However, when working with multiple Scenario tabs, the answer may be confusing. Suppose there is a signal from COO7 to all GMS drawn in the Master Chart. The signal is then deleted in one of the Scenario tabs, which is applicable to GMS = A5. Would it still exist in the other Scenario tabs, e.g. GMS = ZF? This is a very important piece in the user’s mental model. To be transparent and consistent, for this use case the system must be improved with a confirmation dialog, a status bar, or even extra visual components.

There are also lower priority issues to fix in the Master Chart. The signal arrow’s body is now too small and difficult to target with a mouse cursor (Issue 1.41). A bigger transparent shape could be added as its container to solve this problem. The grey signals have been reported to be not very readable (Issue 2.46) so the color should be adjusted. As for the drawing method, a few participants have reported their willingness to use both the signal library and direct drawing method (Issue 1.42). Indeed, current drag and drop interaction to draw signals is very intuitive. It would be beneficial for the tool’s wide adoption to allow such direct drawing method on top of the signal library.

The formative test has brought to light the major usability issues in the Master Chart has. This is also shown by the fact that only three new, medium to low issues were uncovered in the comparison test. Furthermore, the findings in the formative test also served as a foundation to build the Master Chart’s training materials. In the comparison test, Scenario tabs were taken away, forcing the users to draw solely using Master Chart. Yet after five minutes of training, despite its significantly different approach to today’s practices, the Master Chart were used by all participants successfully. The time it takes to complete the MSC varies. Five out of six participants finished their MSCs faster with the old MSC Drawer. Yet comparing the exact time is misleading. After all, Master Chart is optimized for functions with high number of variants, not to mention that the Function Owners have been working with MSC Drawer for years. The more important take-away from the test is that they understood the model good enough to be able to complete the tasks within reasonable amount of time.
Signal library vs direct drawing

Just like the Master Chart, the Signal Library is a new component in the MSC Drawing process. Yet its usage needs little introduction to the users. All Function Owners were able to interpret what it means in the beginning of the formative testing. However, removing the filters after successfully adding a signal turns out to be not as straightforward. Most users select a signal by selecting a source component, a destination component, and then the signal. To add the next signal, even if the user selects a different source component, the destination component and the signal from the previous actions remain selected. The illustration below shows the case:

![Image of Signal Library vs Direct Drawing](image)

This makes source-destination-signal combination that shouldn’t be possible and crashed the application (Issue 1.11). After the formative test this bug was fixed and the application no longer crashes. The filters, however, still have to be deselected by control clicking the list items. Most participants did not like having to select and deselect three list boxes every time. They often forgot to do deselect the filters after adding a signal (Issue 1.31). This is worsened by the visually unclear filters (Issue 2.35), as only one list box filter is blue and the other grey. Aside for making the highlight clearer, another possible solution is by resetting all filters automatically after each signal is added or after a new source component is selected.

One design element that has not been implemented was mentioned in the study as well. The library should be able to tell which signals have been added and which have not (Issue 2.45). This will ensure that no registered signals are left unused.

Obviously, the Signal Library testing is still simplified. In this prototype there is no way to register new signals or add any signal as free text (Issue 1.32). The Function Owners have expressed willingness to take that extra authority. It should be explored further how exactly this will look and how the System Architects can approve it. Furthermore, in this prototype some signals go through COO as a mere gateway and it is not yet clear how that is handled (Issue 1.34).
Master Chart and signal library integration

After all, the most important thing is not either the Master Chart or the Signal Library but the integration of them. Because it merely retrieves data from the SesammDB, the current Signal Library is no more than a collection of signals filterable by its source and destination. It can then be added to any component configuration in the Master Chart, regardless of what source and destination components are used as its filter. For example, consider this signal:

![Image](image.png)

**Figure 52 – DLN5_K.AutomaticNeutralRequest arrow from COO to GMS**

DLN5_K.AutomaticNeutralRequest signal has been added by filtering COO7 and OPC4v2 (a type of GMS) while the Master Chart is showing an abstract instance of all GMS. Was the signal added only to OPC4v2 or to all GMS? This case confused many participants both in the formative and comparison testing (1.21). Synchronizing the Signal Library and the Master Chart could solve this issue. For instance, if the current chart shows the abstract component GMS, only signals applicable to all GMS should be shown in the Signal Library. Validation or suggestion message could also be employed to suggest to the user if a specific signal that has just been added should be applied to other variants as well. The synergy between the visual components is the key to the usability of the whole tool.

Although it is a good starting point, the two tests of course cannot reveal all the usability issues that the Master Chart may have. More complex MSC with alt boxes, Actions, and other MSC components should be tested to assess the Master Chart’s scalability. Moreover, this round of testing focused more on the general concept of Master Chart and the ability to traverse along the component dimension. The date dimension still leaves room for testing, especially if it is no longer a property of each Scenario tab. Decoupling the SOP date from Scenario and attaching it as a property of individual MSC object requires a bigger change in both user’s mental model and the underlying data structure. Further testing should be conducted to see if the Master Chart design is still usable in that context.
5. Conclusion

The intuitiveness of MSC has brought it into wide adoption for documenting system behavior. Yet it is not completely free from limitations. As shown in other studies, some have attempted to introduce extensions to MSC to get around the problem. Similarly, this study attempts to enhance MSC visuals to incorporate additional dimensions.

The current MSC format has limited expressiveness for documenting groups of instances that behave similarly. Classes of instances have been suggested, for instance in LSC and Symbolic MSC, to abstract real instances having parallel signals or alternate execution paths. The idea of instance abstraction is taken in this project to build interactive visualization called Master Chart. The Master Chart allows its instances to be declared on flexible abstraction level. By interactively drilling up and down through component dimension, users are able to document signals on any level. The usability test showed that after few minutes of training, the users could draw traditional MSCs in the form of Master Chart.

On the other hand, approaching the date dimension is less straightforward. It can, for instance, stay as a property of the whole chart. The UI should then be able to open many MSCs at the same time (multimaps) or embed them in separate tabs, with a slider as date selector. A more radical approach is to assign the date property to each MSC object, e.g. in individual signals. However, this may require a separate view to visualize and select a date point, where all relevant signals are then retrieved to generate the MSC on demand.

The component and date dimensions are particularly relevant in Scania, where similar MSCs are produced to document a high number of vehicle function variants. Contextual inquiry and PACT framework revealed the company’s practices and uses of MSC. Then the Master Chart prototype is developed to integrate component and date dimensions into traditional MSC. The benefit of having the Master Chart, in Scania’s context, is to have group of MSCs that are less vulnerable to variability. In other settings, the Master Chart design may also be applicable if the component and date variants create many almost-identical MSCs.
6. Future Work

There are three important development items Scania may consider to do in the future in addition to this thesis work.

Integration of AED and MSC creation process.

Creating and maintaining two drawings for similar data model is not a sustainable process. Currently there is ongoing effort to develop a new generation of Sesamm Tool which will include both AED and MSC editing capability. The underlying data model is also being restructured. Therefore it is a good chance to ensure that the application present AEDs and MSCs synchronously by the same data source. For instance, tagging components and signals at Use Case level can produce basic AEDs and Master Charts. An AED editor is used probably only to reposition the objects as an MSC editor is to reorder the signals. Creating Master Charts will be less about drawing on a canvas but more about ordering registered objects, e.g. by the Signal Library, on different variant conditions.

Scenario only as a set of component configurations

Today’s Scenarios have Start and End SOP date properties (illustrated in Figure 39). If a Change Request on a SOP date introduce minor changes in one or two signals, a whole new Scenario and MSC are created. A potentially better model is to assign Start and End SOP date to individual signal or action item (see Figure 42). The AED viewer in Sesamm Tool is already capable of showing validity of each AE and FV based on the History tables in Sesamm DB. Similar functionality could be adopted for the Master Chart. This will be a step to have a common drawing in which SOP date is selected to produce list of signals and generate MSC on demand.

A clearer feedback system for more complex variants

Master Charts as a higher abstraction of MSCs naturally hide some details. For simple cases, drawing a Master Chart on different abstraction levels is not a problem, as shown in this project’s tests. However, on a more complex situation an extra feedback system may be needed. For instance, if a signal is assigned to a complex set of variants (e.g. with component X, without component Y, for a certain period of SOP dates only) what does the system show when the user hover over it (see Figure 35)? Furthermore, if a signal is created on a higher abstraction level (for all family Z) and then deleted on a more concrete level (the user switches to Z1 and deletes the signal), it should be obvious whether or not the signal will still exist for Z2, Z3, and so on. This may require additional system feedback, such as message boxes.
7. References


[34] Scania, “Vägledning för Message Sequence Charts.” [internal]. Available:
  \%guran\Archive\REST\REST15007_bilaqa1.docx.
  \%guran\Archive\REST\REST10019.doc.
  \%guran\Archive\REVRESA\RESA08011.docx.
[37] Scania, "Instruktion för funktionsägares arbete med MSC:er.” [internal]. Available:
  http://wiki.inline.scania.com/wiki/Instruktion_f%C3%B6r_funktion%C3%A4garen_med_MSC\:er.
[38] Scania, "Guidelines and rules for UC, SCN and MSC documentation.” [internal]. Available:
  \%guran\Archive\REST\REST12002.docx.
  \GLOBAL\DFS04\05_General\ECU_systemkurs\Kurser2015\Dag 1 Systemutveckling
  Arkitektur, Process och Teknik\6 Funktionsutveckling_2012 09 11.ppt
[40] Scania, "Elsystem i Kurser 2015.” [internal]. Available
  \GLOBAL\DFS04\05_General\ECU_systemkurs\Kurser2015\Dag 1 Systemutveckling
  Arkitektur, Process och Teknik\5 Elsystem process intro.pptx
[41] Scania, "Andringshantering EXAMM.” [internal]. Available:
  http://wiki.inline.scania.com/wiki/%C3%B4ndringsshantering_EXAMM
[42] Scania, "MSC Drawer.” [internal]. Available:
## 8. Table of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Four stages of visualization</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>GA to GB overview and its detail view equivalent</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>“Multimaps“ for showing diagrams in different date dimension</td>
<td>12</td>
</tr>
<tr>
<td>4</td>
<td>Controlled animation with discrete time steps</td>
<td>13</td>
</tr>
<tr>
<td>5</td>
<td>Basic MSC</td>
<td>14</td>
</tr>
<tr>
<td>6</td>
<td>Inline expression symbol</td>
<td>14</td>
</tr>
<tr>
<td>7</td>
<td>One instance from the server and 3 instances from the client</td>
<td>15</td>
</tr>
<tr>
<td>8</td>
<td>An MSC and its symbolic equivalent</td>
<td>16</td>
</tr>
<tr>
<td>9</td>
<td>Comparison between different types of usability testing</td>
<td>20</td>
</tr>
<tr>
<td>10</td>
<td>Clustering the affinity diagram from interviews with FO, SA, and ST</td>
<td>24</td>
</tr>
<tr>
<td>11</td>
<td>Summary of the three Function Owners’ personas</td>
<td>27</td>
</tr>
<tr>
<td>12</td>
<td>Function Owner distribution inside R&amp;D’s groups</td>
<td>28</td>
</tr>
<tr>
<td>13</td>
<td>User Functions, Use Case, and Scenario distribution across groups. The four letters are group codes. The number inside the bracket are the number of FOs under that group</td>
<td>29</td>
</tr>
<tr>
<td>14</td>
<td>Official workflow for creating MSCs</td>
<td>30</td>
</tr>
<tr>
<td>15</td>
<td>User Function may have several Use Cases</td>
<td>31</td>
</tr>
<tr>
<td>16</td>
<td>Use Case may have several Scenarios</td>
<td>32</td>
</tr>
<tr>
<td>17</td>
<td>Variant structure at Scania</td>
<td>32</td>
</tr>
<tr>
<td>18</td>
<td>Most User Functions have one AED. Major change in vehicle architecture (e.g. NGS vs NCG) may give a User Function another AED</td>
<td>33</td>
</tr>
<tr>
<td>19</td>
<td>AED for UF499 – Load Transfer (sanitized). Some AEs, represented by yellow boxes, are actually variant to each other</td>
<td>33</td>
</tr>
<tr>
<td>20</td>
<td>Each SCN has exactly one MSC</td>
<td>34</td>
</tr>
<tr>
<td>21</td>
<td>Each MSC, which is a drawing of its SCN, is specific to a point in variant and SOP date dimension</td>
<td>34</td>
</tr>
<tr>
<td>22</td>
<td>Relationship between Sesamm database, Sesamm Tool, and MSC Drawer</td>
<td>35</td>
</tr>
<tr>
<td>23</td>
<td>Part of the SesammDB data model</td>
<td>36</td>
</tr>
<tr>
<td>24</td>
<td>SesammTool</td>
<td>36</td>
</tr>
<tr>
<td>25</td>
<td>MSC Drawer</td>
<td>37</td>
</tr>
<tr>
<td>26</td>
<td>MSC Drawer: Open MSC</td>
<td>37</td>
</tr>
<tr>
<td>27</td>
<td>Drawing a signal</td>
<td>38</td>
</tr>
<tr>
<td>28</td>
<td>Main User Interface in wireframe</td>
<td>39</td>
</tr>
<tr>
<td>29</td>
<td>Selecting a Use Case</td>
<td>40</td>
</tr>
<tr>
<td>30</td>
<td>Master Chart</td>
<td>40</td>
</tr>
<tr>
<td>31</td>
<td>Switching to component from the same ECU family</td>
<td>41</td>
</tr>
<tr>
<td>32</td>
<td>The ECU tree is configurable in the application</td>
<td>41</td>
</tr>
</tbody>
</table>
Figure 33 - The three top signals are drawn for all GMS. The last signal is drawn only for A5...

Figure 34 - On a higher level, variant-specific signals are greyed.................................

Figure 35 - Hovering over a signal reveals its configuration information......................

Figure 36 - Going to a signal’s configuration from the context menu..........................

Figure 37 - The start and end SOP date is written in the SCN info detail tab.............

Figure 38 - The same SCN validity data, visualized.................................................

Figure 39 - Four scenarios, with different validity period........................................

Figure 40 - Scenarios shown in separate tabs...........................................................

Figure 41 - The tab headers respond to the SOP date slider...................................

Figure 42 - Illustration of selecting a SOP date and component configuration..........

Figure 43 - Signal Library.........................................................................................

Figure 44 - Object property.......................................................................................

Figure 45 - Illustration of the Object Validity box.................................................

Figure 46 - Severity level........................................................................................

Figure 47 - Identified usability issues......................................................................

Figure 48 - Pre-test questionnaire result.................................................................

Figure 49 - Task completion time with MSC Drawer and Master Chart..................

Figure 50 - Three new issues appeared. Some issues from the formative test have been solved or are not applicable in this test..................................................

Figure 51 - After adding DW.AutomaticNeutralRequest from BC11 to A5, COO7 can be selected........................................................................................................

Figure 52 - DLN5_K.AutomaticNeutralRequest arrow from COO to GMS.............
9. Appendices

9.1. Interview questions

Introduction

- Self and project introduction
- User’s background and role
- User’s typical workday

General questions

- How frequent do you work with AEDs and MSCs?
- What do you do with them?
- When and how (e.g. continuous, sporadic) do you do it?
- What tools do you use? What for?
- What keeps you from spending more time in it?
- Please walk me through typical or the most recent AEDs and MSCs work
- What are today’s biggest problems?
- What are the best things about the tools and processes today? What works well?
- What should this project try to accomplish?
- What worries you about this project?
- How many User Functions are you responsible for?
- How familiar are you with your User Functions?
- How much do you know about AE?
- How do you keep track of changes in MSC?
- How consistent is the information between one document and another (e.g. AED-MSC, FAD-AED)
- How much do you know about variants, compared to Function Owners? (for non-Function Owners)
- What does the current system architecture look like? (for System Architects)
- Are there any technology constraint in this project? (for System Architects)
- Is there anyone not in my list that you think I should talk to?
- What is the best way to contact you in the future?
9.2. Formative test task

Two MSC to draw during the Formative test (sanitized)

9.3. Comparison test task

9.3.1. MSC Drawer Training

In the taskbar, you can find two MSC drawer applications containing blank MSCs.
Use them to replicate the MSCs in the next page.
You can leave all signal arguments blank.

(You may flip this page when you’re ready)

An MSC is currently open in the MSC drawer.
Add the highlighted two signals at the bottom of the MSC.

(You may close the MSC drawer and flip this page when you finish)
9.3.2. MSC Drawer to draw 2 MSC

In the taskbar, you can find two MSC drawer applications containing blank MSCs. Use them to replicate the MSCs in the next page. You can leave all signal arguments blank. (You may flip this page when you’re ready)

9.3.3. Master Chart tutorial

Introducing Master Chart

Launch "Tutorial" application from desktop and follow the basic tutorial in the following pages (You may flip this page when you’re ready)
The filters remain active until they are deselected. Deselect filters by CTRL + Click.

Right click on a component to switch it into another component from the same family.

The signal you add will be applied to a variant according to the top component configuration.

Try to add the signal above.

Switch into TMS1 and add a signal there.
Switch back to GMS and observe the result. You just created an equivalent of two MSC:
one for OPC4v2 and one for TMS1!

Right clicking a signal allows you to delete a signal or go to its specific variant

Dragging a signal allow you to rearrange its position

You have reached the end of the Master Chart basic tutorial

You may spend some time to try it around.
Close the application and flip this page when you’re ready
9.3.4. Master Chart to draw 2 MSCs

Launch “Task” application from desktop to open a blank Master Chart.
Use it to replicate dummy MSCs shown in the next page.
You can leave all signal arguments blank.

(You may flip this page when you’re ready)
9.4. Formative test transcription

9.4.1. Formative Test P01 Transcript

This is the software right now. Can you tell me what you think you see, how do they look to you, and what do you think they are for before we begin?

I see the ECUs you want me to use maybe, and.. can I click?

Yes.

(clicking on filters) OK now I see the signals related to the ECUs.. and probably I can see where they go to..

Yes. OK. And here are the different scenarios (clicking on tabs)

OK, you got that part right. I printed this MSC. This is from Automatic Neutral user function. If you could try to draw the same diagram.

OK.. what happens if I... (clicking filters.. crashed)

In this case we will have to deselect the filters first. For example if we have chosen something like this, we need to deselect them back by control clicking. (demonstrating control click)

Ctrl click?

Yes.

OK I will try to remember that... (trying to add signal but without filtering destination)

And also.. I will also fix this but right now we will have to select all three columns to add the signal.

OK (starts adding signal)

(Added EMS to COO signal) That should actually go to BCI. This (COO) is just a gateway.

When you draw GW do you draw it all the way to destination or?

In the old software you draw directly. And then it added gateway (continue adding COO to BWE signal) (hovering over signal name trying to see full signal name because it is trimmed on the chart) I don’t know if it is the same signal, I guess so.

I think they are the same.

I need to mark all the way to get the signal? (marking all the filters) I can’t just get a signal from here (the signal column)?

Now you can’t.

That would save some time.

I will note that.

(continue filtering signal to look for signal) Oh I am in the Master Chart! I shouldn’t be in the Master Chart Can I change now?

Yes.

OK let’s do this. (filtering source and destination) (hovering over signal name in the in message signal list) I can’t see the signal name fully. Maybe you should at least have a tooltip or something (in the library columns) (Can’t find a signal) It should be to OPC4 in the list here.

In the source or?

In the destination.

Which signal are you looking for?

Maybe I am in the wrong diagram (changing tab, expecting library filters to change if he changes tab) Now I’m in the right tab.
Are you looking for this signal from COO?

Oh it’s from the COO!
(double clicking argument) How do I.. can I just add it (the signal's argument) here?

I haven't really implemented it, but how would you expect to add the arguments.

If you want to improve the old one I’d like a dropdown list to choose the valid (arguments) or if it’s a number value, a possibility to write something. But those signals that have fixed values they should be dropdown.
(start typing argument) Do you think it will crash? (laugh) I’ve made one so I've shown you I can do it.

Yes you can ignore the other arguments.

Now let’s do this (changed to A5 tab). So now something is missing... (observing the A5 tab)
I'm just figuring this stuff out now (changing tabs and then back to A5) (observing the A5 tab)
So from this one it goes directly to... directly from BCI to A5, that is not possible since you need to gateway through the coordinator. Oh, direct wire! Oh now that works. (laugh)

Oh, the expertise.

Hmm this shouldn't be in the Master Chart. (Continue switching tabs and looking for signal)

What are you trying to do?

hmm I'm just looking. Let's do this.. hmm BCI (clicking on BCI1, looking for signal but destination filter is still active)

You will have to deselect the filter first.

deselecting destination) BCI to A5.. (the signal is missing in this test)

Let's just ignore that (signal). I think I forgot to add that.

This shouldn't be in the database. Can you see the direct wire in the AE diagram?
I think so but I can check. How is it currently?

We have them in the MSC but now we just draw them and if it’s not in the list we only write it DW.

Can it be in the list?

No, I don't think so.

OK.

Oh I can read it. (hitting arrow keys to read signal name in the chart)
So I can't do anything else with this. But this one (BWE to COO signal) shouldn't be here.

Would you like to delete it?

If I delete it here (one of the scenario tab) I will delete this too (from Master Chart), or?
Umm because with the OPC you actually send the CAN message from BCI but do you actually.. I don't think you do it with the A5. Since you have a direct wire for this function, maybe it sends the message but not to get the neutral.

Can you go back to the Master Chart. Since this signal....

It's grey.

Why do you think its grey?

Because it only applies to one of the scenario. Bus isn't it in both? (Switching to A5 tab) This is A5 and you have it here.

Yes but that doesn't exist in the OPC I think.

(Switched to OPC tab) Then I put it in the wrong one.

Shouldn't it be there?
Probably should because you have CAM message from the COO, and it’s the BCI that’s..

Can you try to fix this?

Oh it’s a trap! *(laugh)* Hmm can I do something.. I can’t do anything with this because the list is already.. (the signal is missing in this test)

Why do you think you can’t do it with the list?

I don’t have the signal in the list. I don’t have from.. Oh. *(crashed the program again)*

Ooh. OK lets end that there. Can I get back to the screen then.. Ah it’s okay, I recorded the screen

That’s an interesting...

It should’ve been in this scenario with OPC4. The CAN message between COO and BCI for Automatic Neutral. It goes to COO and then the gearbox. But now it’s in the A5 diagram. (He seemed to have drawn an incorrect MSC and realized it)

I see. What is your general impression? I know there’s still a lot to fix.

How do I fix that without using the signal in the list?

That’s a good question...

Can I use free text?

Not right now. Right now you’re limited to the list. But it’s possible to delete and add something. For example.. how do you normally delete a signal now?

I think I.. right click maybe. I need to test it in the real environment then I will remember it.

You can actually right click and delete *(shows right click, delete)*

So it worked before it was wrong in the end.

Yes, you seem to grasp the general idea of having Master Chart and scenarios. May I know why you.. if there were no two tabs here and you can only work with Master Chart, do you think you will work with that and then switching components *(demonstrate how to switch comps in the Master Chart)*

Hmnm. Can you add ECUs in that list or is it fixed? Because in the Master Chart you probably want.. Oh it’s TMS so.. OK. OK then I understand. Then I don’t want to.. in this case. So..

So which way would you prefer to work, on separate tabs or in the Master Chart?

Maybe I’d just use Master Chart if it’s possible but I don’t know if it is. Why is this (a signal in the Master Chart) grey?

Because I added that to A5. If you try to right click and go to signal configuration...

*(Right click and go to signal configuration)*

I added that signal only on A5.

*(Trying to add signal from OPC4v2 filtered library into a Master Chart with A5)* So you can add this one too. It’s here (in scenario A5) Actually I chose OPC (in the library)

Exactly. Now it is only filter to the signal but where its applied to is according to what is active on top.

So if I.. delete that one and *(deletes a signal in scenario OPC) (adds a new signal) (observing result on Master chart level)* (There were two grey signals in Master Chart, one for A5 and one for OPC)

What do you think about this way of drawing? I can still change it.. they way to draw this.

*(Thinking..) I’m not sure if this way is faster than the old way, but that’s not maybe the goal with the whole thing. Maybe it’s more about getting all the scenarios.

Both actually. At least we don’t want it to be slower.

In the old one you just draw line (direct drawing) and then you get something like this.

Let’s say I implement that so you can also draw in here.
Then you get the old one (*laugh*) I don’t know, maybe because I’m not used to working with this one. Maybe if you’re used to it you will get the feel of it.

The idea.. the main difference here is that we have a Master Chart (*Clicking on Master Chart tab*) and the scenarios here are using the same data. They are synchronized instead of separate drawings, but I don’t know if that is understandable or?

In that case if you have 5 scenarios this probably would be faster because you don’t need to draw 5 different MSC. You saw how good it was with the old program when we wanted to copy something.

I’d like to implement the direct drawing here. Probably some will use it, don’t you think?

I think.. maybe. Let’s see what the others say. How many will test this?

I have 2 more today but next week I’m approaching more people.. maybe 8.

It feels faster to just uh.. draw arrows and fill in the gaps, when it works, because it doesn’t work all the time.

OK. Anything else you want to tell me?

No, not now. So in this case.. (inside scenario A5, with library filter OPC4 active) when I don’t have OPC4 maybe this list should react by taking away the ECUs that I don’t use, since that is not in the SCN.

That’s a good point.

Do you get the ECUs automatically, depending on..

SesammDB?

Yes?

Yes.

OK. It’s okay otherwise.. Maybe when we have the whole list of signals (signal column in library), because nothing is chosen maybe we should be able to see where the signal goes. And if it is valid for both gearboxes it should show both possibilities... signal and destination. And if I choose this one (one specific signal), it will automatically choose the other one (the other filter columns) because when I know which signal I want to use I don’t want to do this (*clicking all filters*) But.. maybe that doesn’t work either.. How do I get.. Oh, now I know (playing with filters) This is interesting.. What have I done (*selecting and deselecting signals*) You see my problem?

That the signals don’t come back?

(*Deselecting everything) (finally got all signals back)

Yea it wasn’t clear what was selected.

I can actually see the source here in the name but..

How helpful do you think the filter is. What if we can just list all the signals.

O it depends on the MSC and the Function. This is quite a simple MSC. So you can’t compare it with bus stop brake for instance.

Any last suggestions?

In a perfect world you don’t need a free hand to write the signal names and so on. But in reality maybe you should have a possibility to, as it is in the old one because.. or maybe not, because if something is wrong in this list then something is wrong in the ae diagram.

There’s been suggestion that the FO may have more authority to add new signal. What do you think about that?

Will that work in Scania where everybody should approve everything? Maybe it should only be.. there should be a good way to communicate with the one responsible for the systemization that this signal should be added.. I don’t know.

OK thank you. That is all.
9.4.2. Formative Test P02 Transcript

I see what I think is the library of the message and signal involved in this function from Sesamm database. The property and validity I don’t know what it is.

You can operate it.

Yes. so I should draw this one. Should I draw it in this in this exact order?

It depends on what you want.

Yes because if I have been drawing this I’d have drawn it in another way.

Feel free to do it your way then.

Ah but I can do it just like this.

What I’d normally do, because I’m used to MSC drawer, is this (drags n drops) but I presume it is done in another way (laugh after the software doesn’t respond to his actions)

So I will just take the parking brake applied (clicking on the signal name in the signal column)

(clicking on the add to diagram button) (no result)

Today you’d have to select all three columns.

Oh. (clicks all three columns and add signal button) (types argument)

How would you expect to enter an argument? Can you think of your way of adding argument?

I usually sort of copy paste it from Sesamm tool, but if it’s possible to choose from a dropdown that’d be great. But I don’t think that’s stored in AED today. Does that answer your question?

Yes, thank you.

How do I go back? (want to get another signal)

This time you’ll have to deselect everything again by control clicking them.

OK OK. (continues to add signal)

Is there another way to add the argument?

Not right now.

OK. (continues to add signal)

(copy pasting arguments across signals) (thinking)

(clicks BCI1) Adding BCI to the... hmm here I have the External control message just sent to.. do you have special feature when it’s not just gated? Because this is what you call bridge gate repackage.

How do you do it in the current MSC drawer?

In the current MSC drawer the tool sometimes understand the signal is from here to there, so I can draw BCI to COO and then I added GW block explaining it’s a bridge gate repackage. And it added the bridge repackage. Sometimes it doesn’t, depends on how it is drawn in the Sesamm tool sometime I do it by free hand.

Can you draw straight to the end destination?

No, that’d have to be a simple gateway. If it is pure gateway then it can add automatically. But when it is added and I edit the first one the second one doesn’t update. So it’s just auto paste in the beginning, it doesn't support the gateway feature in the tool.

Here if I mark the BCI I only get A5 and not the OPC. (a signal is missing in this test)

Oh that’s another mistake. I’ll take a note and you can just ignore that signal.

Also Here you have added DW that would be a nice feature but I don’t think that is possible in the Sesamm tool today.

Right.

(Continues drawing) Yes. (finished)
And.. the second one?

The second one? OK... now I have no idea (realizes he's in Master Chart) So I've only made the Master Chart now, if I go here (switching tabs) and its already automatically filled that in (the other scenario tabs) Can I just add stuff here now? (the A5 tab)

Yes.

And that would be BCI.. (Adding BCI to A5 signal) (Typing argument) Then I’d try to move this perhaps but I don’t know.. (want to reposition a signal)

Try to drag it without control.

Ah just like that. (Finished) So.. did I miss something or?

No. Oh that's quick.

Oh that was very nice.

Well you get this quickly maybe you understood the concept already? If you can choose drawing in the Master Chart or in the tabs, which way would you prefer to do?

The unique signal its very nice to do in the actual scenario because I can see what’s specific to that scenario to see what is used there. On the other hand you have filtered it out but it should be possible to see in the Master Chart to see that this signal goes in the variant and the other does not.

Is it possible to do it (drawing signals to specific variant) in the Master Chart? (Adding BCI to A5 signal in the GMS level Master Chart)

Hmm but if I have wide range of gearbox, how do I draw for two but not all of them?

There are 2 things here. First is the (library) list. Even though we can choose OPC or A5 that’s only a filter to help you choose the signal. Which configuration that signal applies to, that depends on the configurations in the chart. That’s how it is now. To change it (the configuration in the chart) you can do.. (right clicking GMS)

Oh! I didn’t try that. I guess I’m biased with the MSC drawer today.

That’s interesting because that’s one of the main questions I’m looking to answer. In the future we can get rid of the scenarios (demonstrating drill up and down)

That’s a very nice feature. I don’t think you need the scenario then. So when I draw a DW between the BWE and GMS how does tool know that is not connected there but its connected there (applies to one variant and not the other) Is it defined by the way I draw it when its filtered like this?

Now it’s defined by what is shown on the top when you add the signal but I’m looking for a way to do it the future. It’s probably validated with Sesamm tool.

Oh ok if I choose GMS (in the Master Chart) even if I filter only A5 and I added it will be added to all?

With the current tool, yes. What do you think? Right now the library is only a filter.

Oh OK... when I first look at it I was under the impression that I was drawing for exactly that signal. But now when you explain it I get the thought behind it, that it’s just a filter and this is what you actually doing (point at chart)

On the other hand I want to be able to do anything so it would be nice to have this feature if you have a signal that you want to add to all signal... it would be nice if you could do it to all GMS in the same time and not add multiple times.

But even though now I only do this (simple test) I think it is very easy to understand. (Thinking) Can I ask you a question? If I’d drawn this. Say we skip the scenario tabs. how would I know... should I just draw the Master Chart and then export it to different scenarios or how do I actually control if the actual scenario is correct?

In the future we will not be so scenario based. Let’s say we have configured this into a certain combinations like this (Modifying top comp). This is a scenario. Scenario is now just combination saved. Then you can just switch between them.

Oh so you save the special view of the master chart as scenario?

Yes, then you can open it as separate tab or window. That’s for future work where scenario is just a configuration saved.

So you don’t have database of scenario then. So you just have the master chart and when you update master chart every scenario is updated? That’s nice.
Of course the question is how you navigate.. Now I’m exploring how the Function Owner will be able to navigate around this signal collection. If you can choose between adding a signal from here (direct drawing) and library, which way would you do it?

I think I like this feature with adding signal from the side because I personally get better overview of what is possible. Because even though I know the function I don’t have everything in my mind, that this signal goes from here to here. Here I can see signals from and to each component instead of trying. Then again I’m used to drag and drop but maybe that’s because I’ve done it so many in MSC drawer but I think this is a... I think I will work quicker with this.

Is there any feature in here that you think probably doesn’t work?

No.. I was thinking about it but it’s hard.. you always come up with special cases when you’re doing the bigger ones. But I haven’t thought of anything right now that makes it not work (trying to click around)

If I have this wire to 2 out of 3 (components in the same family), how would this feature then work?

Right. What I have in mind is first of all this should be configurable by the Function Owners with a separate popup window, where you can customize and subgroup them (right clicking components)

Oh that’s nice. I’m impressed. I think it’s pretty nice.

Thank you.. this is still far from the actual thing though.

I think you heading in the right direction.. is it just one signal at a time or can you.. (trying to multicolonk)

I was thinking of this auto-populate feature, I don’t use it but I was curious if you can add several signals at the same time. (thinking)

One thing.. you said it’s not the scope of your work but when we come to optional signals depending on parameter settings etc. or depending on if you do this or that, how is that handled? You know like the alt boxes they have today. In our ECU very much of the function is optional. So when you come to a certain line, depending on what they have chosen this could or couldn’t happen. Is there a tag or something where you can say this is or isn’t optional?

I don’t know if that’s just my problem because I have a lot of that in my function. We have a lot of options even if I have one scenario with one specified configuration I still have combination of parameters, like 50, so I have a lot of those alt boxes.

What do you think of today’s way of handling them with the alt boxes?

Today we have a problem because we use alt box both for component and parameter variants. So they.. I mean if you have done it you should be doing it if it’s allowed to have alt inside alt. It will be very complicated so I have to specify in text that this is an alt depending on comps.. or parameters etc. So now I have many alt boxes. It’s not optimal. But in this (Master Chart solution) I get rid of variation alt boxes then I can use something similar to just, ah, this signal block right here is optional, it doesn’t have to happen.

Do you understand what I’m thinking?

Yes I didn’t really have it in my scope, so I’m thinking of still implementing the other objects and we will still do it that way with the alt boxes. Although I understand the main purpose is to see the overview first and with the alt boxes it’s difficult to see what actually takes place.

Yeah the biggest one of my MSC is hard to follow because almost every signal is in alt box. But that’s the nature of the function, you can configure it in so many ways. Now parameter is not part of systemization. RESA (architects) has no ideas about the parameters.

A nice feature for me would be a tag in the future, not now, maybe like you’re sorting or exporting scenarios depending on FPCs. We should be doing it by the parameter, in the future depending on the SOP date you can sort out what MSC is used. But now I’m probably talking too far in the future...

Yes. I’m trying to tackle another variant here which is the SOP date but I haven’t got far. For example I can select one signal and then it will show here (in the validity box) when until when does this signal apply to, and another signal from when to when. But then how do we get the overview again? (presenting one idea on the white board)

That’s very nice.. often the MSC is used by the workshop too. The info in Sesamm tool is stored in Sesamm tool but it’s hard to find, for example I know this signal is used but when does this change? But I think for me the most use of this timeline is if I could just choose a SOP and I directly see which signals are used for that SOP rather than when is this signal valid.
9.4.3. Formative Test P03 Transcript

Before we begin can you tell me what you see there and what you think they are?

I see number of different components. BWE and GMS and EMS. I see library of different sources: gearboxes, BCI, engine, coordinator, and as well as destinations for different components, messages, and signals. Umm yeah.. that’s about it.

OK you can also operate it now. This is the MSC I would like you to draw. (hand in a printed MSC)

So you want me to just replicate this in the program?

Yes, exactly.

OK then perhaps something like this. I’m choosing a source, a destination, and a signal (clicking on the library column one by one). I see in the properties (looks at the property box). I’m trying to get it to the diagram (trying to drag but failed)... I didn’t see the button at first.

I can change the argument in the MSC but not before I added the signal (highlighting the argument in the chart) OK so next signal is cruise control vehicle speed from COO to EMS (looking for destination EMS but can’t find it)

In this case we will have to deselect the filters first by control clicking.. (the program crashes)

Oops. one of the perks with this now is that after every signal we have to deselect them all and then select them again. (demonstrates ctrl click)

OK I understand. So in the next signal it goes from COO to EMS.. I don’t know if you can select both of them (trying to multicolor both S6 and S8 but it is not possible). No. (Added one signal for S6) Only one of the EMS at a time. If I choose the other signal (selected S8 and added a the signal) There’s a new signal and it doesn’t look the same with the MSC.

Next signal is from EMS to BCI, CruiseControlVehicleSpeed... again I’d just do that (added messages twice, for both S6 and S8) if I added the signal once again.. Can I delete it? (right click, delete) Yes by right clicking and deleting it.

Then I have signal ExternalControlMessage from BCI to COO.. (added the signal) added. And the message DLN5 from COO to GMS.. and finally GMS to ENV CurrentGear signal (added the signals) And then we change the arguments according to the sketch (typing arguments)

*He never changes the active configuration

It’s okay to leave all the other arguments.

OK. So then I’m satisfied with the result.. perhaps (laugh)

Then I have.. there are 2 different scenarios containing different gearboxes in this case (switching between tabs, scrolling up and down)

They do look the same signal wise.. and that perhaps shouldn’t because.. Can I delete the signal from this only, perhaps? (right clicked, deleted one signal inside A5 tab) Which one am I supposed to delete? (trying to delete some signals but somehow the delete stops functioning)

oh the delete doesn’t work?

no, at least when I’m here in the scenario...

All right, that’s fast.. (cutting) and very good. OK I can give you some intro now. There are actually 2 different ways to draw the diagrams. First we can draw in the tabs twice like its it today or in the master chart like what you’ve done. I should also have mentioned that in the master chart it is possible to manipulate components (show how to switch component and add specific signals) Do you think you’ll use individual charts?

I’ll use master chart but I’d prefer instead of using the different components in the top view, I’d want perhaps want to, you know, like to add some arguments to the signal. Because if I choose the A5 here or the OPC variant (in the top components) the signal is hidden and I think I should be able to see it in the master chart always. I’m not sure but.. because if I choose one of the variants then I could as well use the diff tabs. My point is.. in the master chart.. I should be able to see all signals all the time in the master chart.

There are some views to eliminate scenario in the future.. so we’ll only have master chart and then draw there and if we happen to have a different variant this time, it will just be applied there. What do you think about that, if we only have one master chart?

I think It’s good to see the different variants, the complete different variants, you know, so.. I like the tabs actually to see how the scenarios actually looks like when you sort irrelevant things away. So I like the tab view actually. Perhaps you shouldn’t be able to change anything in tab views but at least have it like a visual representation of the result. At least as long as the scenarios are defined the way they are now.
Hmm that’s one way to do it, certainly. Let’s say we have a signal.. a general signal like from COO to EMS for example, regardless of the GMS. Let’s say we go down from here (switching GMS to A5 in the master chart) and delete it here. Would you still expect it to appear in the OPC scenario?

No... I don’t think so. But I’m not sure actually. Since it’s not, you know, connected to the GMS component, I’m not sure but I think it shouldn’t be affected by.. hard question. I think I’d prefer if you add those types of conditions to the signal itself rather than to change another component (hovering on the top GMS component) because in this case if the signal isn’t connected to that component, it can get quite complicated if you first choose A5 here and S6 here, and you do a signal and then you change to S8 and OPC and delete it there.. it can be quite confusing. So I prefer if you can add conditions to the signal itself.

Would that look like similar to today’s MSC with alt box for variants?

Yes, precisely.

Hmm interesting. Another thing I have in mind is this filters for example. It’s intended to help to get the right signal. However there could be difference between the top components and the filter columns, e.g. I can choose OPC (on the top components) while in the library I choose A5.

I think this should just be a visual representation of the chart and shouldn’t be able to change. But if you were to change it you should only be change the visible components, because it will be confusing if you would add the signal but not see the result. You’d be confused and wondering how and when it took place so I would prefer that you can only add signals with the components you’re actually seeing.

If you can draw signal both here (by directly drawing arrows) and here (adding from the signal library), how would you draw your MSC?

Good question. I’m used to the current implementation where you draw in the chart. I would probably use both methods actually because I don’t have any problems with the current solution, so I’d probably use both.

One of the ideas why this appear here is to.. in the future we probably can show to function owners what signals have been added and what haven’t in order to synchronize it better with the AED.

What about the arguments, How’d you add them?

I’d like the arguments to take form in the.. you should have the variable.. It’s one of the most used argument that’d probably good to.. otherwise you can have the arguments taken according to can specification taken from Sesamm tool, because today it’s quite easy to copy or write an argument that is misspelled or anything, that is just an unnecessary.. um.. mistake.

If you can get a program that helps you get the correct arguments that’d be helpful, so I’d like it to be extracted from database.

.. and add it how?

You can have a table here (indicating a dropdown) as list of available arguments and here (in the signal library).

Yes. And about free text, some people have suggested that the function owners will be able to introduce new signal. Then they will have the authority to add signal and the architects can just approve it.

I think that’s the right way to do it. It is too complicated today to add a signal to the AED.. not complicated but it takes time to find an architect and they have to sit down with you and do it together. I’d prefer if function owners can, as you said, propose a signal and they (the architects) are more like reviewing and approving it.

The other dimension that I’m trying to tackle here is the time dimension, the SOP date. Because not only the components change but if there’s SOP date update we also generate new diagrams. I don’t have a solution yet but for example one the easiest way is to have validity period property for each signal so when you choose a signal you can see validity period of the signal from when to when. The other way is to have an overview (draw design ideas) like this, but I don’t know if it would help.

Good question. There is a risk that master chart would be extremely big if you have everything in it, both regarding components and variants and time as well. If you have, you know, for example master chart reaching from.. well, during 5 or 8 years of time, a lot of signals will be added and removed that would be.. and you will have a lot of irrelevant info in the master chart.. umm. So that is a tricky questions.

One way is as you said you can add validity for each signal and then to visualize it you choose which period you want to see for the moment and hide everything else that isn’t valid for that time. That is perhaps one way perhaps to reduce the complexity.

Another suggestion here.. if you’re able to draw a line in the chart view for example from COO to EMS it would be fine if you can have suggestion, e.g. if you want to draw both to S6 and S8 instead of drawing two times, that’d be neat.
## 9.5. Comparison test task log

### Participant 1

<table>
<thead>
<tr>
<th>Training</th>
<th>From 00:00</th>
<th>P1</th>
<th>Spent around 20 seconds before realizing that the signal is not registered</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To 01:35</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Duration</td>
<td>01:35</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task Old</td>
<td>From 02:00</td>
<td></td>
<td>Spent some time making sure that he has to do it in free text (clicking all combo boxes). Especially for the first few signals.</td>
</tr>
<tr>
<td></td>
<td>To 06:15</td>
<td></td>
<td>Then for the remaining signals he headed straight for free text.</td>
</tr>
<tr>
<td></td>
<td>Duration 04:15</td>
<td></td>
<td>No major problem</td>
</tr>
<tr>
<td>Tutorial</td>
<td>From 06:20</td>
<td></td>
<td>9:38 Checking why signal disappears when switching from OPC4v2 to TMS1. Exploring GMS tree.</td>
</tr>
<tr>
<td></td>
<td>To 13:30</td>
<td></td>
<td>10:52 Trying to modify signal name. Signal is difficult to right click.</td>
</tr>
<tr>
<td></td>
<td>Duration 07:10</td>
<td></td>
<td>Deleting and reading signal and exploring its effect to the tree</td>
</tr>
<tr>
<td>Task New</td>
<td>From 13:46</td>
<td></td>
<td>14:30 Ponders on S6 and S8 of EMS but figured it out to choose one.</td>
</tr>
<tr>
<td></td>
<td>To 18:50</td>
<td></td>
<td>15:50 stops and realizes he’s added A5 specific signals to GMS.</td>
</tr>
<tr>
<td></td>
<td>Duration 05:04</td>
<td></td>
<td>Fixing it by deleting, adding, and rearranging signals</td>
</tr>
<tr>
<td>Feedback</td>
<td></td>
<td></td>
<td>Free edit on signal names</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Signal name too greyed out. Add tooltip</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Users can forget deselecting. Auto deselect filters?</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Signals hard to right click or drag</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>EMS(S6&amp;S8) filter: allows multiselect / multifilter</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Auto add GW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Will use signal library over direct drawing, but both are okay</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>He was able to recover and write subsignals after mistakenly drawing global signals. It showed his understanding of the model</td>
</tr>
</tbody>
</table>

### Participant 2

<table>
<thead>
<tr>
<th>Training</th>
<th>From 00:00</th>
<th>P2</th>
<th>Checking if signal is registered</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To 01:20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Duration</td>
<td>01:20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task Old</td>
<td>From 01:40</td>
<td></td>
<td>2:40 Wrong signal destination, close dialog box again</td>
</tr>
<tr>
<td></td>
<td>To 07:48</td>
<td></td>
<td>3:00 trying the green button</td>
</tr>
<tr>
<td></td>
<td>Duration 06:08</td>
<td></td>
<td>4:37 SA Name written in the message as underscore</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>5:22 Finished first MSC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>6:50 Frustated? on not finding registered signal. If it is</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Environment he headed straight for free signals</td>
</tr>
<tr>
<td>Tutorial</td>
<td>From 08:23</td>
<td></td>
<td>11:38 Trying to go to signal configuration</td>
</tr>
<tr>
<td></td>
<td>To 14:10</td>
<td></td>
<td>12:20 Failed to drag multiple times but finally managed (coincidentally) after he switched to a subconfiguration. Might give him the wrong idea on being able to drag only on a signal's precise configuration.</td>
</tr>
<tr>
<td></td>
<td>Duration 05:47</td>
<td></td>
<td>13:30 Finished but kept trying it around</td>
</tr>
<tr>
<td>Task New</td>
<td>From 14:29</td>
<td></td>
<td>15:05 Stops and ponders on EMS</td>
</tr>
<tr>
<td></td>
<td>To 21:50</td>
<td></td>
<td>15:38 Added two signals, one for S6 and one for S8</td>
</tr>
</tbody>
</table>
### Duration

- **07:21**

#### Feedback
- Highlight missing signals
- Filter highlight in library not clear
- Will use both library and drag. Suspect that power users will use lib

---

### Participant 3

<table>
<thead>
<tr>
<th><strong>P3</strong></th>
<th><strong>Tutorial</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>From</td>
<td>00:10</td>
</tr>
<tr>
<td>To</td>
<td>05:20</td>
</tr>
<tr>
<td>Duration</td>
<td><strong>05:10</strong></td>
</tr>
</tbody>
</table>

**Task New**
- From 05:30
- To 11:20
- Duration **05:50**

**Training**
- From 12:00
- To 13:00
- Duration **01:00**

**Task Old**
- From 13:10
- To 16:45
- Duration **03:35**

**Feedback**
- Likes the thicker border
- Not clear which configurations the grey signals belong to. Keeping it clean vs adding more info
- Arguments: show possible options, or range for e.g. speed
- Thinks SOP variance will be useful
- Combine both drawing methods

---

### Participant 4

<table>
<thead>
<tr>
<th><strong>P4</strong></th>
<th><strong>Tutorial</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>From</td>
<td>00:00</td>
</tr>
<tr>
<td>To</td>
<td>06:40</td>
</tr>
</tbody>
</table>

**01:00 Trying to add signal outside tutorial**
- Right clicking everything
- When he clicks on library filters, he was confused why the active
### Task New

**From** 06:50
- 06:50 Switched to OPC
- 07:00 "I'm supposed to do this (MSC) right?"
- 08:00 Stopped to think on EMS
- 08:20 Added 2 signals, for S6 and S8
- 08:30 Checking S6 and S8
- 08:45 Trying to double click
- 08:54 Deleting one of the signals
- 09:30 Asking about EMS, sounds confused
- 11:42: Finished OPC MSC
- 13:22 Deleted an incorrect signal and added a new one
- 14:20 Realizing A5 is blank
- 16:20 Start doing A5 again. Moving all signals up
- 17:20 Finished

**To** 17:20
- Duration 06:40
- Configurations did not change
- 4:54 Aha? moment when switching back to GMS

### Training

**From** 18:00
- 19:00 Fixed a typo in the first signal
- 19:17 Checking that no signal is registered for the second signal

**To** 19:40
- Duration 01:40
- Expect auto deselect
- Likes the signal library
- (Aries: in the old MSC drawer it’s not obvious which signal has been registered)
- Did not get Master Chart concept during the test
- Sync library with the active configurations
- Likes being able to move signals

### Task Old

**From** 20:00
- 20:10 Checking registered signals
- 20:54 Switching back and forth between 2 MSC
- 21:00 Interestingly he is doing 2 MSC in parallel
- 22:00-23:00 Did auto-populate and lost all. I helped to recover
- 25:10 Finished one MSC

**To** 21:00
- Duration 06:36

### Feedback

- Expect auto deselect
- Likes the signal library
- (Aries: in the old MSC drawer it’s not obvious which signal has been registered)
- Did not get Master Chart concept during the test
- Sync library with the active configurations
- Likes being able to move signals

### Participant 5

#### Tutorial

**From** 00:00
- 01:00-01:30 Trying to deselect but hasn't learned how
- 04:30-05:05 Failed to right click multiple times
- 06:00-08:18 Finished official tutorial but continue trying around

**To** 08:18
- Duration 08:18

#### Task New

**From** 08:53
- 09:45 Switching to OPC4v2 before drawing second signal
- 10:19 Deleted the first signal

**To** 20:00
| Task Old | 24:35 Go straight to FT after a glance | 24:55 Trying to find registered signals on each signal add |
| Task Old | 27:37 Finished first MSC | 29:47 Fixed a typo |
| Task Old | 30:09 Finish |

**Feedback**
- Thinks signals from db is good
- In the beginning it wasn't clear what black and grey means
- Sync library and active configurations
- Notification on new component variant
- Prefer to use signal library because it's harder to get the source and destination wrong
- Deselecting takes time
- Manual gearbox is confusing (e.g. could be part of COO6)

### Participant 6

<table>
<thead>
<tr>
<th>Training</th>
<th>P6</th>
</tr>
</thead>
<tbody>
<tr>
<td>From</td>
<td>00:00</td>
</tr>
<tr>
<td>To</td>
<td>01:20</td>
</tr>
<tr>
<td>Duration</td>
<td>01:20</td>
</tr>
<tr>
<td>Task Old</td>
<td></td>
</tr>
<tr>
<td>From</td>
<td>01:45</td>
</tr>
<tr>
<td>To</td>
<td>07:45</td>
</tr>
<tr>
<td>Duration</td>
<td>06:00</td>
</tr>
</tbody>
</table>

**Tutorial**
- 08:00
- 12:11
- 04:11

**Task New**
- 12:55 13:35 Stopped at EMS
- 14:01 Trying to multi-click S6 & S8
<table>
<thead>
<tr>
<th>Duration</th>
<th>05:39</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feedback</td>
<td>Likes library better than direct drawing</td>
</tr>
<tr>
<td></td>
<td>Free text could be an issue, how to add</td>
</tr>
<tr>
<td></td>
<td>The effect of deleting a shared signal in one variant should be more obvious</td>
</tr>
<tr>
<td></td>
<td>Argument could be added in the property box</td>
</tr>
<tr>
<td></td>
<td>Likes the opportunity to have more authority to add new signal</td>
</tr>
</tbody>
</table>

15:05 Understand that only the active configuration matters
15:58 Switched down GMS to OPCv2 and draw specific signals
16:20 Switched to A5 for specific signal
16:35 Checking GMS at A5 and OPC level
Deleting a signal from OPC. Found out it is also deleted in A5.
Fixing