NTNU – Trondheim
Norwegian University of
Science and Technology

Implementing Lean Systems Engineering in the DNV Fuel Fighter project

Oluf Roar Bjørset Tonning

Master of Science in Engineering and ICT

Submission date:	June 2012
Supervisor:	Terje Rølvåg, IPM
Co-supervisor:	Cecilia Haskins, IØT

Norwegian University of Science and Technology
Department of Engineering Design and Materials

Norwegian University of Science and Technology
Faculty of Engineering Science and Technology
Institute of Engineering Design and Materials

Master's thesis

Implementing Lean Systems Engineering in the DNV Fuel Fighter project

Author:
Oluf Tonning

Supervisor:
Prof. Terje Rølvåg, NTNU
Co-supervisor:
Post-Doc Cecilia Haskins, NTNU

June 11, 2012

NTNU – Trondheim
Norwegian University of Science and Technology

DNV FUEL FIGHTER 2

THE NORWEGIAN UNIVERSITY
OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF ENGINEERING DESIGN
AND MATERIALS

MASTER THESIS SPRING 2012
FOR
STUD.TECHN. OLUF TONNING

IMPLEMENTERING AV LEAN S.E. I DNV FUEL FIGHTER PROJECT
Implementing Lean Systems Engineering in the DNV Fuel Fighter project

The DNV Fuel Fighter is NTNU's contribution in the Shell Eco Marathon competition. It has previously achieved very good results; most notably the current world record for the longest distance travelled consuming an amount of energy equalling the energy of one litre of petrol.

Systems Engineering is a newly adopted focus for the project. The motivation for implementing Systems Engineering is to reduce risk and cost by devising proper verification and validation processes and by reducing the knowledge gap from one year's team of students to the next.

The main task of this master's thesis is to assess the project from the perspective of Lean Systems Engineering, identify problem areas and propose solutions or improvements. More specifically, the following tasks should be carried out:

1. Describe Lean Systems Engineering and the process of assessing a project with regard to Lean principles.
2. Perform assessment of the project from a perspective of Lean Systems Engineering by interviewing multiple members of the team, preferably from different engineering disciplines.
3. Evaluate the results and identify the areas where improvement is needed.
4. Propose a solution.
5. If there is time; implement the solution.

The thesis should include the signed problem text, and be written as a research report with summary both in English and Norwegian, conclusion, literature references, table of contents, etc. During preparation of the text, the candidate should make efforts to create a well arranged and well written report. To ease the evaluation of the thesis, it is important to cross-reference text, tables and figures. For evaluation of the work a thorough discussion of results is appreciated.

Three weeks after start of the thesis work, an A3 sheet illustrating the work is to be handed in. A template for this presentation is available on the IPM's web site under the menu "Undervisning". This sheet should be updated when the Master's thesis is submitted.

The thesis shall be submitted electronically via DAIM, NTNU's system for Digital Archiving and Submission of Master's thesis.

The contact person is (navn på veileder i utlandet, bedrift eller lignende)

Torgeir Welo
Head of Division

Terje Rølvåg
Professor/Supervisor

NTNU
Norges teknisk-naturvitenskapelige universitet
Institutt for produktutvikling og materialer

Forord

Dette siste semesteret på NTNU har vore litt av ei reise. Å vere med i NTNU sitt Shell Eco-maraton-lag er ingen spøk. Det har vore lange dagar med mykje arbeid, fortviling og glede. Suksess og nederlag. Forsvunne sokkar og bussturar ein ikkje heilt er sikker på at faktisk fann stad. Mange nye kjenningar. Venskap som eg håpar vil vare livet ut.

Dette semesteret har eg tilbrakt i ei feltseng på eit loft i Åsbakken Asylmottak. Kan trygt fastslå at ting ikkje alltid går som planlagt. Men noko naud har eg aldri lidd.

Det har vore seks lange år i Trondheim, og mange å takke. Først vil eg takke rettleiarane mine. Terje for gode idear, og for å tvinge meg til å tenkje kritisk. Cecilia for sine gode råd og upåklagelege entusiasme.

Takk til Shell Eco-maraton-teamet, og særleg takk til Itsaso for samarbeidet i Systems Engineering-avdelinga.

Takk til alle på PLM-laben. De har gjort skuleåret til eit eventyr. Kan berre unnskylde for alle kilo de kan ha lagt på dykk etter alt snopet eg har pressa i dykk.

Takk til Anders, Tor og Andreas. Åsbakken er og blir det beste asylmottaket i landet.

Og sjølvsagt takk til mamma. Alt maset ditt om lekser og pugging opp gjennom åra er grunnen til at eg no kan levere denne masteroppgåva. Vel verdt det, faktisk. Og så skulle eg gjerne ha takka pappa, men du fekk dessverre ikkje sjå meg fullføre seks år i Trondheim. Utan di støtte hadde eg heller ikkje vore her.

Oluf Roar Bjørset Tonning
Trondheim, June 11, 2012

Abstract

This thesis is a practical application of Systems Engineering in a full-scale project. The author has participated as a member of NTNU's Eco-marathon team, who has spent the last two semesters designing and manufacturing an energy-efficient car for the Shell Eco-marathon competition 2012.

The author made an assessment of the project to find the areas where he should focus his effort. The author also applied Lean Thinking to find the Systems Engineering methods that would best aid the team improve within the areas uncovered in the assessment.

The areas where the team needed improvement were Knowledge management and Continuous improvements of the product development process.

By using Model-based Systems Engineering processes the author produced a system model acting as a knowledge repository. Then, the author used methods from Lean Thinking for knowledge capture, namely Knowledge Briefs and learning events.

For improving the product development process, the author introduced the team to Visual Workflow Management to help the team become more agile, and to visualize project progress from multiple perspectives in order to discover parts of the system which is not evolving at the desired rate.

Key concepts in this thesis include: Lean Thinking, Lean Systems Engineering, Model-based Systems Engineering, Model-based Documentation, Knowledge management, Knowledge briefs, A3 method, system modelling, functional analysis, functional flow block diagrams (FFBDs), system architecture, Visual Workflow Management, Stand-up meetings, Visual project board, risk management

Abstract

Denne oppgåva omhandlar praktisk innføring av Systems Engineering i eit fullskala prosjekt. Forfattaren har vore medlem i NTNU sitt Eco-maraton-lag, som har nytta dei siste to semestera til å designe og byggje ein energieffektiv bil som skal prestere i Shell Eco-maraton 2012.

Forfattaren gjorde ei vurdering av prosjektet for å finne dei områda kor han burde fokusere hovudtyngda av arbeidet sitt. Forfattaren brukte også Lean Thinking for å finne dei Systems Engineering-metodane som best ville kunne hjelpe gruppa i å forbetre seg innanfor dei problemområda som vurderinga avdekte.

Områda kor gruppa trengde forbetring var Kunnskapshandtering og Kontinuerlege forbetringar i produktutviklingsprosessen.

Ved å nytte Model-based Systems Engineering-metodar produserte forfattaren ein systemmodell som skal fungere som eit kunnskapsdepot. I tillegg nyttar forfattaren metodar frå Lean for oppfanging av kunnskap, nærare bestemt Knowledge Briefs og læringssesjonar.

For å betre produktutviklingsprosessen introduserte forfattaren gruppa for Visual Workflow Management for å gjere gruppa meir responsiv til endringar, og for å visualisere prosjektframgong ifrå fleire perspektiv for å oppdage delar av systemet som ikkje utviklar seg i ønska rate.

Nøkkelkonsept i denne oppgåva inkluderer: Lean Thinking, Lean Systems Engineering, Model-based Systems Engineering, Model-based Documentation, Knowledge management, Knowledge briefs, A3 method, system modelling, functional analysis, functional flow block diagrams (FFBDs), system architecture, Visual Workflow Management, Stand-up meetings, Visual project board, risk management

Contents

Acronyms

BOM Bill of Materials.

CAD Computer-aided Design.

DNVFF DNV Fuel Fighter.

DNVFF2 DNV Fuel Fighter 2.

FFBD Functional Flow Block Diagram.

I/O input/output.

INCOSE International Council on Systems Engineering.

IPM Institute of Engineering Design and Materials.

K-brief Knowledge Brief.

LEfSE Lean Enablers for Systems Engineering.

LPD Lean Product Development.

LSE WG Lean Systems Engineering Working Group.

MBD Model-based Documentation.

MBSE Model-based Systems Engineering.

NASA National Aeronautics and Space Administration.

NTNU the Norwegian University of Science and Technology.

SBCE Set-Based Concurrent Engineering.

SE Systems Engineering.

SysML System Modelling Language.

TPS Toyota Production System.

UML Unified Modelling Language.

VPB Visual Project Board.
VSM Value Stream Mapping.
VV&T Verification, Validation & Testing.
VWM Visual Workflow Management.

WBS Work Breakdown Structures.
WWI the First World War.
WWII the Second World War.

Chapter 1

Background

This thesis is a practical study of how to perform Systems Engineering (SE) in a real, full-scale project. The author has a background with SE courses at the university, and wrote a project work on Model-based Systems Engineering (MBSE), fall of 2011. The author joined the Shell Eco-marathon team of the Norwegian University of Science and Technology (NTNU) as an extension to the SE part of the team, which already counted one participant. The author will be referring to this other thesis written by Itsaso Yuguero Garmendia, called *Development and construction of car for eco-marathon for participation in competition: Verification, Validation and Testing activities of the DNV Fuel Fighter 2* at multiple points.

In order to identify the areas where the author should focus his efforts to make the best contribution to the team, he performed an assessment of the team's current and desired states based on theories presented by Bohdan Oppenheim in the book *Lean for Systems Engineering* and a series of articles of the same name. Oppenheim presents a multitude of enablers that systems engineers may apply to an organization to help them become more value-added. With help from Professor Terje Rølvåg, the author picked out the best-suited enablers to help the team in the areas where the difference between the current and desired state was large.

This thesis starts with a very broad perspective, but narrows down to focus on the fields of Knowledge management (an area where the author chooses to employ MBSE techniques to help the team, with guidance from Post-Doc Cecilia Haskins) and Visual Workflow Management. The two are even integrated with each other at one point. The study is a practical one, and the author's focus has all the way been to participate in the team and to reach the Eco-marathon competition and produce the best possible result.

1.1 The Shell Eco-marathon and NTNU

The DNV Fuel Fighter (DNVFF) [i] is NTNU's contribution to the Shell Eco-marathon, a competition which *"[...] challenges high school and college student teams from around the world to design, build and test energy efficient vehicles."* [21] The cars' energy consumption is measured in Joule and converted into kilometres travelled per litre of petrol (km/l). The competition has its roots from a competition held by engineers at Shell in the US as far back as 1939. The modern version of the competition started in Europe in 1985. [22]

The NTNU contribution is called the DNVFF and has participated since 2008. A new team of graduate students participate every year. The project has gained much success - most notably the world record of 1,246 km/l from 2009. In 2011 the team finished second, just short of 1,000 km/l. The project has also won a number of other awards, such as the PR awards in 2009 and 2011.

However, the story of the DNVFF has a backside. Despite the strong results, the vehicle has barely made it through the competitions, twice only producing a single approved result and once even failing to produce any result.

In January 2011 the project employed its first systems engineer. The motivation behind this, was to have one person in the team whose responsibility it was to focus on the system in its entirety. This means concentrating on fulfilling requirements, integration of subsystems and testing and verification. The positive feedback from the 2011 team on the role of SE made it clear next year's team also needed systems engineers, and thus the 2012 team currently employs two systems engineers.

The 2012 team is designing and building an all-new vehicle, called the DNV Fuel Fighter 2 (DNVFF2) [ii]. This is a big challenge with a higher associated risk, compared to improving an already existing vehicle. Also, the team is larger than the last years' teams with a total of 14 members, compared to last year's 6 members.

The DNVFF2 will also compete in a new category from the one NTNU has competed in previously, namely the Battery-Electric class, meaning vehicles with Lithium-ion batteries as energy source. The new category is more competitive than the old one (hydrogen) and much more prestigious.

After the initial assessment of the DNVFF2 team, the research questions for this master's thesis became:

- How does one design methods and a healthy environment for successful knowledge capture and transition?

[i] Det Norske Veritas (DNV) is the main sponsor of the project

[ii] When this thesis refers to DNVFF it means the project in its entirey, including previous and future teams. When it refers to DNVFF2 it refers only to the 2012 team or vehicle.

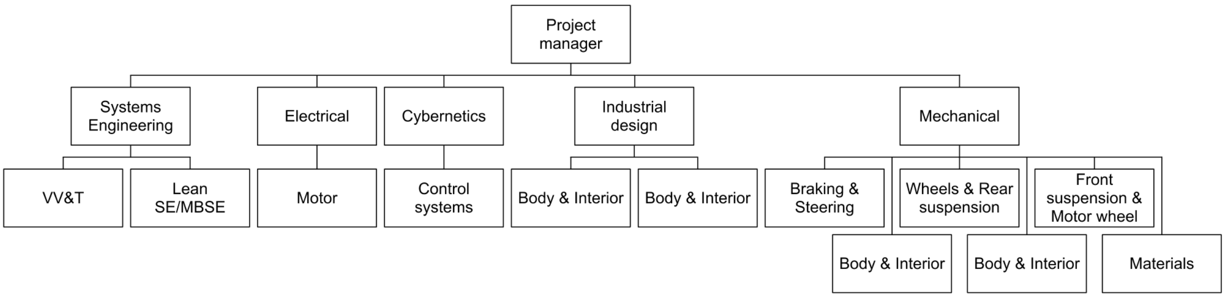


Figure 1.1: Hierarchy displaying the six engineering disciplines employed in the project, as well as the main responsibilities of every team member.

- How does one coordinate and visualize project progress on a system, sub-system and component level?
- Is it possible to combine Lean and Systems Engineering without conflict?
- How does Lean principles affect projects of short duration?

1.2 About the team and the competition

During the first semester of the DNVFF2, the team included one systems engineer. The SE tasks that semester consisted of requirements analysis and interface management.

In January 2012, the author of this thesis joined the team, as the second systems engineer. From this point on, the team had two systems engineers. The initial systems engineer focused on Verification, Validation & Testing (VV&T) activities, while the author focused on implementing Lean SE methods in the project - with particular focus on the up-coming transition from the 2012 till the 2013 team. Working as a team, the two systems engineers collaborated on requirements and risk analyses, communication and Work Breakdown Structures (WBS).

The project was divided in two main phases, one phase for each of the two semesters (fall and spring semesters). The fall semester was spent doing concept exploration and evaluation, and design of solutions. The spring semester was spent doing production of parts and VV&T activities. The team recruited a cybernetics engineer at the start of the spring semester, which added some concept and design work also in that semester.

The sub-systems were designed in parallel. Particularly two sub-systems were designed in a very efficient and impressive manner; the body and the motor. These two sub-systems involved engineers from different disciplines; electrical and mechanical for the motor and industrial design, materials and mechanical for the body. The body shape was designed by two industrial design students, and validated by the mechanical engineer through structural analyses based on input from the materials specialist. The process involved numerous iterations and resulted in a top-quality product. The production was also performed concurrently for every sub-system, resulting in the sub-systems finishing at approximately the same time.

The team did not reach all of its goals. The motor was unfortunately not finished on time due to a series of repeated production errors, and the team had to compete with the old motor solution. The team had foreseen the problem and designed the vehicle to also fit the old motor.

Mechanically, the vehicle is a very strong contribution to the marathon. Upon reaching Rotterdam - where the competition was held May 14 to 19 2012 - the team saw that they were by far one of the lighter cars of the competition with the DNVFF2's 87 kg. Most competitors were 100kg+, even one as heavy as 160kg.

The team made a very good competition finishing 5 th with a result of 163 km/kWh, which constitutes 1581 km/L of petrol. The team learnt that the vehicle is lighter than its competitors and pushing the margins to the limits, but has an inferior power train. The power train would not have been much better with the new motor, as the team realized an all-new design for the power train is necessary to win the competition. The reader will find more details about the competition in Figure C.1 and C.2.

Chapter 2

Theory

2.1 Systems Engineering

> *"Systems Engineering is an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem. Systems Engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs."*

- International Council on Systems Engineering (INCOSE)[8, p. 1.5]

> *"[A system is] a set of of interrelated components working together toward some common objective."*

- Kossiakoff & Sweet[9, p. 3]

2.1.1 Historical background

SE principles have been practised in one way or another for as long as man has developed systems. From the making of early stone age canoes to Swiss Army knives, work has been conducted according to some plan derived from a need to cross a river or to whittle a stick. However, Systems Engineering as a discipline has only existed since the the Second World War (WWII) [9, p. 6].

The First World War (WWI) changed the face of warfare, industrializing it. Complex systems entered the battlefield in the form of tanks, fighter planes, rockets and self-propelled guns. By the outbreak of WWII the military leaders had learnt the lesson from WWI; to win the war one had to win the race of technological prowess.

New weapons had to be developed quickly. Deploying the right weapon at the right time and before the enemy could field a counter was imperative. Innovation and development had to be streamlined. At the same time, the complexity of the systems grew. From the basic cannon of the 19th century who could do little else but fire a projectile at the enemy and allow its users to wheel it around - to the tank of WWII who could fire, move, traverse all kinds of terrain, turn its turret independently of its body, house soldiers for days, withstand direct hits from other tanks or anti-tank guns and at the same time deliver enough firepower to destroy enemy tanks and vehicles. And yet, the systems had to be developed in shorter time spans than before - and in sharp competition, not only with the enemy but also with competing suppliers to the military. The need for a structured development process surfaced.

The systems engineers added the system perspective that such ambitious projects demanded. These engineers allowed integration of all the different fields of specialization among engineers and scientists. They applied "systems thinking"; considering the system as a whole, analyzing the needs of the users, comparing design alternatives, performing trade-off analyzes and evaluating the risks. Before implementation and production could ever start, the system had to be fully designed, all possible outcomes thought through and all pitfalls covered.

The rate of technological development spurred by WWII did not end in 1945. Instead, the military kept requiring a growth in technology all through the Cold War [9, p. 6]. The wars in Korea and Vietnam, the Space Race, spy planes such as the U-2 and Reagan's Star Wars program of the 80's all drove the technological development onwards. The complexities of the systems continued to grow, especially with the dawn of the computer age.

The Cold War is over, but the technology race has long ago been picked up by the civilian sphere. The development of software, passenger planes, Formula 1 racing cars or the building of massive constructions like the Øresund Bridge joining Denmark and Sweden [8, p. 3.5.2] are all examples of complex, civilian systems in need of systems engineers to integrate specialists from numerous engineering disciplines working together to build one system.

2.1.2 General Systems Engineering principles

The SE process encompasses the entire life cycle of a product. The systems engineer's task is to design the life cycle of the system [8, p. 3.1]. In the process, the systems engineers run the development phase, making the final design decisions based on the work and testing performed on different concept alternatives [9, p. 23]. At the same time, the systems engineer plans how to build the resulting fully functional system, how to launch the product, maintain it during its operational life and finally how to retire the product.

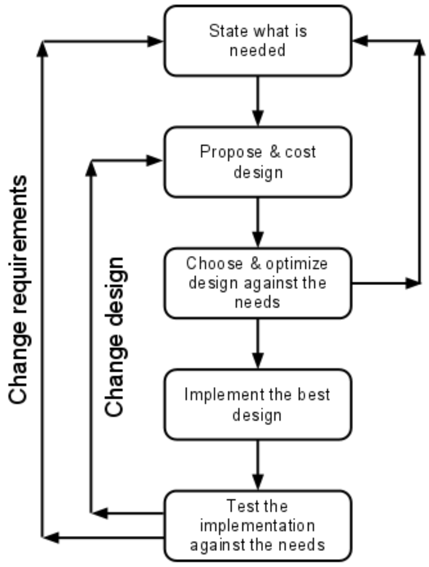


Figure 2.1: The essence of Systems Engineering [23, p. 345]

The SE effort seeks to reduce the amount and impact of the risks that threaten the system. Lowering the likelihood of errors occurring late in the life cycle is imperative, as the cost associated with removing such errors is high. A statistical analysis performed on projects in the US Department of Defence shows that the cost of removing errors is 500 to 1000 times higher in the latest stages of the life cycle, compared to the initial stages [8, p. 2.5]. A study by INCOSE on the return of investment from systems engineering efforts shows that projects using systems engineering principles greatly reduce their schedule and cost overruns [8, p. 2.6].

The product life cycle is described in the ISO/IEC 15288, which is the standard for life cycle processes of man-made systems [8, p. 3.3],[9, p. 53]. This model divides a product's life into six stages; concept, engineering, production, utilization, support and retirement. The stages are represented in Figure 2.2.

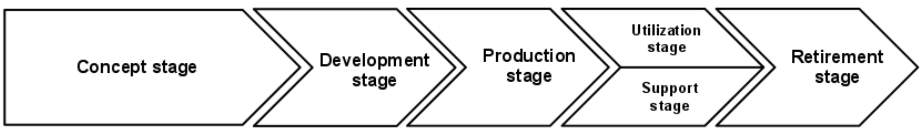


Figure 2.2: The ISO/IEC 15288 standard of a product life cycle.

New projects materialize either from needs within an organization, or from new ideas [8, p. 3.3.1]. No matter the exact origin of the project, the systems engineers use the concept stage to explore the needs of the perceived users of the system, the available technologies and to perform "feasibility studies" [9, p. 58]. Stakeholders' requirements are identified and formulated. Such endeavours are often called the "pre-concept stage" [8, p. 3.3.1]. Projects that are green-lighted from such a pre-concept stage undergoes more detailed scrutiny. The systems engineers now starts designing the architecture of the system, building models of the system-to-be, exploring concept alternatives and building prototypes. A common technique in SE is functional and physical analysis of the system [8, p. 3.3.2] resulting in models describing the system. These techniques will be explored in more detail later in this chapter.

SE is an iterative process [8, p. 2.2]. To reach the goal of risk reduction, the systems engineers work iteratively, following standardized working procedures. Figure 2.3 shows the standard work flow of the concept stage of a project [9, p. 70].

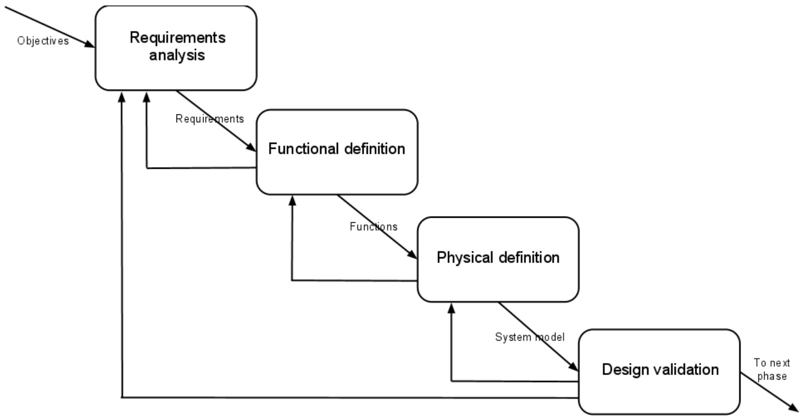


Figure 2.3: N-squared chart of the work flow in the concept stage.

The output from each stage in Figure 2.3 are strongly linked to each other, as illustrated in Figure 2.4. Requirements are satisfied by functions, and functions are realized by physical components. The purpose of this traceability, is to achieve total understanding of why a decision has been made, e.g. why was Component A chosen over Component B, or what functions do Component C perform, or how well is Component D performing relative to its requirements?

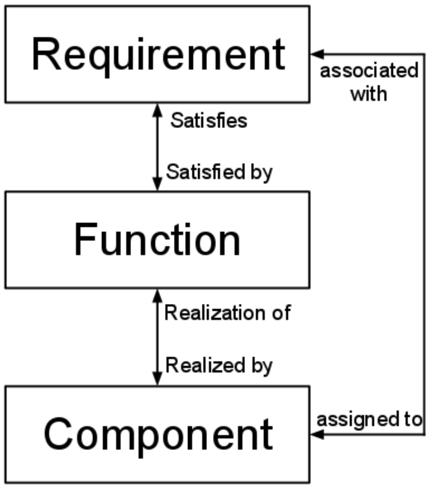


Figure 2.4: Traceability between the output from the different phases of the concept stage.

The systems engineering focus is summarized in a model called the "Vee" [8, p. 3.3], displayed in Figure 2.5. The concept stage - and thus the contents of Figure 2.3 - contitutes the left-hand side of the Vee. This figure also highlights two other concepts that are imperative to the SE effort; *verification* and *validation*.

Verification and validation is the key to avoiding costly errors. The act of validation is to approach the stakeholders or the future users of the system and ask the following question: "are we building the right thing?". Verification is to test the actual system and compare its effectiveness to that of the stated requirements, or - in other words - to answer the question: "are we building it right?". The Systems Engineering Handbook highlights these acts as a part of the development stage [8, p. 3.3.3], yet Kossiakoff and Sweet also include these acts on the left hand side of the Vee [9, p. 70] as a natural way of ending an iteration where one measures the rate of progress and plans work for the following iterations.

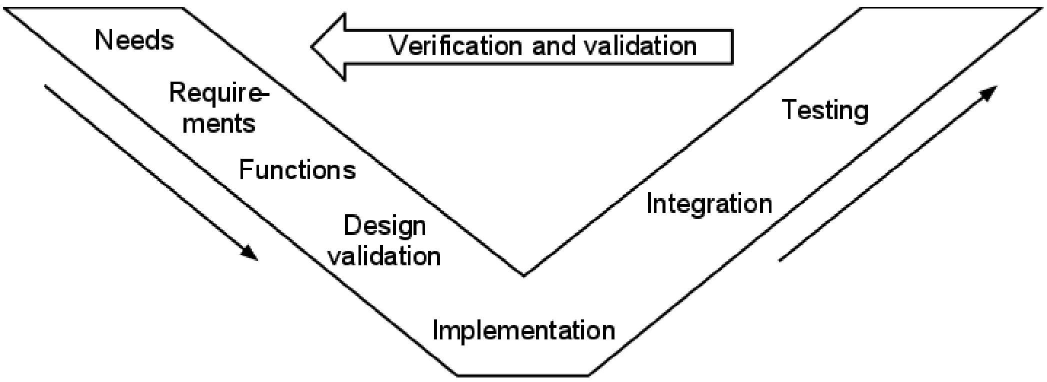


Figure 2.5: The VEE model.

2.1.3 Risk management and Systems Engineering

As stated in the previous section, SE aims to reduce the risks associated with a development program. There will always be many uncertainties early in the program, but the engineers must address these issues to reduce the likelihood and impact of these risks. Lack of knowledge or skill, too high ambitions, lack of funding and unproven technology are examples of issues that all contribute to making a project less likely to succeed. This section will give a more in-depth view of this very important SE focus.

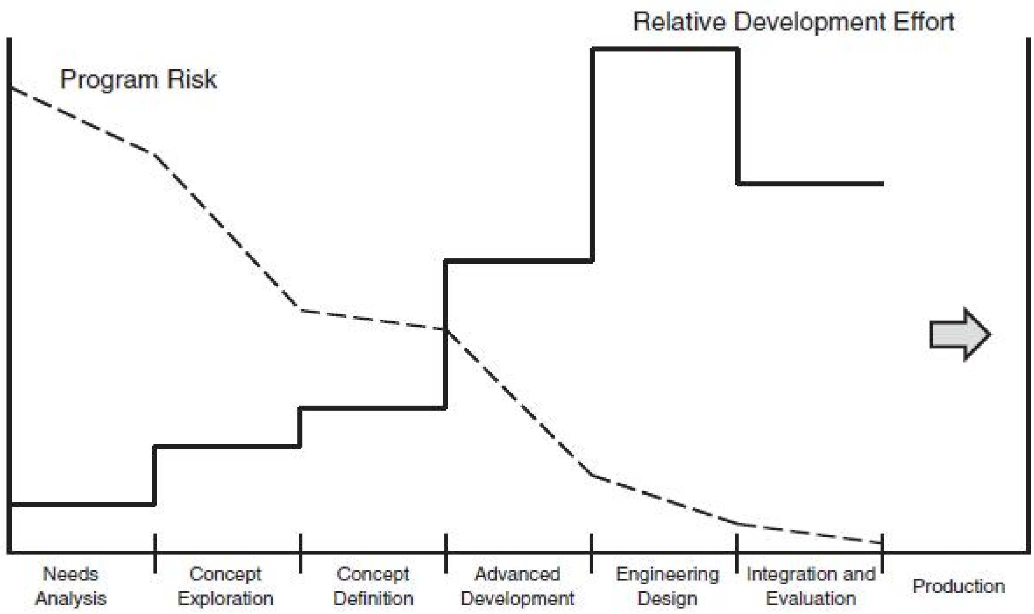


Figure 2.6: Variation of program risk and effort throughout system development. [10, p. 121]

When a project starts, the risks associated are high. As the project wears on the risk level will reduce - as long as the team addresses the issues at hand and solves them before they become unsolvable or it is too late or too expensive to fix them. It is the systems engineer's role to manage risks by coordinating the effort of identifying and analyzing risks. After identification and analysis the Systems Engineer will develop mitigation plans, that is; plans for how to reduce the worst risks.

Given the the risk is realized, what would be the magnitude of the impact?			
1	Minimal or no impact	Minimal or no impact	Minimal or no impact
2	Minor performance shortfall, same approach retained	Additional activities required, able to meet key dates	Budget increase or unit production cost increase <1%
3	Moderate performance shortfall, but work-arounds available	Minor schedule slip, will miss needed dates	Budget increase or unit production cost increase <5%
4	Unacceptable, but work-arounds available	Project critical path affected	Budget increase or unit production cost increase <10%
5	Unacceptable; no alternatives exist	Cannot achieve key project milestones	Budget increase or unit production cost increase >10%

Figure 2.8: Impact levels according to Kossiakoff. [10, p. 124]

What is the likelihood the risk will happen?		
1	Not likely	Will effectively avoid or mitigate this risk based on standard practices.
2	Low likelihood	Have usually mitigated this type of risk with minimal oversight in similar cases
3	Likely	May mitigate this risk, but work-arounds will be required
4	Highly likely	Cannot mitigate this risk, but a different approach will
5	Near certainty	Cannot mitigate this type of risk; no known processes or work-around are available

Figure 2.7: Likelihood levels according to Kossiakoff. [10, p. 124]

According to Kossiakoff, a risk can either be a *technical risk* or a *resource risk*. A technical risk is related to the actual function of the element, e.g. uncertainties about how the material will behave in the system environment. A resource risk is related to money, knowledge, manpower and schedules. Kossiakoff suggests quantifying the risks in their likelihood of occurrence (L) and their impact on the system if they occur or, i.e., consequence (C). E.g., a 1 to 5 scale where 1 means 'not likely' or 'low impact' and 5 means 'very likely' or 'potential project/system failure'. The risk value (RV) is thus:

$$RV = L \times C \tag{2.1}$$

Tables 2.7 and 2.8 show how Kossiakoff explains the different risk levels.

The risk value varies from 1 (very low) to 25 (extreme). Shown graphically the values may be assigned to different risk levels with colour coding. The colors represent the degree of severity, as shown in Figure 2.9.

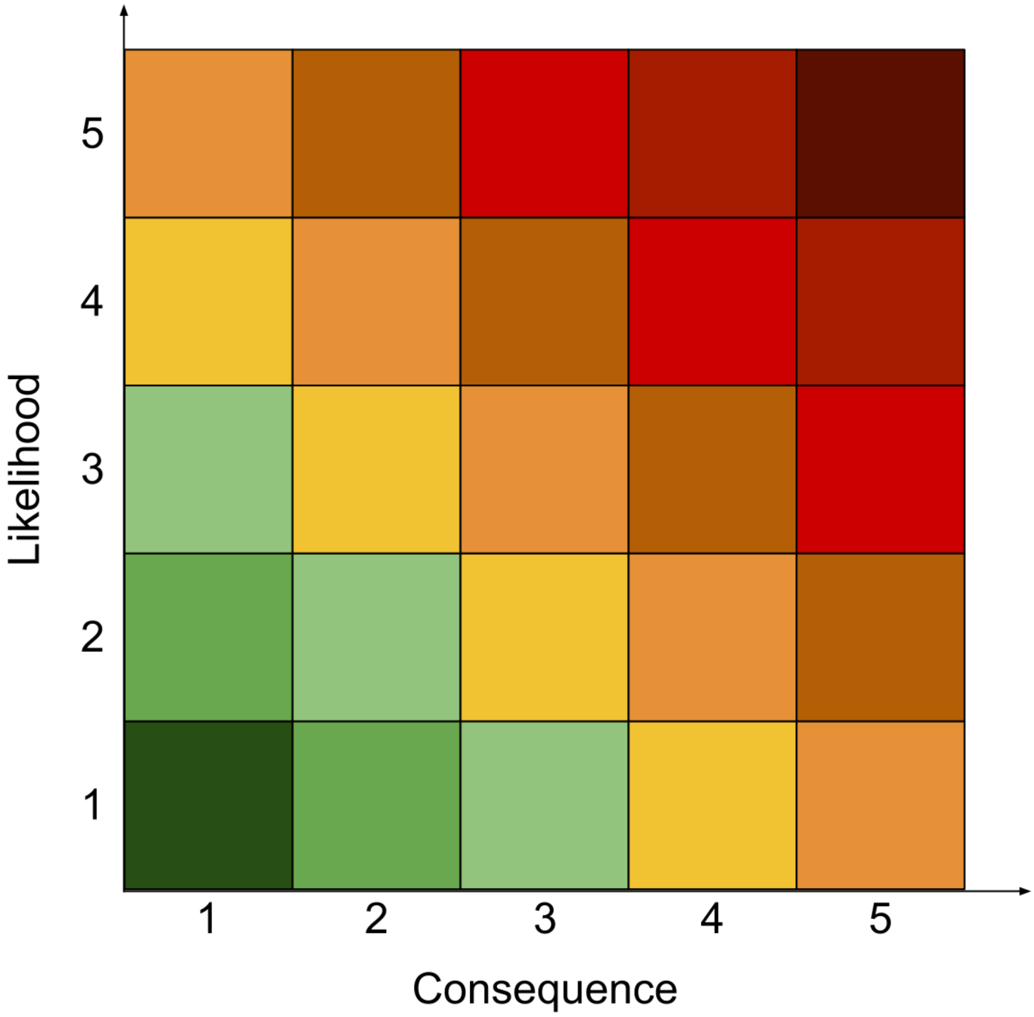


Figure 2.9: Risk level with corresponding colour coding.

Mitigating the risks is - as stated earlier - one of the main tasks of the systems engineer. The risks are prioritized according to their risk value and mitigation plans are developed accordingly. Many risks may be reduced by doing more in-depth analyzes or via more comprehensive software simulations. Others are reduced by designing special VV&T activities that simulate the possible failure modes. Other risks may require the systems engineers to review either the development process or the requirements definition. Changes to either of them may reduce or even eliminate a risk totally. [10, pp. 120-128]

2.1.4 Model-based Systems Engineering

> *"MBSE is the formalized application of modelling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases."*
>
> - INCOSE[8, p. 1.5]

Strictly speaking, MBSE is the application of traditional SE practices in a visual manner. Friedenthal et al. says *"[a] MBSE method is a method that implements all or part of the systems engineering process, and produces a system model as one of its primary artifacts."* [4, p. 21] MBSE is an attempt at standardizing the SE effort by developing a technique for documenting the SE effort through models of diagrams and hierarchies that follow strict rules, and a way of depicting systems through requirements, functions, system architecture and VV&T activites. However, the community has not quite settled on one standard yet, as the reader will learn from this section.

Modelling of systems-to-be has long been common in both electrical and mechanical design. With the dawn of the computer age when databases and more sophisticated modelling tools became available, the models became more and more complex, going from two to three dimensions and gaining the advantage of analyzing design through simulators instead of prototyping.

Software engineers also started modelling their work, and the Unified Modelling Language (UML) was developed to facilitate this effort. This modelling language is still widely used, with its graphical modelling notation, simulation tools and code-creation tools.

However, UML is domain specific for software development and a need was growing within the SE environment for modelling of complex systems, namely the emerging field of MBSE. Based on UML a new modelling language was developed as a general-purpose graphical modelling language. This language is called System Modelling Language (SysML) and supports tools with a wider scope than just software development, with analysis, specification, design, requirements, behavioural analysis, verification, testing and validation.

SysML is one way of modelling, yet far from the only one. However, it needs special mentioning. There is a myriad of software suits that employ their own way of modelling, however, they usually support the same diagrams and the difference is only apparent on a code-level. This is one of the challenges of MBSE today; the community has not derived on one common solution yet. The author wrote his project work on this very topic, and the interested reader may consult that paper for more insight on the efforts being made to unify the MBSE community. [24]

A *system model* is a representation of a system in its elements, these elements being - typically - requirements, behaviours, states, components, test cases, verification requirements, and the interrelationships linking them all. The system model is used to ensure the system being developed is being developed correctly. Also, it acts as an introduction to any engineers new to the system. The model may also be designed to describe the system from a user's perspective explaining the operation of machinery, tools or proper conduct in certain environments.

In addition, it may act as a repository for information, where the documents describing a certain element - i.e. a component or test case - is linked to the element of interest. This gives the reader a logically structured archive where accessing a certain part of the system only reveals information related to that exact part of the system. This way of structuring information is referred to as *Model-based Documentation (MBD)* in this paper. [4, pp. 15-31]

The perceived benefits from implementing MBSE are: [4, p. 20]

- Enhanced communication
  - Shared understanding of the system across the development team and other stakeholders.
  - Ability to integrate views of the system from multiple perspectives.
- Reduced development risk
  - Ongoing requirements validation and design verification
  - More accurate cost estimates to develop the system
- Improved quality
  - More complete, unambiguous, and verifiable requirements
  - More rigorous traceability between requirements, design, analysis, and testing
  - Enhanced design integrity
- Increased productivity
  - Faster impact analysis of requirements and design changes
  - More effective exploration of trade-space
  - Reuse of existing models to support design evolution
  - Reduced errors and time during integration and testing
  - Automated document generation
- Leveraging the models across life cycle
  - Support operator training on the use of the system
  - Support diagnostics and maintenance of the system
- Enhanced knowledge transfer

  - Capture of existing and legacy designs
  - Efficient access and modification of the information

Haskins expands on this, adding that MBSE *"improves knowledge capture and reuse leading to reduced cycle time"* and that it improves the *"capacity to teach and learn SE, to integrate new team members, to minimize loss of knowledge as team members leave [and] to establish shared mental models"* [5, p. 1]

MBSE is suffering from low adoption rates. Haskins explains this by pointing to the problem of not yet having established a common technical ground for the practice of MBSE, there is no culture for practising MBSE and there are few 'success stories' to rely on, and - of course - it is the case of money. MBSE is relying on more research to gain higher implementation to get in a position where anyone is willing to spend money evolving the discipline to its - perceived - deserved position. [5]

2.2 Model-based Systems Engineering practices

The following section will describe the traditional SE practices of requirements and functional analysis, and system architecture with a focus on visual representation or, i.e., MBSE.

2.2.1 Functional analysis

The functional analysis is an effort to identify the behaviour of the system-of-interest, detailing the behaviours to leave no doubt as to the intended purpose of every function. The result is a model describing the system as a whole, serving as a basis for the development of the actual physical solution [8, J.1]. The functional analysis answers the question; "what is the system supposed to do?", i.e. the functions make up the "what". The physical architecture establishes the "how", as will be shown in Section 2.2.2. The system may be a product, such as a car, or a manufacturing process where the functional analysis describes the process of manufacturing the product.

Input to the functional analysis, according to the INCOSE Systems Engineering Handbook [8, J.1]:

- Functional requirements
- Performance requirements
- Architectural requirements

- Program decision requirements (such as objectives to reuse certain hardware & software, or use commercial off-the-shelf items)
- Specifications and Standards requirements
- Concept of operations (description of how the system will be operated to meet stakeholder expectations, from an operational perspective [14, p. 35].)
- Constraints

The functional analysis relies heavily on the functional requirements. Note however, that the functional analysis may be commenced before the performance requirements or constraints are formulated. The functional analyses are in such cases used as tools to help identify and refine these requirements [9, pp. 148-150], [14, p. 46].

The output from the functional analysis varies in format depending on what stage or iteration the project has entered, and on the favoured techniques within the organization. The most common and important output is:

- Functional hierarchy
- Functional flow block diagrams
- Linkage between function and requirement

The functions are structured in a hierarchy [9, pp. 32-34]. An extension of these hierarchies are so-called *functional flow block diagrams*. These will be addressed in Section 2.2.1.

The functional analysis is not finished till every requirement has been allocated to a function. It is imperative that this link is established, as the requirements state why a function exists and how well it is expected to perform.

Functional hierarchies

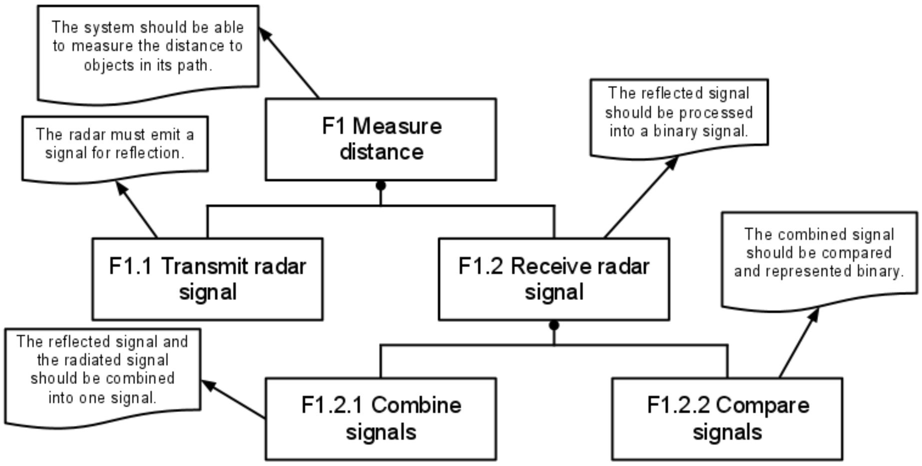


Figure 2.10: A hierarchical tree-structure of functions, with functional requirements.

The functional requirements provide the basis for the functional analysis. The first step is to determine the logical top-level functions. Functions on the top-most levels are not transparent, requiring decomposition in order to gain full understanding of their inner structure. This insight is achieved by breaking - decomposing - the function into sub-functions [20, p. 150]. The functional requirements are also decomposed and assigned to their corresponding sub-functions [8, J.1], [14, p. 49].

Performance requirements are then identified for each functional requirement and flowed down through the function tree [8, J]. This provides a strong link between the two analyses (requirements and functional), where functional requirements act as input to the functional analysis, and the resulting functional hierarchy acts as input to the analysis of performance requirements [8, p. I.6]. The act of identifying system functions (and also the physical architecture, see 2.2.2) helps define the performance requirements. This iteration is visible in Figure 2.3, where all the phases exchange information with each other. Further requirements are identified and distributed subsequently, e.g. design constraints and timing requirements [8, J].

Obtaining non-deterministic models

As stated in the opening paragraph of this section, the functional hierarchy does not express the solution; it is the “what”, not the “how”. It is imperative that the functions and sub-functions are unambiguous [i] and non-deterministic [ii] in order to facilitate the search

[i] Not open to more than one interpretation [19]

[ii] Does not dictate a solution [19]

for solutions [20, p. 150]. The use of *generally valid sub-functions* [20, p. 36], [20, p. 151] is recommended. These sub-functions are listed in Appendix A [9, p. 38], [20, p. 157]. Standardizing the functional structure helps recognizing known solutions, or in identifying recurring sub-functions that may be solved by fewer physical components to avoid redundancy (or the opposite; to obtain redundancy, in the case of Safety Engineering), or to identify functions that require new-development to be solved [20, p. 151]. Standards simplify communication between engineers, within or across disciplines - and among Systems Engineers. They also appear in design catalogues [20, p. 151].

The generally valid sub-functions are based on the three types of media that all systems operate on; *information*, *material* and *energy*. Information - due to its massive variation of elements - is further divided into *signal elements* and *data elements*, leaving four classes of system functional elements (with regards to Kossiakoff and Sweet [9, pp. 36-38]):

- Signal elements, which sense and communicate information
- Data elements, which interpret, organize, and manipulate information
- Material elements, which provide structure and transformation of materials.
- Energy elements, which provide energy and motive power.

The sub-functions supplied in Appendix A, all fulfill three criteria; *significance*, i.e. performing a distinct and significant function, *singularity*, i.e. falls within a single engineering discipline, and *commonality*, i.e. found in a wide variety of system types [9, p. 37].

Adding sequence and input/output to functions

A weakness associated with a strict hierarchical structure is the obvious lack of relations between functions. Functional Flow Block Diagrams (FFBDs) seek to establish these relationships [iii].

An FFBD contains the same functions as the functional hierarchy. However, the FFBD adds sequence and input/output (I/O) to the functions [iv]. Arrows show the order in which the functions are executed. I/O is flowed between functions whenever they communicate or otherwise interact with each other [8, J.2.1], thus providing the relation between functions.

FFBDs support concurrency and selection. Concurrent - or parallel - functions are separated by a node, typically marked “A” or “AND”. The node indicating a selection is typically marked “OR”. Iterations [v] and loops [vi] are also supported, typically drawn as an arrow returning to a previous function from a decision gate [15, pp. 68-70].

[iii] FFBDs were originally developed to describe the ballistic behaviour of missiles.

[iv] When FFBDs contain I/O they are referred to as “behaviour diagrams” [15, p. 70] or “Enhanced FF-BDs” [11, p. 6]. This report will refer to them only as FFBDs.

[v] Repeating an action a certain number of times.

[vi] Repeating an action until a predetermined condition is met.

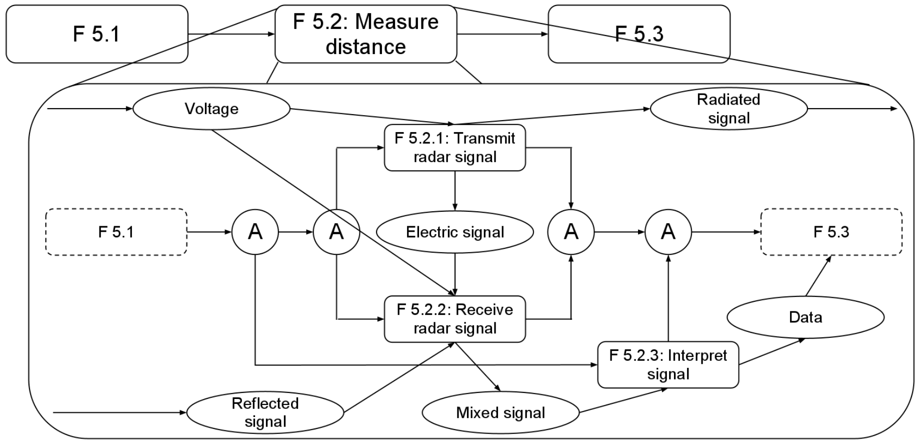


Figure 2.11: Example FFBD of the top level functions of a radar system.

Figure 2.11 takes the top level functions of a radar system as an example of a simple FFBD. Square boxes are functions, ovals are I/O and circles are nodes. The functions have unique IDs indicating where in the functional hierarchy they belong, cf Figure 2.10. The diagram tells the reader a decision has been made to use a radar to measure distance. It also shows how an FFBD may reduce the size of the solution space; the transmission, reception and calculation are parallel functions. This concurrent behaviour describes the functionality of a continuous-wave radar, as opposed to the serial functions of a pulse radar. This choice of technology is typically a reflection of the system requirements. Yet, such decisions provide the systems engineers with more data for the next visit to the requirements stage [14, pp. 42-44].

Finding an optimal solution is an NP-Complete [vii] problem [15, p. 70]. Narrowing down the solution space is the best way of handling such problems, which is exactly what is achieved by working iteratively with the functions and their flow.

2.2.2 Physical definition

The physical definition represents the actual solution, telling how the system shall perform its defined functions [15, p. 36]. The physical representation shows the system components, their parts and how they are coupled. This is the architecture of the system. Building the final architecture is an iterative process, using trade-off studies to evaluate architecture alternatives [8, K].

Where requirements say "why" a system is built and the functional analysis say "what" it is supposed to do, the physical definition tells "how" a system works, i.e. what components and technologies are selected to satisfy the requirements.

[vii] Problems without a known efficient solution algorithm.

Input to the physical definition phase:

- Requirements
- Functional architecture
- Business case, including budget goals or limits
- Technology analysis
- Similar systems

The functional analysis acts as input to the physical definition phase. Every element in the physical architecture shall correspond to a function in the functional hierarchy, i.e. the physical element is the realization of the function.

The business case states the financial constraints for the system. The technology analysis is often performed at the pre-concept stage, laying a foundation for what can be reused and needs to be developed or upgraded. Similar systems are rivalling systems or older systems that may contain reproducible solutions.

Output from the physical definition phase:

- Logical architecture
- Part tree
- Documented definition of interfaces
- Documented justification for the selected components

The result of the physical definition phase is the Part Tree. This tree contains all components, down to single parts such as nails and wires, i.e. the Bill of Materials (BOM). This can be achieved using one of two - or both - strategies, either divide the components into logical categories or in a strict as-built architecture (BOM).

Logical architecture

When exploring alternative concepts, it is useful to regard the system in its logical elements. That is, classifying the system elements in categories [8, K.1], [15, pp. 37-38]. For instance in disciplines (such as “software” or “hardware”) or by the function they perform (such as “sensors” and “processing units”). To sort logically, means assigning “true”/“false” relationships to the sorting elements. In this context, an element either belongs in a category (true) or it does not (false).

The advantages of regarding a system logically are many. For instance, one’s understanding of the technical requirements and the relationships among requirements are further improved during the logical decomposition [14, p. 49]. Also, searching for a particular component in the logical tree is much easier than searching for the same component in the part tree. The difference being, that one must have an insight in the composition of the part tree to find the component. To understand the logical tree, one need only know what type of component one's looking for in order to understand the categorization of the logical tree.

Another advantage, is that the logical architecture may also include the wide range of concept alternatives, making the tree act as a taxonomy of available solutions [15, p. 38]. When trade-off studies are performed and alternatives are subsequently discarded, rationales are added to the different alternatives explaining why the certain concept has been chosen, or not. Knowing which concepts were considered and what criteria they failed to fulfill is valuable during future revisions of the system. Figure 2.12 shows an example of a logical structure for the logical element "Sensors" in a perceived system. Note the rationales added to the discarded concepts. The radar is the only concept that fulfilled all of the top level requirements.

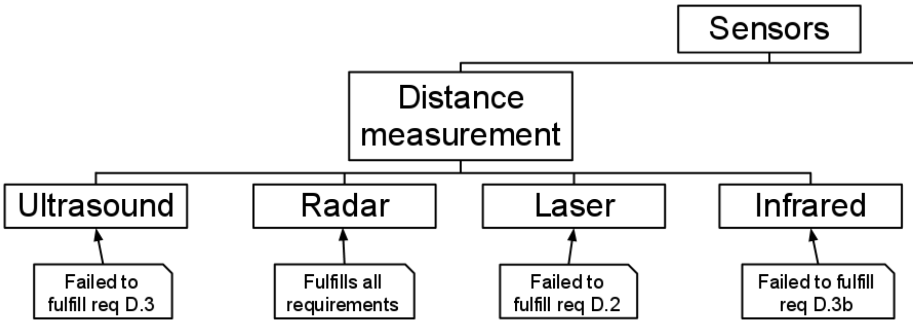


Figure 2.12: The "Sensors" section of a logical architecture.

As-built architecture

The as-built architecture is the hierarchy that describes how to build or assemble the product. This means that all components on one level of the hierarchy together make up the one component on the level above, as illustrated in Figure 2.13.

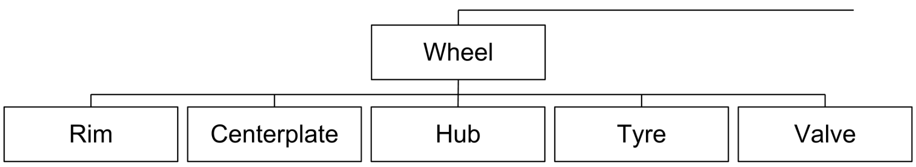


Figure 2.13: The rim, centerplate, hub, tyre and valve make up a wheel of DNVFF2.

Interfaces and modular design

Another important part of the physical definition is the identification of interfaces. A large fraction of system failures occur at interfaces, and identifying and defining them is paramount to secure system functionality [9, p. 46]. Interfaces are, basically, where an interaction occurs, e.g. human-machine, mechanical-electrical or mechanical-mechanical interactions. Interface elements are made up of three types; *connectors*, i.e. transmission of electricity, fluid, force, etc, *isolators* that inhibits interaction, and *converters* which alter the form of the interaction [9, pp. 44-46]. Interfaces should also be clear, stable and decoupled [23, p. 89] in order to facilitate modular design.

A modular design is a design where the interactions between components and sub-systems are kept at a minimum [9, p. 245]. The advantage is a system of wide variety, with a parallel development process where every component or sub-system can be tested alone, independent of the surrounding system. The resulting system consists of components that need no further adjustment upon installation, making upgrading or repairing the system easy [9, p. 245], [23, pp. 89-90]. Disadvantages include an often more expensive system that may be easily re-engineered by rivals. The overall performance of the system may also suffer [6, pp. 163-163].

An example of a highly modular design, is the Walther GSP competition sport pistol. Its frame supports both low and high caliber functionality, simply by exchanging the top piece which contains the barrel, the breech block and the slide casing for either .22 or .32 ammunition. The trigger mechanism and magazine port are designed to work with both top pieces. In addition, the parts which takes the most wear - trigger, grip and breech block - are all easily replaceable, as is the optics and barrel. This makes the weapon highly configurable.

The trigger mechanism is a good example of a particularly modular design. It has only three external interfaces; the human-mechanical interface in the trigger, mechanical-mechanical between the hammer and the pin in the breech block, and the actual shape of the mechanism. Any type of trigger mechanism that maintains these interfaces may be inserted into the frame.

The modular design of the Walther GSP does not affect its performance as a precision weapon, indeed it is one of the best precision weapons in the market. However, it makes it a single purpose weapon with a low overall performance. The square features makes it unattractive to the eye, it is heavy and it takes a long time to disassemble. It is built to last and to be used on a daily basis, using robust materials. It is unfit as a military or police weapon, yet superb for athletes.

Visualizing interfaces with N2-diagrams

The N-Squared - or just N2 - diagram was developed specifically for specifying interfaces. It may also be used as a supplement for FFBDs to show the flow of I/O.

The N2 diagram is an N x N matrix, where the N elements are positioned along the diagonal instead of along the sides. Links between elements are shown by adding an interface element in the cell at the intersection between the row of the first element and the column of the second element. [11, pp. 3-4] Figure 2.14 shows the interfaces of the Wheel sub-system which is shown as a breakdown structure in Figure 2.13.

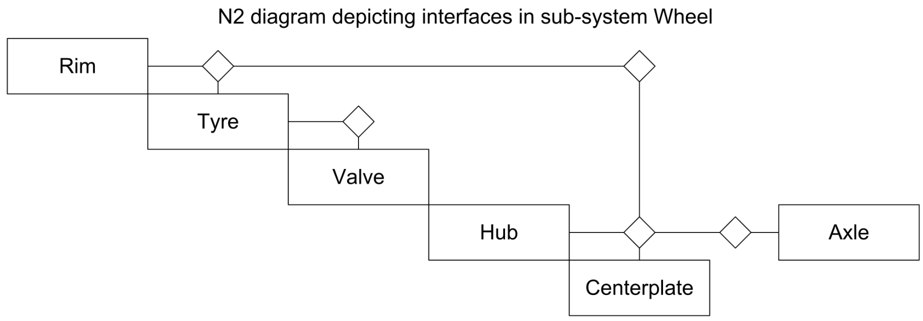


Figure 2.14: Example of a N2 diagram showing the interfaces for sub-system Wheel, which is shown in Figure 2.13. Diamonds indicate an interface. External interfaces are shown off the diagonal on the right-hand side, as shown with the hub-axle interface.

2.3 Lean Thinking

The following section will give an introduction to the field of Lean Thinking. A brief overview of the history of Lean will be presented, as well as a definition of some core techniques in Lean that will be further elaborated in Chapter 3. With these concepts in place, the remainder of this section will elaborate on the six principles on which Lean is funded; Customer value, Value stream, Continuous flow, Pull of value by customer, Pursuit of perfection and Respect for people.

2.3.1 History and basics of Lean Thinking

> *"**Lean Thinking:** the dynamic, knowledge-driven, and customer-focused process through which all people in a defined enterprise continuously eliminate waste with the goal of creating value."*
>
> \- Murman et al, 2002[17, p. 32]

In the 1980s, Womack et al. performed an assessment of leading American, European and Japanese car manufacturers. The results were published in 1990, and showed that the Japanese manufacturers were able to produce cars at a mere 67% and 46% of the time spent by their European and American competitors, respectively. In addition, the Japanese manufacturers' defect rate was lower than that of the Western companies. [7, p. 7]

Womack et al. went on to investigate the reason behind these vast differences. The output was a term Womack called "Lean Production", being a way of producing where the central vision is *an uninterrupted, continuously flowing, value stream which delivers the desired customer value with the least waste of resources in the shortest time possible.* [7, p. 8]

Lean was derived from studying the Toyota Production System (TPS), a system developed at Toyota since the 1950s. Its purpose was to achieve high flexibility in the production system due to Toyota's situation at the time as a supplier of many different products for small-volume markets. [7, p. 7]

To anyone unfamiliar with the practices of Lean Thinking, the mindset may seem awkward and backwards. Lean consists of pushing production start as close to release date as possible, starting the production preparations well before design is finished, prototyping and testing is done rapidly, suppliers are tightly integrated in the production system instead of being played against each other to force lower prices, and a bottom-up approach to planning and control where the employees plan the process as opposed to having this planning done by management.

To the uninvited, this may sound like the recipe for a product that does not reach its release date, whose quality is poor, which fails due to poor testing, is too expensive due to not pushing prices from suppliers and where the responsibility of the failure is put on the employees rather than management.

In fact, when implemented correctly the reality is quite different.

The implementation of Lean Thinking has proven the theory to hold true, especially in the cases of Toyota and Honda. By spending more time in the concept phase, making more detailed planning of the production phase and integrating this planning as a part of the design phase, by developing robust standards for testing and lowering the bar for producing prototypes, by working closely with suppliers and teaching them how to supply in an efficient manner, and by elevating the sense of ownership among employees by trusting them with more responsibility the outcome is a cheaper product with a higher quality and a shorter lead time. [16, pp. 352-374], [7, pp. 7-8]

The application of Lean Thinking consists of continuously improving the enterprise, through identifying and eliminating activities that do not add value to the end-product, and capturing knowledge gained in projects and spread this knowledge across teams and divisions. The idea is to do incremental improvements, meaning improving one small thing at a time, rather than making major leaps at long intervals, [7, p. 12], [16, pp. 367-371] illustrated in Figure 2.15. Continuous focus on improvements keeps people ready and willing to make the effort, - also, being often reminded that the improvements help keeps people motivated to make the effort.

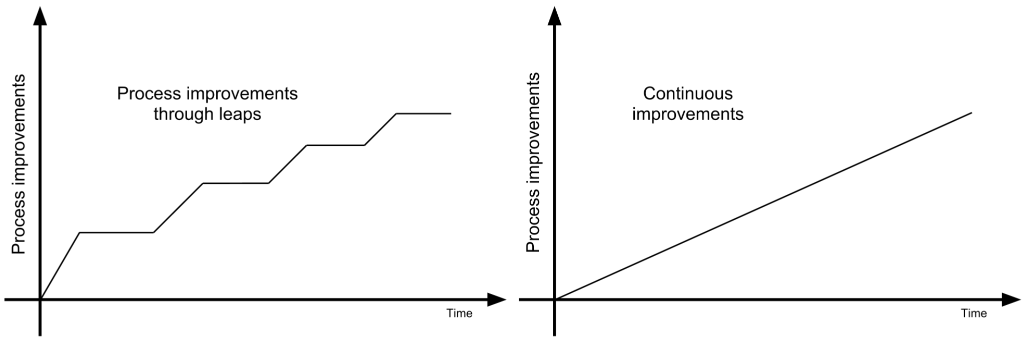


Figure 2.15: Process improvements through leaps vs continuous improvements.

Two important techniques employed in Lean enterprises Visual Workflow Management (VWM) and Knowledge management. Both of these will be addressed in more detail in Section 2.5.

Visual workflow management VWM is showing project progress to project participants in a graphical manner, using whiteboards with color codings, schematics and Knowledge Briefs (K-briefs). In combination with Stand-up meetings, this technique allows project participants to stay up to date on what's going on, communicate to co-workers when and how they can help, and coordinate work efforts. The color coding allows users to quickly get an overview of what's going on and the current status of the project. This technique will be further elaborated in Section 2.5.1.

Knowledge management Knowledge is a resource, like money or material. Knowledge is the treasure of an organization, and should be accumulated and stored like the treasure of kings of old. Everyone knows the expression *"knowledge is power"*. However, knowledge management is a new discipline in Western industry. Not till the 1990's did the competitive advantage of knowledge become apparent to the West, after realizing the Japanese domination was much due to a strong culture of learning within the Japanese companies.

Knowledge means knowing how to do that which you need to do. It means spending more time exploring alternative design concepts, and to test early and often. Knowledge is captured in so-called *"learning events"*, where the goal of the event is to learn, capture the knowledge gained and store it. [13, pp. 191-209] This field will be further elaborated in Section 2.5.3.

2.3.2 A definition of value in product development

A major focus when improving production or development processes using Lean Thinking is on identifying and eliminating *waste* in order to create as much *value* as possible. To fully understand the Lean principles in the next section, one must have a clear definition of value.

Oppenheim defines value as *"the delivery of a complex system satisfying all stakeholders, which implies a flawless product or mission delivered with at minimum cost, in the shortest possible schedule, fully satisfying the customer and other stakeholders during the product or mission lifecycle."* [17, p. 33]

For production systems, Lean operates with three types of activities; *value-added activities*, *required non-value-added activities* (also known as *type 1 waste*) and *non-value-added activities* (also known as *type 2 waste*).

Value-added activities are activities that the customers are willing to pay for and which reduce risk and uncertainty. Also, the activities need to be done correctly the first time to be value-added.

Required non-value-added activities are activities that do not fit into the above definition, yet cannot be eliminated due to law, company mandate, current technology, etc.

Non-value-added activities are pure waste, and should be eliminated. Examples are unneeded reports and inspections, waiting and idle time.

Product development differs from production. Given a definition of customer as one who receives output from a preceding process or phase, production can be said to only have one customer; the end-user. Product development, on the other hand, has to satisfy a range of customers; stakeholders, end-users and down-stream functions such as manufacturing and maintenance. Product development's output is information on manufacturability, functionality, usability and serviceability which is delivered to the next phases of the product's life cycle. Thus, value-creation in manufacturing is limited by the product and production processes, while value-creation in product development must take into account a much wider range of customers. [7, pp. 14-16]

For product development Mascitelli operates with a different definition from that of production, saying *"a design/development activity is value-added if it transforms a new product design (or the essential deliverables needed to commercialize it) such that the product's profit margin and/or market share are positively impacted."* [13, p. 59]

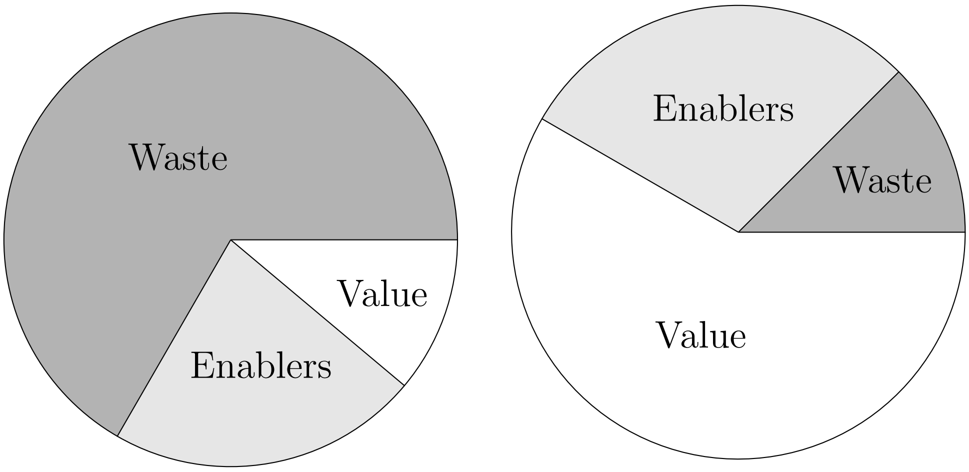
Based on this definition for value in product development, Mascitelli divides activities into the following three categories: *value-added activities*, *enablers* and *waste*. Mascitelli proceeds by defining the three types of activities in this manner:

Value-added activities are activities that meet the definition of value. For product development this means transforming new designs and commercializing deliverables.

Enablers are activities that do not directly add value, yet provide essential support to the process and which yields a positive return on the time invested. I.e. these are activities that enable designers to spend more time on value-added activities.

Waste is everything that does not fit any of the above definitions. [13, p. 60]

The goal of implementing Lean Thinking is to reduce the amount of waste in product development while increasing the amount of value. Mascitelli argues that *"performing enabling activities better and faster should be a key objective of any lean product development process"* to ensure a vast reduction in waste. Figure 2.16 illustrates this point. [13, p. 61]



(a) Current state (b) Lean future state

Figure 2.16: The amount of waste in product development before and after implementing Lean Thinking, according to Mascitelli. [13, p. 61]

2.3.3 The six principles of Lean

Womack and Jones identify five principles that characterize Lean thinking; *Customer value*, *Value stream*, *Continuous flow*, *Pull of value by customer* and *Pursuit of perfection.* [7, p. 9] In addition, Oppenheim includes a sixth principle in his Lean Enablers for Systems Engineering (LEfSE), called *Respect for people.* Although presented as separate entities, these principles are not independent of each other. Rather, on the contrary they are deeply interwoven and dependent of each other. To understand Lean, one must understand all six principles.

Customer value

In Lean Thinking, the customer pulls value through the development process. This means providing the right product in the right way, and is achieved through an intensive focus on defining the customer base, understanding its needs and wants and deriving clear, measurable requirements based on these analyzes. Also, it is important to clarify what value the organization wants to deliver in order to specify a market segment to focus on - often, this is where the development should start. Oppenheim defines value in Lean product development flow as *"a robust product satisfying stakeholders' functional and contractual requirements and expectations within a short schedule and at minimum cost."* [16, pp. 359-360], [7, pp. 9-10]

Map the value stream

Mapping the value stream means to identify all the activities necessary to bring the product to market, the sequence of these activities and the major input and output between them. In this process, all non-value-added activities (enablers and waste) should be identified and reduced to a minimum. In order to achieve this, a solid definition of value in the specific setting is important (output from applying principle one). Also, to ensure a perfect flow, one should consider the value streams of external processes, such as suppliers and transport. [7, p. 10]

To achieve a smooth value stream flow, one should seek the *critical path* using queuing theory. Critical path is the correct sequencing of concurrent tasks, such that the output of one process is immediately used by another process.

Oppenheim recommends mapping the current state and to refine this mapping into a future, desired state of which to aspire for. Generally, Oppenheim always recommends parsing the value stream into Takt Periods of one week each, where every week has clear goals of what should be achieved. Every Takt Period is ended with an Integration Event, where the team gathers to discuss designs, lessons learnt and future work. Oppenheim encourages the use of Program Rooms - "War rooms" - where all team meetings are held and where the walls are dedicated to show the value stream map. [16, pp. 360-365]

This term will be referred to as Value Stream Mapping (VSM) from this point on.

Continuous flow

The idea of flow is to streamline the transitions between phases in production or product development. Oppenheim emphasizes this as one of the main pillars in Lean, stating that *"Lean particularly focuses on streamlining flow between the processes"*. [17, p. 30]

Lean advocates cutting storage time, operating in a *just-in-time* manner where goods are always in motion, and products are put directly into transport after completion. With the Value Stream Map as a basis, the work can be planned to achieve a continuous flow of information in the product development process. Waste in this context means waiting; either a developer lacking information to maintain its work flow - or information waiting to be used, meaning it has been produced too early.

A way of managing flow, is to reduce batch sizes. Work on small batches at the time, and finish this work before starting on the next batch. To achieve this in product development, Oppenheim proposes the use of Takt Periods, as presented in the previous section. The Takt Periods provide absolute, non-negotiable common deadlines, they impose a sense of urgency and they resemble working on a moving line where it is immediately inherent if one is lagging behind. Also, the Takt Periods allows a predictable flow of the Value Stream and program progress. [16, p. 365]

Pull of value by customer

Letting the customer pull value is yet another implementation of just-in-time in Lean. The customer - using the same definition as presented in Section 2.3.2 - triggers all processes along the value stream, meaning that no product - nor a piece of information - is produced without having a customer waiting at the receiving end. Hoppmann divides this into two levels; micro and macro. On the macro level, the customer is the external customer; the end-user. The micro level is thus the internal flow and the customer is the down-stream function in the value stream. The up-stream function does not produce its output without a demand from the down-stream function.

The advantages of pull, is that the organization does not need to rely on uncertain predictions made by forecasts which may lead to over- or under-production. To fully reap the benefits of pull, the organization needs to establish strict communication routines to ensure the appropriate functions are contacted and given the correct information. The correct information includes who is the recipient, what are the needs of the recipient and the details of the transaction. [7, pp. 11-12], [16, p. 367]

Pursuit of perfection

To achieve the successful Lean organization, it is not enough simply to apply the four preceding principles. Applying these principles is itself an iterative process, where the organization stops and considers itself, searching for yet more areas to improve in or at Lean enablers that are not implemented to an adequate level.

In Lean, these improvements are done in a bottom-up fashion, unlike the more common way where management introduce changes to processes and force them onto the lower levels. Toyota utilize so-called “Kaizen” events, where the employees themselves set their competencies and knowledge to use to improve their own working environment with the goal of making their job better and more efficient.

Also, in order to achieve perfection, the project has to systematically avoid errors. The later in the project an error occurs, the more costly it is. An important enabler related to this principle is the identification of risks and the mitigation of these risks. The subsequent directly value-added activities are thus the ones that actively mitigate the risks.

Oppenheim lends the following division of risks from Hastings and MacManus: lack of knowledge, lack of definition/specification, lack of statistical characterization, known unknowns and unknown unknowns. He also lends the following mitigations: margins, redundancy, design choices, design space exploration, portfolios and real options, verification and test, generality, upgradeability and modularity.

Team training is imperative to the success of the fifth principle. The team needs to be schooled in Lean Thinking, and need to be onboard with Value Stream Mapping, Takt Periods and learnt to hate waste and to actively pursue a waste-free environment. Every team member is empowered to halt the project at any time, and bring the team to attention to concerns or issues that need to be resolved. [16, pp. 367-371], [7, p. 12]

2.4 Lean Systems Engineering

There is wide acceptance within academia and industry for the application of SE, although the recommended amount of SE effort varies across the field. Supporters of SE often use the numerous successful National Aeronautics and Space Administration (NASA) missions as arguments for the application of SE, such as the 60 successful military satellite launches in a row without failure, only two unsuccessful space shuttle flights (the Challenger and Colombia accidents) and the construction and operation of the international space station [18, p. 43].

However, critics of SE also use NASA as examples of the shortcomings of SE, pointing to "fuzzy life cycle definitions and hazy boundaries between SE and other domains" [18, p. 43] allowing satellite crashes to occur, the use of hazardous materials in satellites making retirement of said satellites difficult, entire development programs end up producing useless systems, and even the Challenger and Colombia accidents are blamed on poor SE efforts [18, pp. 43-44].

There seems to be a problem related to properly implementing SE. Oppenheim states in his book that the SE effort contains a lot of waste. In fact, Oppenheim proposes that as much as 88% of the SE effort adds no value to the project [18, pp. 44-45]. INCOSE seems to agree with Oppenheim, and has formed a work group called Lean Systems Engineering Working Group (LSE WG) whose purpose is to streamline the SE effort by cutting waste and identifying the value-adding processes of SE. The LSE WG states that:

> *"Traditional Systems Engineering is a practice which has a lot of strengths, but is not as good as it could be."* [18, p. 42]

Oppenheim strongly encourages spending more time in the concept phase, using trained systems engineers and, in particular, spending much time defining crystal clear and unambiguous requirements. [18, pp. 51-53] On these requirements the entire project stands. Section 2.2.1 will cover Functional analysis, a much employed technique used by systems engineers. This is the phase following the requirements definition phase, yet, as the reader will see - and which has already been hinted to in Figure 2.3 - there is a strong link between the two phases. Gaining full overview of a systems functionality is thus a strong value-adding process in SE.

2.5 Lean techniques

2.5.1 Stand-up meetings and Visual Workflow Management

Stand-up meetings

Stand-up meetings are short get-togethers held frequently, where team members communicate to the rest of the team what they have done since the last meeting, what they will have done by the next meeting, issues that prevent them from performing their tasks and what they might be needing help with. These meetings are derived from a need to have more frequent consultations than the weekly or monthly summaries in order to maintain focus on making progress. As the name implies, the participants should be standing up during these meetings.

Engineers and scientists are often poor at being productive. [13, pp. 81-84] Being proud beings always attempting to attain perfection, the engineers may get too immersed in the problem at hand as to stop and regard the project progression. A result may be a project which misses big milestones and deliverables.

The Stand-up meeting is meant to combat such situations. The shape of these meetings are inspired by football huddles and the quick briefings held by factory, restaurant and retail workers at the start of a shift. [13, pp. 81-84] In these short meetings, tactics or daily tasks and goals are communicated, as well as the *status quo*, and issues in need of special attention.

The Stand-up is a form of *just-in-time workflow management*, meaning sudden changes in project status may be communicated and dealt with shortly. Responding quickly to alterations is critical for maximizing productivity, directing the team along the ever-winding path of a project.

The benefits of employing Stand-up meetings are summarized as:

- Frequent resynchronization of efforts
- Coordination
- Communication
- Allows a team to flexibly adapt to current conditions [13, pp. 81-84]

It is important to note that *the Stand-ups belong to the team, not the team leader.* [13, p. 87]

Performing the Stand-up meeting

An idea behind the Stand-up, is to impose a sense of urgency on the worker. This is achieved by keeping the meeting short and efficient, preferably no longer than fifteen minutes. Also, it is important not to deep-dive into technical detail. [13, p. 84] Instead, when a discussion arises the team should react to this and request that the participants of the discussion continue after the meeting. [16, p. 362]

Engineers respond best to strict routine, meaning the time limit should not be exceeded as the engineers may respond negatively to this over time and lose interest in the Stand-up meetings.

Each team member should address three specific questions, namely; *"what has been accomplished since last meeting"*, *"what actions must be completed by next meeting"*, and *"what issues or obstacles might prevent the team from achieving those goals"*. [13, p. 84]

The ideal time for having these meetings is early in the work-day. Alternatively, just before lunch. The time must be chosen based on when most of the team members are regularly available. Meetings late in the day is discouraged. Location is unimportant as long as it does not cause inconvenience. [13, pp. 84-85]

The frequency of the Stand-ups should vary throughout the project, based on the current project phase. As the meetings are intended for keeping the team updated between weekly or monthly meetings, the need for these updates is decided on how rapidly things change in the project. Just before deliverables or milestones, the need for such meetings is high as the collaboration intensifies. In such cases the meetings should be held on a daily basis, maybe even twice a day. In slow periods once a week may be enough. Generally, the ideal number of Stand-ups is three times a week. [13, pp. 85-86]

In summary:

- Keep it short, no more than 15 minutes
- Location is unimportant
- Do not go into technical detail
- Team members should address these 3 questions:

1. What has been accomplished since last meeting
2. What actions must be completed by next meeting
3. What issues or obstacles might prevent the team from achieving those goals

- Decide frequency based on project type and status
  - Ideally 3 times a week
  - At least once a day during crunch periods
- Have the meeting in the morning or just before lunch

Visual Workflow Management

The Stand-up is only the first half of managing workflow. To get full benefit from the Stand-up, the communication needs to be visual. Visual communication means using colors, graphics and symbology to make information clear and readily understood. [13, p. 83]

A popular form of visual communication, is the Visual Project Board (VPB). This is a wall-space, blackboard or equivalent dedicated to a project. The contents of this wall-space is intended to capture the current status and progress of the project. It is important that the contents are intuitively understandable and interactive, to achieve real-time resolution of issues. [13, p. 88]

The VPB should display the current tasks being worked on and when they are due, as well as the milestones of the project and the progress relative to these dates. In addition, the VPB should capture unplanned work, and also act as a platform for sharing knowledge, e.g. through a space for K-briefs.

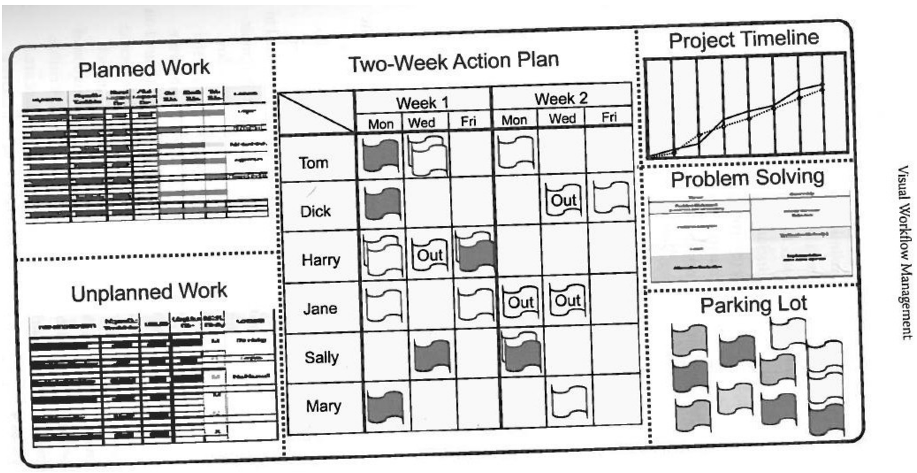


Figure 2.17: A VPB from Mascitelli, which he says is "[a] well-tested format for a VPB that captures the status, progress, and plans for a single development project". [13, p. 89]

A form of visualizing tasks, responsibilities and due-dates is through a tool called the *Wall-Gantt.* [13, pp. 97-102] This is a modification of the classic Gantt diagram, which uses lines to show start and end dates of tasks. These diagrams are not devised to provide the reader with the type of quick understanding which is required during a Stand-up meeting, nor is it easily updated on a whiteboard. The Wall-Gantt attempts to mitigate these shortcomings. It is made up of a grid, where every row is assigned to a team member, and every column is one day, with columns enough for two weeks. A task is written on the board on the day it is planned to be finished by. [13, pp. 97-98]

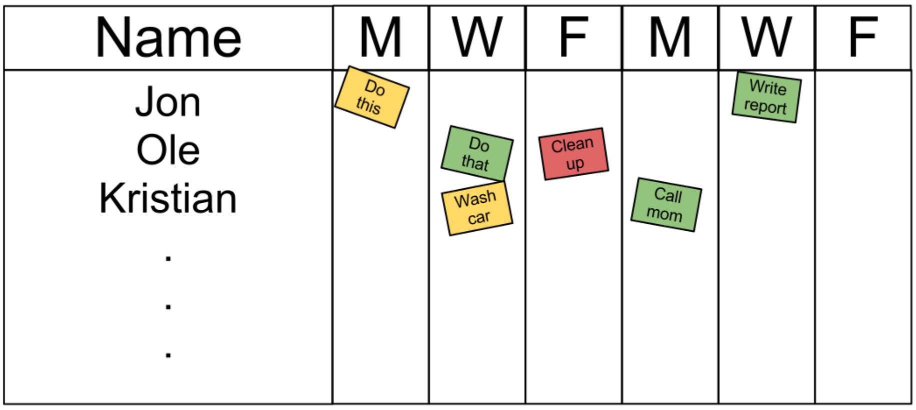


Figure 2.18: A Wall-Gantt.

In order to manage exceptions - such as tasks slipping, milestones being broken or important issues in need of quick resolution - it is recommended to use a range of colors for status indication. [13, p. 93] E.g. green means a task is on schedule, yellow means it has slipped once and red means it is critical.

For the Stand-ups to reach its objective of making team members aware of changes, the VPB must not be altered between meetings.

2.5.2 Responsibility-based planning

Responsibility-based Planning is the practice of involving the engineers in the planning process. The project manager only sets the major milestones of the project, the engineer details the development process up to these dates. Oppenheim recommends - as mentioned in Section 2.3.3 - dividing the project into Takt Periods, of one week durations.

In traditional development programs, activities are decided and delegated by management, so-called top-down planning or scientific management. However, it is strongly argued that this procedure is inferior to Responsibility-based Planning, despite the lower coordination effort. [7, p. 32]

By involving the engineers in such a manner, a sense of ownership is induced in the individual. The engineers are given a chance to comment on the deadlines, giving the progress plan a more realistic estimate. In exchange of being involved, the engineer is held accountable for his or her progress rate; they are responsible for holding their own deadlines. As the engineer has a good insight into the problem, potential risks and his or her own work rate, the result is a more robust schedule which is perceived to be less error-prone than the one provided by scientific management. [7, pp. 31-33]

The engineers must be able to track their own - and other's - progress. This may be achieved with tools such as VPBs.

The perceived advantages of Responsibility-based Planning are summarized as follows: [7, pp. 31-33]

- Tasks and activities are planned up-front
- Project progress can be verified against the planned activities
- Higher motivation and accountability of the individual engineer through a sense of ownership
- Reduces risk of schedule overruns
- Contributes to the continuous improvement of processes

2.5.3 Knowledge management

Organizational learning

Knowledge is the treasure of an organization. Projects start and end, people move between projects, they leave - or even die. The knowledge gained from a project often remains with the people who experienced it. In today's competitive world, knowledge becomes a resource more valuable than money.

Having much knowledge is also a way of mitigating risk. Many of the greatest risks a project faces is not knowing the possible outcomes of a decision. Having the possibility to search through a rich knowledge database - treasure vault, if you will - to find solutions that have already been tested and documented, is a great advantage to the organization. If the knowledge has not been created yet, the task of the current project is to create this knowledge, document it and store it in an accessible manner. [13, pp. 196-199]

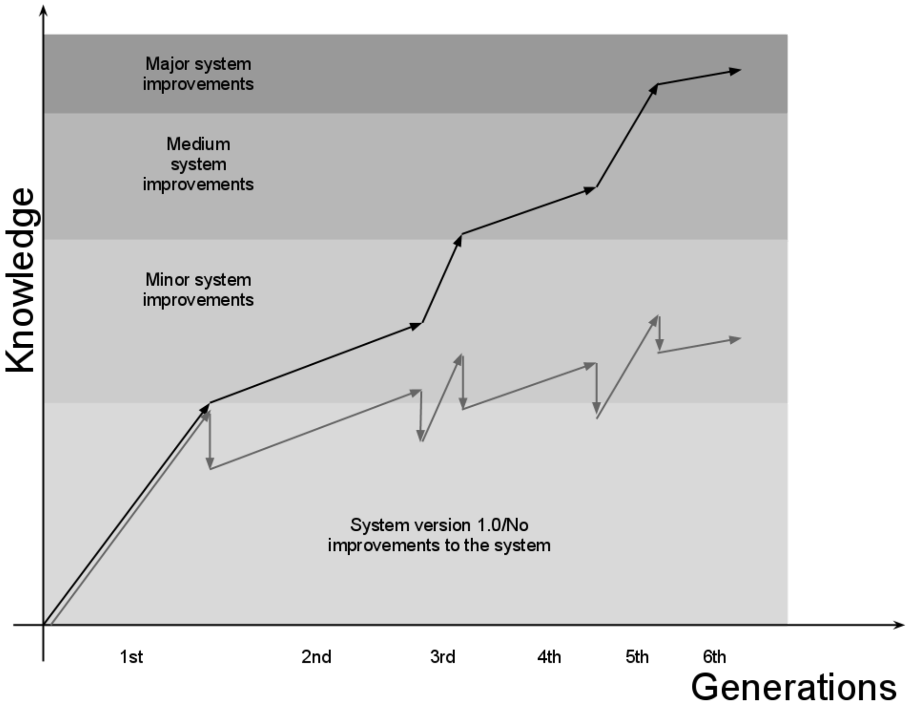


Figure 2.19: Chart showing the impact of knowledge loss on potential system improvements.

Figure 2.19 shows how knowledge affects the potential improvements to a system when new generations of engineers take over. The system may also refer to an organization, and the improvements are, e.g., enhancements in product development methods, group dynamics or new technical solutions. The lower curve shows how a drop of knowledge may occur when new members join the organization without the inherent knowledge of the ones who came before. Organizations that experience this drop does not capture and share knowledge; they keep recreating knowledge - a waste of time. And time is indeed the issue here; when knowledge must be created again, time becomes scarce and the new team is never able to build upon the knowledge gained by the old team. The result is a system that never manages to improve beyond its first version. The upper curve represents utopia; of course a drop of knowledge will always occur. The mission of knowledge management is to reduce the drop as much as possible.

Mascitelli identifies three essential elements that must be present in an organization to achieve organizational learning: [12, p. 127]

1. A commitment of time; learning takes time.
2. A strong motivation; is there a substantial benefit?
3. A humble attitude; noone knows everything.

To capture knowledge, one must standardize the work process to some extent. This is not met with enthusiasm in product development communities, where creativity and playfulness is encouraged. Mascitelli suggests using milestones and deliverables as opportunities for capturing knowledge. Deliverables are any tangible and transferable items that contributes to the commercialization of a new product, such as documents, drawings, decisions, reports or prototypes. [12, p. 95] At these points, the work performed over the last period, the lessons learnt and problems solved can be summarized and added to the knowledge database. [12, p. 128] This is one way of standardizing. Another way will be presented in the next section.

Knowledge briefs; the A3 method

One method of capturing knowledge that is gaining widespread support, is the *K-brief*, or A3 template. This is a method that stems from Toyota, which employs visual representation of a problem and its solution on a large sheet of paper, preferably A3. The idea is to represent the knowledge in a brief, visual and easily accessible format. The best way of doing this is by describing the process like a story, with illustrations and short pieces of text. [13, p. 205] The point of it all is to make the K-brief inviting to the reader, and not to scare the reader off with massive amounts of text and irrelevant background theory. If the reader needs more background theory, this is best presented by adding references to sources.

These K-briefs should be easy to make, and preferably standardized by a template. Mascitelli recommends a template that reflects the "plan-check-act-do" process. This is a way of solving problems through analyzing the problem, solving it based on the analysis, verification of the solution and implementation. Mascitelli proposes dividing the sheet into two columns. The left-hand side is devoted to the problem and the range of solutions, and the right-hand side is devoted to the best solution and implementation. The left-hand side contains the following topics: Problem statement, Goals and Alternative Evaluation. I.e. an understanding of the problem, what should be achieved by solving it and a list of alternative solutions. The right-hand side contains Countermeasure Selection, which is choosing the best solution, Verification Method(s), which describes how the solution will be tested, and Implementation and Follow-up Plan, which describes how the solution should be implemented in the organization and how the organization should behave in order for the solution to be effective. [13, pp. 204-209] Figure 2.20 shows an example of how such an A3 may be structured. [13, p. 206]

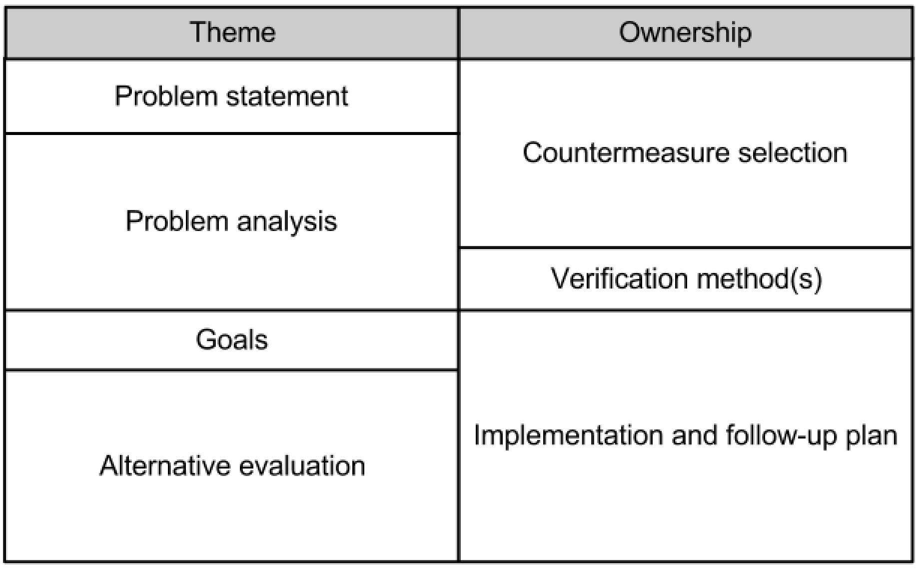


Figure 2.20: K-brief template according to the Plan-Do-Check-Act process.

Standardizing the K-briefs decreases the number of mistakes and omissions, and it saves time. [12, p. 129]

Mascitelli speaks of *process knowledge* and *technical knowledge* as the essential categories of information relevant to a product development organization. Process knowledge is best recorded using the Plan-Do-Check-Act template. For recording technical knowledge, though, Mascitelli recommends being more detailed in what type of data should be recorded. He proceeds with suggesting the following list of relevant data:

- Important design trade-offs and decisions
- Reusable design elements
- Solutions to critical-to-quality issues
- Solutions to critical-to-cost issues
- Performance curves
- Raw material/component data
- Test results for common design elements
- Reliability/environmental data
- Factory design rules/capability data
- Supplier design rules/capability data
- Frequently used parts/raw materials

These are the data that - according to Mascitelli - has the best long-term utility for an organization, being the type of data that withstand the decay of time. Data on specific types of technology may be obsolete within a year. The data represented in this list concerns mainly the decisions made and the data supporting these decisions. They say why a certain technology has been chosen and another discarded. [12, pp. 131-134]

Chapter 3

Methods

3.1 Lean assessment

3.1.1 Performing the assessment

The assessment was performed on the 30th of January 2012, at the DNVFF office. Nine team members participated. The assessment lasted four hours.

The participants were first given a short lecture about Lean thinking, then five different areas within Lean were presented to them. After lecturing about an area, the participants were given a questionnaire with questions about how their organization was utilizing the Lean principles presented in the lecture. The participants ranked the utilization with values from 1 to 5, 5 being full implementation and 1 being no implementation at all. The participants rated both the current state of the organization and the state that they would have liked to have achieved (the desired state). The difference between the two is called the "gap", and the bigger the gap the more urgent the implementation. The questionnaire is supplied in Appendix B.

Professor Terje Rølvåg supplied the author with a spreadsheet that included all the Lean enablers that Oppenheim proposes in his book. Upon entering the results from the assessment, the spreadsheet picks the most relevant Lean enablers based on the gap between the values depicting the current and the desired state of the organization. The author used the same gap limit as Rølvåg uses when the professor performs these assessment with companies from the industry, of 1.5. This means that any gap between current and desired value greater than 1.5 points to a need for improvement.

The author started the assessment by presenting some of the core ideas concerning Lean Product Development (LPD) in a Power Point presentation. The contents of this presentation was based on two Power Point presentations provided by Prof. Torgeir Welo which the professor has used when performing the same assessment on Norwegian industrial companies. Based on the contents of these two, a tailored presentation for the DNVFF2 was designed.

The presentation included the following subjects, in order of mentioning:

- Lean principles
- a definition of value-added activities, enablers and waste
- a definition of Lean Product Development
- the Lean Product Development Model
- Customer Focus
- Knowledge
- Stabilization
- Continuous Improvements
- Culture

After consulting with Prof. Welo, the subject of standardization was removed from the assessment. The lack of a manufacturing department at DNVFF led to this decision.

To make the presentation and topic as relevant and interesting to the participants as possible, the author divided the presentation into sections, ending every section with a discussion around the content, then allowing the team members to assess the current and desired state of the project. These scores were also discussed in plenum. The team responded well on being invited to discuss, and the session became an open discussion where the entire team participated. The author views this as a great benefit to the entire team; both in team building, in providing a common understanding of the team's goals and wishes, and in strengthening the team's unity.

3.1.2 Results of lean assessment

This section will present the results of discussions and assessment scores provided by the team. Firstly, the author would like to share one of the discussions held before the team started their assessments. The discussion was intended to increase the team's understanding of value-added activities, as well as "warming up" the team for the coming discussions. The outcome of the discussion was a definition of value in the context of the project.

When addressing value-added activities versus waste, the author chose to bring up a subject of which the team had had a number of discussions over the past week - but introduced a new perspective. The discussion was regarding whether or not the team should participate in a publicity stunt at the local Science Center. The stunt was a five-day appearance at the center, meaning all production activities would have to be moved there, with all the complications this may cause. The team was positive to the concept, yet fearful of the delay it could cause as well as of the elevated risk of damage to components. The new perspective the author wanted the team to assess, was one of value-added activities; was it value-added for the project to actually perform the stunt?

The team concluded 'yes'; the sponsors are paying the team for public appearances. Also, doing stunts of this sort increases the team's chances at winning the media award at the competition. It may even be argued that the stunt is an enabler; goodwill from this year's sponsors may help next year's team in renewing contracts. From this point of view, the stunt should be performed.

Before assessing each question, the questions were read out loud, and any uncertainties were addressed. The meaning of the term “functional department” varied from question to question, yet the team decided whether this meant team members, sub-systems or successive teams (team 2010, 2011 and so on) based on the context of the question or discussion.

Customer focus The team discussed the role of the customer in the project, and - indeed - who the customer really is. This is something the team had not addressed previously. From the discussion, the team concluded that they themselves are the customers. Yet, the sponsors and Shell - who are merely stakeholders - set the requirements. This makes the project unique, as the developers are also the customers. Yet, the developers have to adhere to requirements set by stakeholders.

With the developers also being the customers, the question of to what extent the team is working with the customer to understand current and future customer needs and wants became a question of the team members ability to communicate their desires and needs for the project, from their subjective views, as well as how the team considers the vehicle as a “deliverable” for the succeeding team.

The team assess their communication as free, open and unrestricted, feeling that everyone has been able to set their mark on the product. Also, the goal of winning the competition and setting a new world record is the common motivating factor that drives the team. The desire to achieve a high score on the thesis, as well as delivering a superb vehicle to the next year's team are also motivating factors shared by the entire team. Future customer needs and wants are thus addressed by building a modular vehicle that can be continuously improved over the following years. The team assess themselves at level three, desiring level four.

When asked how customer wants, needs and requirements reach the design engineers, this became a question about how the team reaches a common understanding of where the team wants to go and what they want to achieve. The team concludes that given the good communication within the team, the group knows well the wants and needs of every team member, driver, race host, etc. The team assess themselves at four, desiring five.

Figure 3.1 show the results of this part of the assessment.

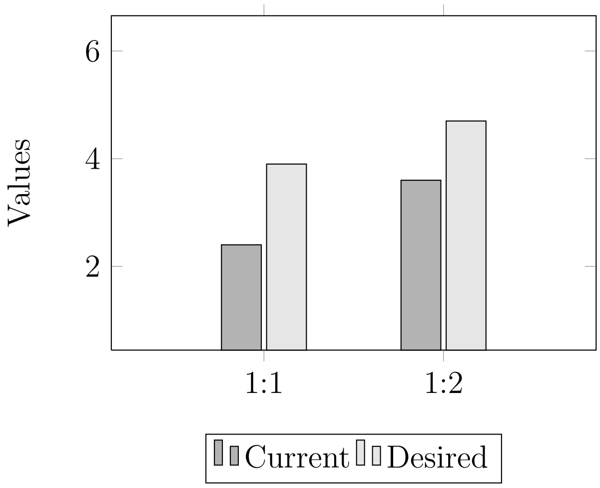


Figure 3.1: Average Current and Desired values

Knowledge When asked how the team rates knowledge, and whether the team considers its own collective knowledge as an asset, the team assessed themselves a two, desiring to be at level five. The reason for this low current score, is that the team does not focus on generating knowledge, nor document lessons learned for the next generation. The knowledge stays with the graduating team, apart from that which is written in the thesis. The thesis is not an easy read, making the knowledge difficult to find. Also, the contents of the thesis are chosen for the sake of a good thesis, not for spreading team knowledge.

The team does not have a database for storing knowledge, nor a "knowledge manager" responsible for gathering knowledge, structuring it and design standards for knowledge collection. The team thus rated themselves a two. The desired state is four, as the team perceives one knowledge manager for the entire team as sufficient.

When asked how knowledge is transferred between successive teams (called "functional departments" in the questionnaire), the team assessed themselves at a two, desiring to be at five. The team states that, sharing knowledge is not done systematically except for the one get-together with the graduating team at the beginning of the first semester. Where to search for knowledge is also unclear, beyond having access to the previous year's server area and thesis. However, there is no distrust between the teams, which elevates the project from being a category one. The team desires a structured system for storing and sharing knowledge, and to adopt a culture where knowledge is viewed as a common asset benefiting the entire project.

The team was introduced to Set-Based Concurrent Engineering (SBCE) during the presentation. When asked whether this is employed in the project, they assessed themselves at a level three, wanting to be at level five. A stronger focus on exploring multiple design concepts, defining clear requirements and doing trade-off analyzes is desired.

Figure 3.2 show the results of this part of the assessment.

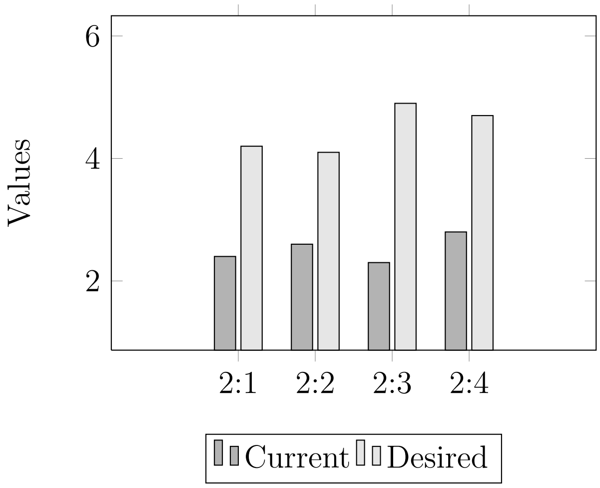


Figure 3.2: Average Current and Desired values

Stabilize The team members were divided on whether or not they get the resources they need, and whether the resource plan is adequate. Also, a discussion ensued whether a resource plan is applicable for this project. This was the only discussion that got a bit heated during the workshop. In the end, the team converged on a combined score of three, desiring four.

The team assesses its communication practice as very good, giving it a score of four. The flat management style allows free communication, and the fact that most of the team is co-located in the same office (or close by) helps a lot. However, the team is ambitious and wants to become even better at communication, assessing the desired state as a five. To achieve this the team wants to see more "walking management" and direct, brief communication.

Upon addressing the role of key suppliers and how they are treated in the organization, the team discussed the difference between suppliers and sponsors. The conclusion was that these are often the one and same; sponsors are also often suppliers. The team co-operates closely with a few of the major sponsors, sharing both technology and experience. The team thus scores itself at a level three. The team desires to achieve a level four, believing that involving the sponsors more may help seal new sponsor contracts the following year.

Figure 3.3 show the results of this part of the assessment.

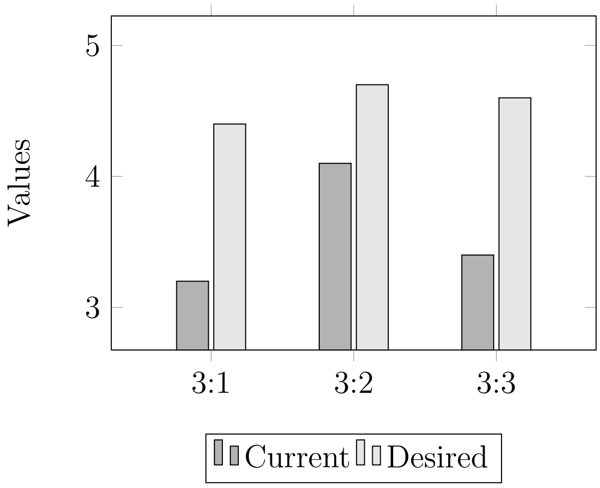


Figure 3.3: Average Current and Desired values

Continuous improvements The team does not feel they have a focus on continuously improving the product development process, recognizing the phrase “[Product development process is sporadically improved], as isolated ’improvement packages’ ”, nor does the team have a person that is responsible for improving processes or making sure agreed-upon practices are followed. The team assess themselves at a level two, desiring to reach level four. The team wishes to have a stronger focus on product development methodology and to try out industry practices as a part of the learning process of participating in the Shell Eco-marathon.

When assessing how the team is using metrics and productivity measures in the development, the team assess themselves a level three, desiring level four. The team keeps track of progress using percentages, which are displayed on the windows of the office.

Figure 3.4 show the results of this part of the assessment.

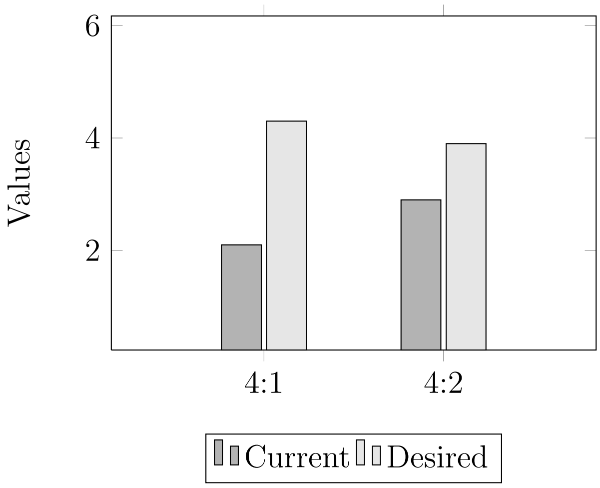


Figure 3.4: Average Current and Desired values

Culture The team feels that Trust, Respect and Responsibility are core values in the group. Given the flat structure and the joint commitment, the team assess themselves at level five on this point.

When asked whether project decisions are based on fact, the team assess themselves at a level three, desiring to be at level five. However, the team also mentions that being students participating in a project where they may make the big decisions themselves without external pressure from customers, making decisions based on what the team *wants* to do compared to what the team *should* do is one of the major perks of the project.

When it comes to using simple and visual communication, the team assesses themselves at level four, desiring to be at level five. The team wants to see more use of A3-reports and the SE Wall.

Figure 3.5 show the results of this part of the assessment.

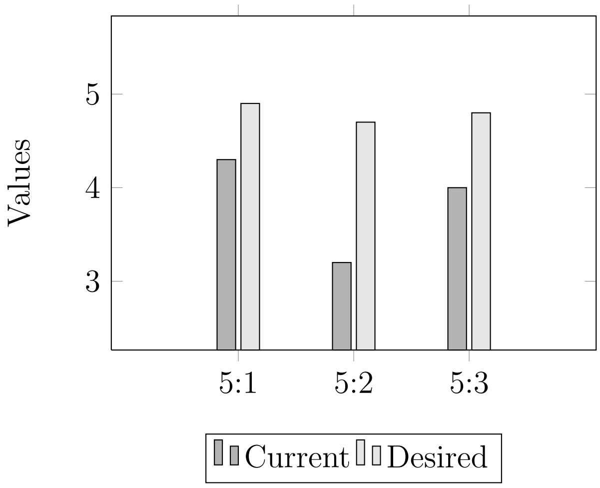


Figure 3.5: Average Current and Desired values

3.1.3 Gap between current and desired state

The bar plots in Figure 3.6 show the gaps between the current and desired state of the project. It clearly states which areas the team desires improvement. Every question in the Knowledge category and one question from the Continuous Improvements category have a gap between the current and desired state that exceeds the limit of 1.5.

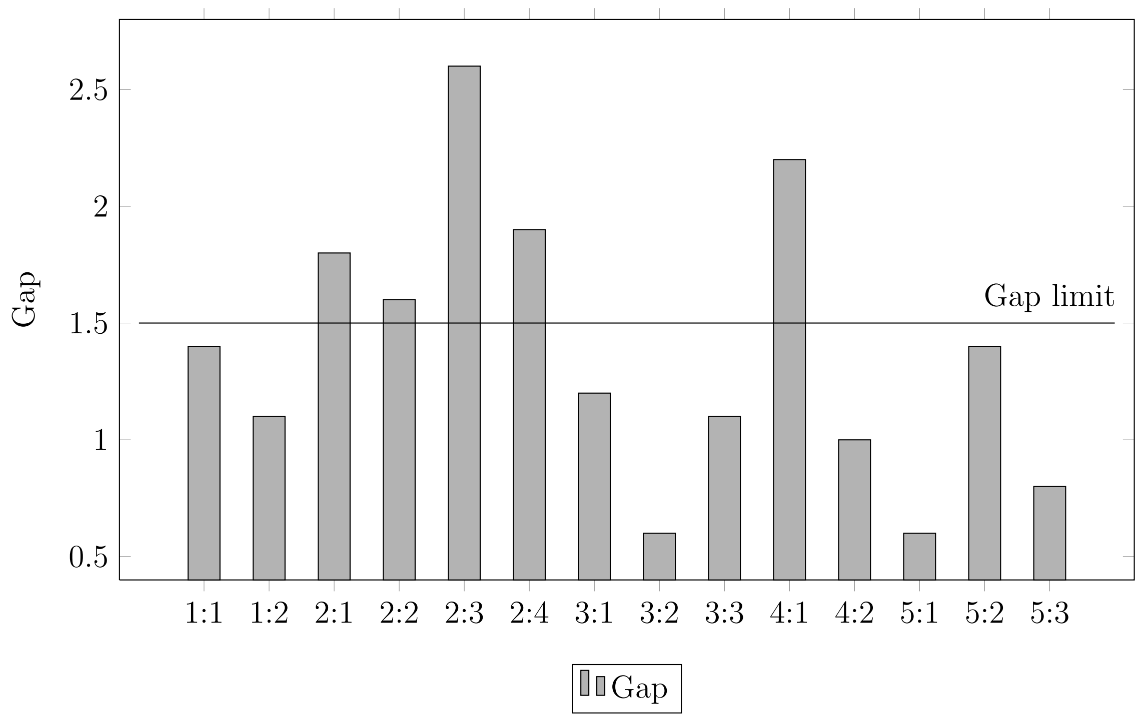


Figure 3.6: Gap between Current and Desired state

3.1.4 Discussion of assessment results

The discussion around value-added activities in DNVFF helps define what really are value-added activities in this environment. Given the lack of an end-user and customer, the sponsors are the only ones laying money on the table. The definition of value-added activities state that all activities that the customer is willing to pay for, is value-added. The sponsors take over that role. The sponsors are paying the project. In return, they expect representation. Attending public events, is thus value-added to the project. Also, the potential goodwill gained from making the effort of attending these events may benefit the next team of students.

Customer focus The team in general felt it is not relevant for this project to extend the customer focus. Given that the developers are both the design engineers and the customers, the customer focus is already given. As long as the communication flows freely allowing the desires and needs of each team member to become known to the group as a whole, then the project has the appropriate customer focus.

Although feeling that customer focus is at an adequate level, the team did benefit from discussing who the customer actually is. This was not something the team had addressed earlier, yet it is - from both a Lean and a SE point of view - an essential exercise in any project. Determining who the customer is, helps clarify requirements, gain insight in the problem at hand, and paths the way onwards. Also, emphasizing the fact that every team member is equally important as a customer shows the group that they are all decision-makers and have the right to voice their opinions. As a team exercise, this discussion did good for the team.

None of the questions in this section exceeded the gap limit of 1.5.

Knowledge Knowledge is the section where problems were really unearthed. Every assessment score in this section exceeded the gap limit of 1.5, meaning knowledge is an area that will need special attention and effort.

The team states that they know too little about the earlier project iterations, having to learn the system themselves and to "learn by doing" through the year. One team member expressed to the author, *"upon coming here the first day, we were shown the vehicle and the workshop, and basically told 'to figure things out ourselves'."* Having to obtain fundamental knowledge without guidance may quickly constitute a lot of wasted time, pushing back the formal project start by several days, even weeks.

Generally, a lack of knowledge of the system, the product development process and important decisions made by previous teams is a source of risk; a risk of making wrong decisions, a risk of wasting time on copying work, of performing the same errors made by previous teams and of failing to improve the system, even making it reduce its performance. Knowledge transfer is an issue in need of special attention.

The team requests a knowledge manager that collects knowledge, who designs standards for knowledge capture and is responsible for storing the knowledge in an easily accessible format.

Also, employing more front-loading is deemed important for future projects. Especially spending more time on defining requirements and studying multiple solutions.

The LEfSE Tool suggests the following enablers:

- Create mechanisms to capture, communicate, and apply experience-generated learning and checklists
- Maintain team continuity between phases to maximize experiential learning. Capture and absorb lessons learned from almost all programs: "never enough coordination and communication". Synchronize work flow activities using scheduling across functions, and even more detailed scheduling within functions.

- Adopt and promote a culture of stopping and permanently fixing a problem as soon as it becomes apparent.
- Plan to utilize cross-functional teams made up of the most experienced and compatible people at the start of the project to look at a broad range of solution sets. Explore trade space and margins fully before focusing on a point design and too small margins. Anticipate and plan to resolve as many downstream issues and risks as early as possible to prevent downstream problems. Plan early for consistent robustness and ”first time right” under “normal” circumstances instead of hero behaviour in later “crisis” situations.

Based on the enablers and the team’s wishes, the following measures are proposed:

1. Design standard A3 K-briefs for capturing knowledge during the development process
2. Design standard reports for documenting interfaces
3. Capture lessons learnt from trade-off analyzes.
4. Assign a knowledge manager in the team
5. Decide on a common practice for writing and storing K-briefs
6. Build a culture of solving problems as they become apparent

Stabilize The team had a heated discussion around the value of having a more detailed resource plan. In the end, the majority of the team voted they were satisfied with the existing plan, also feeling they get the resources needed when needed.

The division in the team indicates that there may be need for some action concerning a more detailed resource plan. However, due to the fact that most of the team members voted for making only slight improvements, resulting in a gap of less than 1.5, this has only been communicated to the project manager and has not been prioritized further in the report.

Communication internally between team members and externally towards sponsors or key suppliers is very good. Being co-located maintains the team relationship, allowing everyone to get acquainted with each other and thus “soften up”, lowering the threshold for making contact and asking question or voicing opinions in discussions.

Members of the team have visited sponsors, exchanged knowledge, and even worked with them in their workshop. This helps building strong relationships between the team and the external resources. Main focus for the team are the sponsors, and these are continually updated on the progress, and also queried when the team needs consultation. The main sponsor also provides the team with reports about their standard working procedures for the systems engineers to read and employ as required. This is a good example of knowledge sharing between sponsors and the team.

No changes suggested in this category.

Continuous improvements The team is satisfied with the use of metrics showing project progress. The author is of the impression that the team assess themselves slightly higher than what is actually the case, yet the discussion did show that the team is satisfied with the current system, regardless of the actual score. However, the team has no way of showing project progress on a sub-system level. Especially when taking the tight time constraint in mind, the author feels having total control of project progress may aid the team in identifying sub-systems that are lagging behind before getting critical. Project manager may thus allocate extra resources to these “stragglers”, and prevent last-minute rescue missions.

The team realizes their product development strategy is not anchored in any specific methodology. The group wishes to improve their methods, and try out theories from LPD. Also, the team fears its methods have stagnated since the project started, leading to a lack of focus. By continuously focusing on following the right procedures, the team hopes to reduce waste and increase its efficiency.

Continuously improving the product development methodology of the team is an area the team wants help with and is willing to focus on. The LEfSE tool recommends the following enabler:

- Pursue Continuous Improvement according to the INCOSE Handbook Process. In addition: Promote the idea that the system should incorporate continuous improvement in the organizational culture, but also.... ...balance the need for excellence with avoidance of overproduction waste (pursue refinement to the point of assuring Value and ”first time right”, and prevent over-processing waste). Treat any imperfection as opportunity for immediate improvement and lesson to be learned, and practice frequent reviews of lessons learned. Use the formal large Six Sigma teams for the problems which cannot be addressed by the bottom-up and Kaizen improvement systems, and do not let the Six Sigma program destroy those systems. Use formal value stream mapping methods to identify and eliminate SE and PD waste, and to tailor and scale tasks.

However, the author feels this enabler is aimed at bigger companies and projects of a higher longevity. Also, the project has come too far in the school year to employ Kaizen events to any great effect. The author has spent some time looking at other enablers and techniques from LPD, and proposes the team starts having Stand-up meetings using a VPB designed especially for the purpose of DNVFF, in addition to spending more time identifying potential risks and early mitigation of these risks. The team should be introduced to the concept of *critical path*. By combining VPB and an continually updated risk analysis, the engineers may identify the most crucial tasks at hand and prioritize them.

The proposed action is as follows:

1. Arrange Stand-up meetings
2. Design VPB
3. Perform risk analysis and keep it updated.
4. Visualize project progress on a sub-system level, possibly also component-level

Culture The team has a strong culture, at least when it comes to communication and democratic decision making. Also, being a young team of students, the team is willing to learn new methodologies and thinking "outside the box". However, whether the team really has the right culture for capturing and sharing knowledge is unknown. The author needs to pay close attention to this aspect, and work continuously on writing K-briefs and updating risk analyzes, mitigation plans and similar.

Model-based Systems Engineering and the DNV Fuel Fighter project

Comparing the list of perceived benefits from MBSE in Section 2.1.4 with the results from the assessment, shows that MBSE touches on aspects of which the DNVFF is struggling, namely; enhancing knowledge capture and transfer, to simplify the training and integration of new team members and to reduce cycle time.

The output from the MBSE effort may be combined with the enablers for knowledge capture and risk analysis. One way of doing this, is to link knowledge to corresponding model elements to help structure the information. Also, software types that are based on SysML or similar normally support risk management. In this manner, the process of applying Lean principles to the project may be tightly integrated with the MBSE effort.

3.1.5 Conclusion of assessment

From this assessment, knowledge transfer appears to be the the main problem for this project. Also, the team needs to focus more on improving the product development process, to ensure progress efficiency.

By listening to the team's wishes and consulting the lean enablers in view of the survey scores, the final assessment is that the team needs to assign a knowledge manager whose responsibilities are to design knowledge briefs and make sure these are easily accessible and understandable. Also, the team needs to have frequent, short meetings where the team members inform the rest of the team what they are working on, what they need help with and what might be hindering them in making progress. For these meetings the team will need a VPB tailored for the project. The team members need familiarization with queueing theory and the term "critical path". By performing risk analyzes at regular intervals, the team members may find help in identifying prioritized tasks - as well as gaining a better understanding of the system.

MBSE aims to solve many of the problems uncovered by this assessment. The MBSE effort may thus be tightly integrated with the effort of introducing Lean principles to the project, through linking model and information, and using MBSE software tools to manage risk.

3.2 Knowledge management in DNV Fuel Fighter 2

The results of the assessment points to an existing need for an increased focus on knowledge sharing within the organization. A team works on the project for two semesters, then graduates and leaves behind a modified vehicle and a master's thesis. The thesis is written to please the censors, not to share knowledge with the team's successors. The knowledge one gains during such a project, needs to be transferred to the next team; the knowledge is the team's legacy. The group desires a method for capturing and sharing knowledge.

In order to share knowledge, one needs a way of structuring the knowledge captured; one needs to know where to actually put the information, or - later - where to find it. This thesis employs MBD - described in Section 2.1.4 - to structure information, where documents are linked directly to the element within the system model of which they describe.

Some of the perceived advantages of MBSE is indeed increased learning, knowledge transfer and easier training of new team members or users, as stated in Section 2.1.4. This is the reason for the author's decision to use MBSE to structure knowledge.

The system model is meant to act as a portal to the knowledge database, as well as act as a tool for describing the system to the team members. The model contains a hierarchy of requirements, a functional description and a system architecture which also includes all interfaces between components. Upon reading the model itself, the reader should be able to understand why the system acts as it does, what it is capable of doing and how it achieves this. Also, the functional description of DNVFF2 describes the environment in which the vehicle operates.

First, this thesis will present the K-brief templates designed specifically for this project.

3.2.1 A knowledge brief template for technical data

Knowledge should be captured as it is created, while it is still fresh. Mascitelli recommends the creation of K-briefs at every decision gate, which captures the decisions made and lessons learnt over the period. To ensure a standardization of knowledge capture, a K-brief template was designed especially for the DNVFF2 project.

The author interviewed members of the project and showed them Mascitelli's list of technical data (see Section 2.5.3). The members expressed they would have appreciated knowing the following pieces of technical data from previous years:

- Important design trade-offs and decisions
- Reusable design elements
- Raw material/component data
- Test results for common design elements
- Reliability data
- Supplier design rules/capability data

In addition, they would like to have an easy overview of interfaces and potential risks.

From these wishes the author designed a K-brief meant to describe a sub-system or major component of the DNVFF2. The K-brief extends across three A3 pages. The front page is dedicated to giving an overview. The second page concentrates on material data, important requirements and structural analyzes performed by software. The final page focuses on risks and testing. The K-brief is intended to be easy to fill out and easy to read.

The front page

The front page is shown in Figure 3.7. This page gives information on components, interfaces, important design trade-offs and decisions, reusable design elements, reliability data and suppliers.

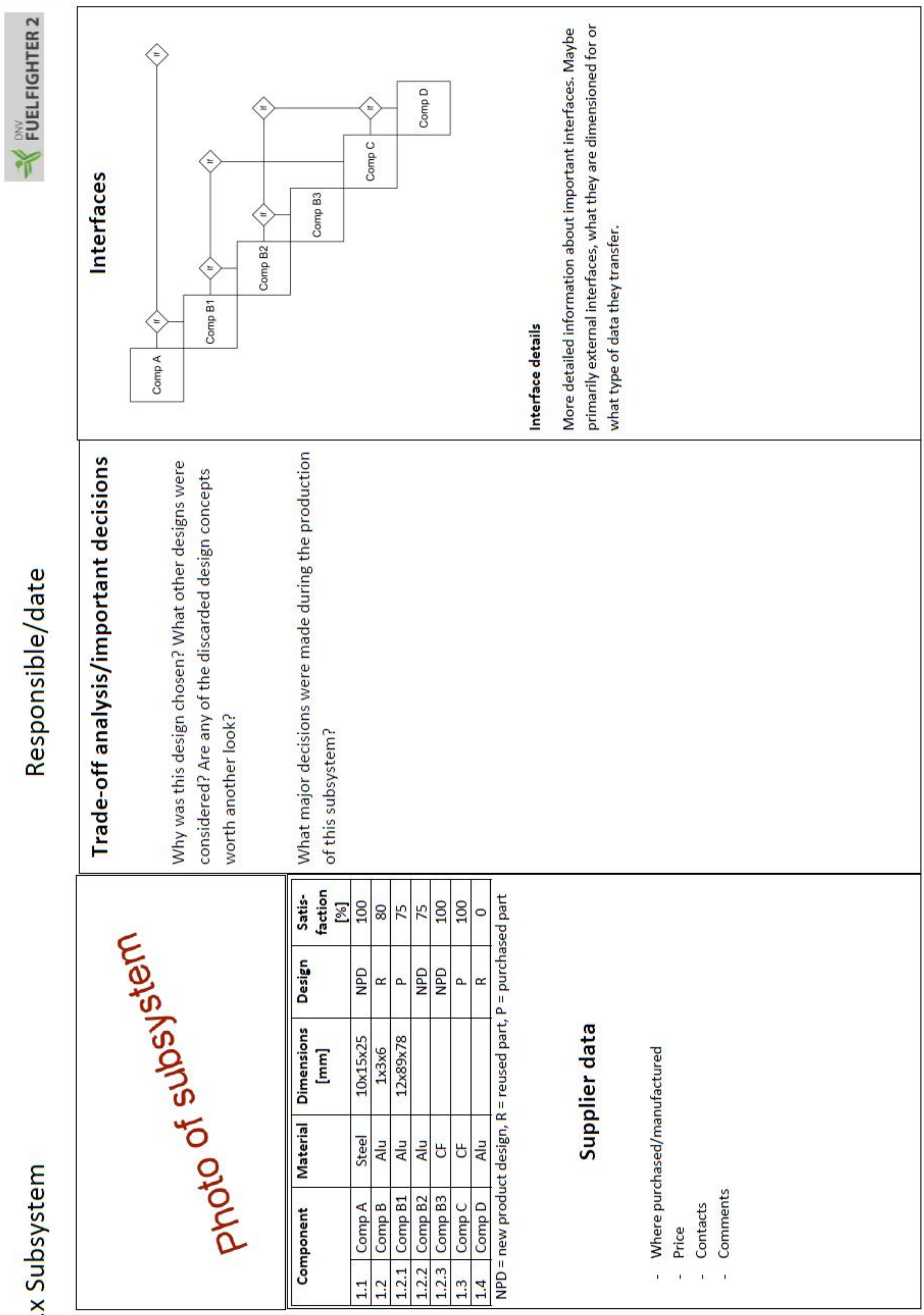
The page is divided into three vertical sections. The left-hand section contains at the top a picture of the sub-system. Below is a table listing the components that make up the sub-system (in the left-hand-most column). The second column contains material data. The user should enter as specific data as possible, for instance 'Alu6060'. This provides the reader with instant knowledge on where to search for more data on the material used. Since weight is such a crucial factor in the Shell Eco-marathon, the third column contains this information.

The fourth column contains one of three acronyms; *NPD*, *R* or *P*. NPD means 'New Product Development', and tells the reader that the component is designed during the last iteration. R means 'Reused' and says the component is unchanged during the last iteration. P means 'Purchased' meaning the component is an *off-the-shelf-item*. The last columns gives a ratio of satisfaction for every component, as a sort of reliability data. If a component fails, or has weaknesses due to imprecise manufacturing this should be reflected in the rating depending on the severity of the faults. This column guides the new engineer in where to concentrate his efforts for improving the design.

Below the table the writer should provide information on the suppliers of materials, off-the-shelf-items, and if the components are manufactured in-house it should also contain information on the manufacturing methods used.

The mid-section is dedicated to important design trade-offs and decisions. The writer is free to present this information as he or she wants, yet is encouraged to keep it visual and simple. Where applicable the writer should use figures, curves or lists for explaining.

The right-hand section is devoted to interfaces. At the top of the column, is an interface n2 diagram. Information on how to read such diagrams is provided in Section 2.2.2. From this diagram the reader should easily gain insight on how the sub-system is put together, and to which other sub-systems it connects. Below this figure the writer should give insight into the most important interfaces, providing data on e.g. what an interface is dimensioned to handle, what type of data is exchanged or what tolerances it employs.



S.x Subsystem

Responsible/date

DNV FUELFIGHTER 2

Photo of subsystem

Component		Material	Dimensions [mm]	Design	Satis-faction [%]
1.1	Comp A	Steel	10x15x25	NPD	100
1.2	Comp B	Alu	1x3x6	R	80
1.2.1	Comp B1	Alu	12x89x78	P	75
1.2.2	Comp B2	Alu		NPD	75
1.2.3	Comp B3	CF		NPD	100
1.3	Comp C	CF		P	100
1.4	Comp D	Alu		R	0

NPD = new product design, R = reused part, P = purchased part

Supplier data

- Where purchased/manufactured
- Price
- Contacts
- Comments

Trade-off analysis/important decisions

Why was this design chosen? What other designs were considered? Are any of the discarded design concepts worth another look?

What major decisions were made during the production of this subsystem?

Interfaces

Interface details

More detailed information about important interfaces. Maybe primarily external interfaces, what they are dimensioned for or what type of data they transfer.

Figure 3.7: The front page is dedicated to listing components, interface diagrams, describing major decisions and the manufacturing process.

The second page

The second page is shown in Figure 3.8.

This page is dedicated to the design process, with focus on 3D modeling and analyzes performed with software. The page is divided into two sections. The left-hand section is much smaller than the right-hand one, and should contain textual information on important requirements, assumptions made, materials and software used. The right-hand side is dedicated to figures. It is strongly encouraged to present the design/analysis process as a story, using arrows or boxes like in a comic strip. If the component is made from a composite laminate, it would be wise to include an illustration of the lay-up here.

The third page

The third page is shown in Figure 3.9.

The page contains two sections. The left-hand section supplies a table listing potential risks to the sub-system, with values telling the reader the likelihood of the risk occurring, and the level of consequence if it does.

The right-hand section is dedicated to the performance of the sub-system, and the proposed future work for new engineers. The performance should contain information on testing and race. The testing history is important. The new engineer may identify weaknesses in the testing procedures, and design better procedures. Knowing how the sub-system performs is imperative for new design iterations. The future work section may give the new engineers a head start on their development process.

Together, these three pages contain all of the desired pieces of information listed earlier in this section.

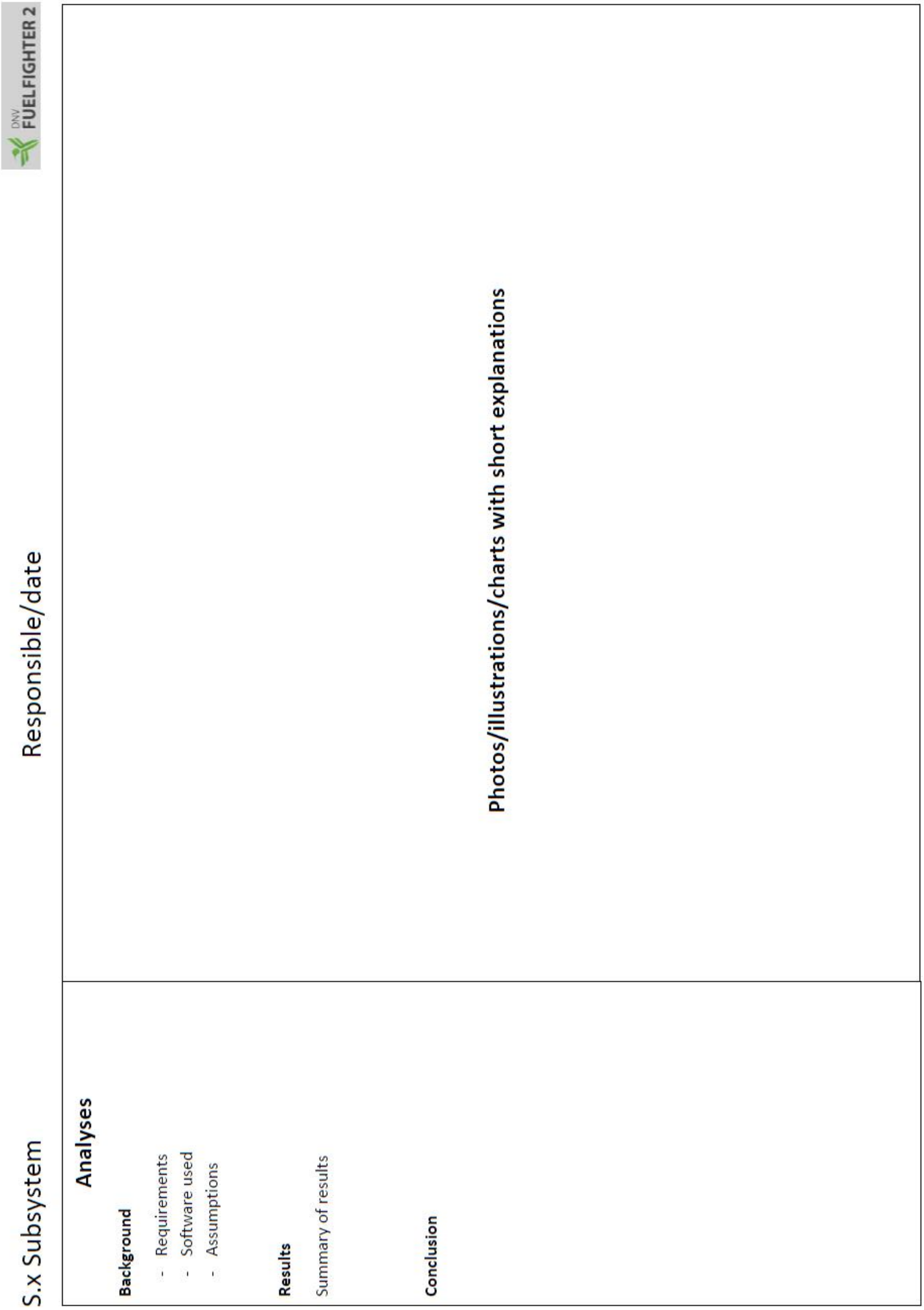
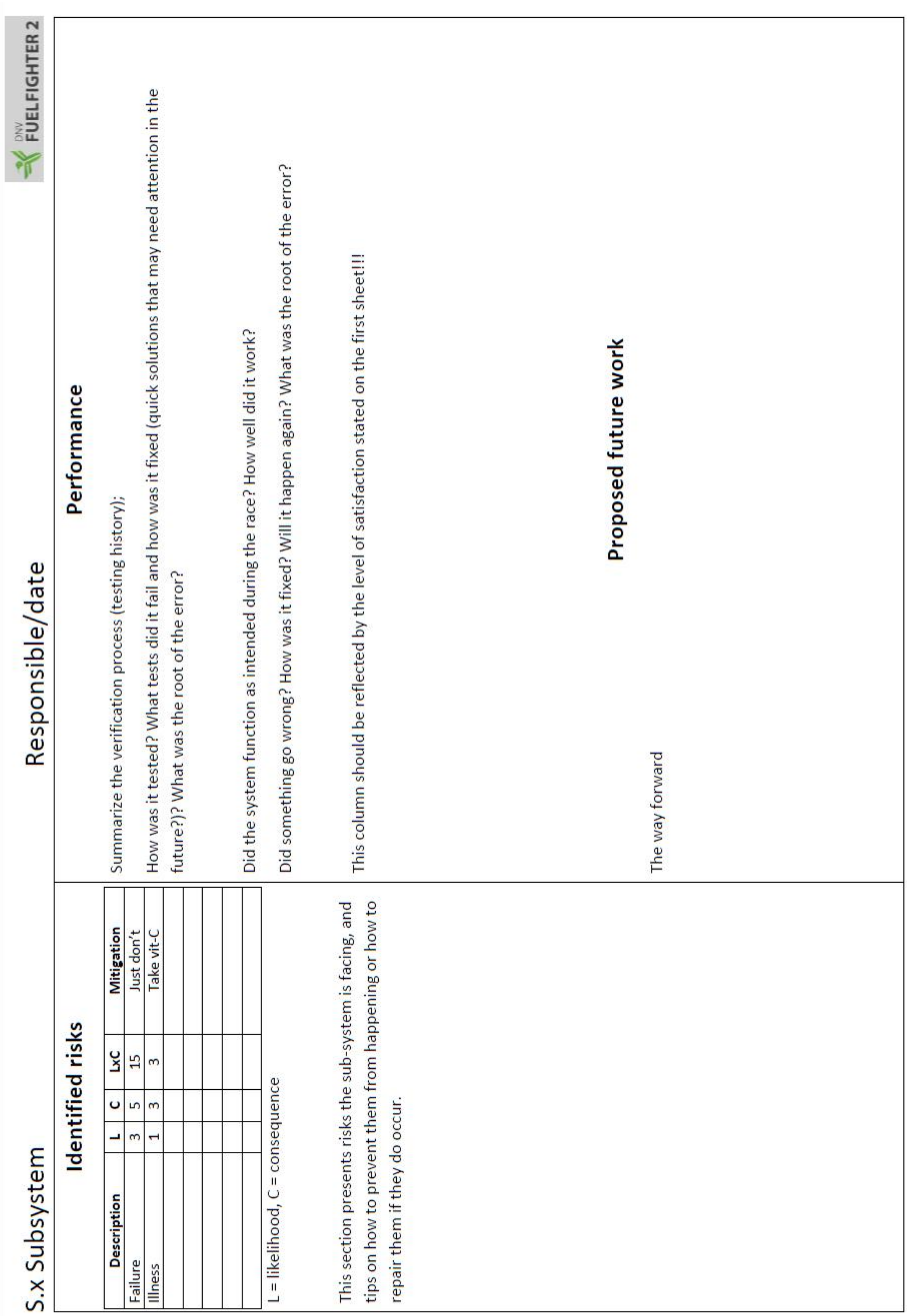


Figure 3.8: Page 2 is dedicated to the 3D-models, simulations and analyzes.



S.x Subsystem

Responsible/date

DNV FUEL FIGHTER 2

Identified risks

Description	L	C	LxC	Mitigation
Failure	3	5	15	Just don't
Illness	1	3	3	Take vit-C

L = likelihood, C = consequence

This section presents risks the sub-system is facing, and tips on how to prevent them from happening or how to repair them if they do occur.

Performance

Summarize the verification process (testing history);

How was it tested? What tests did it fail and how was it fixed (quick solutions that may need attention in the future?)? What was the root of the error?

Did the system function as intended during the race? How well did it work?

Did something go wrong? How was it fixed? Will it happen again? What was the root of the error?

This column should be reflected by the level of satisfaction stated on the first sheet!!!

Proposed future work

The way forward

Figure 3.9: Page 3 is dedicated to perceived risks, a description of problems and how they were solved, and proposed future work.

3.2.2 Other knowledge briefs

For other K-briefs, e.g. for lessons learnt or as a deliverable the author use either the Plan-Do-Check-Act template in Section 2.5.3 or the simple template that Institute of Engineering Design and Materials (IPM) use for student projects, shown in Figure 3.10.

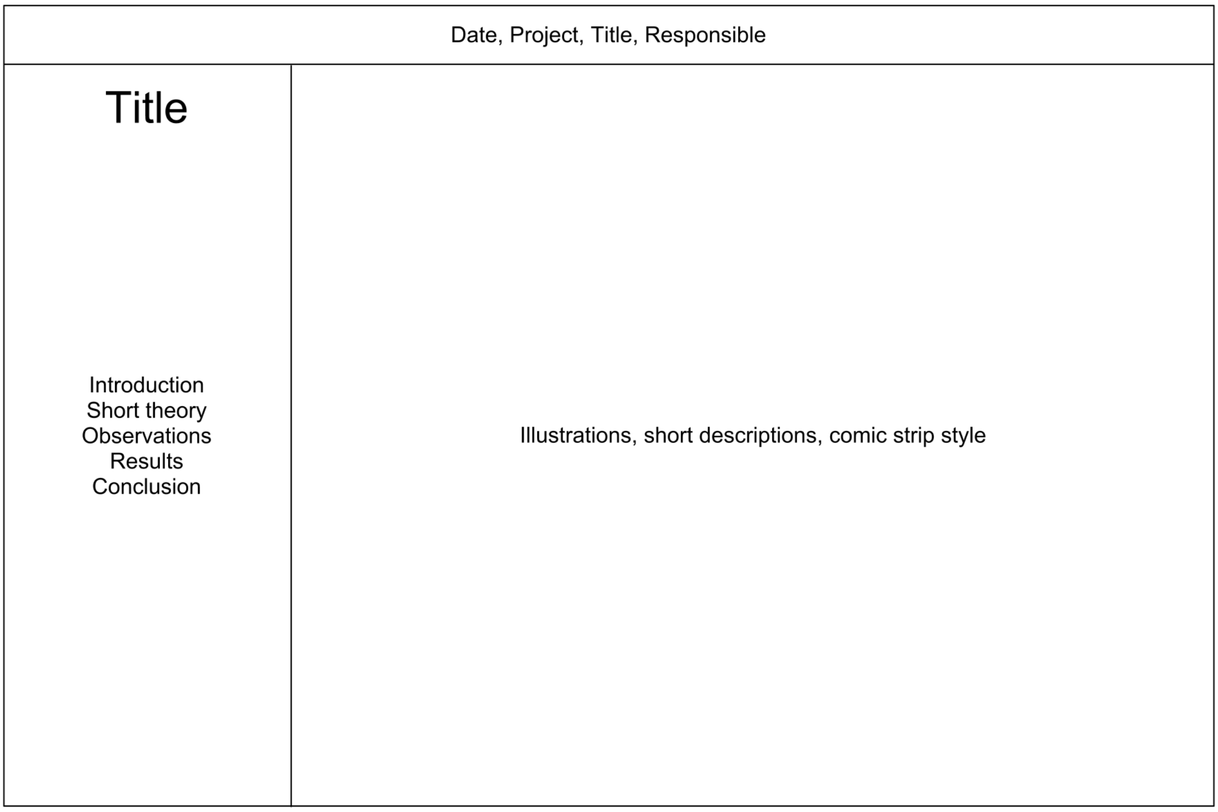
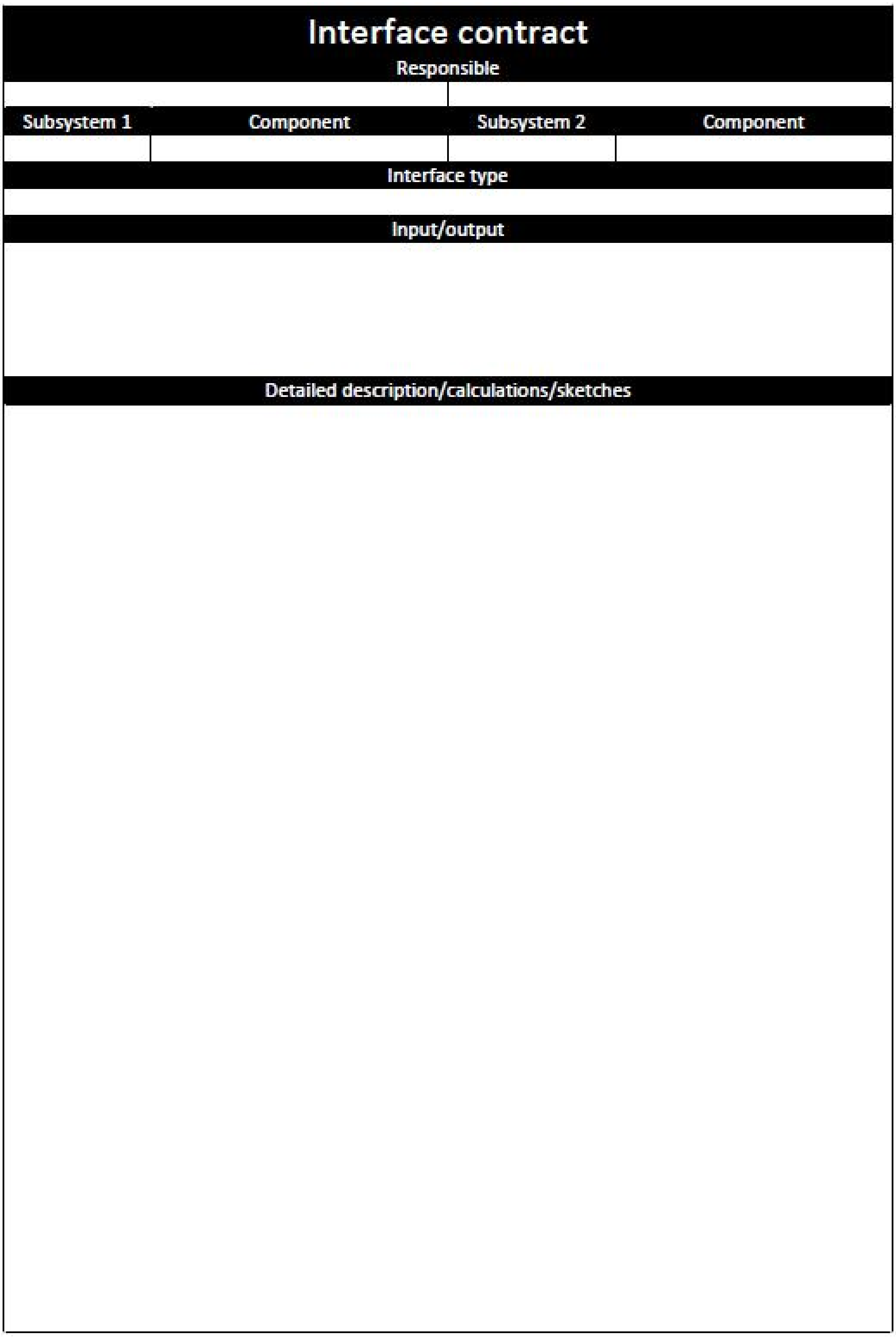


Figure 3.10: The simple template used by IPM.

In order to document interfaces, the author designed a sheet also for this, but in A4 format. This document is shorter, and ready to be filled out with the most relevant information, also utilizing a visual manner of presenting information, as shown in Figure 3.11. The author decided to make this template as he realized no documentation of interfaces existed within the project. The document is intended to be a sort of contract between the people responsible of the two sub-systems interacting. It documents what they have agreed upon, and by actually filling out the contract the engineers are forced to really regard the interface properly. If a decision is made to change any parts of a sub-system, this contract should be updated to match the alteration.


Interface contract

Responsible

Subsystem 1	Component	Subsystem 2	Component

Interface type

Input/output

Detailed description/calculations/sketches

Figure 3.11: A template for an interface contract that establishes the responsibilities, interface type and input/output of the interface in question.

3.2.3 The system model

This section will describe the purpose of the system model, its structure and important decisions made by the author leading up to the resulting model.

Requirements

The requirements were gathered from the Shell Rule book. Only the rules which applies to the DNVFF2 were gathered. The requirements were often stated in long sentences, which some times contained more than one requirement. Hence, these formulations had to be decomposed into single requirements of one short sentence, and represented with parent-child relationships in a hierarchy. The team also decided on additional requirements themselves, such as weight limits for every sub-system, strength and design decisions. These requirements were not easily accessible for anyone but the one who had written them, as they were kept in a number of spreadsheets and not structured in any standardized layout.

The leaf nodes of this resulting hierarchy each represent a single requirement, and fulfilling the requirements in each leaf node means also fulfilling the above node(s). The requirements diagrams were also utilized by the rest of the SE team during testing and verification. Testing procedures were planned according to the requirements, and the diagrams acted as checklists during testing.

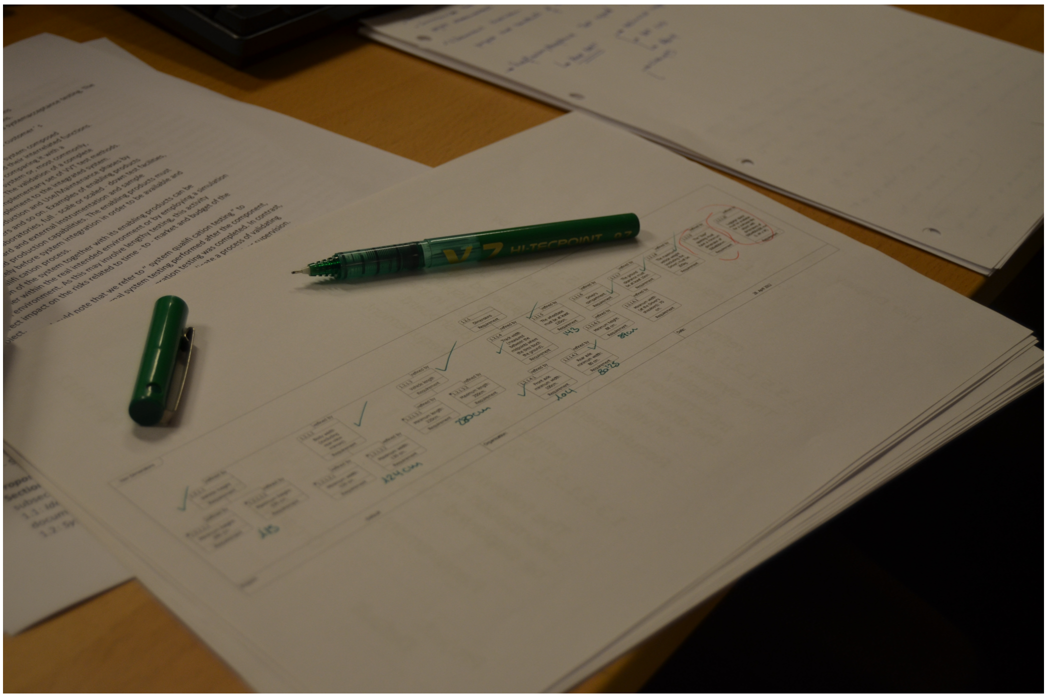


Figure 3.12: Crossing off requirements as they are verified.

Functional description

Functional descriptions may describe a number of things depending on what part of the system they focus on. They may describe the steps of the manufacturing process, or focus on the product itself and what functions components and software must perform.

Originally, the focus for the functional description for this project was to describe how the car acts mechanically and electrically. However, after working with the project for a while, it became clear that the way the car works is not complex enough to require a functional description in order to understand it quickly. What is not clear for the team though, is the competition. So, the resulting functional description will focus on describing the system surrounding the car; transporting the vehicle, preparing the vehicle for competition, and the actual race. Of special interest to the team is the process of qualifying the vehicle through technical inspections at the race site. The Shell Rule book is rather vague on how this is done, what types of tests are performed and how strict the inspectors really are. The functional description will attempt to capture and describe this process.

During the planning of testing and verification, the level of doubt increased within the SE team on whether the team would be able to test the vehicle the same way that the Shell inspectors would. In particular, the team knew nothing about how the rear view mirrors, the windshield wiper, seat belt nor the electrical system would be inspected. Upon reaching the event in May and speaking with other teams, the team realized the rear view mirrors would not pass the inspection. Hence, the team had to manufacture new and bigger mirrors the day before the inspection. Failing one of the tests means having to go through the entire inspection again. The team passed the inspection on the first attempt - much due to the extensive testing performed in Trondheim - but it took four hours even though the team was one of the first in the queue. The reader will find more about the VV&T activities of this project in Itsaso Yuguero Garmendia's master's thesis.

The purpose of the functional description of DNVFF2 is to remove all traces of doubt within the team. The legacy teams will now know how the process of technical inspections work, what they have to test and how to test it. The stress of not knowing what is coming has a negative impact on team morale, which is what this model will try to repair.

Also, the model describes the process of packing up and transporting the vehicle. This is not as straight-forward as it may sound, as the rig on which the vehicle rests is built especially for this project and the vehicle itself needs special attention when it gets strapped onto the rig.

System architecture

The system architecture is of use to the team in many ways. It divides the project into parallel work efforts, it may be used to visualize project progress - as is shown in Section 3.3 - and it adds insight into the physical aspect of the system; what components are used and how they interface with each other. The model also includes discarded solutions, e.g. the hydrogen fuel cell that was replaced by a battery.

The process of constructing the system architecture involved interviewing team members, studying 3D-models and observing the building process. The result is an architecture that divides the vehicle into its main sub-systems, and which uses interface diagrams to illustrate how the components are put together.

The interface diagrams are N2-diagrams, and give an easy overview of the internal and external interfaces of a sub-system. By reading these diagrams in addition to the hierarchy of the system architecture, the reader gets full insight into how the system is assembled, preparing the reader for working with the actual system or its Computer-aided Design (CAD) models.

Building the system architecture was performed by interviewing team members, by studying drawings and models, and by being present in the workshop constantly taking notes.

3.2.4 The software

To reach the goals of making a system model that supports MBD, the team needs a software that can perform the following tasks:

1. Create and display hierarchies and FFBDs
2. Support relationships between requirements and functions or physical elements.
3. Specify interfaces
4. Link model elements to information contained in other file formats, such as PDF or CAD files.
5. Publication of system model in an open format for sharing.
6. Should support a collaboration environment
7. Should support VV&T

The systems engineer of 2011 delivered a simple system architecture with a depth of three levels, created in Microsoft Visio. Visio is a very good tool for drawing flow charts and hierarchies, yet does not support collaboration, relationships between requirements and functions or physical elements, or any form of publishing the model for sharing.

The author was familiar with two software products that fulfilled the requirements stated earlier in this section; TopCased by the TopCased Organization, and Core by Vitech.

TopCased is a freeware by a development forge by programmers from across the world. It is a SysML and UML editor. It is a powerful tool which also support executable models. [1] The author have extensive experience with this software from his summer internship at a major Norwegian defence contractor and from using the tool in his project work in the fall of 2011. The software demands quite a lot of training to get used to, and may appear untidy and a bit frightening even after some use. Also, the fact that it draws the diagrams with SysML makes the diagrams a bit uninviting to engineers with no software development background. Figure 3.13 shows an FFBD created with TopCased.

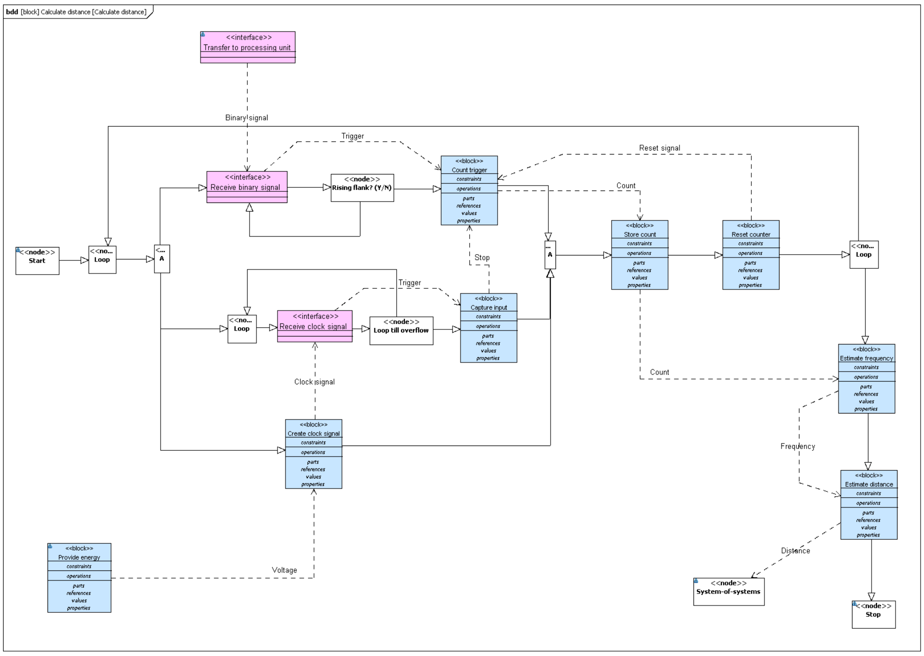


Figure 3.13: An example of an FFBD from TopCased.

Vitech's Core is a tool developed especially for MBSE. It is not based on SysML, but on a more metabased form of drawing diagrams, making them more accessible to humans. Basically, the diagrams consist of boxes and arrows without any complex indicators. The program has standardized the diagrams, and the user has little influence on the visual design of FFBDs or hierarchies. Vitech's MBSE methodology is built on the same three areas of which this thesis is focusing, plus VV&T activities which are important for the DNVFF2's other systems engineer. [3, pp. 30-35] The author had a comfortable amount of experience with the software from attending Haskins' SE course at NTNU. Haskins has long experience with Core, and could provide consulting if need be. Figure 3.14 shows an FFBD created with Core.

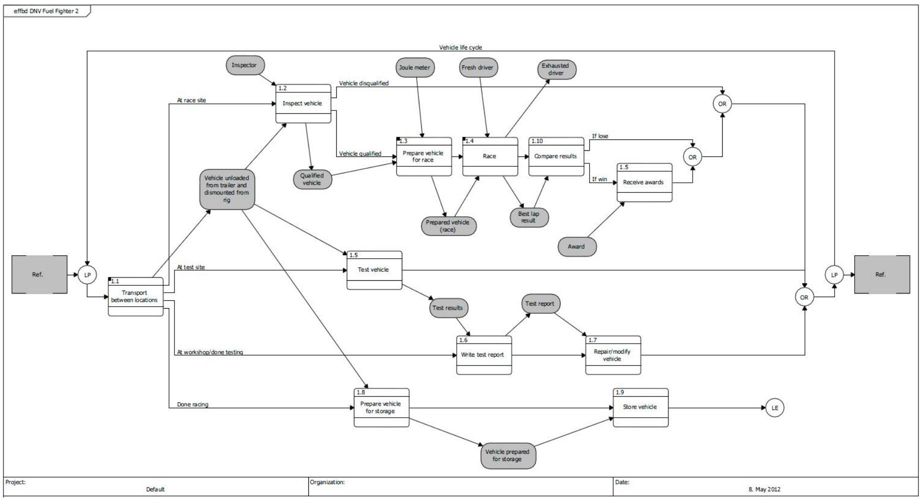


Figure 3.14: An example of an FFBD from Core.

Both programs fulfill the requirements stated above. Both of them solves requirement 5 by exporting an HTML webpage which can be viewed in any web browser. The webpage displays all the diagrams, and going from one diagram to the next is achieved by clicking on a node. Documentation related to a system model element is listed in an element's property screen, and are accessible as a hyperlink which opens the file in its required software upon clicking, e.g. a PDF reader or a CAD program.

However, Core stands out with its simplicity and the way it displays interfaces and relationships. The standardized format helps with simple training of new users and readers of the model, which is a great advantage when training a new team. Haskins' extensive experience with the program adds a "safety net" for the author which would not exist for TopCased.

The final decision thus fell on Core.

Cecilia Haskins says *"a fool with a tool is still a fool"*. Even though the author had a brilliant piece of software available, the good old fashioned practice of pen, paper and sticky notes was the favoured way of working. Building a model is not something that is done on a whim. It takes a lot of planning, and one needs to know where to start and where to go. Big sheets of paper with sticky notes on that can be moved about, or a big glass window and a marker are creative ways of working in the early phases of building a model.

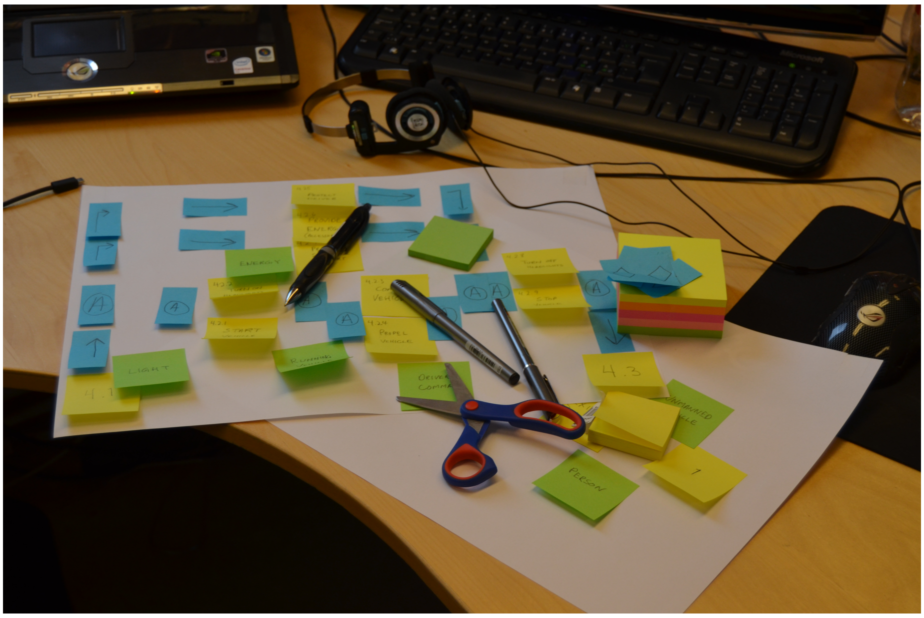


Figure 3.15: Paper, scissor and pen is necessary before proceeding with complex software tools.

3.3 Visual Workflow Management in DNV Fuel Fighter 2

This section will describe how VWM was implemented in the DNVFF2 project, how the methods evolved over the course of the project and how it affected the team and project.

3.3.1 Methods of implementing Stand-up meetings

Stand-up meetings were commenced at the earliest convenience after the Lean assessment. They were held three times a week, on Tuesdays, Wednesdays and Fridays at 11:45, to accommodate the slightly different time schedules of the team members. The design student were not located with the rest of the team, yet attended the stand-up meetings when they were present at the workshop. Typically, ten people participated in the meetings.

On Mondays the team held a weekly meeting, led by the project manager. On this weekly meeting the current status of the system was reviewed and updated goals were set for the week. Important information was also broadcasted. The Stand-up meetings were thus an extension of these meetings, keeping the team up to date throughout the week.

The author tailored a VPB for the DNVFF2 based on examples from Mascitelli and personal experience with VPBs from different internships in Norwegian industry. The result can be seen in Figure 3.16.

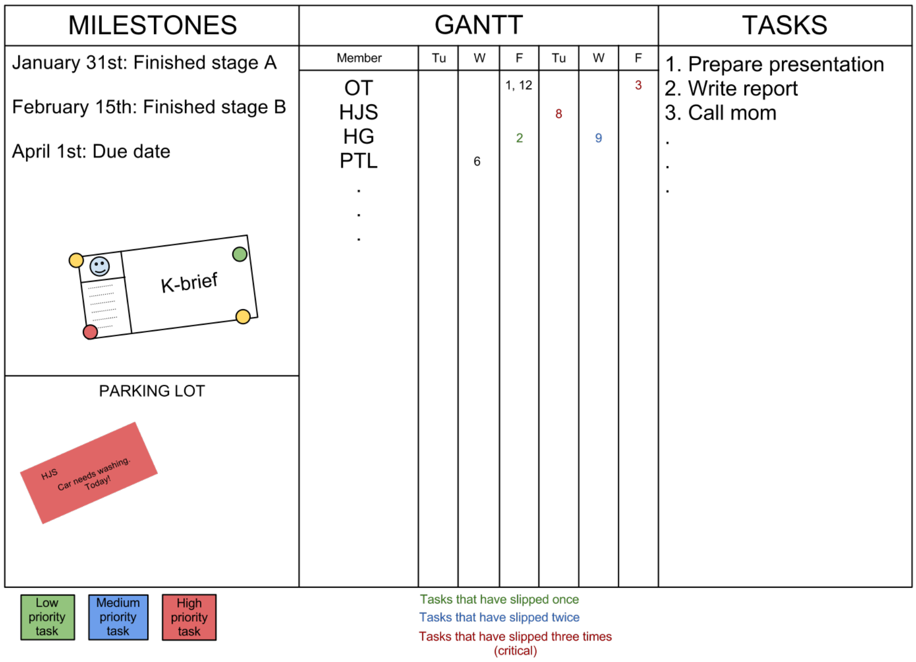


Figure 3.16: The Visual Project Board tailored for DNVFF2. Tasks are listed on the right-hand side, and the corresponding number is added on the Wall-Gantt to indicate due date. The color coding indicates task status and exceptions. The left-hand side contains project milestones, and a parking lot where the users may add tasks that need to be done.

The VPB utilizes a slightly different Wall-Gantt then the one suggested by Mascitelli which employs sticky notes. Instead, this one has a task list on the right-hand side, with corresponding numbers to each task. The number is added to the Wall-Gantt on the date it will be finished by, and on the row indicating whose responsibility it is. The left-hand side contains the major project milestones, and a parking lot where users add tasks in need of doing.

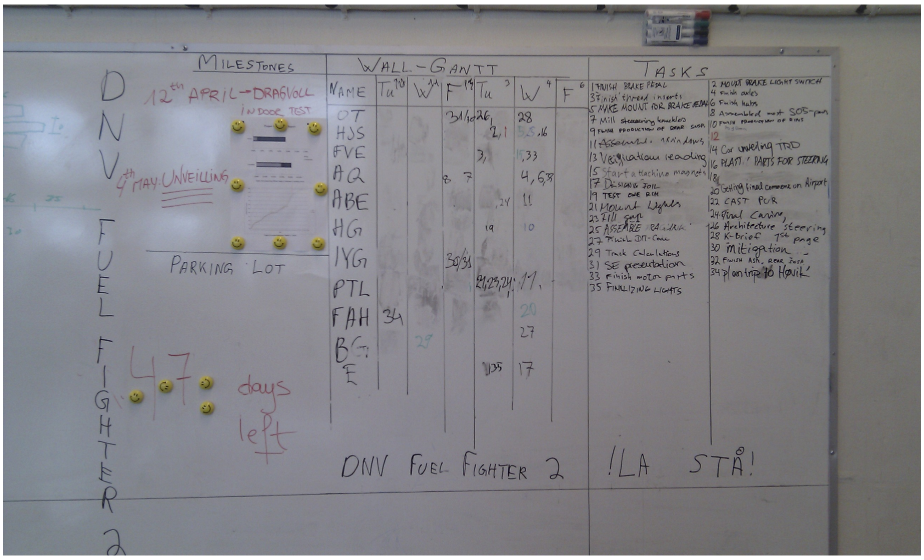


Figure 3.17: The actual VPB at one point during the project.

The parking lot was utilized to a low extent, and transitioned into being a place for hanging up all kinds of information, mainly from the project manager about project status, spendings and so on.

Originally, the design also included a column for tasks more than two weeks into the future. This column was removed early, after feedback from the users who feared tasks in this column would be pushed back endlessly.

Stand-up meetings at the university

The location for the Stand-up meetings were first at the team office on the 2^{nd} floor. This proved inconvenient, as the team members spent their days in the workshop on the ground floor. Having to run up four flights of stairs to attend a quick meeting was not popular and added undesired stress to the meetings, which resulted in an unwillingness to attend. Also, the couches and chairs in the office presented too much of a temptation and the author's requests to have the team members stand up during the meetings went unheeded.

It came to the attention of the author that there was a whiteboard on a wall just outside the team's designated workshop space that was unused. It was hidden from view by factory junk, but a quick tidy-up gave the team access to one end of the whiteboard. The VPB was redrawn here, and the Stand-up meetings moved downstairs. This immediately made things better, and the feedback from the team members improved.

Figure 3.18: The team performing a Stand-up meeting at the workshop.

The author instructed repeatedly that the participants should answer the three questions stated in Section 2.5.1 when they had the “stage”. When the team members started to get the hang of this technique, the meetings improved - both in duration and quality. However, the meetings still lasted close to 15 minutes and feedback suggested this was still too long.

The author decided to redefine the SE team and project manager to mere spectators at these meetings, thus reducing the active participants to seven. This reduced the duration to 8-11 minutes, much to the appreciation of the team. The SE team and project manager only supplied information during the Stand-up meetings whenever something important had to be broadcasted, yet participated actively by asking questions and adding relevant input to the short discussions that sometimes arose during the meetings.

Stand-up meetings in Rotterdam

Before going to Rotterdam the team had not really learnt to appreciate the Stand-up meetings, and some members were still critical of them. The team needed a demonstration of its effectiveness, and in Rotterdam they got one.

The author did not organize Stand-up meetings during the first days of the stay in Rotterdam. Honestly, it did not occur to the author that this would be necessary. However, the competition proved to be an ever-changing environment for the team, where new mirrors had to be made, the motor had to be adjusted and the ever-constant need of knowing what anyone was supposed to do, when anyone could take the day, evening or night off, who is responsible for getting groceries, where and when to have dinner, and so on. Dinner may seem irrelevant, but a team of thirteen people who wants to spend time together -

especially during meals - takes a lot of organizing. Figure C.1 shows a K-brief about the race days in Rotterdam.

After experiencing to fail test-runs on the first test-day and to spend the night working on the car in order to test on the next day, and then to almost fail two attempts on the first race-day only due to poor team organization (the vehicle itself performed fine) the team morale was low. During a hasty team meeting where the failures were discussed, a team member uttered: "We need Stand-up meetings."

For the last two days of the competition, the team held six Stand-ups. Team organization improved dramatically. The Stand-ups allowed the team to respond quickly to the ever-changing environment that a competition is. The team had to stop and discuss the results that ticked in, what vehicle modifications had to be done and also how to make the new time limit when Shell decided to close the track early. After the competition, one team member said "Stand-ups saved us in Rotterdam."

The team kept employing Stand-ups for the remainder of the stay in Rotterdam. The team attended events for their main sponsor in and around Rotterdam, which took a lot of organizing. One to two Stand-ups a day was enough to keep everyone informed and aware of their roles.

3.3.2 Methods of visualizing project progress

The team was already visualizing project progress as the percentage of tasks finished. This number was printed out in large letters and posted on the windows of the office, facing out.

This way of visualizing project progress gives a good understanding of how the project is progressing on a system level. However, it says little or nothing about how the different sub-systems are progressing, and which ones are in danger of slipping behind.

A brain-storming session in the SE team resulted in two additional ways of visualizing project progress that displays how sub-systems are evolving. These two ways of visualizing progress have different fields of focus, and will be referred to as the *Timeline* and the *Wall-Architecture*. The SE team also decided to utilize the Risk management effort as a way of visualizing project progress.

Proper VWM is founded on a strong WBS. A WBS is a detailed overview of what tasks must be done, when they must be done and also - preferably - who performs then and how. A fully detailed WBS also describes the flow of resources between tasks. [2, pp. 662-671]

Risk management and project progress

Risk management was one of the main focuses of the 2012 SE team. The risk analysis and mitigation plans was important for performing proper VV&T activities, and for the author's area it was used in order to visualize project progress and to enable the team to identify problems early and deal with these problems rapidly.

The SE team employed the processes described by Kossiakoff - see Section 2.1.3 - of first identifying risks, then analysing them and rating them, before making a mitigation plan. As the project evolved, the risk analysis was updated with reassessments of the risk values, yielding either an improved or worsened state of the project.

Figure 3.19: Average risk level development from February to April 2012.

Identification of risks was performed in workshops where the entire team participated. Risks were listed, then discussed and rated in order of likelihood and consequence. The risks and their associated values were kept in a Google spreadsheet were everyone could enter details. The spreadsheet consisted of a number of sheets, one for each sub-system including project management. The risk level for each sub-system was measured as the average of the risks within that sheet to gain a better understanding of how this sub-system was evolving. The values were updated on a weekly basis. On the front sheet the SE team put the values into a plot, showing the development of the project week by week, as Kossiakoff shows in Figure 2.6. Figure 3.19 shows how the average risk level of the

DNVFF2 developed over a two month period (February to April 2012). Figure 3.20 shows the spreadsheet for sub-system "Front suspension".

System			Technical				Resources				Comments	Mitigation plan	Status	Responsible
			Risk	L	C	LxC	Risk	L	C	LxC				
S.5 Front suspension														AQ/HS
	S.5.1	**Linkages**	Misalignment	2	4	6					Higher rolling resistance and greater wear on tires and mechanical links	Fine tune alignments		
			Rods may bend	2	3	5						Use spare		
			Rods may break	1	4	5						Use spare		
	S.5.2	**Shock Absorbers**	Ref S.6											
	S.5.3	**Axle**	Axle may bend	2	2	4					Can't drive or brake disc misalignment	Use spare		
			Axle may break	1	2	2					Can't drive	Use spare		
	S.5.4	**Hub**	Lug threads wear out	2	4	8					Can't attach wheel	Tape over 2nd set of lug bolt holes, then after damage: use 2nd set of lug threads on same component		
			Brake disc threads wear out	1	4	4					Can't brake or pass inspection	Use 2nd set of threads on same component		
			Retaining ring groove creeps	1	3	3					Retaining rings can't cage bearings, but there will still be enough friction to contain them			
			Retaining ring fails	1	2	2						Use spare retaining ring		
	S.5.5	**Tie rods**	Tie rod may bend	1	5	5						Spare		
			Tie rod may break	1	5	5						Spare		
			Misaligned	3	3	9						Realign		
	S.5.6	**Front knuckle**	Delamination on upper swivel bolt	1	3									
			Lower swivel bolt pull-out and resulting delamination in knuckle	2	5	10					Should only happen in extreme cases	Glue bolt back in, and try again. Otherwise use spare. Ideally avoid switching to spare, if the knuckle was subjected to adverse loads which cannot be avoided in the future.		
			Toe delamination	1	4	4					Should only happen in extreme cases	Glue toe back together and wrap in carbon fiber or perforated strips, depending on the situation.		
		Acc risks				72				0				
		Total								72				
Average risk		5.14285714285714												

Figure 3.20: Spreadsheet showing risks for sub-system "Front suspension".

The SE discovered that the average risk values were just increasing for this project, and chose to stop tracking progress this way. New risks were constantly being discovered, and some of the risks that were rated as low actually occurred, which of course increased their value (the best examples here is when one of the newly produced rims suddenly cracked after the first test day and when a magnet in the newly produced rotor suddenly jumped loose). The risk management process was never terminated, but the ever-increasing risk level acted as a stress inducer on the team and did not yield the positive effect the SE team was hoping for.

For more information on the risk management procedures of this project - with more focus on mitigation plans - the author recommends the reader to consult Itsaso Yuguero Garmendia's master's thesis for DNVFF2.

The Timeline

The Timeline is an idea that works well with visualizing both the WBS and the project progress, as well as allowing an implementation of Takt Periods. In the DNVFF office, one of the walls is covered with a whiteboard. On this whiteboard a timeline was drawn along the entire length of the wall. This timeline was divided into columns, one for every remaining week of the project. Horizontally, every sub-system was given a row spanning the entire timeline. For every week, the main tasks for that week was listed, sub-system by sub-system. Every sub-system was represented by a small cut-out of the DNVFF2 car, which acted similar to a piece in a board game; as tasks were finished, the piece moved closer to the finishing line. The pieces would move independently of the weeks, showing the relative completeness of the sub-system. Important messages and their due dates are written above the corresponding week. The idea is represented in Figure 3.21.

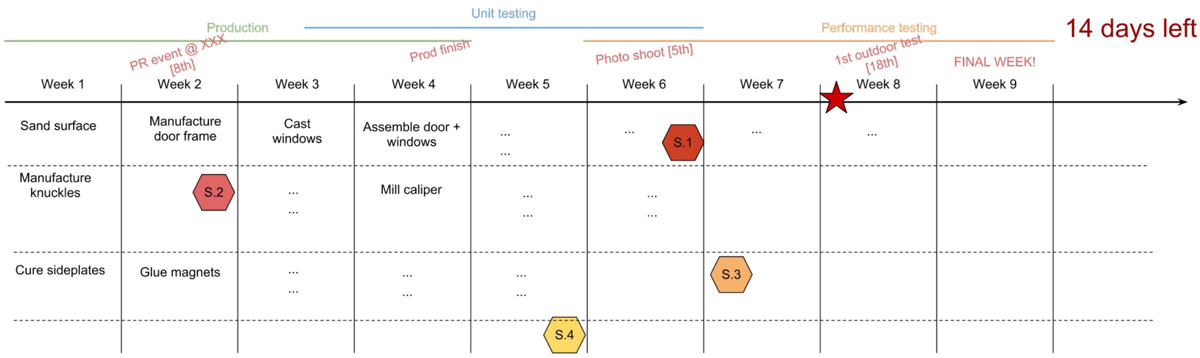


Figure 3.21: Project timeline divided into Takt Periods of one-week durations. Tasks for every sub-system are listed in the weeks they are due. The hexagonal pieces indicate sub-systems and their progress towards project finish. Important messages are listed above the timeline. The red star indicates the current date.

The Timeline was updated every Monday at the weekly meeting. One notable effect - that the author noticed immediately - was that the Timeline reduced the need for communication; team members - particularly the ones not permanently located in the DNVFF office - consulted the Timeline for information. This made a positive impact on the duration of the Monday meetings, as no time was spent asking “when does this or that happen?”.

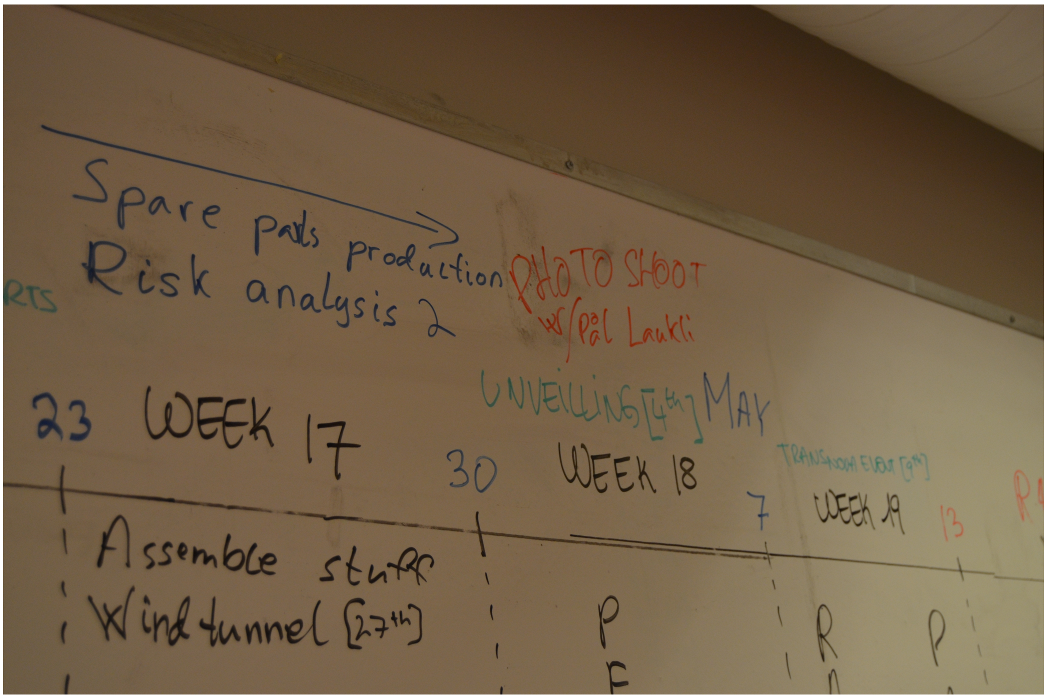


Figure 3.22: Close up of the top part of the Timeline.

The pictures in Figures 3.23 and 3.24 show the actual timeline as of week 13 and week 16 of 2012. By week 16 it is clear that sub-systems 9 and 10 are straggling due to issues with the motor. The motor had been progressing slowly between week 13 and 16, yet made a leap in week 15. Unfortunately, the stator short-circuited, which halted the motor from progressing yet again, which also hindered the control systems in their progression. By week 17, the motor is again ready for testing. The motor wheel as well - sub-system S.11 - cannot progress without the motor finishing, explaining its lagging behind.

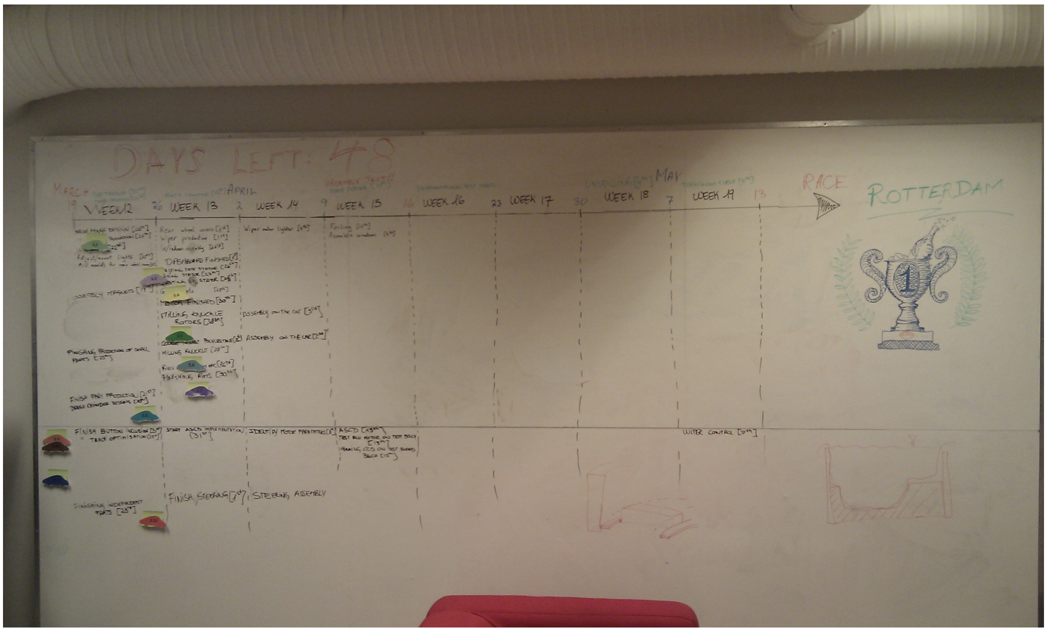


Figure 3.23: The project timeline as of Week 13. The WBS was planned up till week 15. From week 15 and out, the weekly activities were planned continuously based on the outcome of testing.

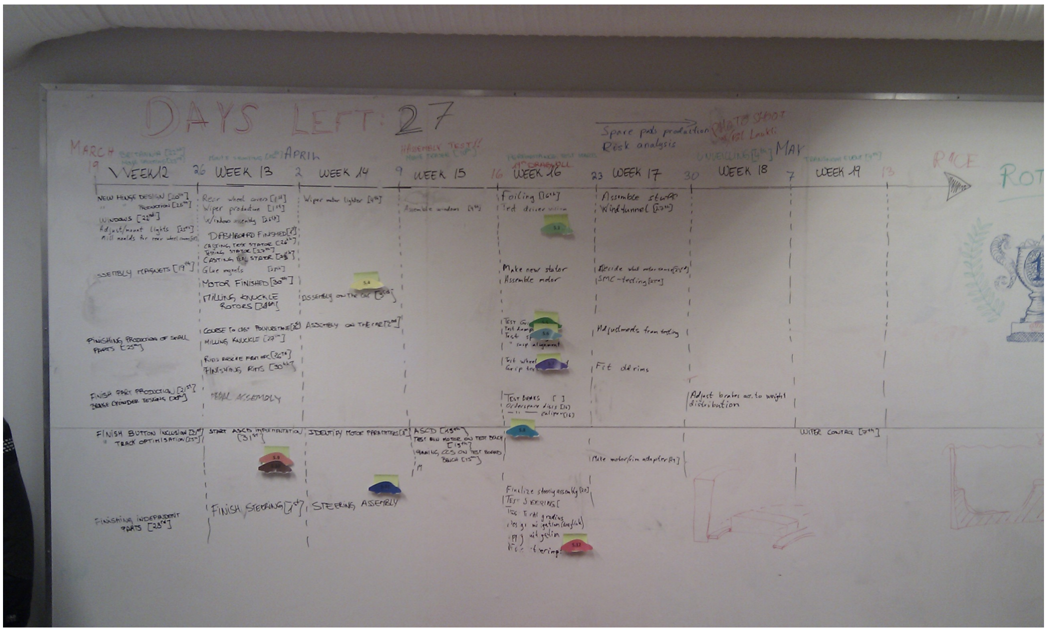


Figure 3.24: The project timeline as of Week 16.

Figure 3.25: The project timeline as of Week 17.

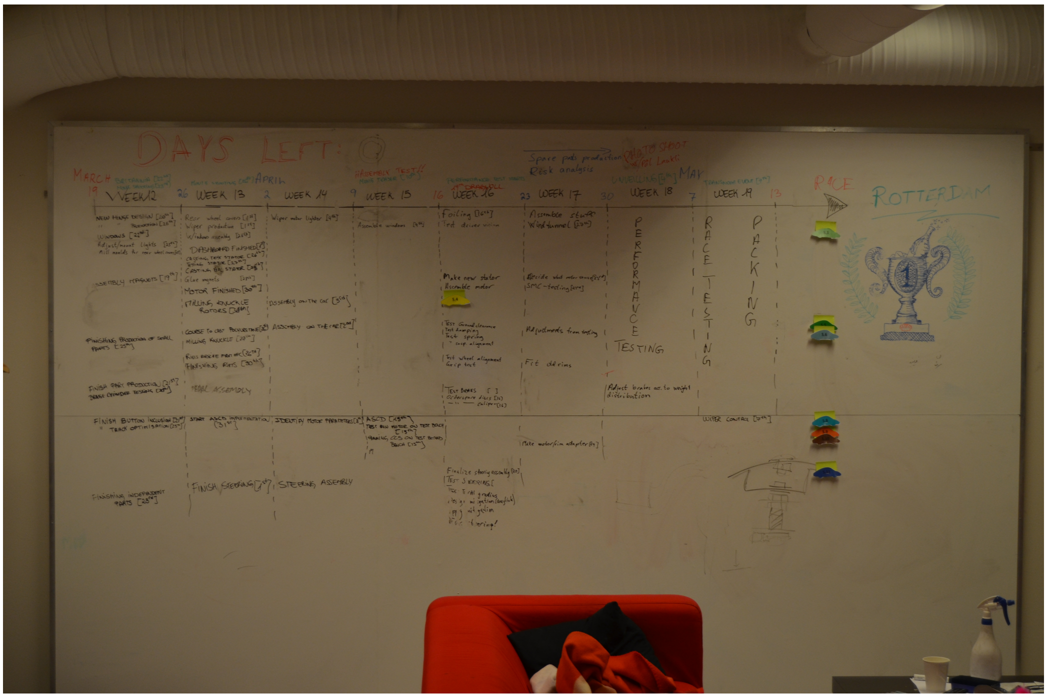


Figure 3.26: The project timeline as of Week 19, just prior to leaving for Rotterdam.

Figure 3.26 shows the status of the project just before leaving for Rotterdam. The motor is lagging, due to it never passing the testing in week 17. The team decided to go with the old motor, which was in backup.

The Wall-Architecture

The other way of visualizing project progress - called the Wall-Architecture - displays project progress all the way down to a component level, with particular focus on manufacturing and integration. The Wall-Architecture is an alternative application of the system model which was built for the Legacy program. This tool employs the system architecture, which is a hierarchy of every component in the system. This hierarchy is displayed on the wall. By using red, yellow, orange, green and blue sticky notes the current status of every component can be visualized;

- Orange - under construction, being manufactured
- Yellow - suspended; waiting for something or someone stated on the sticky note
- Green - finished, meaning manufactured
- Blue - assembled, meaning fitted to the vehicle
- Red - critical, urgent!
- None - nothing happens

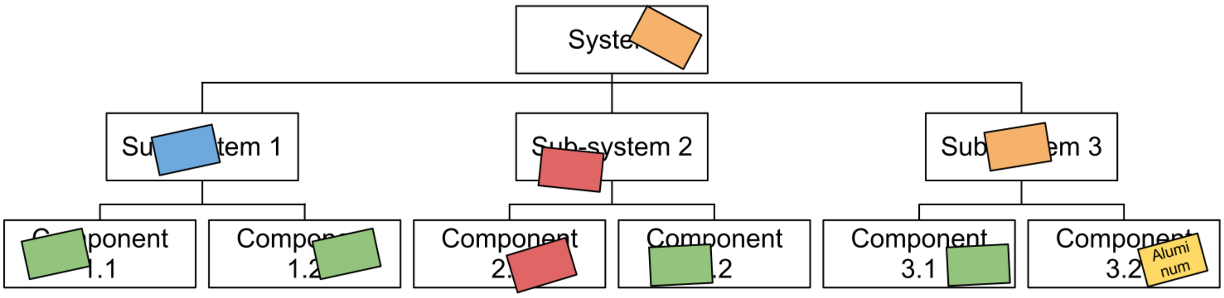


Figure 3.27: Illustration of the principle behind the Wall-Architecture. The system is currently 'under construction' told by the orange sticky note on the root node. Sub-system 1 is finished and assembled, sub-system 2 is critical due to an issue with component 2.1, and sub-system 3 is under construction, yet waiting for more material for component 3.2.

This tool was located in the workshop, on a wall dedicated for SE purposes. The author would visit the workshop daily and talk to the team members to get updates on the progress. The wall was updated in a rather ad-hoc manner - as opposed to the more formal Stand-up meetings - sometimes bringing one or two members over to the wall, sometimes updating the sticky notes after just speaking with them at their work stations. The Wall-Architecture was clearly visible from a distance, the colours demanding attention.

When green or blue colours were added to the wall cheers would go up from the people in the room.

The background for developing this tool was originally a desire from the author to find an extended use of the system models. A member of the team told the author he often had available time at the workshop (waiting for machinery, carbon fibre to cure, etc) that he would like to spend helping others. However, there was no efficient way for him to know where to apply himself. The Wall-Architecture was devised to help the team increase its collaboration and efficiency. [i]

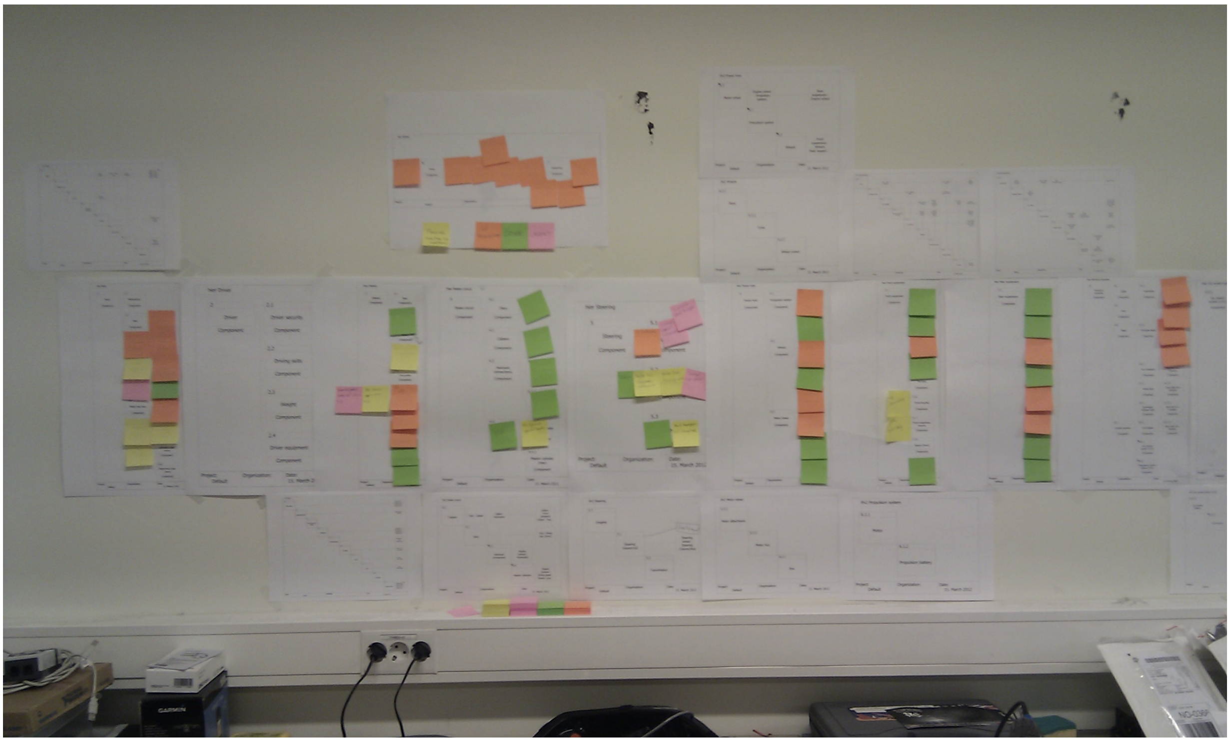


Figure 3.28: Project status on March 12, 2012.

[i]The Wall-Architecture is by no means a new invention, yet in this environment it had its outspring from these two converging thoughts.

Figure 3.29: Project status on April 11, 2012.

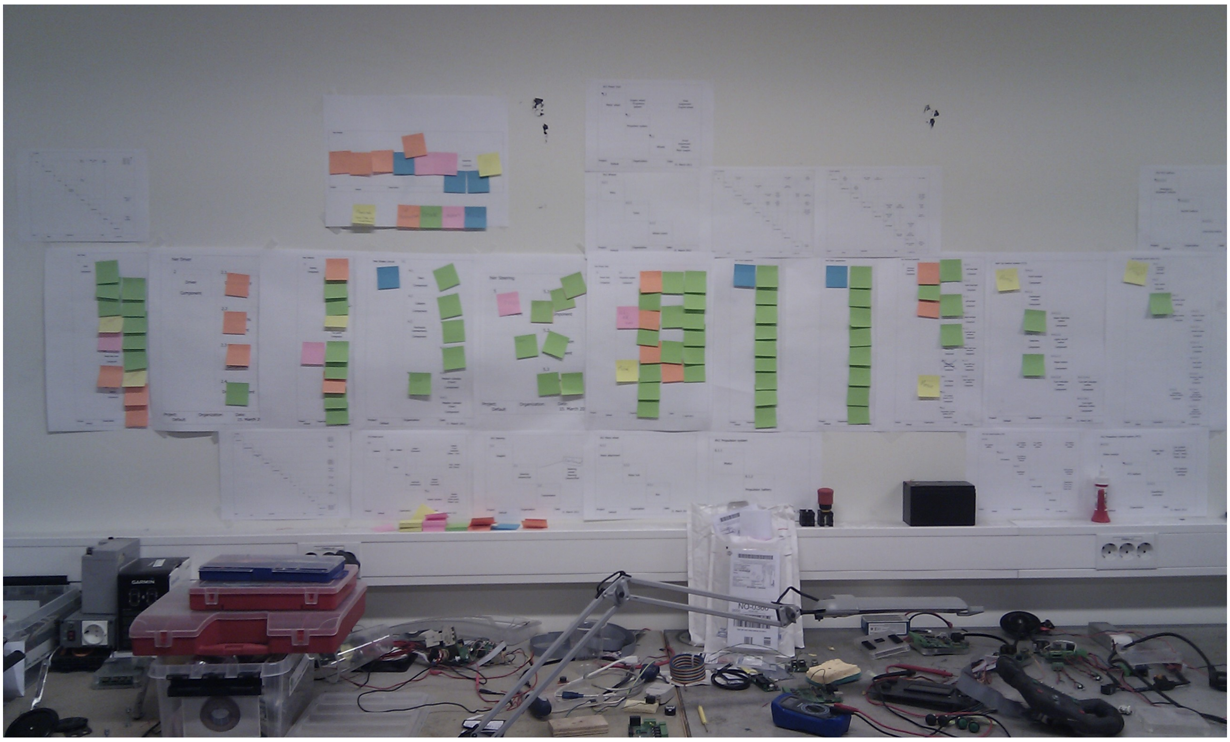


Figure 3.30: Project status on April 23, 2012.

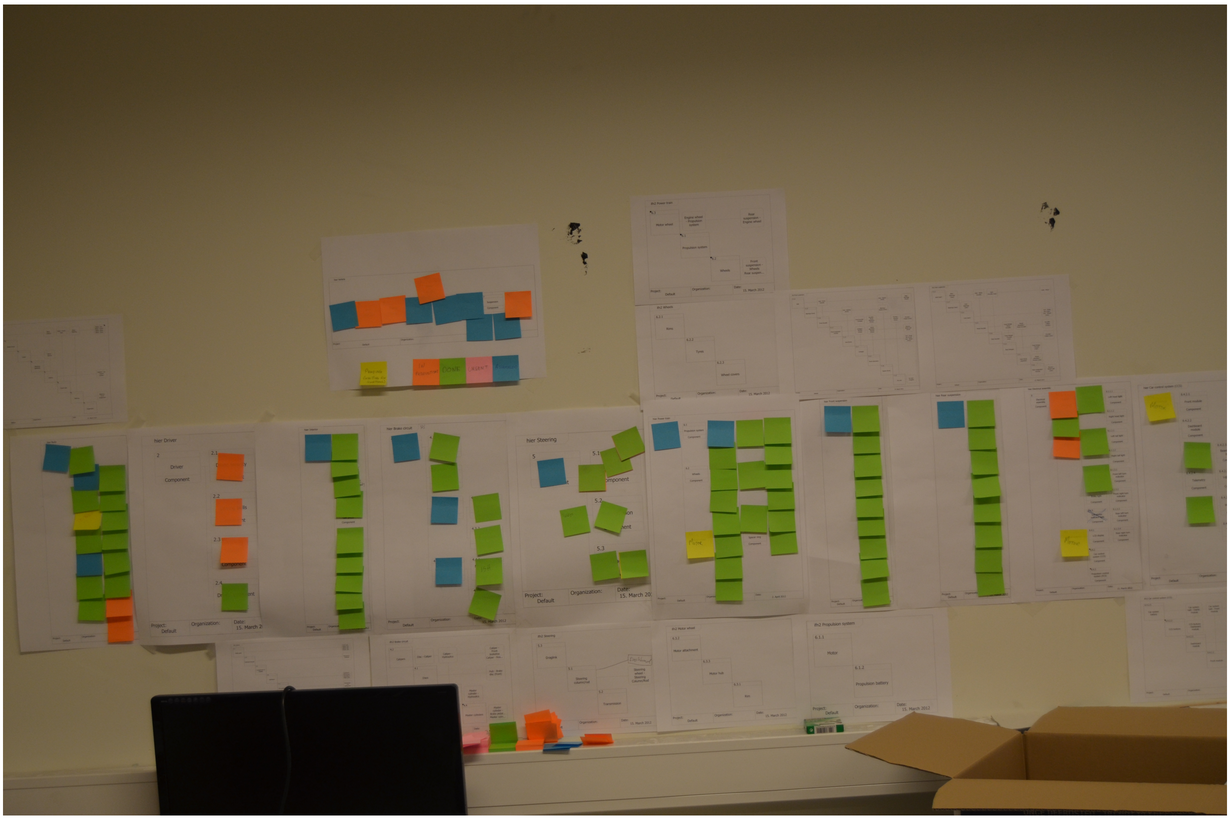


Figure 3.31: Project status on April 30, 2012.

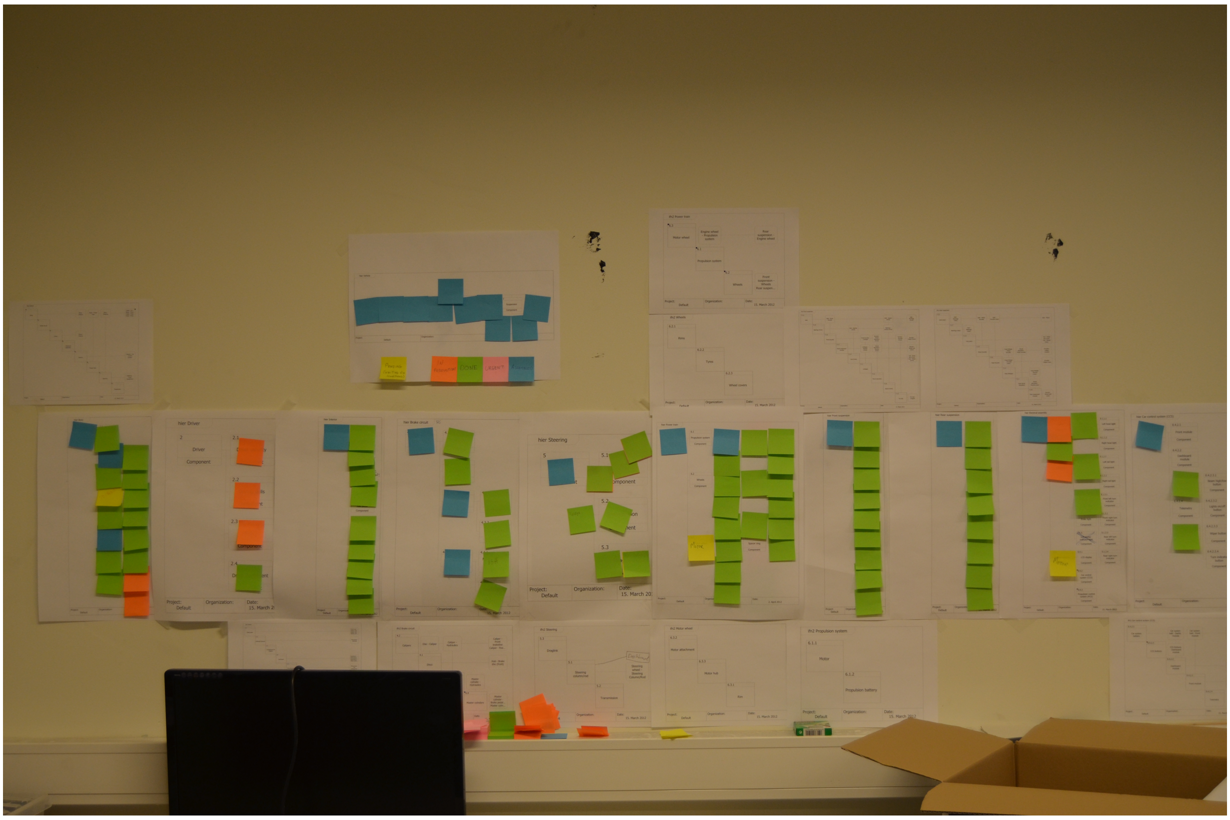


Figure 3.32: Project status on May 12, 2012, the day before departure for Rotterdam.

3.4 The author's participation in DNV Fuel Fighter 2

The author participated in the team as one of two Systems Engineers. Together with the project manager, these three made up the organizational part of the team. The other systems engineer - Itsaso Yuguero Garmendia - concentrated her main efforts on VV&T activities. Together the SE team worked on WBSs, risk analysis and requirements analysis.

The SE team focused heavily on communication. The team saw communication as one of the most important aspects for reaching project goals after witnessing that slow or imprecise communication caused the project to suffer. The various ways of visualizing project progress was also a way of responding to this need of improving communications.

The author's main tasks were system modeling, knowledge capture and updating visual project progress tools. The author was also the main contact point between the technical team and the communications officer, who had offices on another campus. The author also organized all excursions, e.g. all events where the team had to appear with the vehicle. There were quite a few of these events, and every event consisted of many hours of preparing. The rest of the team also bestowed the author with the responsibility of speaking whenever the team presented their work to sponsors, interested parties and at the official unveiling.

Working in this project was a job more than a master's thesis.

The SE team viewed themselves as the "glue of the team", stepping in whenever they saw the need for intervening. Communications, project progress and "first time right" were key aspects in their work. They also experienced that SE can be a tough job. It is long hours that do not result in a tangible product, like the work of the other engineers who produce parts and code. Of that reason, the other engineers may not always understand the work of the Systems Engineer. Being two was a great support for the author.

Chapter 4

Results

The author chose to focus on two of the four Lean enablers proposed by the Lean assessment tool for Knowledge management. For Continuous improvement the author chose to focus on a set of enablers that better matched the size of the organization than the one proposed by the tool. The enablers in focus and the chosen method of implementation is summarized in Table 4.1.

Lean Systems Engineering effort	
Lean enablers	Elected implementation
Knowledge management	
- Create mechanisms to capture, communicate, and apply experience-generated learning and checklists - Maintain team continuity between phases to maximize experiential learning. Capture and absorb lessons learned from almost all programs: "never enough coordination and communication".	The technical K-briefs, the IPM template and the 'Plan-Do-Check-Act' template, the interface contract template and the MBSE effort with system modelling and MBD.
Continuous improvement	
- Arrange Stand-up meetings - Design VPB - Perform risk analysis and keep it updated. - Visualize project progress on a sub-system level, possibly also component-level	Stand-up meetings three times a week Task overview and messaging on the tailored VPB Risk management Component-level progress control through average risk level, Timeline and Wall-Architecture.

Figure 4.1: Table showing how solutions are mapped to Lean SE enablers.

4.1 Knowledge management in DNV Fuel Fighter 2

4.1.1 K-briefs

Technical K-briefs were made for each sub-system of the DNVFF2, as well as for some prominent components or sub-sub-systems. Each one was made in collaboration with the responsible engineer for each sub-system. A collection of K-briefs are available in Appendix C, both technical and process K-briefs.

4.1.2 Final system model

The system models contains four tiers; requirements specification, functional analysis, system architecture and verification procedures. Together they make up over 900 elements. The main effort has been on the requirements and the system architecture. Verification has been handled by Itsaso Yuguero Garmendia. Appendices D and E show a collection of diagrams from the system model.

Requirements

Figure 4.2 shows one of the requirements diagrams created in Core.

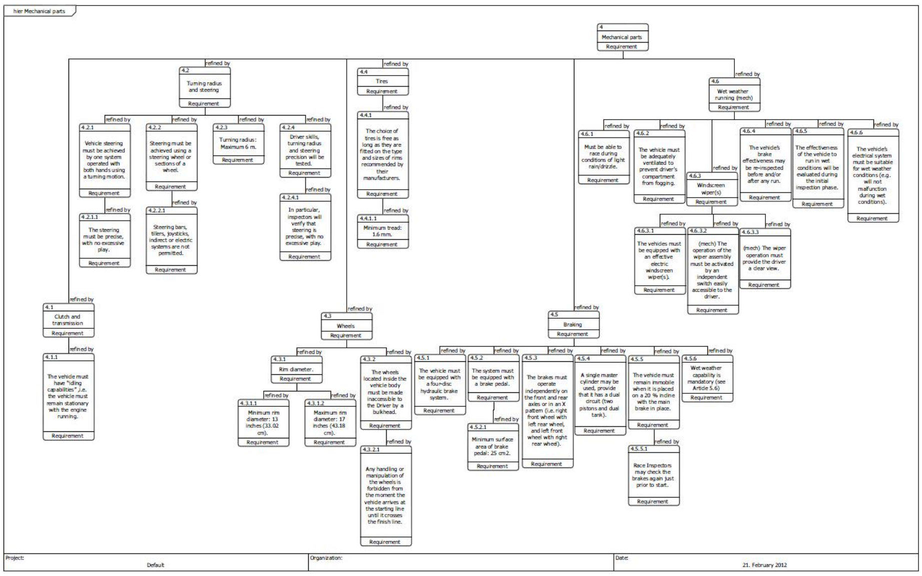


Figure 4.2: An example of a requirements diagram, showing some of the requirements for the mechanical part of the DNVFF2.

Functional analysis

Figures 4.3 and 4.4 show two of the FFBDs created for the DNVFF2. The FFBDs tell how the inspections are performed in Shell Eco-marathon. The I/O say who and how many must be present, and what equipment is used at every stage.

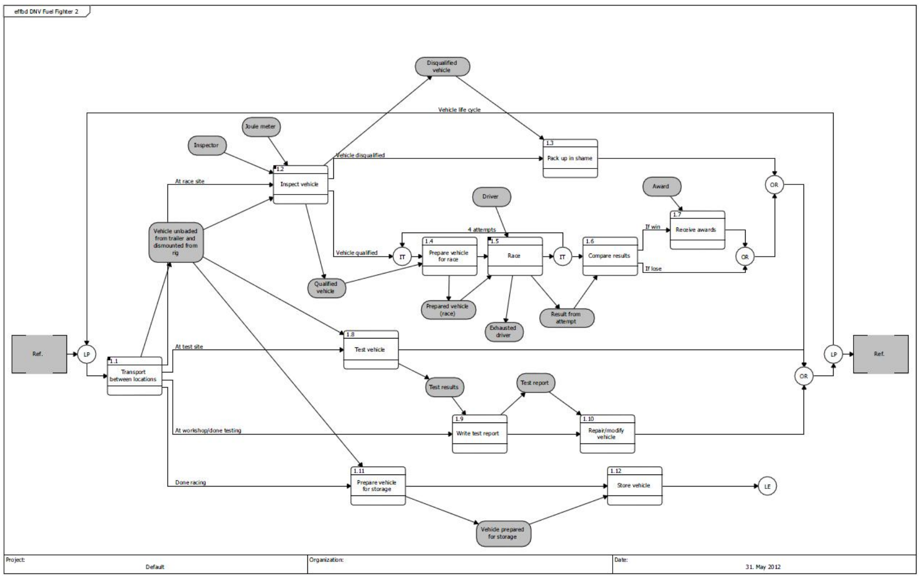


Figure 4.3: An example of an FFBD developed for DNVFF2. The FFBD shows how the inspections are performed at Shell Eco-marathon. The function "Full technical inspection" is shown in Figure 4.4

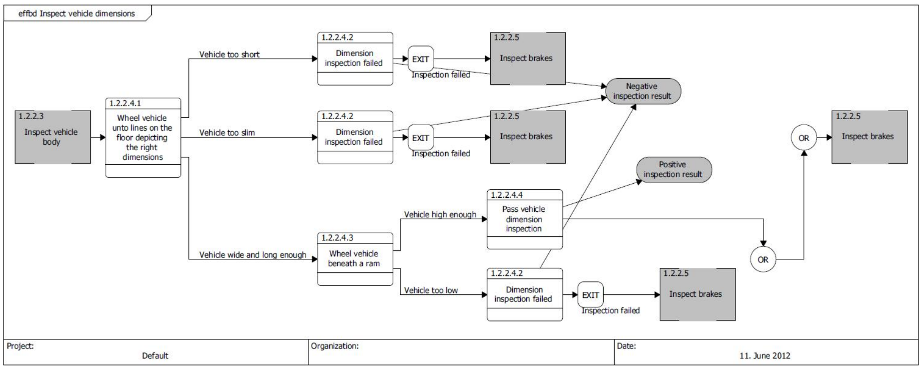


Figure 4.4: The FFBD shows the function "Full technical inspection", describing how the technical inspection takes place at Shell Eco-marathon.

System architecture

The system architecture expands on the one made by the systems engineer of 2011, which only had one level of detail. This system architecture is an as-built description of the vehicle, going down to single components, yet not quite down to bolts and nuts.

The architecture of 2011 divided the vehicle into the sub-systems Body, Driver, Interior, Brakes, Steering, Wheels, Motor wheel (wheel containing the motor), Front suspension, Rear suspension, Car control system and Propulsion control system. The new architecture is not much different, yet builds on an example by Friedenthal, Moore and Steiner [4, p. 68], resulting in the division shown in Figure 4.5. This division collects Motor, Wheels, Motor Wheel and energy source into one category called the "Power train", the control systems are collected into the "Electrical assembly" along with the lights, horn and wiper. The front and rear suspensions are also collected in a category called "Suspension".

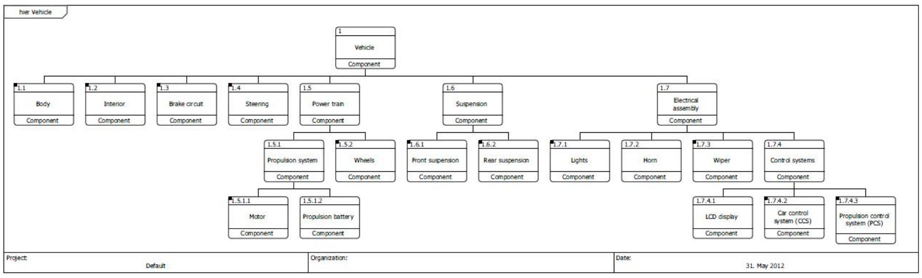


Figure 4.5: The top-level system architecture of DNVFF2.

The vehicle itself is not the top level of the architecture. In fact, it has a top level which also includes non-technical elements from the system-of-interest. The sub-system "Driver" - due to not being a technical sub-system - was moved into this non-technical category, which is called "Race" which describes elements that are important in a race situation. The top-most level of the architecture is shown in Figure 4.6. The decision to include non-technical elements is linked to the MBD; if the model is to be a repository for knowledge, it must also include process knowledge or other non-technical knowledge. The word "system" is a wide concept, and in this case the system has been defined to also include what surrounds the vehicle, just as has been done in the functional analysis.

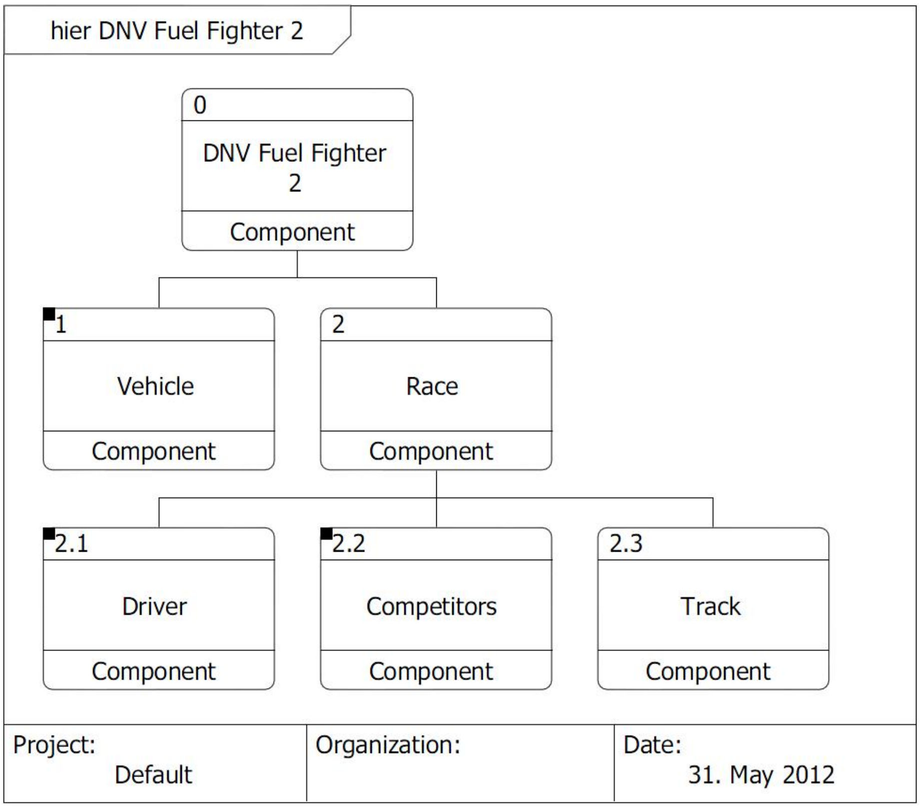


Figure 4.6: The top-level system architecture of DNVFF2.

Figure 4.7 shows how the architecture displays alternatives in the design or component choices made. The green boxes are chosen alternatives, the orange indicate are discarded ones.

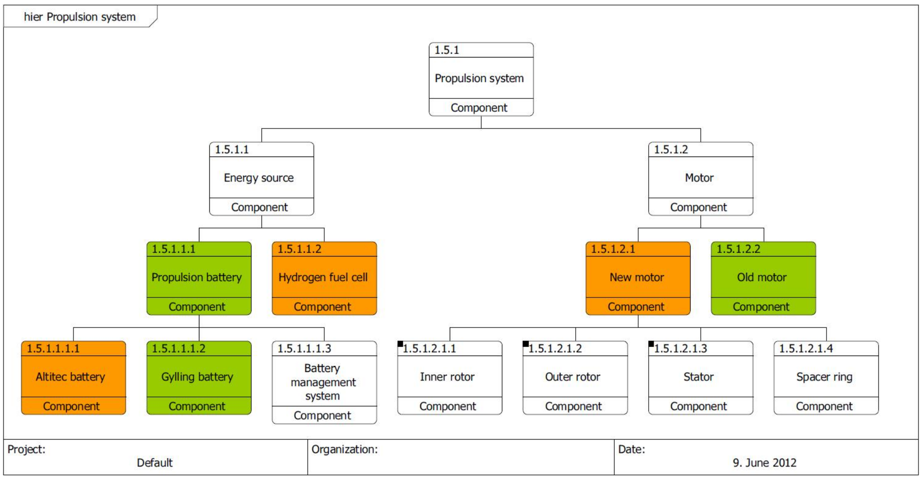


Figure 4.7: The architecture also shows alternative design concepts or existing alternative components. Colour codings tell which solution is preferred; green is the chosen alternative, orange is the discarded one.

Figures 4.8 and 4.9 show the architecture and interfaces of the front suspension.

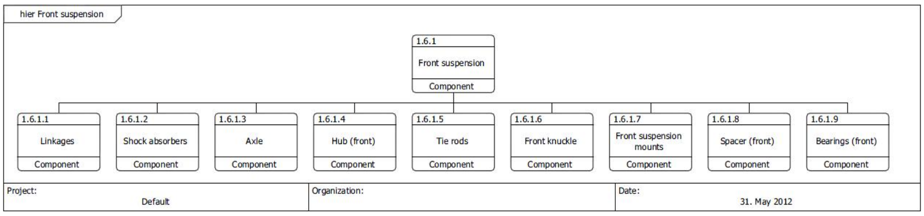


Figure 4.8: Architecture of the front suspension.

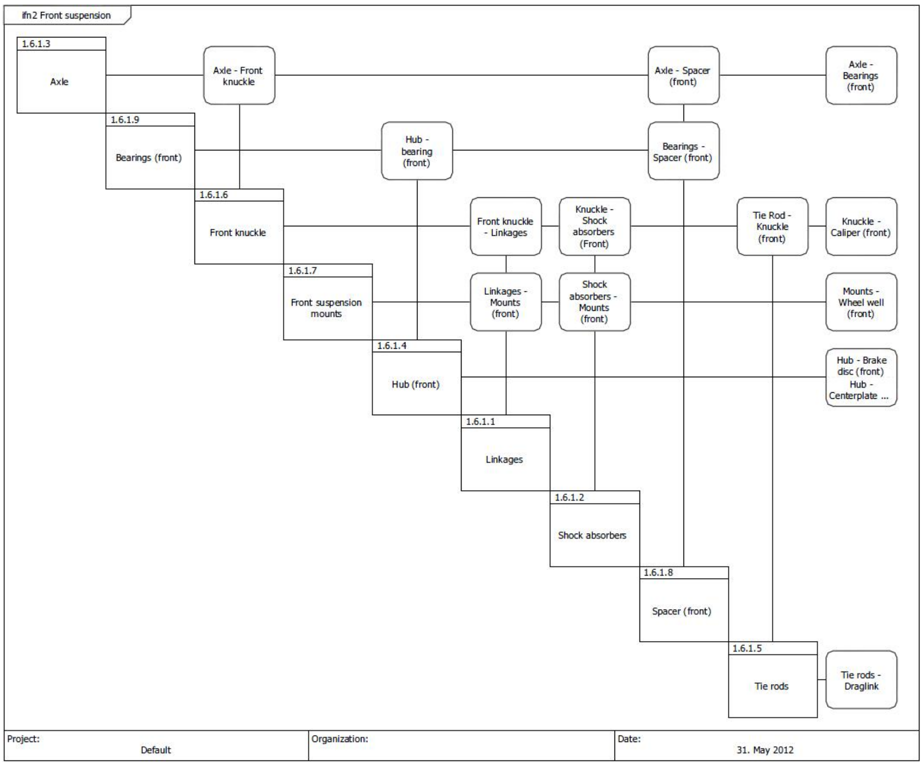


Figure 4.9: Interface diagram for front suspension.

Published model

The published model is a HTML webpage that much resembles Wikipedia in the way that the user makes use of hyperlinks within both text and illustrations to navigate. Every element comes with at least two different screens; a property sheet and a hierarchy, and either a FFBD (in the case of functions) or an N2 diagram (in the case of physical elements). In the property sheet, one can read the descriptions of the element, access external files (e.g. K-briefs or CAD files) and see an overview of interfaces, linked requirements, physical elements or functions. The model is easy to use and navigate.

However, the model only support reading; the user may not add files to the repository. This must be done by the one who possesses the Core files. Anyone familiar with HTML may add the hyperlinks themselves. However, this decreases the robustness of the model as the links only exist in the HTML files, not the Core model files and will be deleted whenever a new version of the model is exported.

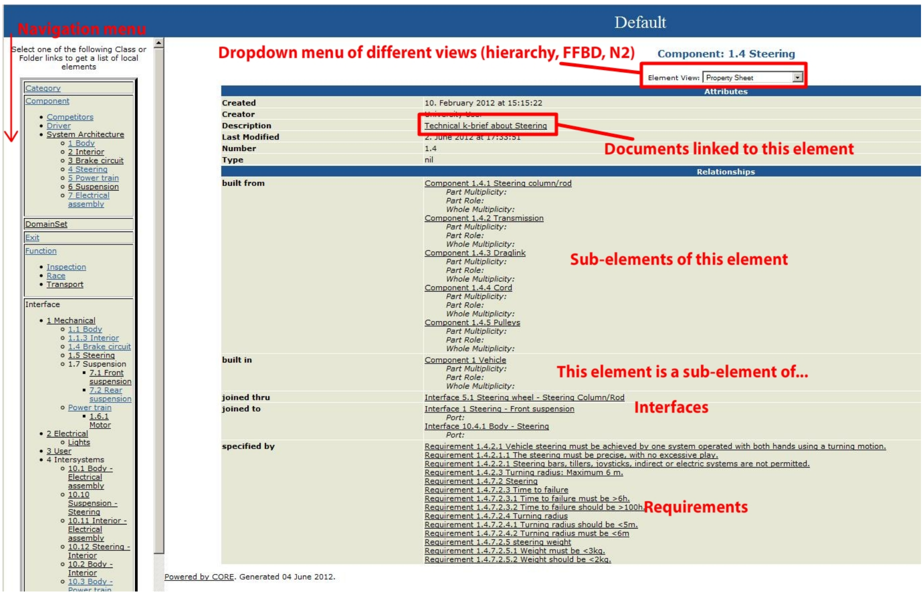


Figure 4.10: A typical view of an element's Property Sheet in the published model.

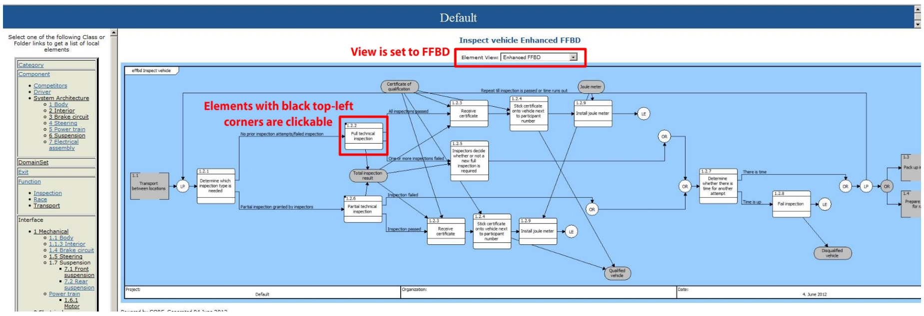


Figure 4.11: A view of a FFBD. Elements with black top-left corners are clickable and opens the property sheet of the element being clicked.

4.2 Visual Workflow Management in DNV Fuel Fighter 2

For this project, the following has been observed with regard to Stand-up meetings and the VPB:

- Three times a week is sufficient
- Maximum number of active participants is 7 or 8
- 15 minutes is stretching it. 12 is enough.

- Location matters during production periods. The meetings must be held in or close to the workshop.
- Addressing the three questions (see Section 2.5.1) makes the meetings shorter and more relevant to the listeners
- The meetings are best held in the morning, during production periods due to the accessibility of machines.
- Stand-up meetings are very important during the competition, due to the ever-changing environment that such a competition is.
- VPBs are great tools for managing team work over a short period, and for communicating important messages between team members.

Regarding visualizing project progress, the following has been observed:

- Tracking project progress in more than one way gives a better and more diverse overview of what is actually happening
- Visualizing the project progress acts as a message board for the team. It is especially important for the team members who only visit the main office a few times a week.
- Visualizing project progress creates a sense of urgency within the team members. Moving forwards on the Timeline or adding green or blue colours to the Wall-Architecture strengthens morale.
- Average risk level is a difficult way of tracking progress. It requires a mature team who has dealt with risk management before, and who has time available to properly analyze the identified risks.

4.3 Second Lean assessment

The second assessment was held on May 30, a week after returning from Rotterdam. The purpose of this assessment was to measure the effect of the steps taken after the first assessment. The author started with a recap, where the results of the first assessment were repeated to the team, before doing a more in-depth presentation of the steps taken to improve the project from the aspect of Lean Systems Engineering. The team then filled out a new assessment, where they only regarded the new current situation for the five questions that exceeded the gap limit in January.

The questions reassessed are:

- 2:1: How does the team rate the value of knowledge? Does the team consider its own collective knowledge as an asset?

- 2:2: How is knowledge ownership defined and organized in the company, including responsibilities for knowledge system, standard, creation and capture skills, and improvement?
- 2:3: How is knowledge transferred between functional departments (teams, team members)?
- 2:4: To what extent is Set-Based Concurrent Engineering (or other equivalent methods) generally accepted and adopted as the company's main philosophy for designing products?
- 4:1: Is continuous improvement, waste elimination and pursue for perfection in product development an integrated part of the team's philosophy?

The desired values are unaltered from the first assessment. Figure 4.12 shows the results of the second assessment compared to the results of the first assessment and the desired values. Figure 4.13 shows the gap between current desired values, for both January and May.

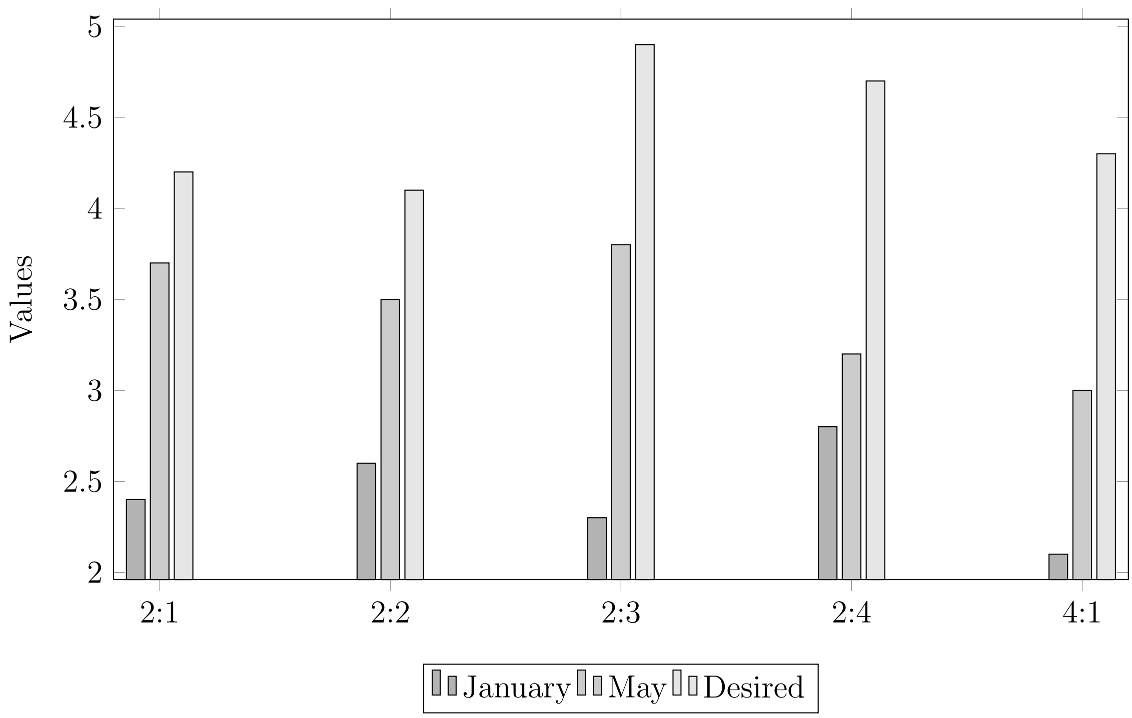


Figure 4.12: Average Current and Desired values

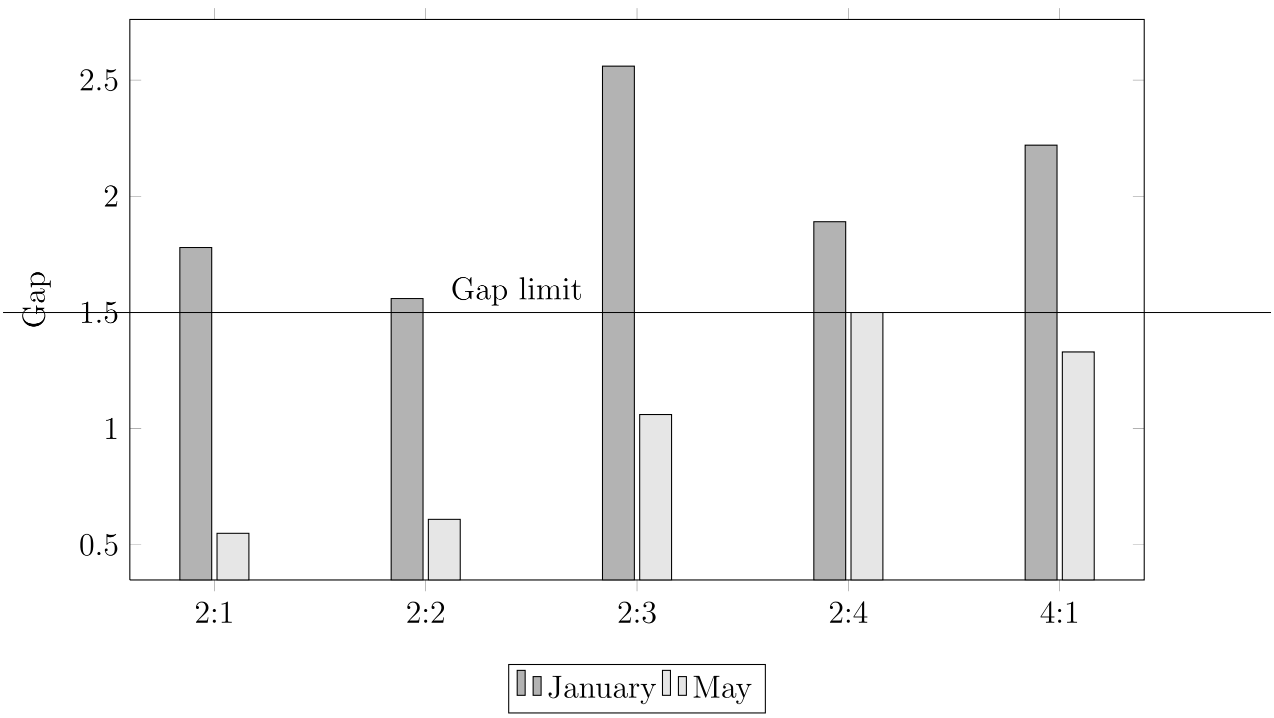


Figure 4.13: Gap between Current and Desired state

The results show an improvement in all categories, however the desired value is never actually reached. However, three out of four categories drop below the 1.5 gap limit. Question 2:4 has a gap of exactly 1.5, yet improves with 0.72 from January to May.

The team expressed that the Stand-up meetings were a good way of organizing the team, especially in Rotterdam. They also indicated that the Timeline was a good tool, yet implemented too late in project. For future use, the team would have liked a more precise production plan or WBS before using the Timeline. The Wall-Architecture is a tool the team stated they utilized at the workshop. Posting green and blue sticky notes were occasions that could be celebrated and which increased morale.

The team also responded well to the K-brief templates, and decided to even put these in their own master's theses. One team member even stated that "the K-briefs makes me want to read its contents. It is a good tool." This was also the first time the team got to see the system model, and they agreed with the way of structuring information. They were uncertain whether the model would actually be used as much as it should, though, given that the information database of DNVFF is not that huge, yet.

When asked to assess how much time the project may save at the onset of a new iteration given this knowledge transfer is in place, the team agreed upon two to three weeks, based on the time they themselves spent in the early phases of the 2011-2012 iteration.

Chapter 5

Discussion

Lean Thinking has done wonders for Toyota and Honda in Japan. Yet, one must not forget that these companies have spent generations anchoring this philosophy in their organizations. Starting up with Lean principles is not something that is done overnight, as this thesis well shows. The real value of Stand-up meetings was not clear to the users till the going really got rough and the team needed to improve their communication to make it. Up till that point the benefit was still elusive for some team members.

The idea of Lean is to make things faster, better and cheaper. However, the thought of, e.g., integrating engineers from all phases of the project (design, VV&T and manufacturing) in the design process, or running all project phases in parallel, increases the amount of communication exponentially.

Communication is very hard to manage. This year's SE team focused particularly on the flow of communication. And it was difficult, indeed. All of the tools employed for Workflow Management - and indeed also for Knowledge management - were meant as tools for communication. But it takes time, a lot of time, to work such routines into the habits of people, and not till the latter phases of the project did this start to take proper effect. Lean contains several ideas - and some concrete ways - of managing communication, but they only add to all the other things that must be learnt when starting in a new organization.

The team employed Responsibility-based planning. The author chose to add this point to this thesis merely because he saw how high the risk of failing at this point really is. The project experienced not reaching many of its goals - some big, some small - due to team members not being able to reach their own deadlines and not realize that they were missing them until it was pointed out to them. Responsibility-based planning just does not fit everyone. The project manager shall have much credit for employing such a modern tool, but this is a method which relies on the individual's ability to assess its own capacity - and the individual's ability of insight just varies too much from person to person. Not everyone is able to set their own deadlines and keep them, some actually needs to be told what to do - such is nature. Maybe it is something that must be learnt and this project is too short to learn the lesson in time, but it may also be that this is something that does not work as well in Western as in Eastern culture.

Proving that something works, that is the real challenge. No one will employ an unproven technique, and the utilization of a new technique will drop if the benefit does not present itself soon. That is the challenge for SE and Lean, both. If implemented correctly, nothing goes wrong and you are none the wiser. The value of the enablers are not apparent till something breaks in their absence. The team responded slow to VWM, yet jumped immediately on Knowledge Management. The reason is simple; they had been in the situation where they knew nothing and there was nowhere to turn for information. The idea of K-briefs and an information repository (the system model) was something they understood at once and the benefit was apparent.

Oppenheim states that SE is too difficult to implement properly in its current form, and gains support from within the SE community. In the author's view this also applies to Lean. It is a mindset that collides with the common way of regarding development processes, and it needs time to properly sink in. However, Oppenheim realizes that just stating that something does not work is not enough, and comes a long way by proposing specific Lean enablers that may improve the SE effort. But simply using Lean enablers is not enough; it still takes time, commitment and a focus on culture to make it work. Just as with traditional SE.

Combining Lean and SE did not give any conflicts to the application of SE in this project. The two disciplines share many of the same thoughts and goals. The idea of both is to reduce risk and to produce the best product in the eyes of the customer. Lean is a product development method, SE is a field of engineering specializing on making certain the applied development method is performed correctly. A link between the two appears natural to the author.

As both systems engineers in this year's team has experienced; SE is difficult. It is a discipline populated with different approaches to its application. If viewing the discipline through the goggles of Lean helps remove the dead weight of the discipline, SE may come a long way. Lean adds a new perspective to the debate; value. If a SE practice is not considered to add value to a product - neither directly nor as an enabler - then this practice should be terminated. Oppenheim provides a good roadmap for SE with his LEfSE.

And sharing is where MBSE seems to struggle. The published models from both Core and TopCased do not support collaboration. The shared models are HTML files which do not link back to the database, making the users only readers and not writers. Whenever a systems engineer has to update the model with new documents or elements that could have been added by other members of the organization, the effort is a waste. The time spent adding the element is time that could have been spent in other areas. MBSE needs to agree on a standard, and it needs to follow the development in other areas where anyone can share contents in a shared repository.

All in all, MBSE is value-added, but its implementation must be discussed and tailored for the system it meant to describe.

The results of the second Lean assessment, shows that the team has belief in Knowledge management, and in the way this thesis suggests performing it.

The K-brief is a very easy way of presenting information, but will not serve its purpose without the right context. The K-briefs need to follow a template to save time for the users and encourage filling them out, the organization needs to adopt a culture for making and using the K-briefs, and the users need to know where and how to find or store them.

What is the right culture? Standardization is key. Capturing knowledge should be done at specific points, preferably at important deliverables, and enforced by a strong leader. For this project, twice a month during the design period is recommended. Making use of the templates presented in this thesis will enable the team to do this quick and without forgetting important information.

A "consult the knowledge database first" attitude should be encouraged whenever something is unclear. Whenever something new has been learnt of general interest or an important problem solved, the next step must always be to write a "Plan-Do-Check-Act" K-brief. Making sure this culture is adopted is a responsibility for project management and SE together. Utilizing the VWM tools - especially the VPB - for sharing the newly created knowledge is strongly encouraged. Increase the value and utilization of the VWM tools by making them "alive". Employ colours and few words in the K-briefs, put them on the VPB and "lure" people over to read them.

This thesis employs MBSE to build the surrounding system that stores and shares the knowledge. The decision to use MBSE was based on it being an already defined standard that is gaining support in the industry. It is always a risk when making such systems that the resulting system is only understandable for the maker, and appears alien to the user. By employing an already existing standard, the author hoped to reduce this risk.

The published HTML version of the model is an easy and very accessible tool for a reader. However, it does not facilitate adding new material without going via the systems engineer that built it. Nor does it support a search engine. These are negative points for an otherwise great tool. However, for a system as small as the DNVFF2, this problem is a minor one. The amount of information created in this project is not likely to surpass an unmanageable level for the team.

The uncertainty that this thesis does not uncover, is whether the system model will actually be utilized by the new team. The diagrams will take some learning, and the focus of the model may not be the correct one. On the other hand, there will not be any one else there to tell the new team how the system works, except the system model. The 2012 team understood the concept of K-briefs immediately, and it is believed that the next team will find these very helpful and immediately understandable, hopefully to the extent where they decide to keep up the work with knowledge capture.

For Knowledge management to become a success story, the teams need to have a member that is Knowledge manager. Preferably, this is a systems engineer who knows system modelling and has insight into the various sub-systems that make up the vehicle and surrounding system.

Is MBSE value-added? It depends on its implementation. If the system model is made just for the SE team or it is a too complex description of the system-of-interest for the readers to understand, it will be of limited value to the project or organization. However, if it is shared with the team to enable communication, knowledge transfer and VV&T activities, it will become an enabler and it will help elevate the rate of value-added activities within the organization. It may even become directly value-added if the system model is made available to the customer to help validate the system and reduce the risk of making an undesired product.

To successfully implement VWM the team needs to spend time making proper WBSs. Planning the future effort is the foundation of tracking progress. When the WBS is written the contents of it can be summarized on the Timeline while the more specific WBS should be employed to populate the Wall-Gantt with tasks.

VWM must be employed early. Especially for projects of this type where time is very scarce, early implementation is paramount. It takes time for people to get the Stand-ups into their routine, as well with utilizing the tools for information gathering and sharing. One must not also forget that the participants of DNVFF are students, most likely unfamiliar with such enablers.

Parts of the team will feel that these meetings are a waste of time. It is important to press on in such cases, and look for ways to motivate them. Remind them of past successes using the tool, even minor ones, or perform stress tests where the team is forced to organize their communication in order to finish. For this team, that stress test was the competition - and it was a very good lesson for all.

This paper has addressed the implementation of Lean in an academic project work, a field where Lean has not been tested much before to the author's knowledge. In fact, when speaking to professors at the institute, some showed scepticism towards implementing Lean in this project as it was a student project and not industrial. This attitude surprized the author. The DNVFF is a perfect playground for anyone doing research on both Lean and SE, where the enablers, tools and techniques can be tested in a real environment, not a simulated one. This project actually produces a result in much the same manner as in the industry. The participants work with the same tools, goals and methods as in the industry. Saving time, reducing waste and "first time right" is very important in such a time-constrained project. Applying new theories in the hopes of achieving a better result in less time - and gaining an edge on the competitors - is very much in the desires of the team, project owner and sponsors.

The author recommends using this project for testing out Lean principles in the future. With the knowledge transfer secured, the project will be able to (almost) continue where it left off when the previous team graduated. Also, the fact that new team members enter the team every year makes a perfect platform for always testing new enablers or procedures on “blank sheets of paper”; people uncorrupted by the industry.

Chapter 6

Conclusion

Knowledge management is an important enabler for the DNVFF2 project. The fact that teams only work on the vehicle for one year at a time, attending one competition and then graduate, makes the project vulnerable to knowledge drops every year. The only way of securing an improved system year after year, is to employ Knowledge management.

The system model and K-brief templates designed in this thesis will aid the project in the following years to capture and share their knowledge. The knowledge captured this first year of the DNVFF2 is paramount for the succeeding teams; this was the year that many major decisions were made who will affect the project for years.

The project must employ Systems Engineering and Knowledge management in the following years to make sure the knowledge is not lost. The model must be kept up-to-date, and the templates must be used. The teams should stop and capture knowledge at least twice a month during the design phase, which is when the major decisions are made.

The legacy that this team has taken the effort to capture and share with next year's team will reduce the time needed in the early phases of the next iteration, giving the team more time to explore concepts and design solutions. It will also have a great positive effect on the risk level of next year's effort as many uncertainties are removed already at the onset of the year.

Knowledge is key to reducing risk levels. This makes knowledge management an important SE effort, and employing the Lean enablers is a great way of tackling this challenge.

Stand-ups enable the participants to share information, get help and to respond quickly to changes. The Stand-ups should not take more than 12 minutes, and should have no more than 8 participants. The location must be close to the work station of every participant. A VPB is a very helpful tool for these meetings.

VWM enables the team to follow the progress of every sub-system, down to component level. Being able to follow the progress in different ways eliminate the chance of missing important details or perspectives. The Timeline gives a good overview, while the Wall-Architecture gives a detailed view of the progress of the team. The VPB gives a snapshot of what is happening on a day-to-day basis. Average risk values may be a great tool, yet it was beyond the scope of this team and may require a more experienced team to actually reach its potential. However, risk management is important for VV&T activities.

Employing Takt periods and spending much time early on making a proper WBS and risk analysis is key to project success.

VWM and Knowledge management are not two disjoint efforts, rather on the contrary they are very much linked; they both concern communication and knowledge sharing. The environment created in VWM make a perfect arena for sharing knowledge. Joining the two efforts may yield positive synergies to the learning culture of the organization. Stand-up meetings are all about learning from each other, and the VPB only comes into its proper being when it is employed as a repository for information.

Lean Systems Engineering combines the best of two worlds; the waste-reducing enablers of Lean with the system perspective of SE. It is a strong contribution to the field of SE that is struggling with finding a clear definition of itself. SE is indeed about enabling teams to work faster, better and more precise - which is also the focus of Lean. A combination of the two seems natural.

Implementing the enablers was a process that demanded a large effort, and it confirms why continuous improvements should happen as small steps and not in big leaps. As the author witnessed here, there will always be individuals among the users who are sceptical to the new procedures, and they will revolt if too many or too big changes are made in one go - and you do need to have the entire team with you to make the effort pay off. The enablers implemented in this project were not big, and did not affect the work habits of the team to a large extent, yet they were tough to grasp for some. The benefit became apparent after a while and the effort was well worth it. The team reached the level where they would be accessible for new incremental changes. The team achieved a culture for learning and a willingness to make an effort to capture the knowledge for next year's team, as well as a culture for enhanced and more efficient communication. Changing the culture is essential for success, it takes time, and it is where any implementer of Lean or SE must focus his or her efforts.

Appendix A

Generally valid sub-functions

Class Function	Element Function
Signal - generate, transmit, distribute, and receive signals used in passive or active sensing and in communications	Input signal Transmit signal Transduce signal Receive signal Process signal Output signal
Data - analyze, interpret, organize, query, and/or convert information into forms desired by the user or other systems	Input data Process data Control system Control processing Store data Output data
Material - provide system structural support or enclosure, or transform the shape, composition, or location of material substances	Support material Store material React material Form material Join material Control material
Energy - provide energy or propulsive power to the system	Generate thrust Generate torque Generate electricity Control temperature Control motion

Appendix B

Questionnaire for Lean assessment

This Appendix contains the questionnaire from Lean Systems Engineering assessments one and two, in January and May 2012. The questionnaire consists of 14 cases that the participants had to compare to the DNVFF2 project and rate the project accordingly.


NTNU – Trondheim
Norwegian University of
Science and Technology

DNV
FUELFIGHTER 2

Lean Product Development

Assessment of the implementation of Lean principles in the
DNV Fuel Fighter 2 project

Figure B.1

NTUU – Trondheim
Norwegian University of
Science and Technology

DNV FUELFIGHTER 2

Lean Product Development

Date: ____

Engineering discipline: ______________________________

a) To what degree (in %) does the team generally achieve project objectives?

Performance _______%

Lead-time _______%

c) Do customers or downstream functions experience problems with your products?

Almost always ☐

Often ☐

Rarely ☐

Never ☐

e) **How much of your work time do you spend on value-adding activities?**

Percentage of total work time[1]_______%

b) What percentage of your working hours is usually booked (or scheduled) in advance each week?

Less than 40% ☐

40% - 80% ☐

80% - 100% ☐

More than 100% ☐

f) How many projects are you currently formally assigned to?

Number of projects_______

1 In lean manufacturing, a specific operation adds value to the *part* if (a) the customer will pay for the operation, (b) the operation actually changes the part and (c) it is done right first time. In Product Development (PD) and other white collar areas, however, knowledge and information take the product's role; hence, value added may be defined as *producing the right information in the right place and make it available at the right time when needed, such that the use of this knowledge/information maximizes customer value and minimizes waste throughout the process*. This means that an activity must add value to the *body of knowledge of the company*(not only the individual) such that it either can be used directly to benefit a specific product, or the information is captured within the company to benefit future products with the customer (who 'pays'…). For example, internal meetings, administration, coordination, resource management, traveling, unnecessary testing, etc – i.e. every single activity that can be eliminated without loss of value in the eye of the customer – is considered 'pure waste'. On the other hand, 'necessary waste' are activities required by the customer that do not directly add value to the body of knowledge, e.g. documentation, reporting, etc. On the other hand, iterations in the design process as well as knowledge generation to reduce risk may be considered as value creating activities in PD. Moreover, calculated risk-taking could be a mechanism for adding value (as a basis for innovation) – unlike manufacturing where risk-taking is prohibited.

Figure B.2


Lean Product Development
DNV FUELFIGHTER 2
NTNU – Trondheim
Norwegian University of Science and Technology
Part one:
Customer focus

Figure B.3

NTUN – Trondheim
Norwegian University of Science and Technology

DNV FUELFIGHTER 2

Lean Product Development

1:1	Customer Focus: Role and Values				
	Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**
To what extent does your company work with the customer to understand current and future customer needs and wants?	• Company internal preferences and priorities have a much greater impact on product specifications and design choices than customer needs. • Customers/end-users does not have an integrated role in the PD process. The company has a short-term relationship with individual customers. The company only aims to barely meet predetermined specs and no efforts are made to establish directional product targets together with the customers. • Complaints or warranty claims are solved at contractual or legal level. These are solely considered as *quality* cases, and not *customer dissatisfaction* cases where the main issue is to understand *why* the customer is dissatisfied.	Interpolate between Level 1 and Level 3	• Customer requirements, needs and preferences are generally prioritised but management and design teams sometimes overlook customer preferences due to insufficient field research. • A few customers are viewed as long-term partners and these are to some extent integrated in the PD process. The bulk of customers do not participate actively in the PD process. • The most frequent and serious complaints and warranty claims are assigned to task groups, appointed at management level, resulting in a suboptimal learning process for the organization.	Interpolate between Level 3 and Level 5	• "Customer first" is a core value within the company. Every employee views end-user/customer satisfaction as the primary driver for future company success, and continuously works with adding true customer value. • The customer has a central, integrated role in the company's PD process to proactively create insight in true customer needs and wants. The company has a very good understanding of their products *from a user perspective* and uses various methods to continuously extend user knowledge. • Complaints and warranty claims are seen as excellent input for developing better and more user friendly products.
	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]
Background					

Figure B.4

NTUU – Trondheim
Norwegian University of Science and Technology

DNV FUELFIGHTER 2

Lean Product Development

1:2	Interface between Customer and Design Engineer									
	Level 1		Level 2		Level 3		Level 4		Level 5	
How do customer wants, needs, and requirements reach design engineers (project teams)?	• Design engineers get customer requirements "over the wall" through sales, marketing or other departments, who typically receive these from the purchasing organisation of the customer (or equivalent). • Design engineers very seldom participate in product field studies, causing lack of understanding of how products are used and misused in the field. Knowledge about users and customer needs is based on qualified guesses rather than facts. • Feedback from customers and end-users related to product performance, reliability, etc. rarely reaches design engineers so that they can improve the design.		Interpolate between Level 1 and Level 3		• Marketing and other functions co-operate with PD to understand customer needs. When a design specification is defined, it is considered final and a long process is required to make even minor changes. • Design engineers interact with end users on ad hoc basis. They have an understanding of the customer's product experiences, and work to integrate this understanding in future design considerations. • Feedback from customers and end-users reach engineers only when critical or product design related issues arise. Feedback loops are often filtered through other departments, causing loss of information.		Interpolate between Level 3 and Level 5		• Marketing and other departments co-operate with R&D throughout the PD process to understand customer needs. Designers are rather given a "design space" than a "design spec.", allowing design flexibility to heighten customer satisfaction. • Engineers routinely interact directly with end users. Designers are trained for, and encouraged to spend time studying how end users use the product and what problems end users experience. The company has designated a Chief Engineer role being customer's representative (voice) in the project. • Feedback loops from customers are fast and issues related to product design quickly reach product engineering.	
	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]
Background										

Figure B.5


Lean Product Development
DNV FUELFIGHTER 2
Part two:
Knowledge
NTNU – Trondheim
Norwegian University of
Science and Technology

Figure B.6

NTNU – Trondheim
Norwegian University of Science and Technology

DNV FUELFIGHTER 2

Lean Product Development

2:1	Learning and Knowledge Value Stream				
	Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**
How does the team rate the value of knowledge? Does the team consider its own collective knowledge as an asset?	• PD is not seen as a knowledge generating process but only as a sequence of tasks necessary to design a product that will meet customer specifications. Management promotes knowledge and technology in sales pitches and annual reports, but a knowledge-based approach is generally losing to short-term considerations. • No effective system for sharing and spreading new knowledge in the organization exists. • Knowledge is generated, saved and used by the individual. Key competences are dependent on a few individuals and the team would struggle if one of these individuals left the project.	Interpolate between Level 1 and Level 3	• The project manager has responsibility for project execution and learning. Management does encourage knowledge building, but this is often short-cut and down-prioritized due to other, more ‘urgent’ project prioritizations. • Knowledge is documented in a learning-archive (or equivalent). However, knowledge from the archive is seldom reused in by new teams/members, although the archive is accessible and user-friendly. • The team has a strategy for capturing, generalizing and reusing knowledge but many key competences are still with a few senior individuals.	Interpolate between Level 3 and Level 5	• PD is seen as a continuous learning process where the basis is knowledge generation and emerging products is a natural result. Management actively promotes knowledge based PD. • Any knowledge gained is routinely evaluated and shared team wide. Searched knowledge is always easy to find and ready for use. • Knowledge is captured and stored with the explicit purpose of being reused. The process of generating, collecting and communicating knowledge is standardized and all learning, including failures, are considered valuable.
	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]
Background					

Figure B.7

NTUU – Trondheim
Norwegian University of Science and Technology

DNV FUELFIGHTER 2

Lean Product Development

2:2	Knowledge Ownership and Management				
	Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**
How is knowledge ownership defined and organized in the company, including responsibilities for knowledge system, standard, creation and capture skills, and improvement?	• No knowledge owners exist within the team. • Knowledge creation is done ad-hoc. Knowledge is typically captured by individuals, using different practices. Documentation is stored inside the head of individuals or their PCs. • Functional managers (FMs) are resource providers and administrators, with very limited involvement in projects. Communication and coordination between project managers (PMs) and FMs are sporadic. The projects do not pull knowledge from existing standards, and do not transfer knowledge into future standards.	Interpolate between Level 1 and Level 3	• A specific person is designated with overall knowledge management or coordinator responsibility. • The project has a central R&D database where key reports and technical documentation are stored. However, the data are not optimized for reuse in terms of accessibility and structure. • FMs have an overall responsibility to maintain the database. They participate in integration/gate events, but do not have any formal responsibility in aligning design solutions with design and knowledge standards. (PM has that authority).	Interpolate between Level 3 and Level 5	• Each functional area has assigned a specific person with clear ownership and responsibility to knowledge. • Knowledge owners are trained to (a) create and capture knowledge, (b) validate, generalize and organize knowledge for reuse, and (c) build design standards. • A clear definition of roles exists between projects and knowledge owners: The former owns the product and is responsible for tech. trade-offs, its business success and customer value. The latter owns the knowledge and manages this on the behalf of the team.
	Current [ ] | Desired [ ]	Current [ ] | Desired [ ]	Current [ ] | Desired [ ]	Current [ ] | Desired [ ]	Current [ ] | Desired [ ]
Background					

Figure B.8

NTNU – Trondheim
Norwegian University of Science and Technology

FUELFIGHTER 2

Lean Product Development

2:3	Cross-Functional Knowledge Flow									
	Level 1		Level 2		Level 3		Level 4		Level 5	
How is knowledge transferred between functional departments? Functional departments may be individual team members, members of different engineering disciplines/responsibilities or the previous year's team.	• No systematic approach for sharing knowledge between departments exists. There are difficulties in getting required information. • It is unclear how and where to search for knowledge owned by other departments. (Tracking down expertise and verifying data is an exhausting task.) • Departments have strong ownership of their experience, and there is a resistance in sharing knowledge with other departments. Mistrust and scepticism are commonly present between different functional departments.		Interpolate between Level 1 and Level 3		• A system for sharing knowledge between departments is available but is only sporadically used. More indirect channels are often used to get the information needed. • Frequently used information is easily accessible. Finding less frequently used information, however, is dependent on knowing whom to ask. • Functional information, data and knowledge is viewed as department assets. Access to information and data is open, but underlying data and deeper knowledge are harder to access.		Interpolate between Level 3 and Level 5		• A structured system to continuously share knowledge between department and business areas is used throughout the organization. All necessary information needed to solve problems is easily accessible (regardless of who owns it). • Communication channels are clear and every employee knows exactly where to get needed information. Engineers spend time using and creating knowledge instead of searching for data. • Knowledge is viewed as a common asset that benefits the whole organisation. Departments are closely integrated and everybody sees the mutual benefit of sharing knowledge and experience.	
	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]
Background										

Figure B.9

NTNU – Trondheim
Norwegian University of Science and Technology

DNV FUELFIGHTER 2

Lean Product Development

2:4	Set-Based Concurrent Engineering				
	Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**
To what extent is Set-Based Concurrent Engineering (or other equivalent methods) generally accepted and adopted as the company's main philosophy for designing products?	• One design concept is chosen early on, and the continued development process is focusing all efforts on making design choices and resolving issues associated with the chosen main concept. • Decisions made in early PD phases have a tendency to be less fact based resulting in unplanned, costly loop backs. Unplanned loop backs are seen as a part of the PD process. • Limited reusable knowledge is gained from running individual projects. Individuals transfer experience to succeeding projects in an ad-hoc manner.	Interpolate between Level 1 and Level 3	• A few alternatives are explored in early project stages. Standard tests and tools are used to identify the *strongest* design as early as possible. Further design efforts are focused on solving issues associated with the chosen main concept. • The final design concept is chosen by using decision matrixes. The result is sometimes coloured by individual preferences and traditional solutions. • Company strives to capture knowledge from projects and products. Learning archives are somewhat functional but most knowledge stays with the project members.	Interpolate between Level 3 and Level 5	• Several design concepts are thoroughly evaluated through modelling and testing. Final design choice is delayed as far as the overall project time line allows; 'learning-first' principle. • A concept is chosen by "eliminating-the- weak" instead of "promoting-the-strongest" practice. Decisions made in PD are always based on data and facts. As a safe backup, system borders are set to allow a proven alternative among the concepts. • Knowledge is captured from every single design concept, including the eliminated ones. The knowledge is captured, stored and generalized to enable reuse in future projects.
	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]
Background					

Figure B.10

NTNU – Trondheim
Norwegian University of
Science and Technology

DNV FUELFIGHTER 2

Lean Product Development

Part three:

Stabilize

Figure B.11

NTUN – Trondheim
Norwegian University of
Science and Technology

DNV FUELFIGHTER 2

Lean Product Development

3:1	Resource Planning & Management									
	Level 1		Level 2		Level 3		Level 4		Level 5	
To what extent do functional departments get the resources they need *when* needed?	• No company-wide resource planning exist so neither management nor team members have an overview of what resources are needed for long and short term. • Problem solving is delayed as far as possible and prioritized according to urgency. • The basic philosophy is to allocate resources based on a top-down approach.		Interpolate between Level 1 and Level 3		• A resource plan exists, but no process for updating the plan based on actual needs exists. This regularly leads to large diversions between the plan and the reality. • Planning and resource issues are handled when they occur. However, prioritizations are commonly based on urgency, often at the expense of more long term projects. • The basic philosophy is to allocate resources according to a predetermined plan and use formal follow ups to control the execution. However, the planning and resource prioritizations are task-based rather than challenge-based (risks, complexity).		Interpolate between Level 3 and Level 5		• Resource planning is regularly conducted between functional areas and projects. • Problems are surfaced and handled when they occur. • The basic philosophy is to delegate planning responsibility to the lowest possible level, in order to maximize the involvement, responsibility, and commitment of individuals.	
	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]
Background										

Figure B.12

3:2	Communication and Information Flow between Organizational Levels				
	Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**
What is the communication practice in the team, and how does information flow between organizational levels?	• Most upstream communication is to get approvals, notify changes and provide periodical reporting, typically using heavy and administrative type reports. Communication form is written, following strictly organizational structure. • Downstream communication is usually in the form of sign offs, approvals, orders and 'management news' type information, typically written, bureaucratic and formal. • The organization is relatively top-heavy and hierarchic. 'Walking management' is not practiced.	Interpolate between Level 1 and Level 3	• Most upstream communication is brief and data driven. Formal reports are required in connection with business planning, gate reviews, end-year etc. • Downstream communication is in various formats, depending on message and sender. Formal written information dominates from higher management and a more informal and direct process dominates at lower management levels. • The organization is relatively flat at the bottom with a high peak. 'Walking management' is practiced lower in the organization but top management only practices walking management on special occasions	Interpolate between Level 3 and Level 5	• Upstream communication is direct, brief and data driven. Managers at different levels take a coaching role, delegating authority and responsibility whenever possible and seek direct involvement only when assistance is asked for. • Downstream communication is usually in the form of business objectives and targets that, through an iterative process of active involvement, is broken down into actions by middle management and individuals. • The organization is flat with short reporting lines. Managers' practice walking management ('going to the source') and events where different organizational levels exchange information.
	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]
Background					

Figure B.13

3:3	Key Supplier Role in Product Development									
	Level 1		Level 2		Level 3		Level 4		Level 5	
What is the role of key suppliers and how are they viewed and treated in your organization?	• Suppliers are seen as vendors who are told what to do and not considered as an integrated part of the product development team. • The supplier base is large and suppliers are primarily selected based on price. No categorization of suppliers exists. • No shared benefit agreement exists and suppliers are seldom considered as long-term partners.		Interpolate between Level 1 and Level 3		• A few suppliers are seen as partners. Technologies and experiences are shared, but their access to specific technical data is limited. • The supplier base is large with a few preferred suppliers. The lowest price supplier isn't always chosen. • Some risk and benefit sharing exist with the company's closest suppliers. Long-term partnerships exist in some strategic areas.		Interpolate between Level 3 and Level 5		• Suppliers are considered as cooperative partners who proactively present trade-off curves that show they are making progress in technology, quality and cost. • A few key suppliers are selected based on competence, capability and reliability for long-term cooperation. There is a clear categorization of suppliers (key suppliers, commodity suppliers etc). • Development is mainly done under the philosophy of sharing risk and benefit with selected key suppliers.	
	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]	Current [ ]	Desired [ ]
Background										

Figure B.14

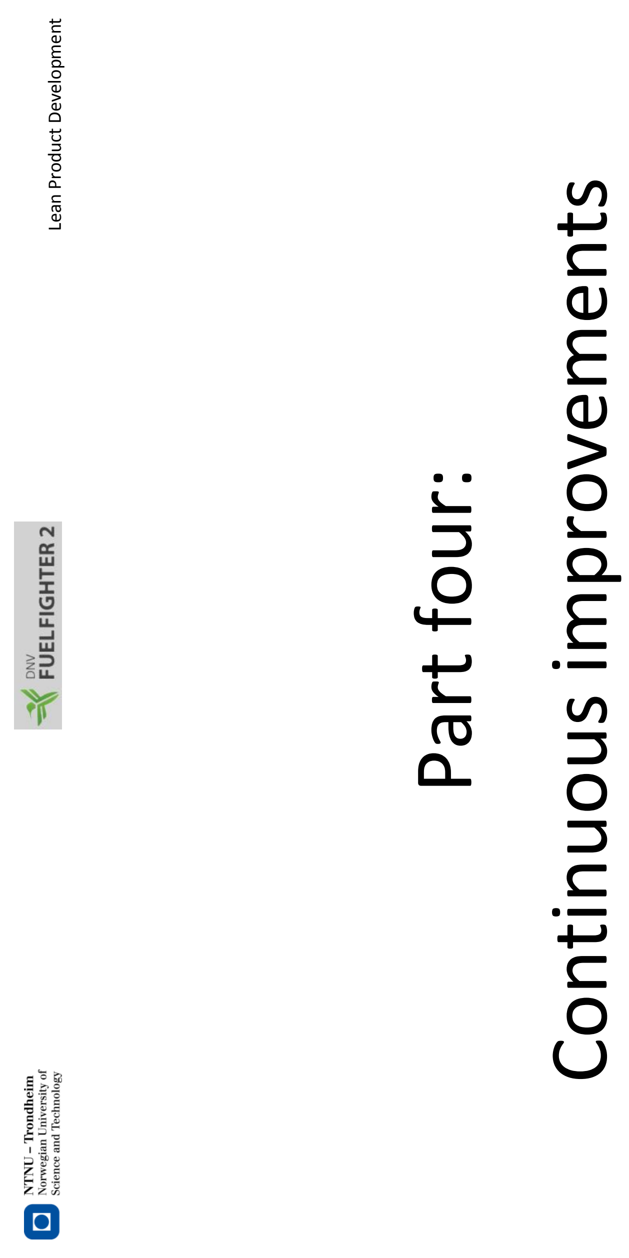

Lean Product Development
DNV FUELFIGHTER 2
Part four:
Continuous improvements
NTNU – Trondheim
Norwegian University of Science and Technology

Figure B.15

4:1	Continuous Improvements in Product Development				
	Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**
Is continuous improvement, waste elimination and pursue for perfection in product development an integrated part of the team's philosophy?	• Improvements of the product development process is done sporadically, as isolated "improvement packages" • There is no clear responsibility for gradually improving process and operations. Affected functions and individuals are seldom directly involved in improvement efforts. • No common definition of "waste" in PD exists within the organization and it is unclear to the design engineers what operations add value and which ones do not.	Interpolate between Level 1 and Level 3	• Continuous improvement of the PD process is encouraged and stated to be a part of the project strategy. However, no resources are earmarked for improvement work, causing lack of momentum. • The responsibility for continuous improvement of the PD process is not well defined. Improvement suggestions are frequently generated but initiatives are more seldom followed through. • An understanding of value adding work and waste in PD is limited among employees. Most design engineers find the definition of "value adding" complex and there are differences in what people consider as value adding work vs. waste.	Interpolate between Level 3 and Level 5	• A systematic approach to continuously improve activities and process is fully implemented and used at all levels in product development. A few work hours every week are earmarked for improvement work. • All organizational levels continuously and systematic work with improving PD operations and processes. Each function, project or group is given freedom and responsibility to improve their own work. • The team has a clear and communicated definition of value added work and waste in product development. A systematic process for identifying and eliminating waste in PD is fully integrated and used throughout the company.
	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]
Background					

Figure B.16

4:2	Productivity Measurements in Product Development				
	Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**
How is your team using metrics and productivity measures in PD to achieve the team's overall goals?	• Functions and projects are measured on short-term or indirect metrics (cost, resources, budget, timing, etc.), where actual performance is less focused. This causes faulty prioritizations and sub optimizing. • Functions and project team (members) have no guideline for prioritizing between lead-time, product performance, development cost or product cost. • Company goals are mainly existent in vision statements, annual reports business plan documents. Functions, projects and individuals see no connection between their work and business goals and visions.	Interpolate between Level 1 and Level 3	• Functional performance is regularly measured. However, the metrics are of varying relevance and even further lack of relevance is experienced when key performance indicators (KPIs) are broken down to lower level activities. • Management has established directional guidelines for how to trade-off lead-time, product performance, development cost, and product cost during project execution. • Company vision and goals are not fully linked to functional and individual goals and tasks.	Interpolate between Level 3 and Level 5	• The company mainly uses metrics to track and follow activities. However, principles instead of metrics are used to manage performance at different company levels. • Trade-offs between lead-time, product performance, development cost and product cost are made available to functions and project teams, who fully understands the implications of the various trade-offs from a company and customer perspective. • Company goals are communicated, visualized and understood across the entire organization. Business goals are broken down into functional goals, project goals and individual tasks in a systematic way.
	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]
Background					

Figure B.17

Lean Product Development
DNV FUELFIGHTER 2
NTNU – Trondheim
Norwegian University of Science and Technology
Part five:
Culture

Figure B.18

NTNU – Trondheim
Norwegian University of
Science and Technology

DNV FUELFIGHTER 2

Lean Product Development

5:1	Trust, Respect, and Responsibility				
	Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**
To what extent are Trust, Respect, and Responsibility core values in your organization?	• Opinions and views are respected and valued differently depending on the messenger (position, education, department, etc). • Decision-making is usually elevated up in the organization. Top-down, task-based management style leads to slow and, frequently, faulty decisions. A culture for controlling rather than coaching reduces trust between levels and functions. • Unclear responsibilities create lack of discipline and accountability and a one-does-as-one-pleases type of culture.	Interpolate between Level 1 and Level 3	• Every opinion is respected and considered but opinions from higher organizational levels tend to rule. • The company seeks to delegate decisions to lower organizational levels. • Decisions and agreements are generally respected and conformed to. Some individuals or departments tend to get away with disregarding agreements.	Interpolate between Level 3 and Level 5	• Every opinion is *equally* respected, valued, and considered regardless of whom delivers it. • The team seeks to delegate responsibilities to the level closest to the problem. Team objectives are achieved by a coaching management style and strong ownership from individuals. Management creates trust by taking responsibility if things go wrong. Control and inspections focus on actual performance. • Decisions and agreements result from a *process* with involvement from different levels. Decisions and agreements are respected and conformed to.
	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]
Background					

Figure B.19

5:2	Fact Based Decision Making				
	Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**
To what extent is there a tradition to make all decisions based on facts in the organization - from top management level down to the factory floor?	• Management's primary role is to take decisions. Decisions are made and obeyed regardless of whether or not they are supported by facts. • The prevalent view in the organization is that fact-based decision is a luxury one can't afford (from a money and timing perspective). • Decisions are often elevated up in the organization even when the decision could be taken at a lower level. The elevated decisions are often less informed, as the decision makers are longer from the problem or issue. Management style is authoritarian.	Interpolate between Level 1 and Level 3	• Management's primary role is to help resolve problems, keep track of progress (controlling), and to ensure that decisions are supported by facts. However, less fact-based decisions do frequently pass. • The company tries to make fact-based decisions but is often forced to cut corners due to time and resource constraints. • Everyday decisions are usually taken at low organizational levels, whereas major issues are elevated. However, information is lost on its way up, as the distance to the source increases.	Interpolate between Level 3 and Level 5	• Management's primary role is being a coach, motivator and functional leader. Decisions at all levels are always supported by data. • A risk taking approach to problem solving, i.e. cowboy mentality is not accepted. Decisions are based on facts; when facts or data are missing, risks are elevated and minimized. • Decisions are made where the best data is available, usually at the lowest possible organizational level. Management style is including, delegating and built on trust.
	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]
Background					

Figure B.20

NTNU – Trondheim
Norwegian University of
Science and Technology

DNV FUELFIGHTER 2

Lean Product Development

5:3	Simple and Visual Communication				
	Level 1	**Level 2**	**Level 3**	**Level 4**	**Level 5**
To what extent is the use of simple and visual communication anchored in the team's culture?	• Bulky reports are the default way of communicating and documenting within the team. Project briefs and follow-ups are done through reports, creating information overflow and associated waste • Project goals, plans, and schedules are stored in computer systems and are seldom communicated, or reviewed, by the project team as a whole. • Simple visual tools such as sketches, models, trade-off curves, A3´s etc. are seldom used in problem solving.	Interpolate between Level 1 and Level 3	• Simple, visual communication is regularly used in the organization. However, a major part of the communication is still done in terms of heavy reports etc. • To what extent visualization is used in a project environment varies between projects and is dependent on the project leader and the preferences of the team. • The team does not consider visual communication as a strategic tool for learning, standardizing, and generalizing information for reuse.	Interpolate between Level 3 and Level 5	• Project briefs and follow-ups are done continuously in a simple visual manner. As much information as possible is visualized and tools like A3-reports, trade-off curves etc. are widely used and recognized. • All information essential to the project (goals, schedules, current workload etc.) is communicated visually and kept updated throughout the project. A project room or board is used to keep updated information available to anyone inside or outside the project team. • Generalizing and visualizing a problem by sketches or product models is a natural part of problem solving and knowledge capture. All engineers have some training in simple sketching and modelling.
	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]	Current [ ] Desired [ ]
Background					

Figure B.21

Appendix C

A collection of K-Briefs

This appendix contains a collection of K-brief made for the DNVFF2 project. Some are made by the author, others are made in collaboration with the team members responsible for the sub-system in question.

K-Brief DNV FF2's performance at Shell Eco-marathon 2012

DNV FUEL FIGHTER 2

Shell Eco-marathon 2012

Background

The competition was held at Ahoy Arena in Rotterdam, Netherlands May 16-19. DNVFF2 drove test runs on May 16 and 17, then competed on May 18 and 19.

Every competitor had 4 attempts. For every attempt the vehicles had to complete 10 laps in less than 39 minutes. For every lap the vehicles had to make a full stop at the start/finish line (referred to as 'launch').

Results

DNV FF2 finished one successful test run and two successful attempts.

The vehicle finished 5th overall in the UrbanConcept category with 1581 km/L, making it the second best newcomer. The vehicle is lighter than most of its competitors, yet has a too weak drive train for efficient launching. However, it has low rolling friction and reaches a higher top speed than its competitors.

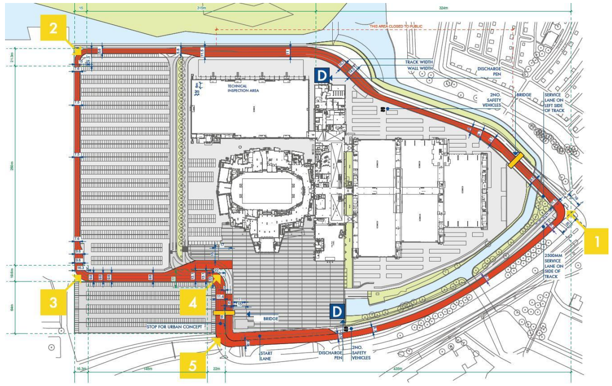
The wind blew the door open on the first test run and again on the first attempt. It had to be secured by double-sided tape. The Kevlar cord in the steering snapped on the last attempt, and the car crashed into the left-hand side wall. The headlights tend to come loose.

Conclusion

The door needs to be properly secured against wind. The steering and suspension needs to be inspected and greased prior to every attempt. Steering cord of steel?

The vehicle spends most of the energy during launch. It is mechanically superior, yet needs a better drive train in order to drive efficiently enough to compete with the best competitors. An extra motor just for acceleration is recommended, with a transmission to increase the torque delivered. Mechanical energy storage should be investigated.

The reader should read the K-briefs about the Steering, Suspension and Drive train.



Map of track. Numbers indicate turns and driving direction (counter-clockwise).

Official length: 1630m, Actual length: 1602m (by calculation). Also see map here.

Attempt 1

Result: DNF

Comments:

The door blew open in the first lap. This also happened in the first test run, and fixed with double-sided tape – a solution that works fine. However, the surface was wet from light rain when the tape was fixed that morning, causing the tape to loosen.

The lesson is to make sure the surface is dry and without traces of dust before putting on the tape. A more robust locking mechanism (new door?) is recommended for the future, as the tape solution may be a violation of Shell requirements. See K-brief about Body.

NB!

Shell allowed a fifth attempt on the last day, yet also decided to close the track 75 minutes earlier than scheduled. Lesson is, make sure Shell knows you are planning to make another attempt late or they will close the circuit. Also, know that a fifth attempt may be granted if time allows it.

Attempt 2

Result: 136 (or 156) km/kWh, Time: 37:50 (10 laps)

Comments:

Due to poor planning, no one counted the laps and the vehicle finished 11 instead of 10 laps. The official result is 136. Estimations say the real result was 156. See also K-brief on how to organize a team during races. The vehicle had to make many over takings every lap during the attempt.

Attempt 3

Result: 163 km/kWh (=1581 km/L), Time: 38:36

Comments:

The team decreased the air gap in the motor before this attempt. The benefit is questionable; if the estimate from attempt 2 of 156 is correct, the improvement is just 7 km/kWh. The vehicle did not have to perform any over takings during the attempt. Also the driving style was different (see K-brief on driving styles). Impossible to say what contributed the most to the improved result.

Results of UrbanConcept (UC): Battery Electric 2012

Rank	Team	Result [km/kWh]	Successful attempts
1	electricar solution	263	4
2	TUS team	189	3
3	Team proTRon	185	3
4	Team Ruppin Jet	180	3 or 4
5	DNV Fuel Fighter 2	163	2
6	mecc-e-	161	?
7	SAITEM	145	1
8	Team ULg	102	?
9	mobileo	93	?
10	CA Marcq	76	?
11	CARBON ZERO	75	?
12	Team Eco Racing Austria	75	?
13	SiMR Team	45	?
14	EcoEmotion	39	?
15	BeeMobile	34	?

22 teams entered, 20 teams passed technical inspection, and 15 achieved a valid result. See 2012 results here, and 2011 results here.

Battery electric (UC), Gas to liquid (P) and Ethanol (UC) were the only classes to see better results than in 2011. The other classes – in both UC and P – saw a reduction in their best results by an average of 10%. See also K-briefs on the main competitors. This may indicate that the Battery Electric class is still evolving; the class was first introduced in 2011. The rough track may have caused the drop in top results in the other classes.

Attempt 4

Result: DNF

Comments:

After 2 laps the driver complaints about stiff steering. After 4 laps the steering cord snaps, and the vehicle hits the soft wall. Only damage to foil.

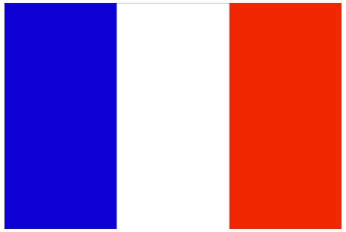
- Increased sliding friction for the draglink due to grease saturated with dust from the track.
- The cord was inspected both prior to and after the attempt, and bore no signs of wear.
- The 'Z-rod' in the suspension experiences a bending moment every time the wheel does not touch the ground (hard cornering). This 'Z-rod' had to be replaced after the 12 laps of testing on May 17 because it had been bent. The force acting on this part may also affect the steering.

Figure C.1

K-Brief

Team electrical solution

Team electricar solution



Nationality: France

Institution: Lycee des metiers Charles Privat (high school)

Participation: 2 times in Battery Electric

Observations

This team came second in the Battery Electric class in SEM 2011 and finished 1st in SEM 2012. Their result was 12% higher than in 2011, which does not match the trend observed in every other class where a reduction of 10% in average was observed.

This team represents a French high school, and appears to be built by teachers/professors at this school, not students. The students participating are obviously only drivers, and skilled at that. We believe they spend much time practicing driving skills based on our observations of their driving style.

Aerodynamically, the vehicle is perfect. It narrows at the back to get the droplet shape, and the flying saucer look also gives it an edge what aerodynamics are concerned.

The team did not even bring any tools to the paddock, and was absent the whole time – except for doing some interviews. The vehicle displays no sponsors, and the fact that the body covers the entire vehicle made it impossible for us to learn anything about the vehicle. It appears heavy, 100+ kg.

Team data

Year	Weight [kg]	Motors	Monocoque	Team size	Valid attempts	Results [km/kWh]	Rank
2011	?	?	No	?	?	230	2nd
2012	? 100+	Prob 2	No	3 teachers, 2 student drivers	4	263	1st

We know little or nothing about this team. We have not been able to see any mechanical energy storage, like a spring or a flywheel. The vehicle appears quite heavy.

Results from SEM 2012

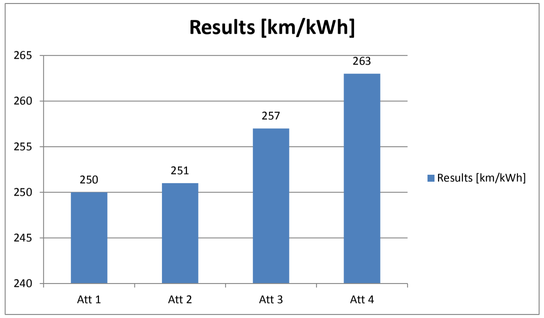


Figure C.2

K-Brief

DNV FF2's driving styles at Shell Eco-marathon 2012

DNV FUEL FIGHTER 2

Driving styles employed at SEM 2012

The reader should consult the K-brief about the team's performance at Shell Eco-marathon 2012 before reading this K-brief.

Driving style and strategy may have profound effects on results. The team observed the driving styles of competitors and saw that practice was important for them. Especially cornering is important – and difficult. Coasting (turning off the motor) is important for conserving energy, yet it demands strong knowledge of both the track, the vehicle and wind conditions.

Green lines indicate that the motor was running. Section 1, 2 and 3 refer to the driving programs accessible via the buttons on the steering wheel.

Pink lines indicate sections where the driver chose to decelerate.

Blue lines are sections where the driver chose to coast.

Control system

The control sections were programmed with these values:

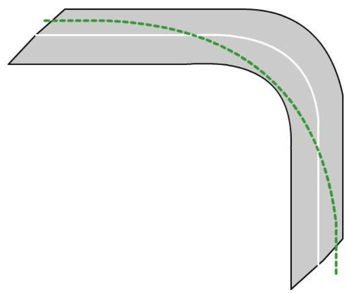
2nd attempt

Section	Reference speed [km/h]
1	35
2	35
3	17
4	?

3rd attempt

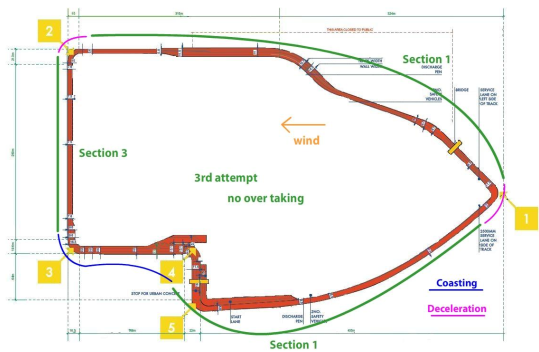
Section	Reference speed [km/h]
1	30
2	35
3	28
4	32

Cornering



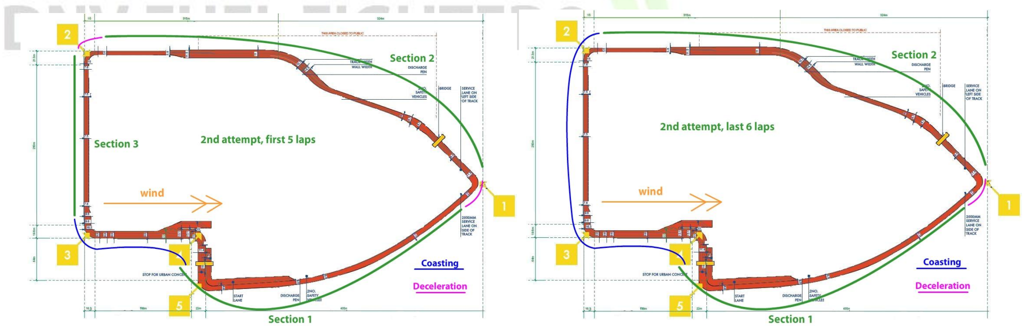
Curve showing proper cornering (green dashed line). Driver must stay far out, then corner in a circular curve, as close to the inner wall as possible, then maintain the circle till the car has reached the far wall and then straighten out. Acceleration should start mid-corner.

3rd attempt, 163 km/kWh



The winds were weaker this day, and blowing in the opposite direction. The driver was not able to coast as much as in attemp 2, but did not have to make any over takings. The motor was modified for increased efficiency before this attempt. The control system was programmed for lower top speed.

2nd attempt, 136 or 156 km/kWh



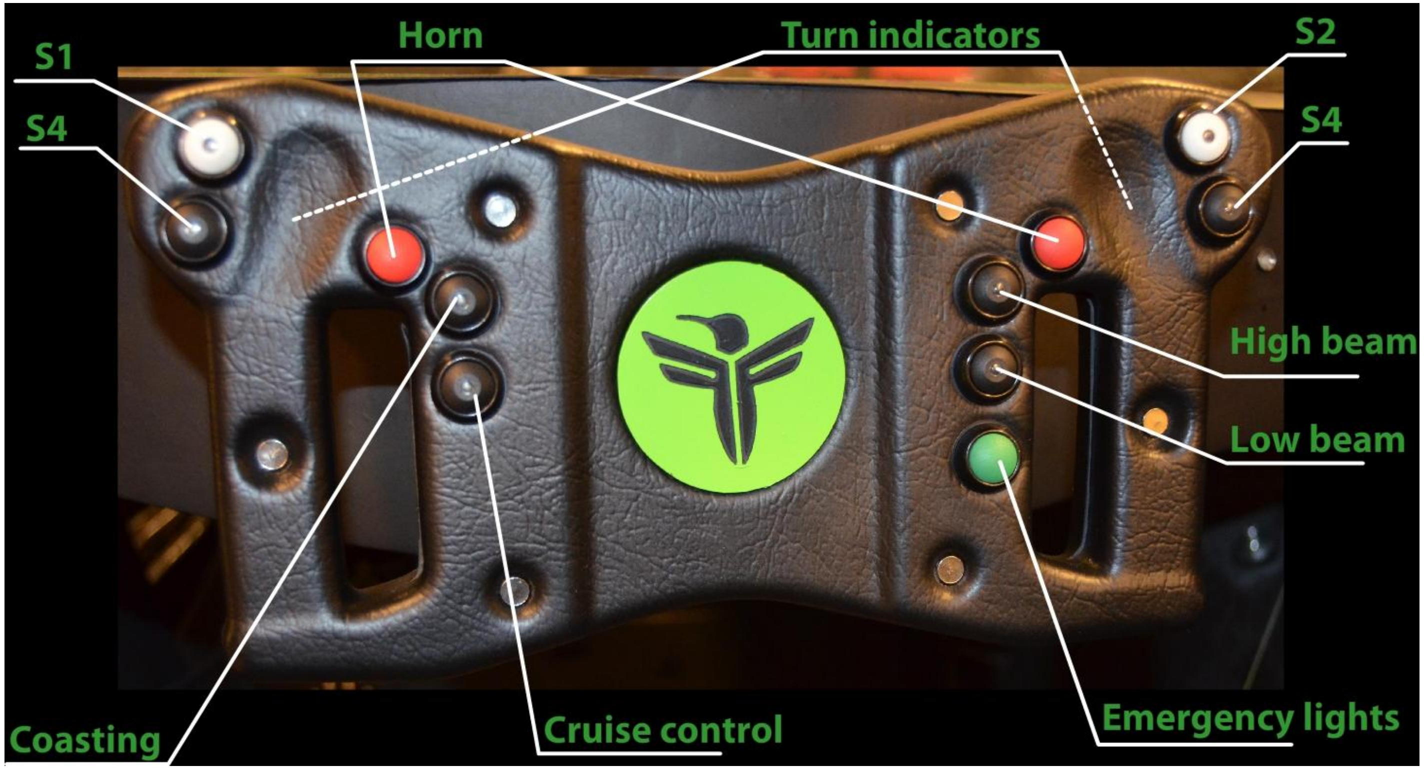
The figures show the driving style of the 2nd attempt. The wind was very strong this day, allowing the car to coast very much. However, this did not occur to the driver till half-way through the attempt. In the last lap, the driver shut down the motor by hitting the emergency button to save as much energy as possible coming into the pit lane. The driver was able to corner at 36 km/h.

The driver had to make many over takings in this attempt.

Figure C.3

K-Brief

Steering wheel as of June 2012



This is how the steering wheel is programmed, as of June 2012. Dashed white lines indicate buttons on the back of the steering wheel (triggers).

Figure C.4

K-Brief

Stand-up meetings and Visual Project Board

DNV FUEL FIGHTER 2

Stand-up meetings and Visual Project Board

Background

Stand-up meetings are short meetings where a team gathers to share information on what and how everyone is doing. A Visual Project Board (VPB) is a tool used during these meetings for visualizing tasks, project status and important messages. The results in this text are based on Stand-ups performed in the DNVFF2 2012.

DNV FF2 held their meetings from January till May by the whiteboard downstairs in the workshop (people were spending most of their time downstairs).

Results

- Three times a week is sufficient
- Maximum number of active participants is 7 or 8
- 15 minutes is stretching it. 12 is enough.
- Location matters during production periods. The meetings must be held in or close to the workshop.
- To make the meetings short and interesting, the speaker should address these three questions: "What have I done since last time?", "What will I do till next time?" and "What is preventing me from progressing?"
- The meetings are best held in the morning, during production periods due to the accessibility of machines.
- Stand-up meetings are very important during the competition, due to the ever-changing environment that such a competition is.
- VPBs are great tools for managing team work over a short period, and for communicating important messages between team members.

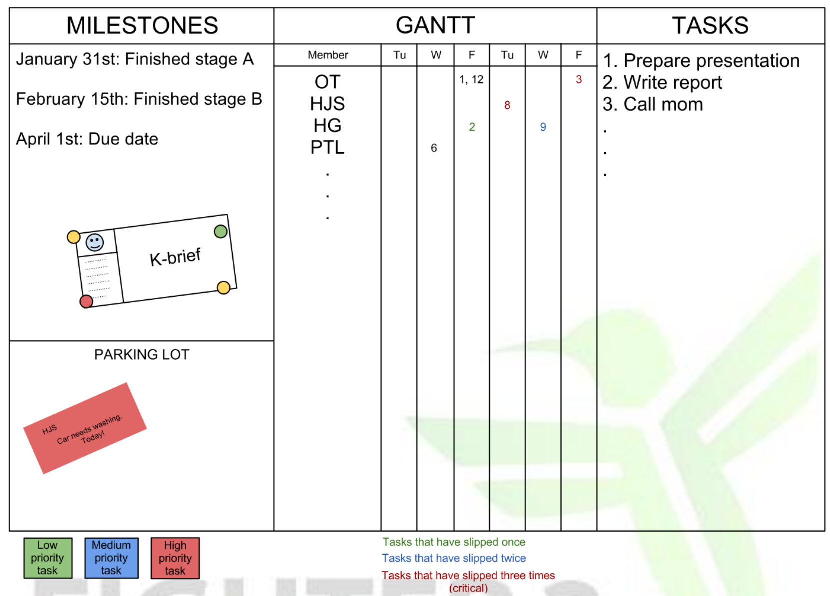
Conclusion

Stand-ups are important for any project. It is important to begin having them early in the process, as it takes the users some time to understand them and work them into their daily routines.

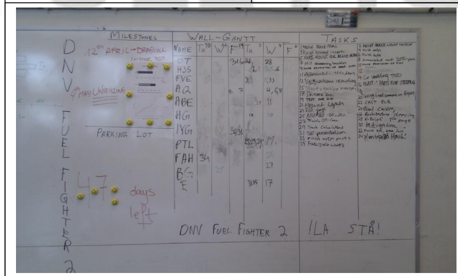
Stand-ups are great for managing a team in environments that are prone to change, e.g. the Shell-Eco marathon competition.

Motivation

- Resynchronization of efforts
- A means for managing an ever-changing environment
- Communication
- Team cohesion
- Induce a sence of urgency into the participants



Visual project board tailored for the DNV FF2. The Gantt diagram shows the dates for two weeks. The tasks list shows the tasks that will be performed over that period. A task's number is put on the Gantt on the day it will be finished by. Colours indicate whether the task has slipped (been postponed) or not. The left-hand side of the VPB is for messaging. Anyone is encouraged to put up relevant information here.



Pictures of the VPB, and the team having a Stand-up meeting some time in March 2012.

Figure C.5

K-Brief

The Timeline as a tool for VWM

DNV FUEL FIGHTER 2

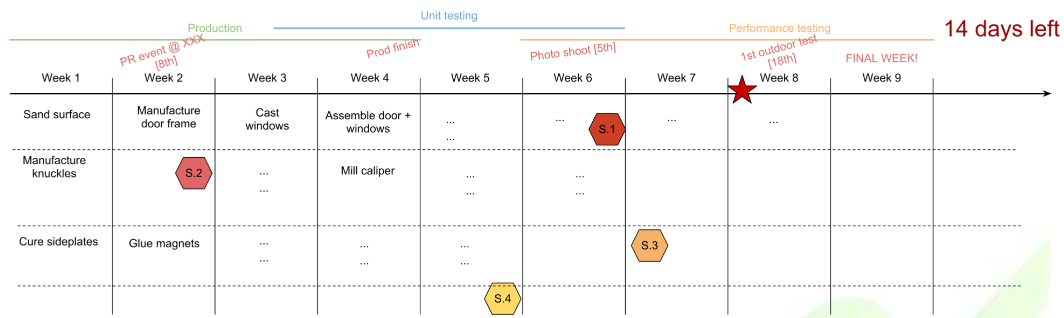
The Timeline as a tool for VWM*

The Timeline is a tool for showing project progress on a system level. The idea is to show the project period as a time line on a white board, divided into periods (preferably weeks). The major tasks that will be performed in each sub-system are listed on a weekly basis. The piece symbolizing the sub-system is moved closer to the finishing line (deadline) relative to how many of the tasks are finished.

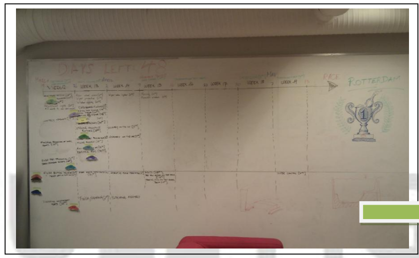
The Timeline was updated every Monday. It is also the view outwards to anyone visiting or passing the office about the current status of the project.

Just showing a total percentage of tasks done says little about the actual progress of the project. This tool shows how the project is doing in its sub-systems. Sub-systems that are lagging behind may be paid extra attention or resources.

*Visual Workflow Management



Layout for Timeline. Every sub-system has its own row. Columns are weeks. Tasks are listed in the week they shall be performed/finished. Every sub-system is represented by a piece (here: hexagonal figures) that move closer to the finishing line as they evolve in the workshop/CAD program. The red star indicates the current date.



The progress of the DNV FF2 from week 13 to week 19 of 2012.

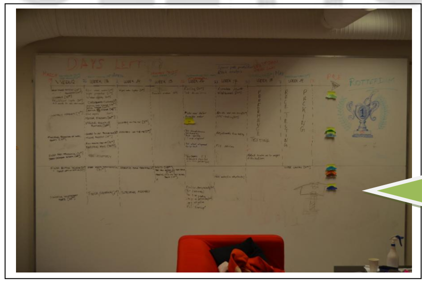


Figure C.6

K-Brief

The Wall-Architecture as a tool for VWM

The Wall-Architecture as a tool for VWM*

The Wall-Architecture is a way of showing project progress on a component-level, as well as being a tool for communication.

The tool utilizes the system architecture produced for the system model of DNV FF2. It is very useful for production, using coloured sticky notes to indicate whether a component is finished, waiting for something, in production or critically late.

It also functions as a tool for communication when more than one person is working on the same sub-system. When a component is finished or started on, the team member indicates this on the wall. The other team members pick components to work on. No other communication is needed. This is helpful e.g. when people have unsynchronized work shifts.

*Visual Workflow Management

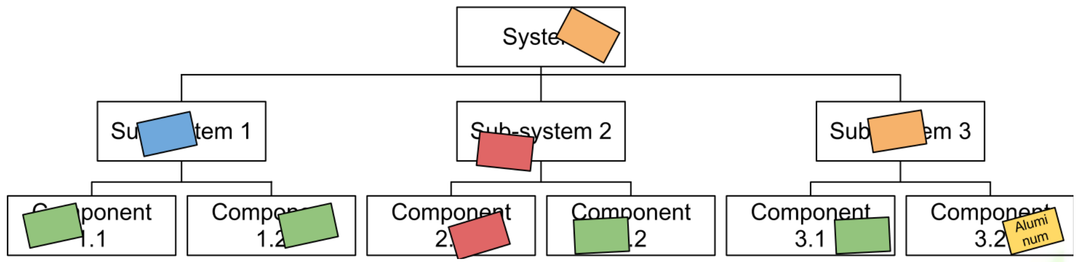
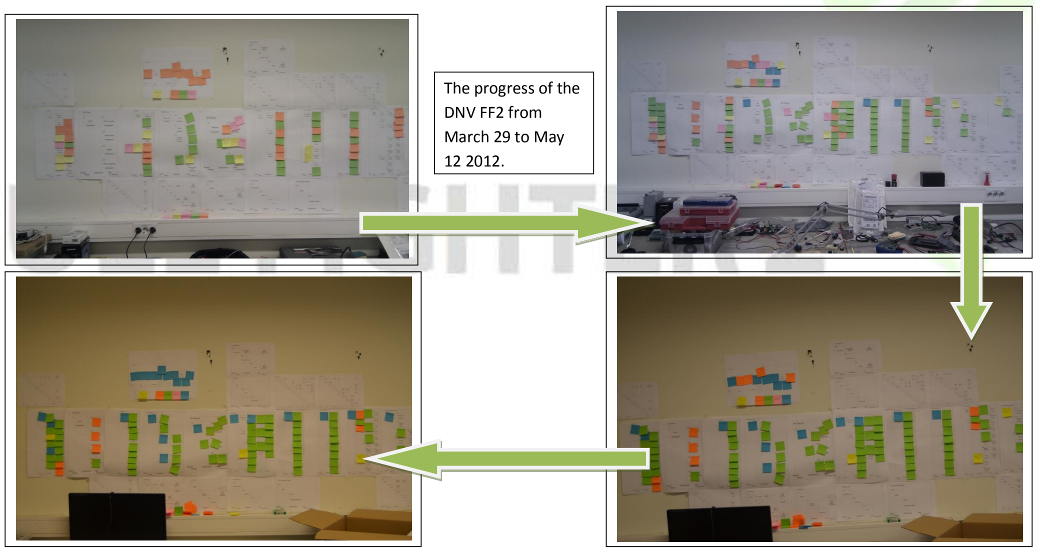


Illustration of the principle behind the Wall-Architecture. The system is currently 'under construction' told by the orange sticky note on the root node. Sub-system 1 is finished and assembled, sub-system 2 is critical due to an issue with component 2.1, and sub-system 3 is under construction, yet waiting for more material for component 3.2.



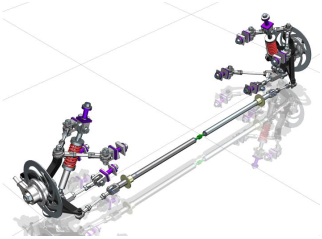
The progress of the DNV FF2 from March 29 to May 12 2012.

Figure C.7

S.5 Front suspension

Aksel Qviller/2012

DNV FUEL FIGHTER 2



Component		Material	Dimensions [mm]	Pro-cured	Satis-faction [%]
S.7.2.1	Shock absorbers	Alu Steel POM	10x15x25	P	
S.5	Front linkages	Alu	1x3x6	NPD	80%
S.5	Axle (front)	Alu	12x89x78	NPD	90%
S.5	Hub (front)	Alu		NPD	90%
S.5	Steering knuckles	CF		NPD	80%
S.5	Bearings	Alu		P	90%

NPD = new product design, R = reused part, P = purchased part

Supplier data

- Alumec 89 was purchased from Uddeholm
  The price is about 206 NOK/kg and about 150 NOK for cutting round bolts to 300 NOK to sawing larger plates
- Aluminium 6082 T6, 6m long rods of Ø10/Ø12mm were purchased from Smith Stål, each rod cost about 60 NOK
- Three hubs were machined by NOMEK AS, who also supplied the alumec, for 6700 NOK
- A fourth hub was self-made in the workshop with an unknown aluminium alloy
- The linkages were machined from Al 6082 T6 alloy
- The four knuckles were machined on the Makino with manual water spray, from carbon fiber slabs made by HPC for 1500 NOK each
- Brake mount standards from Shimano (by phone request)

Trade-off analysis/important decisions

Most important was the decision to make a fully damped suspension.

The design with stiff elements and rose joints was chosen because of its predictable behaviour, mostly easy machining and replacement of any part that could break.

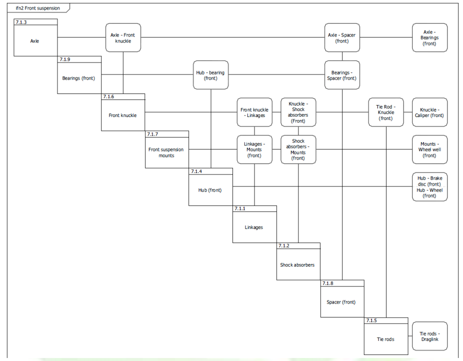
Other primary designs considered were compliant spring mechanisms, such as flexing composite beams (for instance a cross-country ski through the body), or shorter plastic beam harnessess mounted directly to the wheel well wall. The disadvantage with these designs were lack of easy adjustment. Especially, the ability to adjust height was important, and doing that with the flexing beam designs would mean longer bolts on top and lower swivel joints, and thus higher bending moment and risk of failure. Also, a broken element means almost the whole solution is damaged and not so easily repaired.

The short-beam flexing suspension concept as shown is still worth looking into, and could help save weight.

One major decision was to use carbon fiber, as urged by HPC.

The hubs used a five -bolt star pattern and an aiding centering chamfer to allow adjustment of the wheel alignments relative to the hub. Based on previous experience with a single center bolt, the wheels would often wobble without any chance of adjustment. The combination of five bolts with flanges distribute the pressure on the rim's center plate. The number of bolt holes was doubled for redundancy.

Interfaces



Interface details

The suspension has three major external interfaces:

- The Mounts are attached to the Wheel Well (Monocoque) with screws.
- Two types of knuckles:
  - Right-hand side: Normal knuckle interfacing with Brake Caliper
  - Left-hand side: Motor knuckle interfacing with Brake Caliper on Motor Wheel. This knuckle is heavier and stronger than the normal knuckle.
- Only one type of hubs (as rear right):
  - Normal hub interfacing with Brake disc and Centerplate of normal wheel

Figure C.8

S.5 Front suspension

Aksel Qviller/2012

DNV FUEL FIGHTER 2

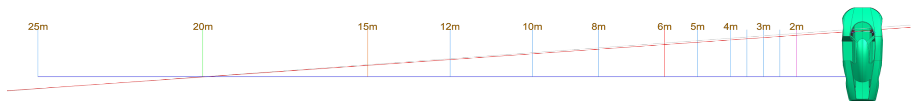
Analyses

Background

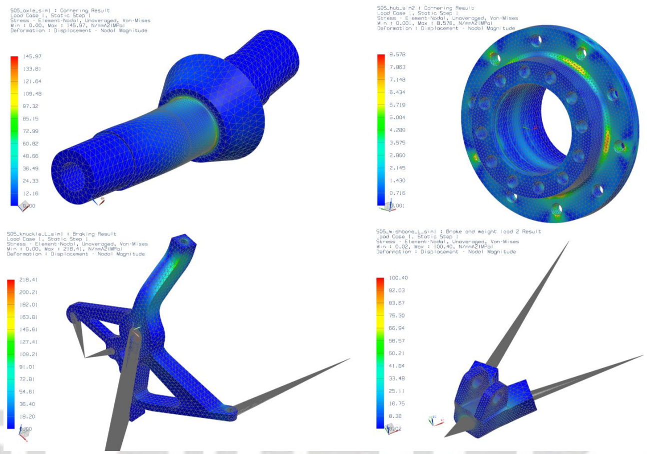
- Requirements were specified with regards to
  - Scrub radius and KPI
  - Toe-in/toe-out
  - Camber andCaster angle
  - Bump steer
  - Ackerman steering
  - Turning radius
  - Camber change
  - Adjustability
  - Strength
  - Brakes
- Software used
  - UGS NX 7.5 with assembly constraints and built-in support for Nastran FEA

Conclusion

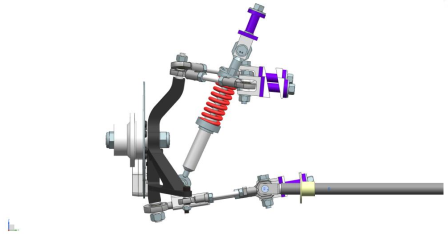
Results showed that parts and joints could be made strong enough and light-weight. A very strong focus on removing material with low stress concentrations would sky-rocket manufacturing complexity with very small gains in weight-saving.



Ackerman steering check using assembly constraints



Snapshots from FEM analyses



CAD integration analysis

Figure C.9

S.5 Front suspension

Aksel Qviller/2012

DNV FUEL FIGHTER 2

Identified risks

Description	L	C	LxC	Mitigation
Misalignment	2	4	6	Fine tune alignments
Rods may bend	2	3	5	Use spare
Rods may break	1	4	5	Use spare
Axle may bend	2	2	4	Use spare
Axle may break	1	2	2	Use spare
Lug threads wear out	2	4	8	Tape over 2nd set of lug bolt holes, then after damage: use 2nd set of lug threads on same component
Brake disc threads wear out	1	4	4	Use 2nd set of threads on same component
Retaining ring groove creeps	1	3	3	Will not happen.
Retaining ring fails	1	2	2	Use spare retaining ring
Tie rod may bend	1	5	5	Spare
Tie rod may break	1	5	5	Spare
Misaligned	3	3	9	Realign
Delamination on upper swivel bolt	1	3	3	See below.
Lower swivel bolt pull-out and resulting delamination in knuckle	2	5	10	Glue bolt back in, and try again. Otherwise use spare. Ideally avoid switching to spare, if the knuckle was subjected to adverse loads which cannot be avoided in the future.
Toe delamination	1	4	4	Glue toe back together and wrap in carbon fiber or perforated strips, depending on the situation.

L = likelihood, C = consequence

Performance

The parts were verified in CAD assemblies for compliance, e.g. whether they fit together, did not collide, and moved within the limits of the joints. Expected loads for the different parts were calculated and used to apply forces in FEM analyses, the results from which were used to improve the part designs.

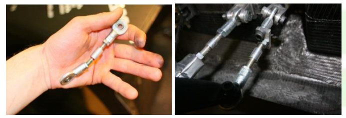
During manufacturing the parts were measured and mated with other parts to check that they were within tolerance. After assembly to the car the parts were first tested as part of the alignment. It was discovered that the clevises attaching the rods ends to the body needed to have rounded corners, or the swing angle would be severely restrained.

Also, the measured caliper width was larger than expected (no bike brake manufacturers share 3D exterior models whatsoever), so the center plates of the rims scratched against them. This was solved by moving the hubs a few millimeters outwards using short spacer rings on the axles, and moving the brake discs away from the hub using M5 washers. The latter one required careful micrometer measurements for selection since washers are produced with low tolerance, and could make brand new brake discs uneven.

The suspension was later put to test by driving indoors, and soon outdoors. At this stage the suspension satisfied expectations and exceeded them with the performance on uneven, mixed asphalt/gravel test site. The brakes were easy to fit, but a completely silent configuration required a lot of careful adjustment.

The left tie-rod buckled due to a bending moment from the toe on the steering knuckle. The reason is that the draglink was located higher than expected because of the foam core in the floor, and when the coilover pushed the suspension down to the lower extreme, the rod end reached its limit and the tie-rod and draglink had to support it.

After 12 consecutive, successful test laps the tie-rod had sustained said damage. This was repaired with a small aluminium plate connecting the rod ends without their shanks being coaxial. It is not an ideal, permanent solution, but strong enough for continued use.



During the competition the combined front and rear suspension is credited with giving us the ability to corner at high speed and eliminate the need to brake just for handling the course.

After coming home from the competition, the front right hub gives a rattling noise for each revolution – a ball bearing might have sustained damage. It can easily be replaced by one of the spares. This might be caused by the bearings being too small for the loads, the distance between them might be too short, their insertion or higher than anticipated forces during the race.

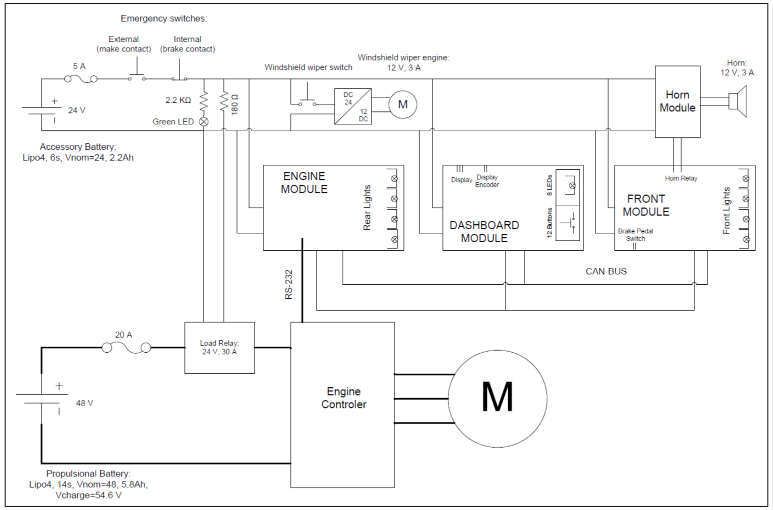
Proposed future work

- Redesign the steering knuckles to support the draglink's higher than expected location. Also, the Ackerman compliance should be reduced to allow sharper turn radii.
- The hubs' weights can be reduced. However a smaller hub may also mean more carbon fiber mass in the rim to accomodate, so the weight-reduction is not that apparent.
- Consider stronger ball bearings.
- The linkages can be improved by using smaller rod ends. They are dimensioned from the strength of an aluminium rod with external threads. Smaller rod ends with male shanks may be used, but then the links might have to be made from steel or a more expensive aluminium alloy. Alumec is sold at diameters less than Ø28mm, so there is a risk of a large waste of material.
- The vibration reduction might be removed, and the epoxy wedges should be replaced with softer polymers of acetal-grade hardness.

Figure C.10

Car Control System + Propulsional Control System Benni Gutjahr,2012

DNV FUEL FIGHTER 2



System architecture

- modular car control system 24V
- two independent emergency switches (internal and external)
- module communication via CAN-Bus
- front module: operating front lights, horn break pedal sensor
- dashboard module: driver interface – operates display and handle button pushes
- engine module: operates rear light and communicates with engine controller via RS-232
- special features: cruise control and automated section control -> no gaspedal
- interface ccs to propulsional control system: RS-232 and load relay
- the car control system was developed in 2010 by Anders Guldahl (see his Master Thesis)
- 2012 modifications: inclusion of windsheeld wiper and change of button handle
- Smart Motors engine controller operating 2011 engine (Master Thesis 2011)

User Manual for CCS:

In order to drive the car:

1. both external and internal emergency switch has to be in ON position
2. all CCS functionality like horn and lights are operational
3. press START button – you can run the car when the button light shines continously
4. Using Cruise Control press lowest black button on the left on the steering wheel! Use the right upper buttons to increase speed – use the left upper buttons to decrease speed.
5. If Cruise Control is disabled the upper 4 buttons are section control buttons. Pressing a button sets a predefined speed and acceleration
6. To stop the car – BRAKE (always release the breake rapidly)
7. Turn off the car by pressing STOP button and either use internal or external emergency button to shut off power

Software used:

- AVR Studio 4
- Br@y++ Terminal v1.9b
- PCAN-View
- USBlyer
- DSPComm

Hardware Used;

- ATMEL_JTAGICE_mkII (dirver: see online)
- PCAN-USB (driver: see online)
- Aten USB to RS-232 (driver: see online, very PC related)

In order to flash the modules:

Use Atmel JTAG mkII and AVR Studio 4 to flash new code via the universal boards microcontroller pins or use P-CAN USB for flashing via CAN-Bus

For more information:

See final project report 2012 / 2011, Master Thesis Andres Guldahl 2010, specialization report 2011, Master Thesis André Dahl-Jacobsen 2011

DNV FUEL FIGHTER 2

Figure C.11

Car Control System + Propulsional Control System Benni Gutjahr,2012

DNV FUEL FIGHTER 2

Track Analysis

Background

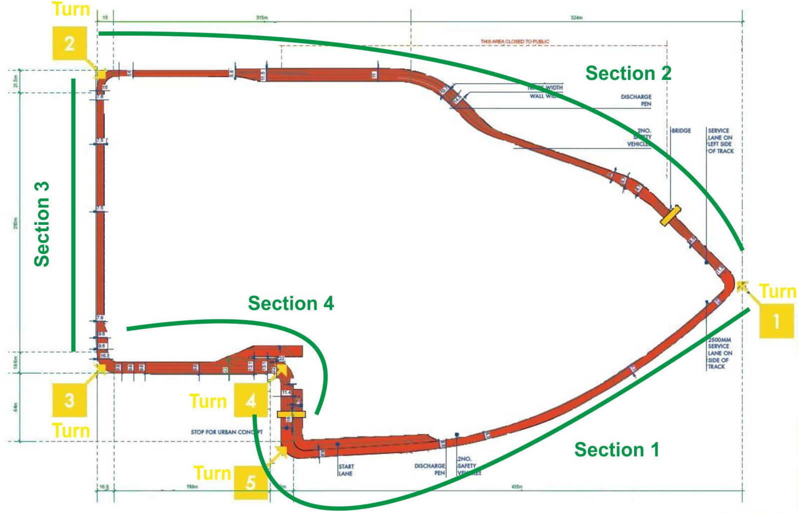
- Requirements:
  Finding the most energyefficient verlocity profile for the given track
- Software used:
  MATLAB and Google Earth
- Assumptions:
  Each section is assumed to be straight – only longituninal car movement is modeled
  The track is assumed to be flat – highprofile was not available

Results

The optimization result for each section ins a optimized maximum speed, optimized minumum acceleration and a time from with the car should stard coasting towards a turn.

Conclusion

It is very crucial to have a prizice track analysis in order to compete in the competition. The inclination should not be considert zero.



Optimization Algorithm

- The inputs of the optimization algorithm are all track parameters like section lenght and turning speed
- The section length was mesured by using google earth
- The max. cornering speed was determined while testruns in rotterdam (v_turn_max_i > 30km/h)
- The algorithm uses the engines efficiency diagram for Power over speed and torque
- The algorithem solves the optimization by first solving a suboptimal problem for each section before optimizing the complete track
- First all possible time combinations for all sections are determined – for each possible combination the power consumption is calculated
- The results of the optimization is used for the automated section controlled drive ASCD using the section buttons
- For each section button the optimized velocityprofile is hardcoded and triggerd whenever a section button is pressed

Figure C.12

Car Control System + Propulsional Control System   Benni Gutjahr,2012   DNV FUEL FIGHTER 2

Identified risks

Description	L	C	LxC	Mitigation
Too windy	3	1	3	Very hard
Over current	2	5	10	Software limits Suitable battery
Under voltage	2	5	10	Software limits Suitable battery
Start routine fails	2	5	10	Higher starting vector

L = likelihood, C = consequence

Performance

Summarize the verification process (testing history);

The system was tested on a testbench at SmartMotors(also see sponsors) for different speeds and torques. Only the blue battery pack has been tested. The blue battery pack performed well when fully charged. After a relativly short time the voltage droped significantly so that the BMS shut down. To avoid a battery shutdown a low voltage limit was set for the controller. Also a torque limit was set. This torque limit can also be used for a over current protection.

During race testing the blue battery pack didn´t deliver the power needed. Testing the white battery pack discoverd that the contollers flystart routine creates a over current on the controllers DC link and caused a battery shut down. Also high engine torques due to track inclination and wind lend to a BMS over current shutdown. This problem was fixed by setting a very conservative torque limit and using a fixed decelleration rate to replace coasting. Doing this prevented further brakedowns.

The overall result in milage was lower than expected. This is mainly due to the big engine airgap and the very energy consuming engine start routine running on the controller.

Proposed future work

In order to improve the cars performace the following points should be considert:

- building a on board car data logging unit
- making telemetry module work
- building a engine testbench
- designing a new drive train that fits the track properties and uses a different start routine
- including windsheeld wiper and horn in one module

DNV FUEL FIGHTER 2

Figure C.13

Appendix D

System architecture of DNV Fuel Fighter 2

This appendix contains the system architecture of the DNVFF2, with some of the interface diagrams. More can be seen in the system model which is supplied as a supplement to the submission of this thesis.

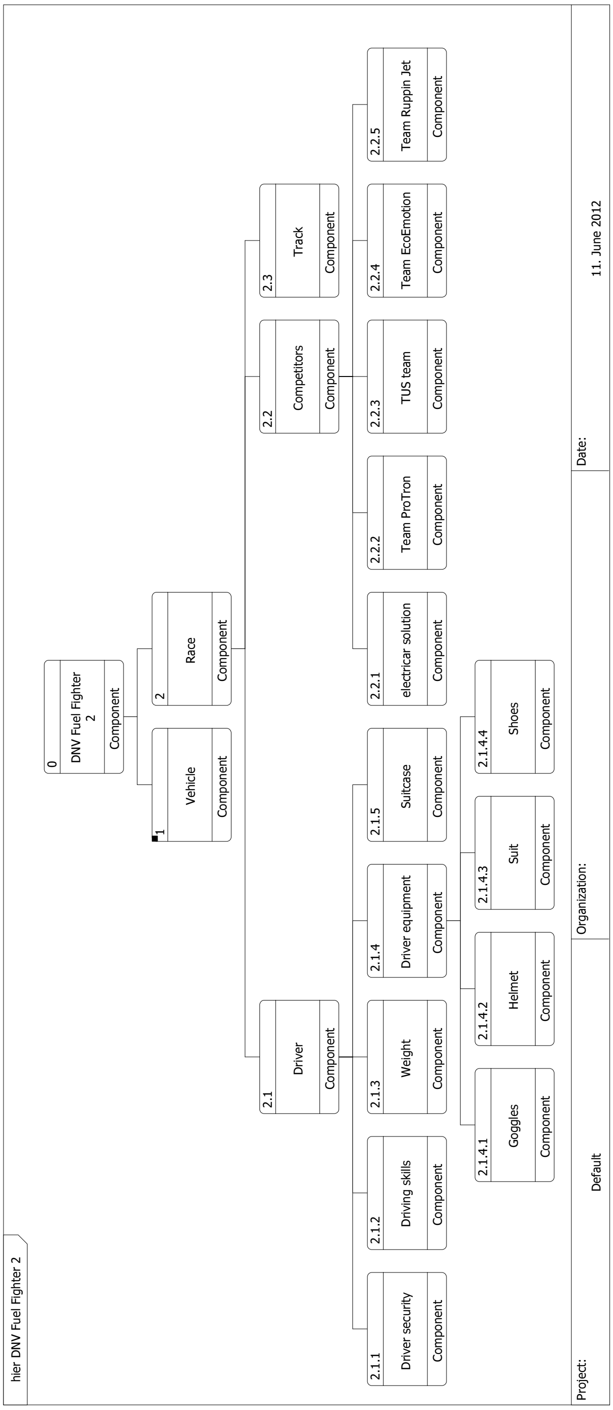

hier DNV Fuel Fighter 2
0
DNV Fuel Fighter 2
Component
1
Vehicle
Component
2
Race
Component
2.1
Driver
Component
2.2
Competitors
Component
2.3
Track
Component
2.1.1
Driver security
Component
2.1.2
Driving skills
Component
2.1.3
Weight
Component
2.1.4
Driver equipment
Component
2.1.5
Suitcase
Component
2.1.4.1
Goggles
Component
2.1.4.2
Helmet
Component
2.1.4.3
Suit
Component
2.1.4.4
Shoes
Component
2.2.1
electricar solution
Component
2.2.2
Team ProTron
Component
2.2.3
TUS team
Component
2.2.4
Team EcoEmotion
Component
2.2.5
Team Ruppin Jet
Component
Project:
Default
Organization:
Date:
11. June 2012

Figure D.1

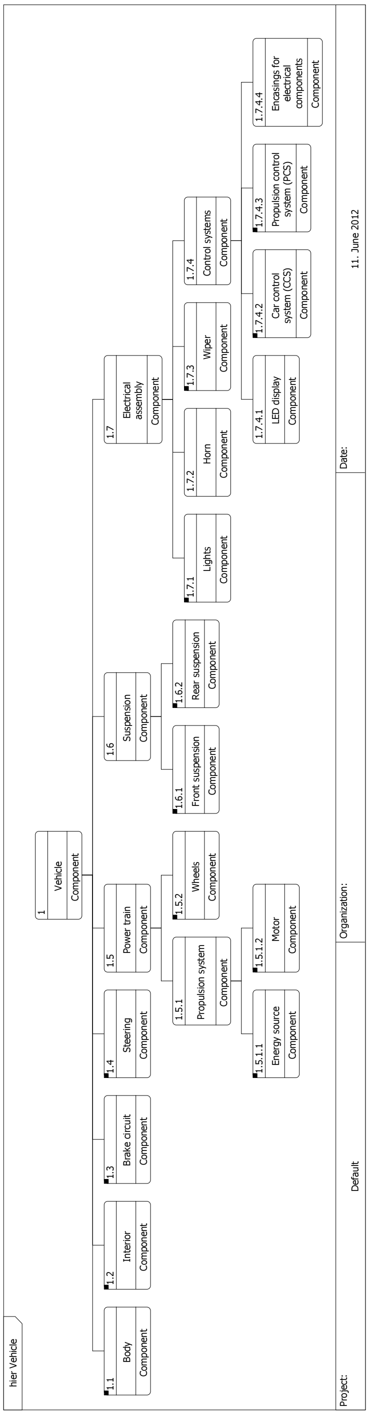

hier Vehicle
1 Vehicle Component
1.1 Body Component
1.2 Interior Component
1.3 Brake circuit Component
1.4 Steering Component
1.5 Power train Component
1.5.1 Propulsion system Component
1.5.1.1 Energy source Component
1.5.1.2 Motor Component
1.5.2 Wheels Component
1.6 Suspension Component
1.6.1 Front suspension Component
1.6.2 Rear suspension Component
1.7 Electrical assembly Component
1.7.1 Lights Component
1.7.2 Horn Component
1.7.3 Wiper Component
1.7.4 Control systems Component
1.7.4.1 LED display Component
1.7.4.2 Car control system (CCS) Component
1.7.4.3 Propulsion control system (PCS) Component
1.7.4.4 Encasings for electrical components Component
Project:
Default
Organization:
Date:
11. June 2012

Figure D.2

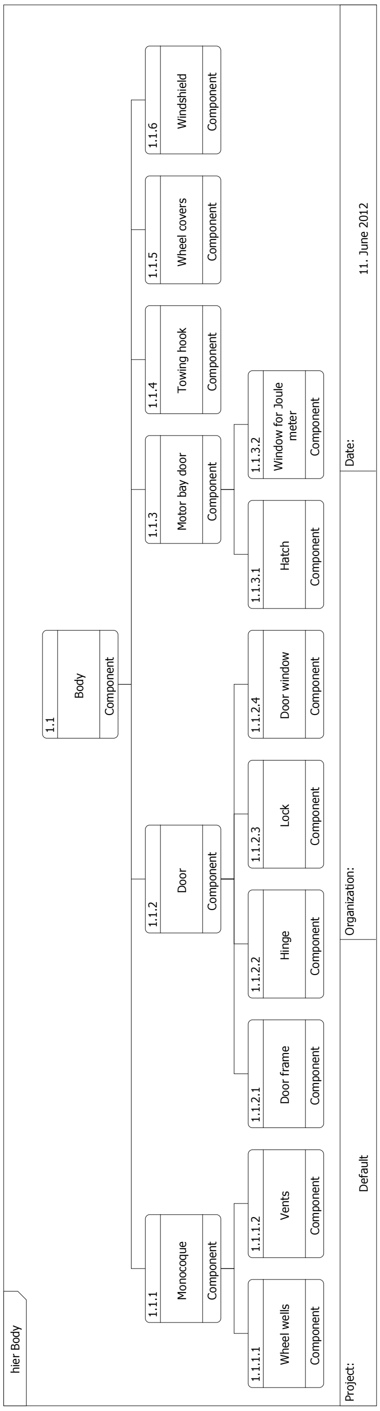

hier Body
1.1
Body
Component
1.1.1
Monocoque
Component
1.1.1.1
Wheel wells
Component
1.1.1.2
Vents
Component
1.1.2
Door
Component
1.1.2.1
Door frame
Component
1.1.2.2
Hinge
Component
1.1.2.3
Lock
Component
1.1.2.4
Door window
Component
1.1.3
Motor bay door
Component
1.1.3.1
Hatch
Component
1.1.3.2
Window for Joule meter
Component
1.1.4
Towing hook
Component
1.1.5
Wheel covers
Component
1.1.6
Windshield
Component
Project:
Default
Organization:
Date:
11. June 2012

Figure D.3

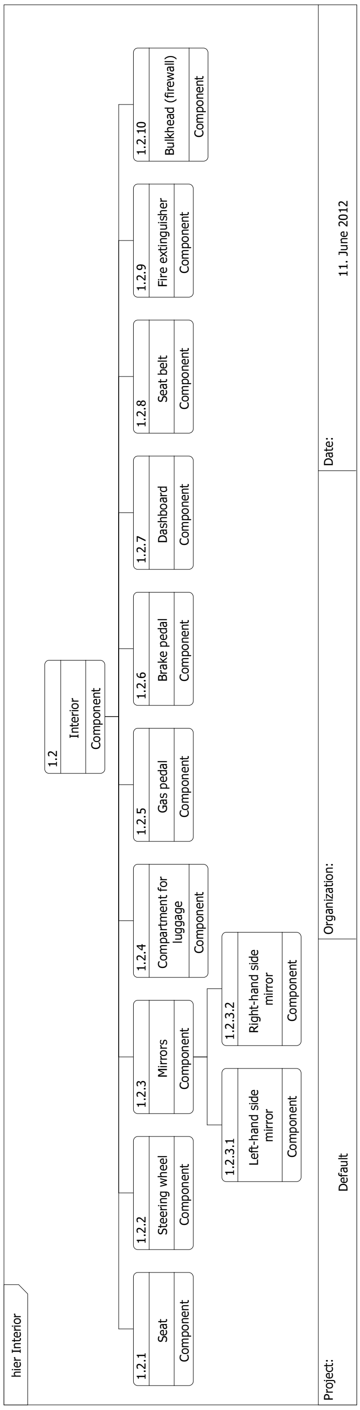

hier Interior
1.2
Interior
Component
1.2.1
Seat
Component
1.2.2
Steering wheel
Component
1.2.3
Mirrors
Component
1.2.3.1
Left-hand side mirror
Component
1.2.3.2
Right-hand side mirror
Component
1.2.4
Compartment for luggage
Component
1.2.5
Gas pedal
Component
1.2.6
Brake pedal
Component
1.2.7
Dashboard
Component
1.2.8
Seat belt
Component
1.2.9
Fire extinguisher
Component
1.2.10
Bulkhead (firewall)
Component
Project:
Default
Organization:
Date:
11. June 2012

Figure D.4

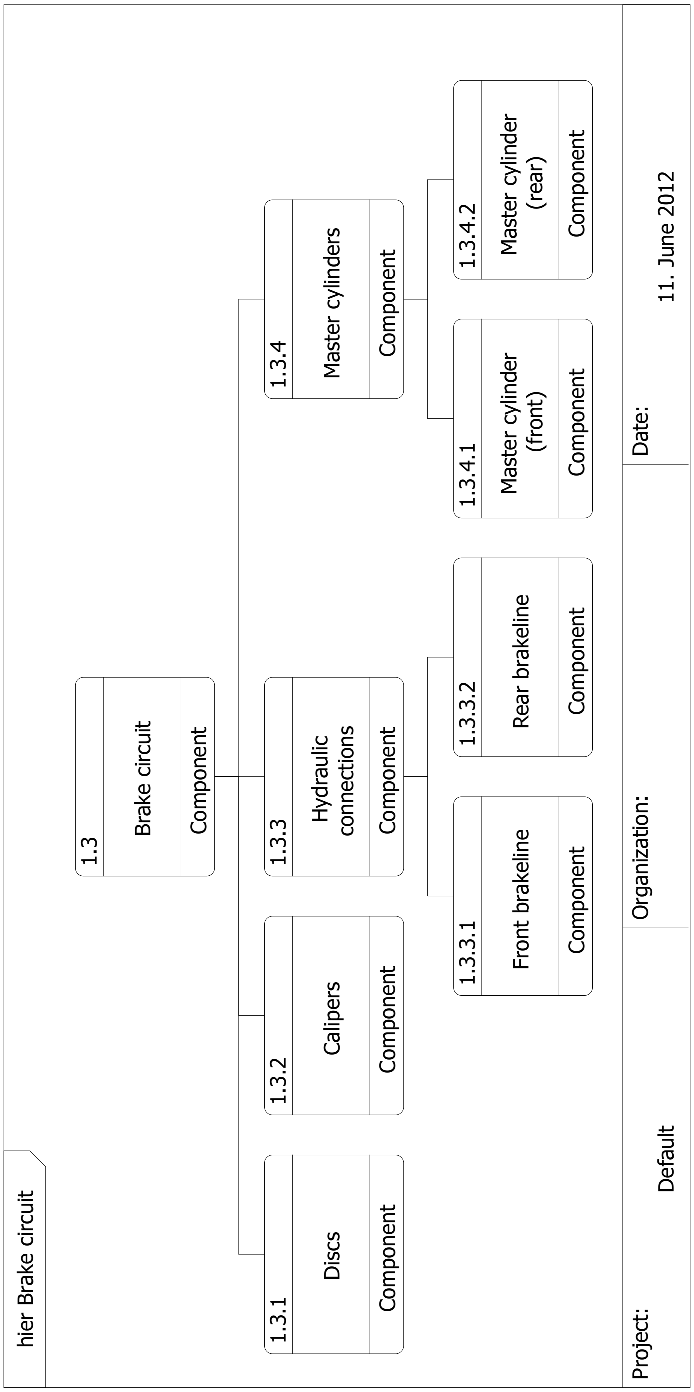

hier Brake circuit
1.3
Brake circuit
Component
1.3.1
Discs
Component
1.3.2
Calipers
Component
1.3.3
Hydraulic connections
Component
1.3.4
Master cylinders
Component
1.3.3.1
Front brakeline
Component
1.3.3.2
Rear brakeline
Component
1.3.4.1
Master cylinder (front)
Component
1.3.4.2
Master cylinder (rear)
Component
Project:
Default
Organization:
Date:
11. June 2012

Figure D.5

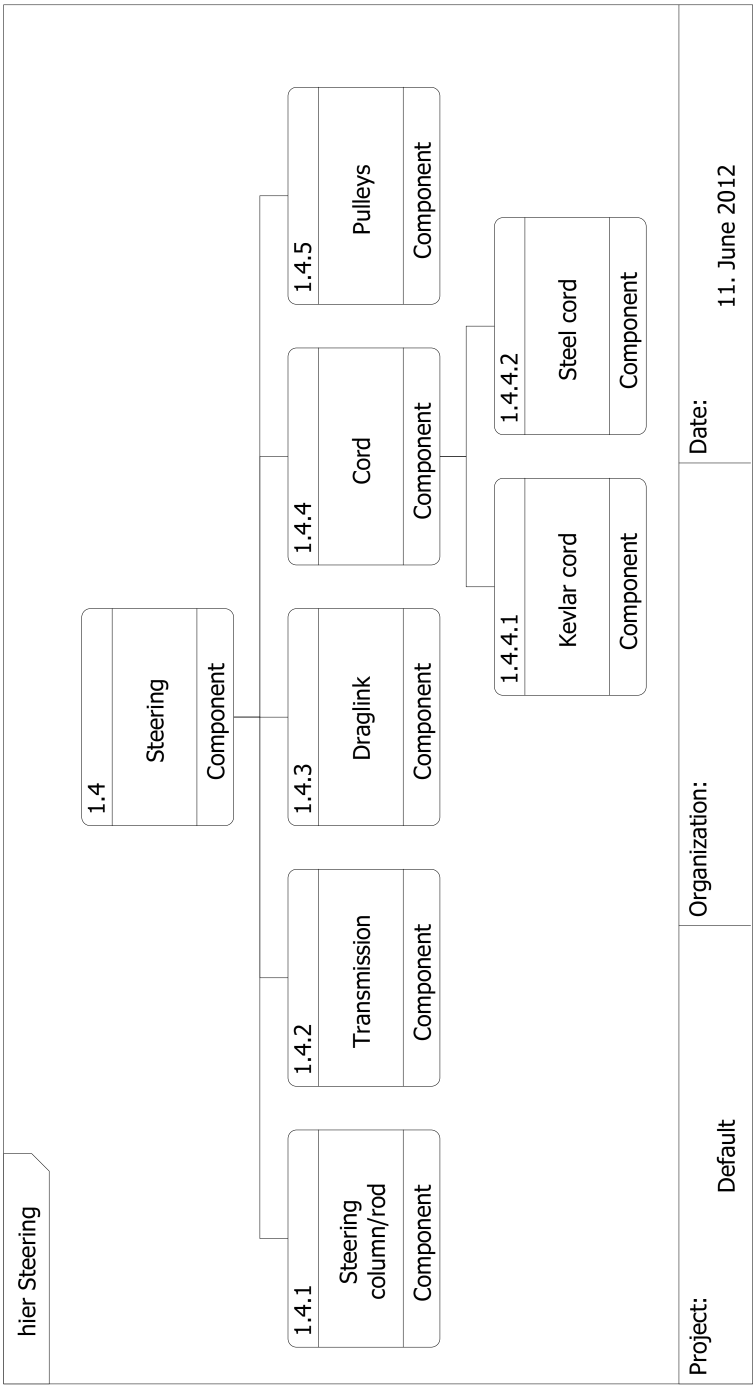

hier Steering
1.4
Steering
Component
1.4.1
Steering column/rod
Component
1.4.2
Transmission
Component
1.4.3
Draglink
Component
1.4.4
Cord
Component
1.4.5
Pulleys
Component
1.4.4.1
Kevlar cord
Component
1.4.4.2
Steel cord
Component
Project:
Default
Organization:
Date:
11. June 2012

Figure D.6

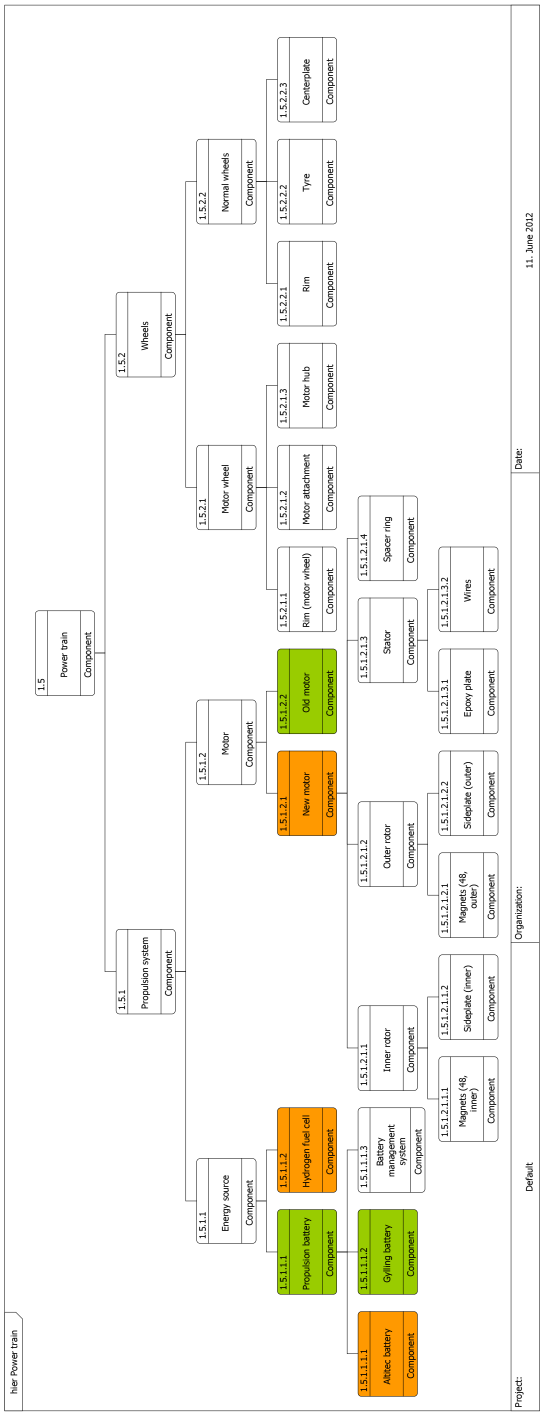

hier Power train
1.5 Power train Component
1.5.1 Propulsion system Component
1.5.2 Wheels Component
1.5.1.1 Energy source Component
1.5.1.2 Motor Component
1.5.2.1 Motor wheel Component
1.5.2.2 Normal wheels Component
1.5.1.1.1 Propulsion battery Component
1.5.1.1.2 Hydrogen fuel cell Component
1.5.1.2.1 New motor Component
1.5.1.2.2 Old motor Component
1.5.2.1.1 Rim (motor wheel) Component
1.5.2.1.2 Motor attachment Component
1.5.2.1.3 Motor hub Component
1.5.2.2.1 Rim Component
1.5.2.2.2 Tyre Component
1.5.2.2.3 Centerplate Component
1.5.1.1.1.1 Altitec battery Component
1.5.1.1.1.2 Gylling battery Component
1.5.1.1.1.3 Battery management system Component
1.5.1.2.1.1 Inner rotor Component
1.5.1.2.1.2 Outer rotor Component
1.5.1.2.1.3 Stator Component
1.5.1.2.1.4 Spacer ring Component
1.5.1.2.1.1.1 Magnets (48, inner) Component
1.5.1.2.1.1.2 Sideplate (inner) Component
1.5.1.2.1.2.1 Magnets (48, outer) Component
1.5.1.2.1.2.2 Sideplate (outer) Component
1.5.1.2.1.3.1 Epoxy plate Component
1.5.1.2.1.3.2 Wires Component
Project: Default
Organization:
Date: 11. June 2012

Figure D.7

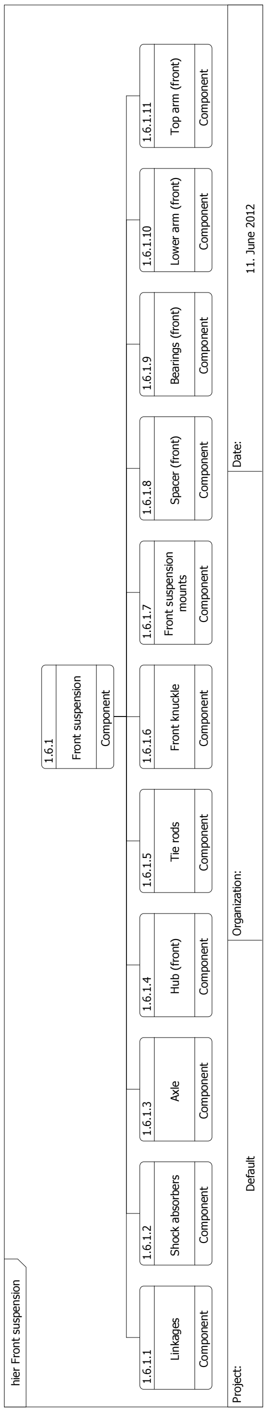

hier Front suspension
1.6.1
Front suspension
Component
1.6.1.1
Linkages
Component
1.6.1.2
Shock absorbers
Component
1.6.1.3
Axle
Component
1.6.1.4
Hub (front)
Component
1.6.1.5
Tie rods
Component
1.6.1.6
Front knuckle
Component
1.6.1.7
Front suspension mounts
Component
1.6.1.8
Spacer (front)
Component
1.6.1.9
Bearings (front)
Component
1.6.1.10
Lower arm (front)
Component
1.6.1.11
Top arm (front)
Component
Project:
Default
Organization:
Date:
11. June 2012

Figure D.8

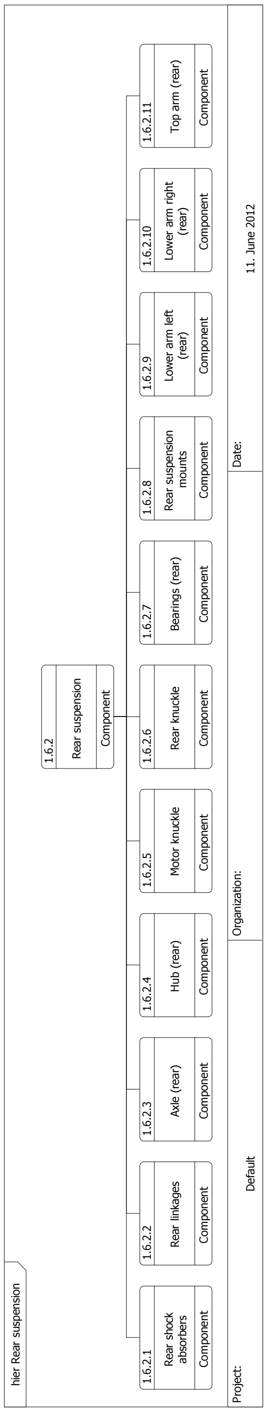

hier Rear suspension
1.6.2
Rear suspension
Component
1.6.2.1
Rear shock absorbers
Component
1.6.2.2
Rear linkages
Component
1.6.2.3
Axle (rear)
Component
1.6.2.4
Hub (rear)
Component
1.6.2.5
Motor knuckle
Component
1.6.2.6
Rear knuckle
Component
1.6.2.7
Bearings (rear)
Component
1.6.2.8
Rear suspension mounts
Component
1.6.2.9
Lower arm left (rear)
Component
1.6.2.10
Lower arm right (rear)
Component
1.6.2.11
Top arm (rear)
Component
Project:
Default
Organization:
Date:
11. June 2012

Figure D.9

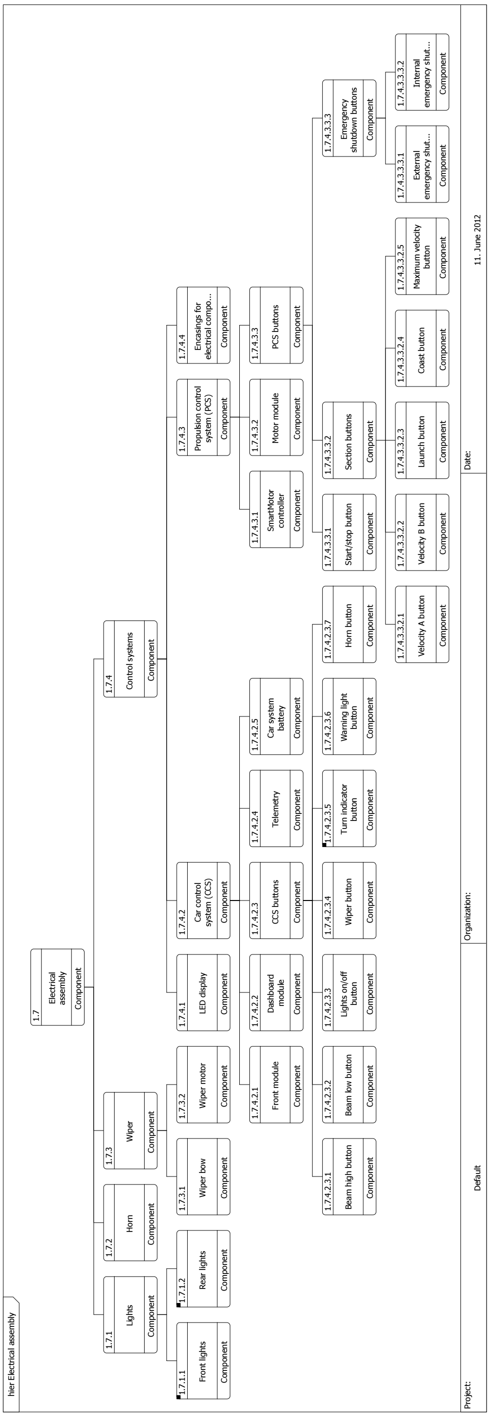

hier Electrical assembly
1.7 Electrical assembly Component
1.7.1 Lights Component
1.7.2 Horn Component
1.7.3 Wiper Component
1.7.4 Control systems Component
1.7.1.1 Front lights Component
1.7.1.2 Rear lights Component
1.7.3.1 Wiper bow Component
1.7.3.2 Wiper motor Component
1.7.4.1 LED display Component
1.7.4.2 Car control system (CCS) Component
1.7.4.3 Propulsion control system (PCS) Component
1.7.4.4 Encasings for electrical compo... Component
1.7.4.2.1 Front module Component
1.7.4.2.2 Dashboard module Component
1.7.4.2.3 CCS buttons Component
1.7.4.2.4 Telemetry Component
1.7.4.2.5 Car system battery Component
1.7.4.3.1 SmartMotor controller Component
1.7.4.3.2 Motor module Component
1.7.4.3.3 PCS buttons Component
1.7.4.2.3.1 Beam high button Component
1.7.4.2.3.2 Beam low button Component
1.7.4.2.3.3 Lights on/off button Component
1.7.4.2.3.4 Wiper button Component
1.7.4.2.3.5 Turn indicator button Component
1.7.4.2.3.6 Warning light button Component
1.7.4.2.3.7 Horn button Component
1.7.4.3.3.1 Start/stop button Component
1.7.4.3.3.2 Section buttons Component
1.7.4.3.3.3 Emergency shutdown buttons Component
1.7.4.3.3.2.1 Velocity A button Component
1.7.4.3.3.2.2 Velocity B button Component
1.7.4.3.3.2.3 Launch button Component
1.7.4.3.3.2.4 Coast button Component
1.7.4.3.3.2.5 Maximum velocity button Component
1.7.4.3.3.3.1 External emergency shut... Component
1.7.4.3.3.3.2 Internal emergency shut... Component
Project:
Default
Organization:
Date:
11. June 2012

Figure D.10

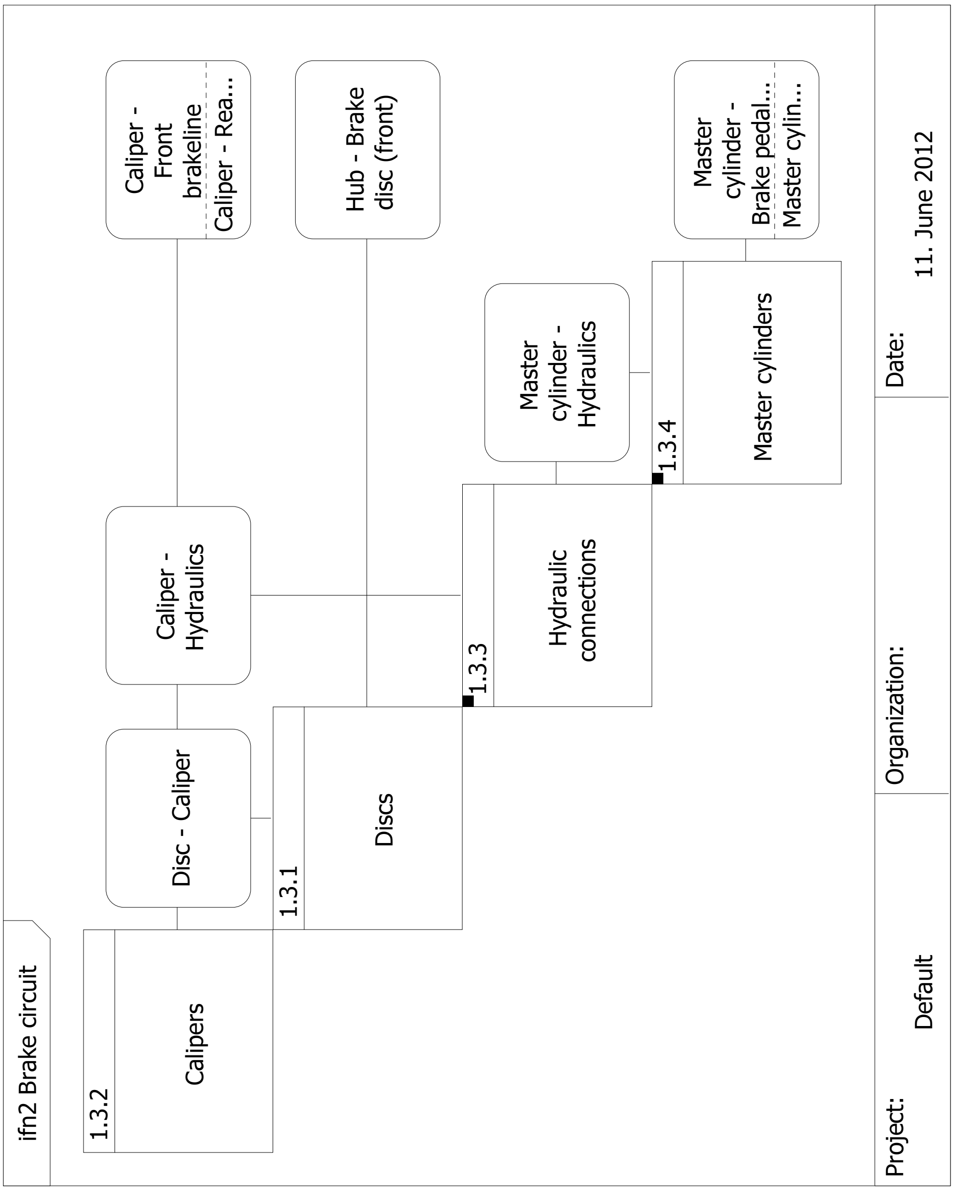

ifn2 Brake circuit
1.3.2
Calipers
1.3.1
Discs
1.3.3
Hydraulic connections
1.3.4
Master cylinders
Disc - Caliper
Caliper - Hydraulics
Caliper - Front brakeline
Caliper - Rea...
Hub - Brake disc (front)
Master cylinder - Hydraulics
Master cylinder - Brake pedal...
Master cylin...
Project:
Default
Organization:
Date:
11. June 2012

Figure D.11

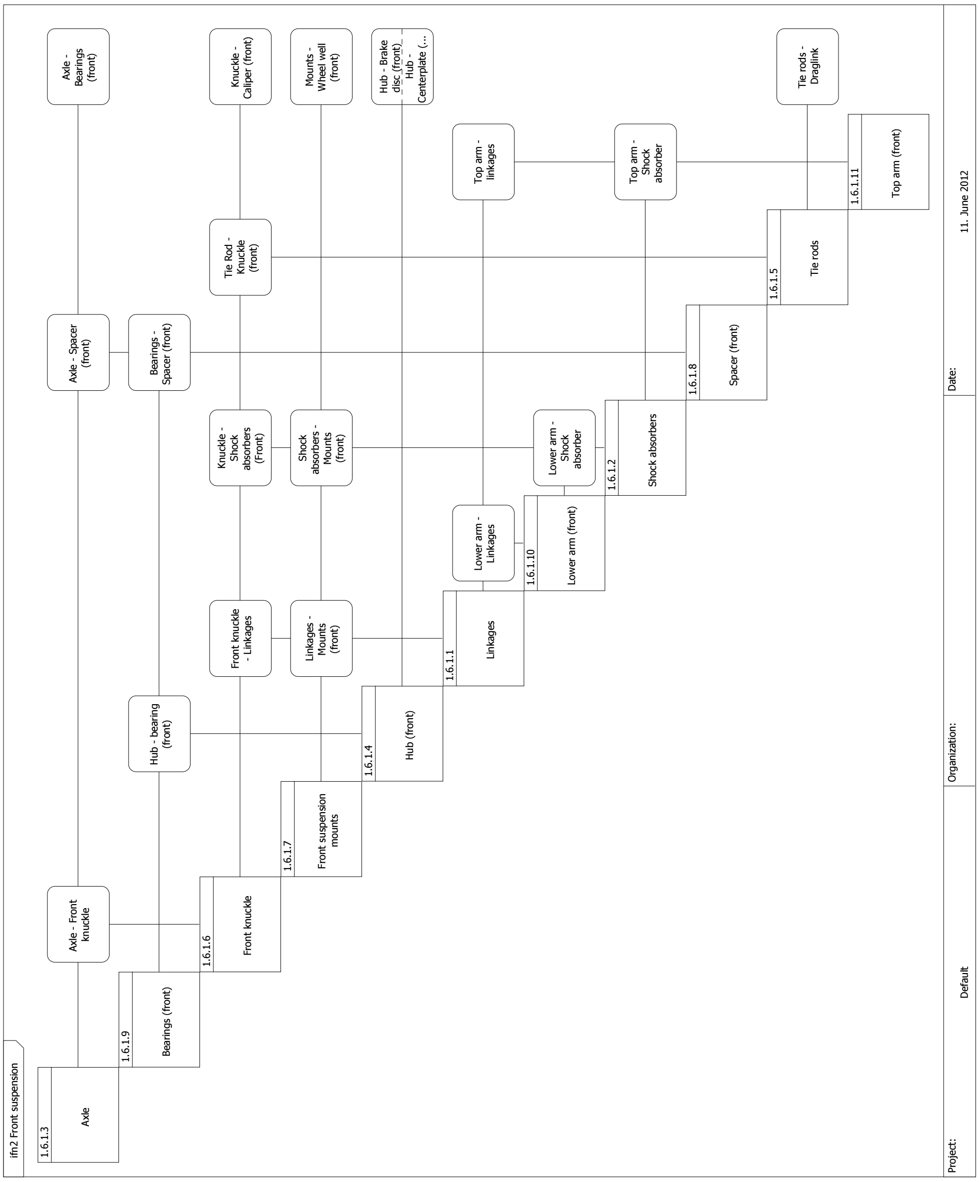

ifn2 Front suspension
1.6.1.3
Axle
1.6.1.9
Bearings (front)
1.6.1.6
Front knuckle
1.6.1.7
Front suspension mounts
1.6.1.4
Hub (front)
1.6.1.1
Linkages
1.6.1.10
Lower arm (front)
1.6.1.2
Shock absorbers
1.6.1.8
Spacer (front)
1.6.1.5
Tie rods
1.6.1.11
Top arm (front)
Axle - Bearings (front)
Axle - Spacer (front)
Axle - Front knuckle
Bearings - Spacer (front)
Hub - bearing (front)
Knuckle - Caliper (front)
Tie Rod - Knuckle (front)
Knuckle - Shock absorbers (Front)
Front knuckle - Linkages
Mounts - Wheel well (front)
Shock absorbers - Mounts (front)
Linkages - Mounts (front)
Hub - Brake disc (front)_ Hub - Centerplate (...
Top arm - linkages
Lower arm - Linkages
Lower arm - Shock absorber
Top arm - Shock absorber
Tie rods - Draglink
Project:
Default
Organization:
Date:
11. June 2012

Figure D.12

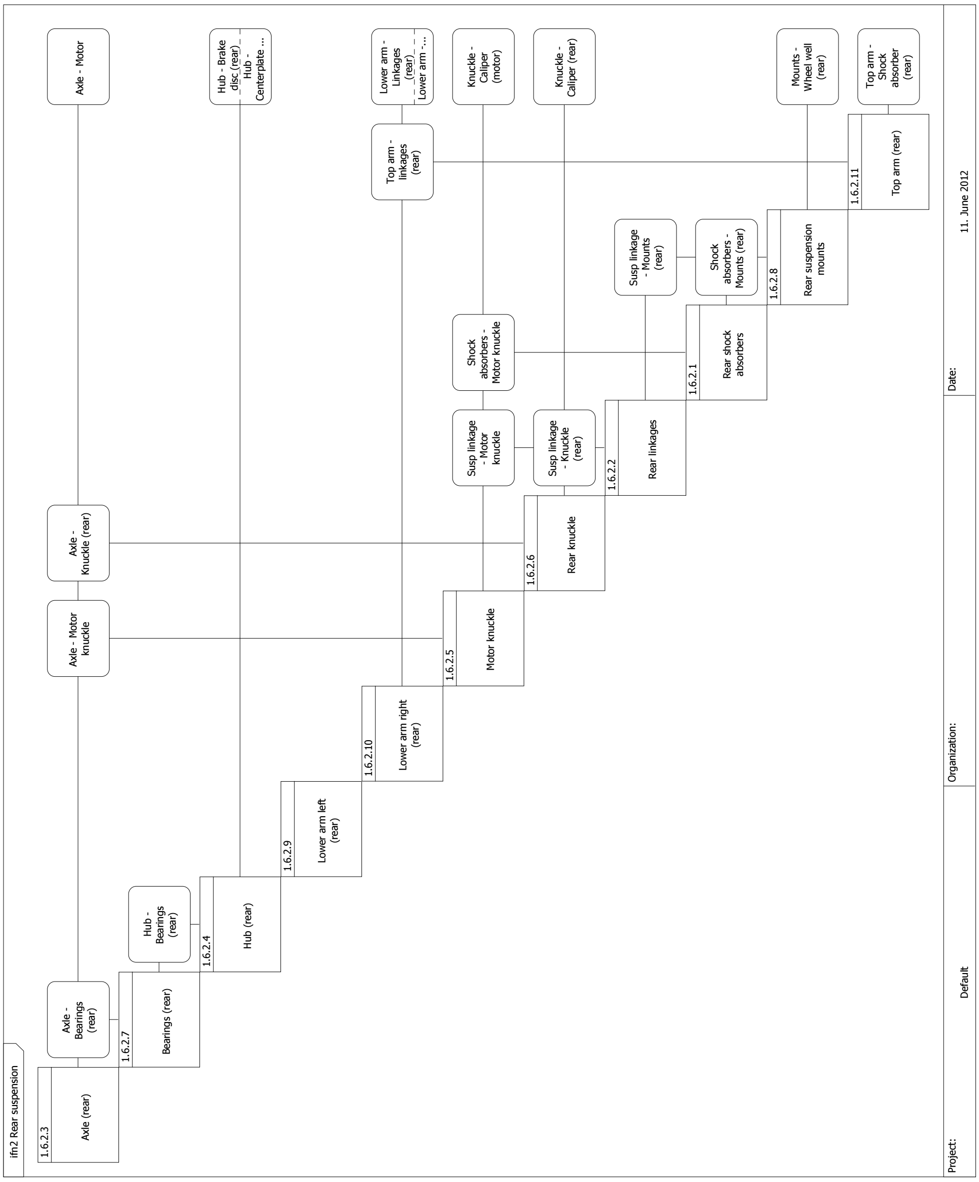

ifn2 Rear suspension
1.6.2.3
Axle (rear)
1.6.2.7
Bearings (rear)
1.6.2.4
Hub (rear)
1.6.2.9
Lower arm left (rear)
1.6.2.10
Lower arm right (rear)
1.6.2.5
Motor knuckle
1.6.2.6
Rear knuckle
1.6.2.2
Rear linkages
1.6.2.1
Rear shock absorbers
1.6.2.8
Rear suspension mounts
1.6.2.11
Top arm (rear)
Axle - Bearings (rear)
Hub - Bearings (rear)
Axle - Motor knuckle
Axle - Knuckle (rear)
Axle - Motor
Hub - Brake disc (rear)
Hub - Centerplate ...
Lower arm - Linkages (rear)
Lower arm - ...
Top arm - linkages (rear)
Knuckle - Caliper (motor)
Knuckle - Caliper (rear)
Susp linkage - Motor knuckle
Susp linkage - Knuckle (rear)
Shock absorbers - Motor knuckle
Susp linkage - Mounts (rear)
Shock absorbers - Mounts (rear)
Mounts - Wheel well (rear)
Top arm - Shock absorber (rear)
Project:
Default
Organization:
Date:
11. June 2012

Figure D.13

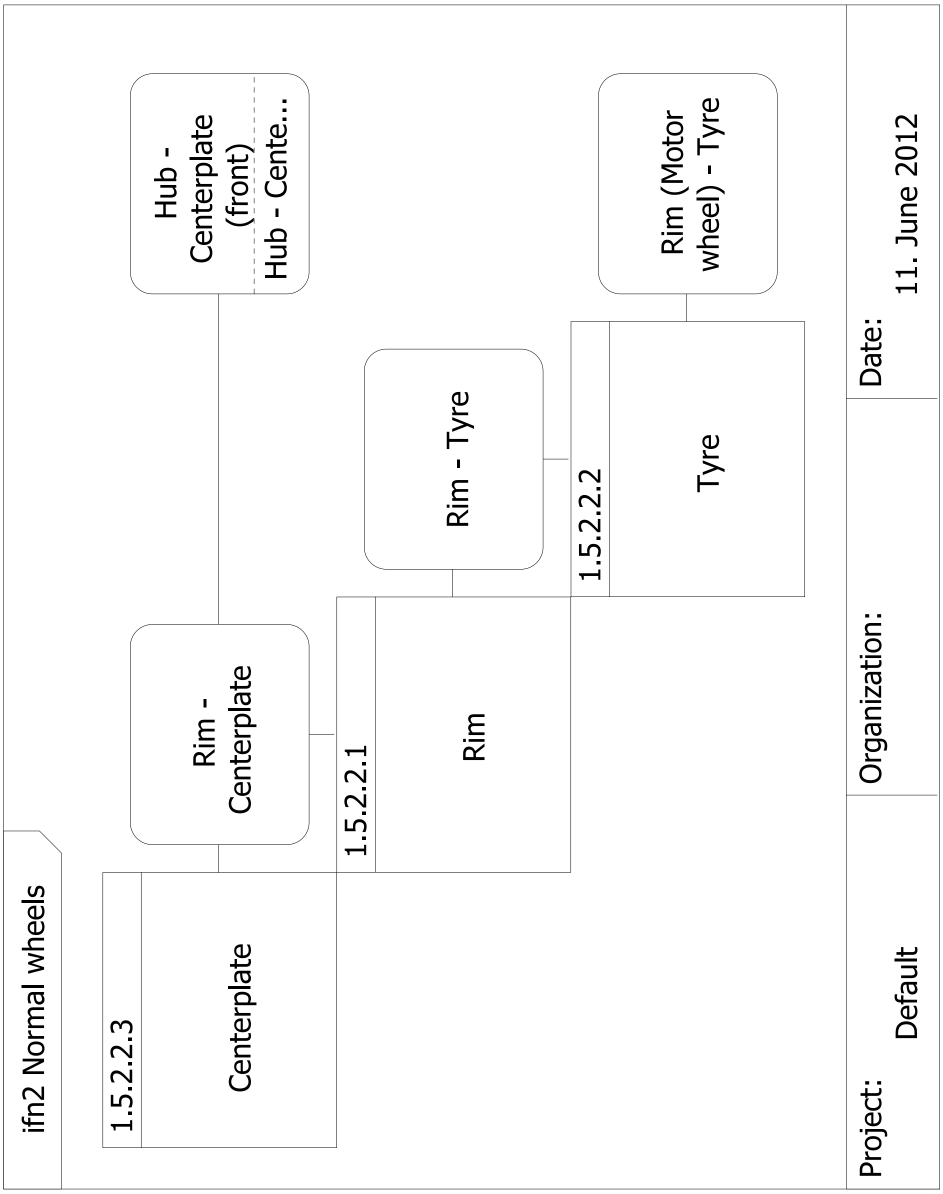

ifn2 Normal wheels
1.5.2.2.3
Centerplate
Rim - Centerplate
Hub - Centerplate (front)
Hub - Cente...
1.5.2.2.1
Rim
Rim - Tyre
1.5.2.2.2
Tyre
Rim (Motor wheel) - Tyre
Project:
Default
Organization:
Date:
11. June 2012

Figure D.14

Appendix E

Functional analysis DNV Fuel Fighter 2

This appendix shows parts of the Functional analysis of DNVFF2. The functional analysis is focused on the handling of the vehicle, and the events at the competition.

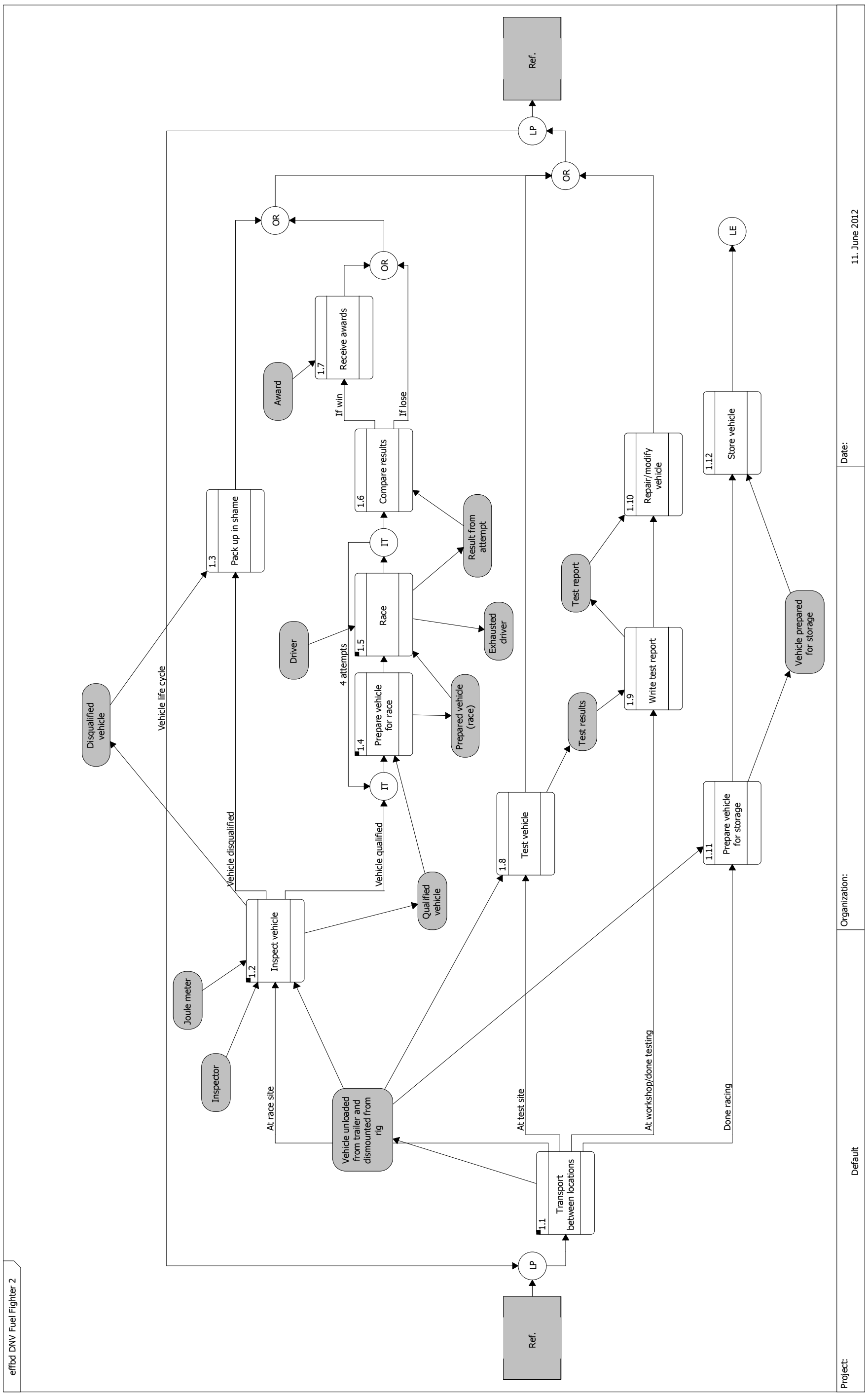

effbd DNV Fuel Fighter 2
Ref.
LP
1.1 Transport between locations
Vehicle unloaded from trailer and dismounted from rig
At race site
At test site
At workshop/done testing
Done racing
Joule meter
Inspector
1.2 Inspect vehicle
Qualified vehicle
Vehicle qualified
Vehicle disqualified
Disqualified vehicle
1.3 Pack up in shame
Vehicle life cycle
IT
1.4 Prepare vehicle for race
Prepared vehicle (race)
Driver
4 attempts
1.5 Race
Exhausted driver
Result from attempt
1.6 Compare results
If win
If lose
Award
1.7 Receive awards
OR
1.8 Test vehicle
Test results
1.9 Write test report
Test report
1.10 Repair/modify vehicle
1.11 Prepare vehicle for storage
Vehicle prepared for storage
1.12 Store vehicle
LE
Project:
Default
Organization:
Date:
11. June 2012

Figure E.1

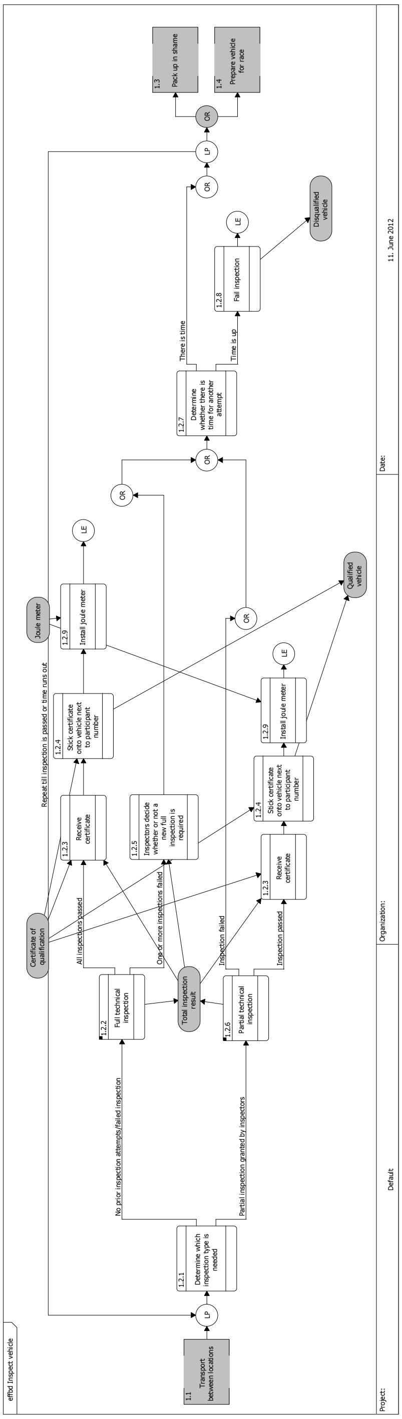

effbd Inspect vehicle
1.1 Transport between locations
1.2.1 Determine which inspection type is needed
No prior inspection attempts/failed inspection
Partial inspection granted by inspectors
1.2.2 Full technical inspection
1.2.6 Partial technical inspection
Total inspection result
All inspections passed
One or more inspections failed
Inspection failed
Inspection passed
Certificate of qualification
1.2.3 Receive certificate
1.2.4 Stick certificate onto vehicle next to participant number
1.2.5 Inspectors decide whether or not a new full inspection is required
1.2.9 Install Joule meter
Joule meter
Qualified vehicle
Repeat till inspection is passed or time runs out
1.2.7 Determine whether there is time for another attempt
There is time
Time is up
1.2.8 Fail inspection
Disqualified vehicle
1.3 Pack up in shame
1.4 Prepare vehicle for race
Project:
Default
Organization:
Date:
11. June 2012

Figure E.2

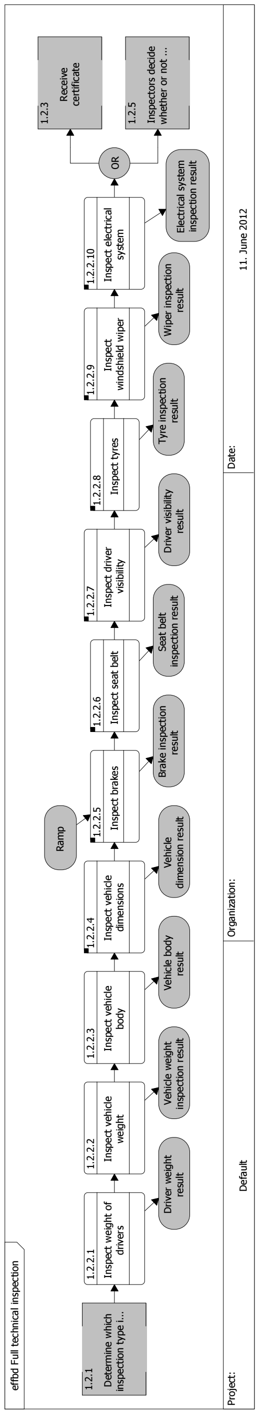

effbd Full technical inspection
1.2.1
Determine which inspection type i...
1.2.2.1
Inspect weight of drivers
Driver weight result
1.2.2.2
Inspect vehicle weight
Vehicle weight inspection result
1.2.2.3
Inspect vehicle body
Vehicle body result
1.2.2.4
Inspect vehicle dimensions
Vehicle dimension result
Ramp
1.2.2.5
Inspect brakes
Brake inspection result
1.2.2.6
Inspect seat belt
Seat belt inspection result
1.2.2.7
Inspect driver visibility
Driver visibility result
1.2.2.8
Inspect tyres
Tyre inspection result
1.2.2.9
Inspect windshield wiper
Wiper inspection result
1.2.2.10
Inspect electrical system
Electrical system inspection result
OR
1.2.3
Receive certificate
1.2.5
Inspectors decide whether or not ...
Project:
Default
Organization:
Date:
11. June 2012

Figure E.3

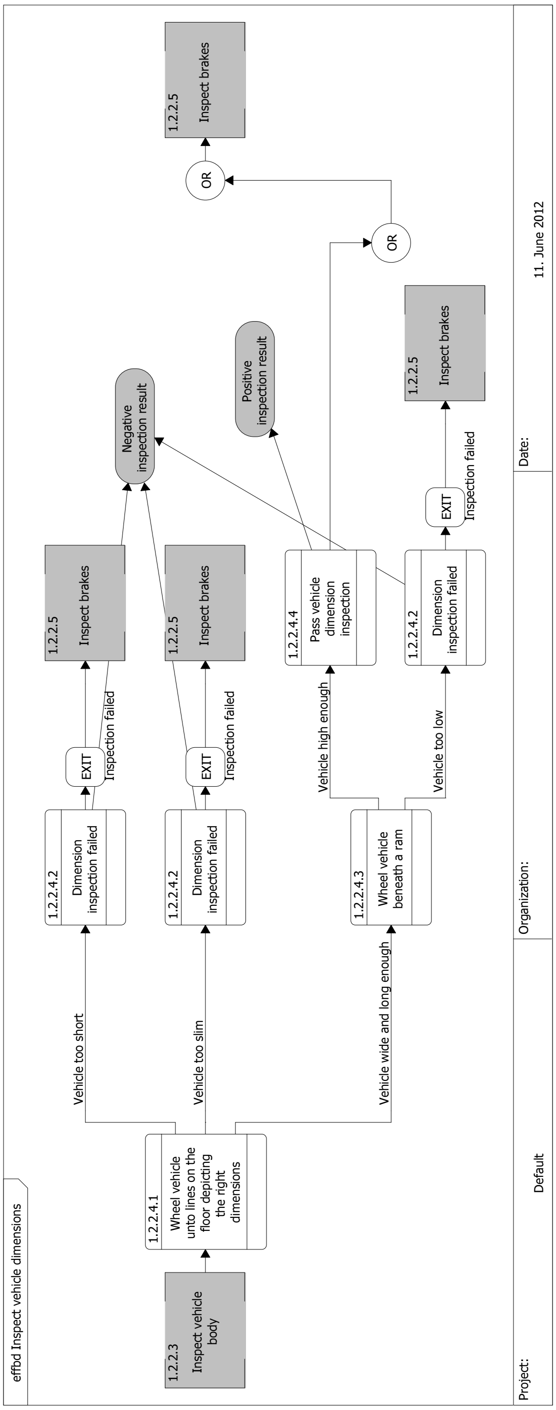

effbd Inspect vehicle dimensions
1.2.2.3
Inspect vehicle body
1.2.2.4.1
Wheel vehicle unto lines on the floor depicting the right dimensions
Vehicle too short
Vehicle too slim
Vehicle wide and long enough
1.2.2.4.2
Dimension inspection failed
1.2.2.4.2
Dimension inspection failed
1.2.2.4.3
Wheel vehicle beneath a ram
EXIT
Inspection failed
EXIT
Inspection failed
Vehicle high enough
Vehicle too low
1.2.2.5
Inspect brakes
1.2.2.5
Inspect brakes
1.2.2.4.4
Pass vehicle dimension inspection
1.2.2.4.2
Dimension inspection failed
EXIT
Inspection failed
Negative inspection result
Positive inspection result
1.2.2.5
Inspect brakes
OR
OR
1.2.2.5
Inspect brakes
Project:
Default
Organization:
Date:
11. June 2012

Figure E.4

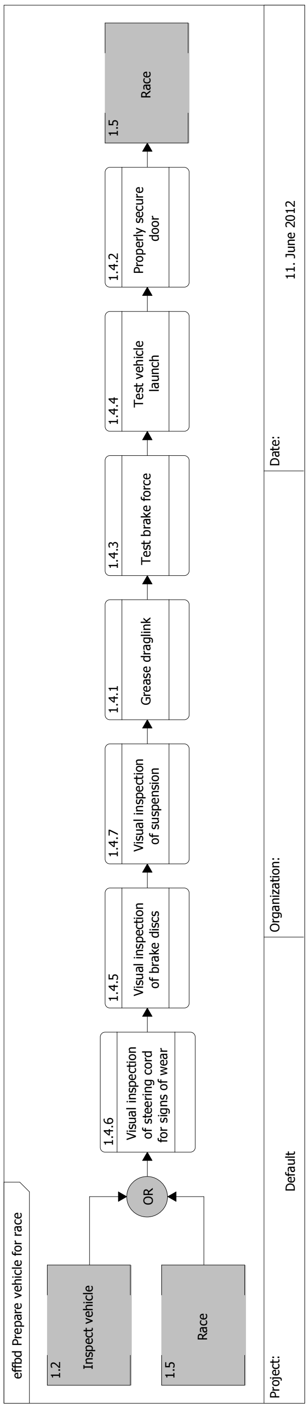

effbd Prepare vehicle for race
1.2
Inspect vehicle
1.5
Race
OR
1.4.6
Visual inspection of steering cord for signs of wear
1.4.5
Visual inspection of brake discs
1.4.7
Visual inspection of suspension
1.4.1
Grease draglink
1.4.3
Test brake force
1.4.4
Test vehicle launch
1.4.2
Properly secure door
1.5
Race
Project:
Default
Organization:
Date:
11. June 2012

Figure E.5

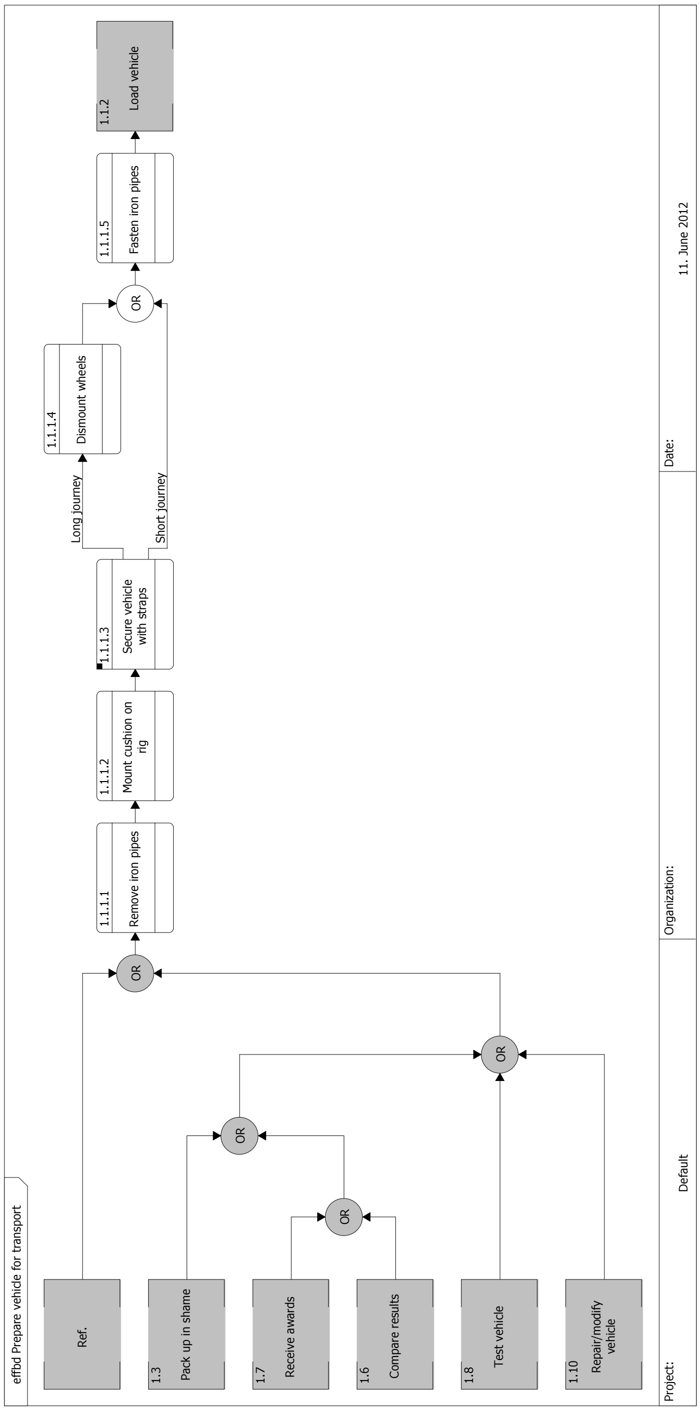

effbd Prepare vehicle for transport
Ref.
1.3 Pack up in shame
1.7 Receive awards
1.6 Compare results
1.8 Test vehicle
1.10 Repair/modify vehicle
OR
OR
OR
OR
1.1.1.1 Remove iron pipes
1.1.1.2 Mount cushion on rig
1.1.1.3 Secure vehicle with straps
Long journey
Short journey
1.1.1.4 Dismount wheels
OR
1.1.1.5 Fasten iron pipes
1.1.2 Load vehicle
Project:
Default
Organization:
Date:
11. June 2012

Figure E.6

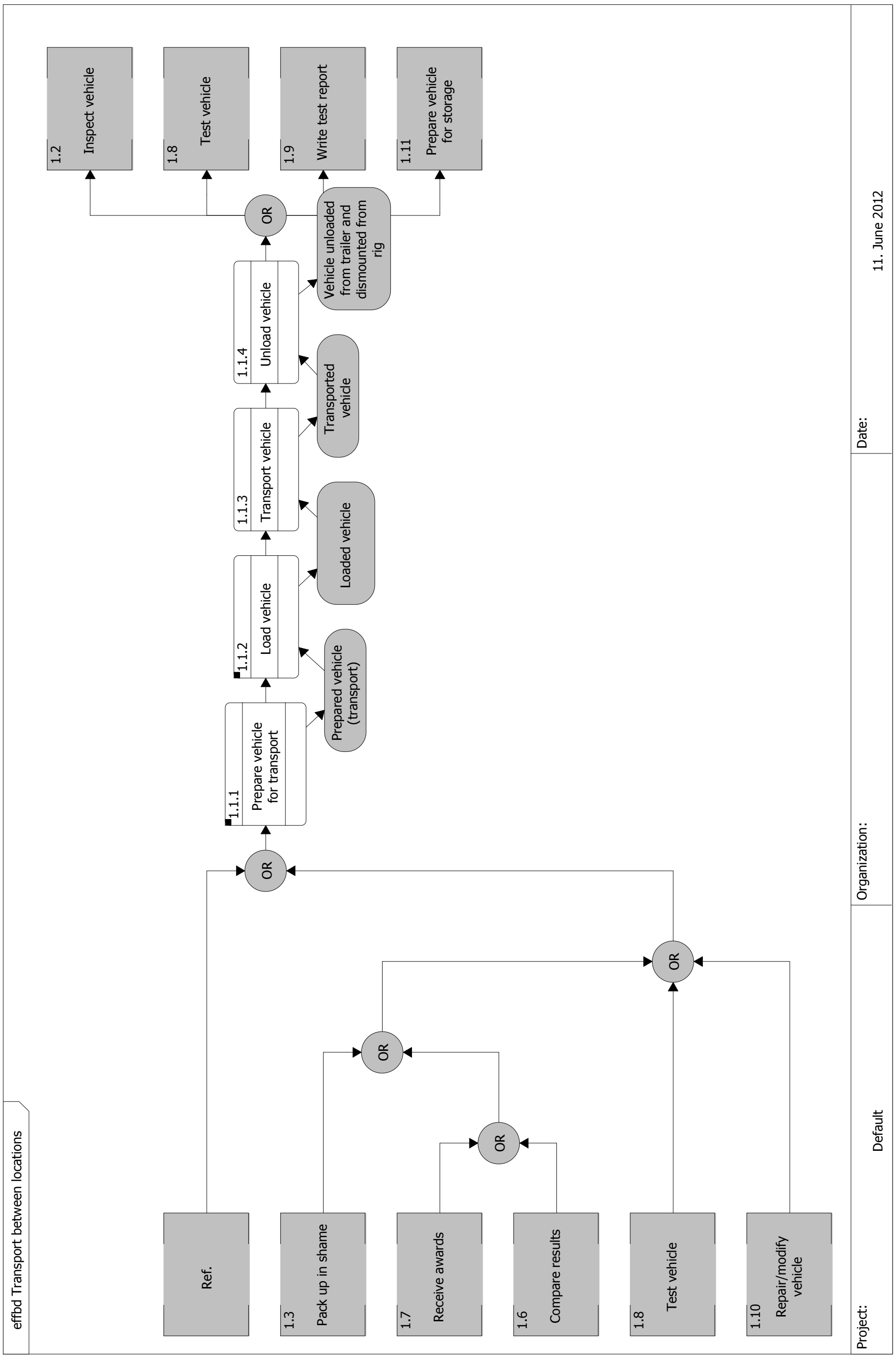

effbd Transport between locations
Ref.
1.3 Pack up in shame
1.7 Receive awards
1.6 Compare results
1.8 Test vehicle
1.10 Repair/modify vehicle
OR
1.1.1 Prepare vehicle for transport
Prepared vehicle (transport)
1.1.2 Load vehicle
Loaded vehicle
1.1.3 Transport vehicle
Transported vehicle
1.1.4 Unload vehicle
Vehicle unloaded from trailer and dismounted from rig
1.2 Inspect vehicle
1.8 Test vehicle
1.9 Write test report
1.11 Prepare vehicle for storage
Project: Default
Organization:
Date: 11. June 2012

Figure E.7

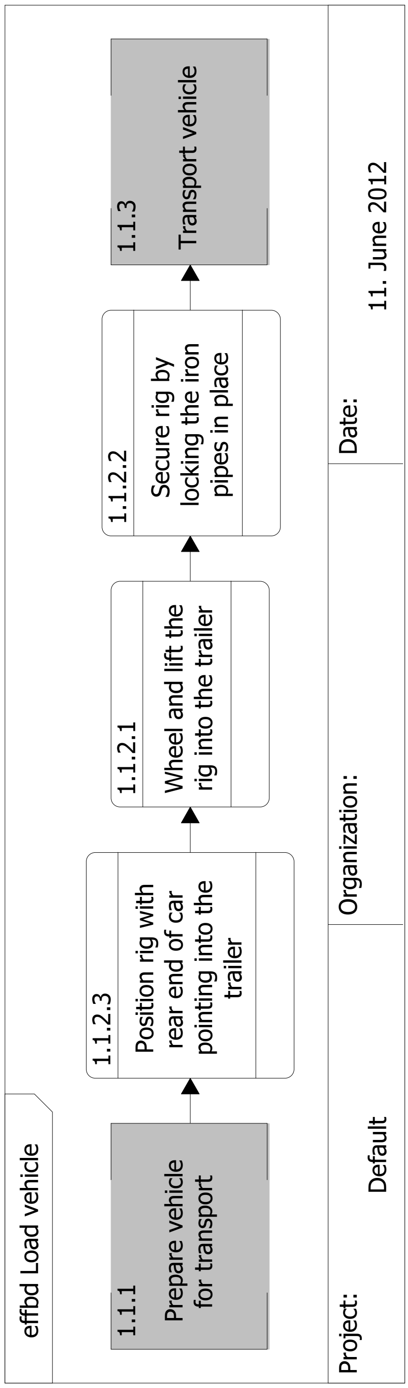

effbd Load vehicle
1.1.1
Prepare vehicle for transport
1.1.2.3
Position rig with rear end of car pointing into the trailer
1.1.2.1
Wheel and lift the rig into the trailer
1.1.2.2
Secure rig by locking the iron pipes in place
1.1.3
Transport vehicle
Project:
Default
Organization:
Date:
11. June 2012

Figure E.8

Bibliography

[1] URL: http://www.topcased.org/index.php/content/view/34/61/.

[2] Wolter J. Fabrycky Benjamin S. Blanchard. *Systems Engineering and Analysis*. Ed. by Holly Stark. 5th edition. Prentice Hall international series in industrial and systems engineering. Pearson Education Inc, Upper Saddle River, NJ, USA, 2011.

[3] Jeff A. Estefan. *Survey of Model-Based Systems Engineering Methodologies*. 2008.

[4] Sanford Friedenthal, Alan Moore, and Rick Steiner. *A Practical Guide to SysML*. Morgan Kaufmann, Waltham, MA, USA, 2012.

[5] Cecilia Haskins. "A historical perspective of MBSE with a view to the future". In: (2011), p. 1.

[6] H. P. Hildre, O. J. Mork, and V. Æsøy. 2010.

[7] Jörn Hoppmann. "The Lean Innovation Roadmap: A systematic approach to introducing Lean in product development processes and establishing a learning organization". PhD thesis. Technical University of Braunschweig, 2009.

[8] INCOSE. In: *Systems Engineering Handbook*. Ed. by Cecilia Haskins. 2007.

[9] A Kossiakoff and W. N. Sweet. In: *Systems Engineering Principles and Practice*. Ed. by Andrew Sage. John Wiley & Sons, Inc, Hoboken, New Jersey, USA, 2003.

[10] A Kossiakoff et al. "Systems Engineering Principles and Practice". In: *Systems Engineering Principles and Practice*. John Wiley & Sons, Inc, Hoboken, New Jersey, USA, 2011.

[11] J. Long. *Relationships between common graphical representations in system engineering*. Vitech Corp, 2002. URL: http://www.coinsweb.nl/wiki/ccdownloads/CommonGraphicalRepresentations_2002.pdf.

[12] Ronald Mascitelli. *Lean Product Development Guidebook*. Technology Perspectives, Northridge California, 2007.

[13] Ronald Mascitelli. *Mastering Lean Product Development*. Technology Perspectives, Northridge California, 2011.

[14] NASA. In: *Systems Engineering Handbook*. 2007. URL: http://education.ksc.nasa.gov/esmdspacegrant/Documents/NASA%20SP-2007-6105%20Rev%201%20Final%2031Dec2007.pdf.

[15] D. W. Oliver, T. P. Kelliher, and J. G. Keegan jr. In: *Engineering Complex Systems with Models and Objects*. McGraw-Hill, 1997. URL: http://www.ap233.org/ap

233-public-information/reference/BOOK_D-Oliver-Engineering-Complex-Systems.pdf.

[16] Bohdan Oppenheim. "Lean Product Development Flow". In: *Systems Engineering* 7.4 (2004).

[17] Bohdan Oppenheim, Earll Murman, and Deborah Secor. "Lean Enablers for Systems Engineering". In: *Systems Engineering* 14.1 (2011).

[18] Bohdan W. Oppenheim. *Lean for Systems Engineering; With Lean Enablers for Systems Engineering*. John Wiley & Sons, Inc, Hoboken, New Jersey, USA, 2011.

[19] *Oxford Dictionary of English*. Oxford University Press, 2005.

[20] G. Pahl and W. Beitz. In: *Engineering Design: A Systematic Approach*. Ed. by Ken Wallace. Springer-Verlag London Limited, 1996.

[21] Shell. *About the Shell Eco-marathon*. URL: http://www.shell.com/home/content/ecomarathon/.

[22] Shell. *History of the Shell Eco-marathon competition*. URL: http://www.shell.com/home/content/ecomarathon/about/history/.

[23] R. Stevens et al. In: *Systems Engineering Coping with Complexity*. Prentice Hall Europe, 1998.

[24] Oluf Tonning. *Common Model-based Platform*. Unpublished project work for the Institute of Engineering Design and Materials, NTNU, Norway, 2011.