



HAL
open science

Model-based User Experience Design and Roadmapping. Application to a Smart Autonomous Vehicle Cockpit

Ilia Iuskevich

► **To cite this version:**

Ilia Iuskevich. Model-based User Experience Design and Roadmapping. Application to a Smart Autonomous Vehicle Cockpit. Automatic. Université Paris-Saclay, 2022. English. NNT : 2022UP-AST057 . tel-03735643

HAL Id: tel-03735643

<https://theses.hal.science/tel-03735643v1>

Submitted on 21 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Model-based User Experience Design and
Roadmapping. Application to a Smart
Autonomous Vehicle Cockpit
*Conception et roadmapping de l'expérience utilisateur
basées sur les modèles. Application au cockpit intelligent
du véhicule autonome*

Thèse de doctorat de l'université Paris-Saclay

École doctorale n°573 : interfaces : matériaux, systèmes, usages (INTERFACES)
Spécialité de doctorat: Ingénierie des Systèmes Complexes
Graduate School : Sciences de l'ingénierie et des systèmes,
Réfèrent : CentraleSupélec

Thèse préparée dans le Laboratoire Génie Industriel (Université Paris-Saclay, CentraleSupélec)
et l'Institut de Recherche Technologique (IRT) SystemX,
sous la direction de Marija JANKOVIC, professeure,
le co-encadrement de Andreas HEIN, professeur,
et la co-supervision de Abdelkrim DOUFENE

Thèse soutenue à Paris-Saclay, le 12 avril 2022, par

Iliia IUSKEVICH

Composition du jury

Emilie Poirson Professeure, Ecole Centrale de Nantes	Président du jury, examinatrice
Eric Bonjour Professeur, ENSGSI - Université de Lorraine	Rapporteur & examinateur
Olivier de Weck Professor, Massachusetts Institute of Technology	Rapporteur & examinateur
Bernard Yannou Professeur, Université Paris-Saclay, CentraleSupélec	Examinateur
Marija Jankovic Professeure, Université Paris-Saclay, CentraleSupélec	Directrice de thèse
Andreas Hein Associate professor, Université du Luxembourg	Co-encadrant
Abdelkrim Doufene Programmes and Strategy Director, IRT SystemX	Tuteur industriel

Title: Model-based User Experience Design and Roadmapping. Application to a Smart Autonomous Vehicle Cockpit

Keywords: Strategic Planning, Technology Roadmapping, Model-Based Systems Engineering, Model-Driven Design, User Experience Design

Abstract: The concept of User experience (UX) includes all aspects, responses, and consequences of using a product or service. The process of designing the UX, UX design, is one of the most important undertakings of the user-oriented business, as a product that offers the best UX on the market is doomed to success. Yet such an activity is also one of the most challenging due to the complex and ambiguous nature of the UX. UX concerns multiple levels of human perception and cognition: a product should comply with high and low-level goals of a user, be desirable, usable, and aesthetically appealing.

The complexity of the UX concept grows drastically when the usage context is taken into consideration. The interaction happens in different cultural and climatical conditions, and the number of situations (use cases and scenarios) in which this interaction occurs is virtually infinite. To develop a plan which respects the complexity and ambiguity of UX, the successful strategic planning methodology focusing on users should marry two conflicting poles: the high level of flexibility with careful planning and comprehensiveness.

One possible solution was proposed under the label of design roadmapping. The essence of design roadmapping is to combine technology and product roadmaps with a roadmap accountable for UX concepts planning. The promise of design roadmapping is to harmonize the efforts of diverse teams by providing a robust yet flexible plan that rather sets goals and priorities than deadlines.

Design roadmapping, however, suggests little on how to address the UX complexity while developing plans. This thesis aims to propose a solution for this problem by complementing design roadmapping with the recently proposed model-based roadmapping. The latter aims to digitize roadmaps by defining a formal syntax and se-

mantics for concepts of planning. The use of models in the roadmapping brings flexibility and reduces roadmap creators' workload by automating routine tasks, providing advanced visual interfaces, and improving integration with user testing data flow.

First, this research work contributes to the technology roadmapping domain by proposing a metamodel of a roadmap. This metamodel defines a roadmap independently from the application context and provides a generic structural pattern for tools supporting roadmapping. This metamodel is designed as the result of inductive analysis of the roadmaps published in the scientific literature and validated by representing three sufficiently different roadmaps.

Within the second contribution, we designed a Domain-specific language (DSL) for UX-focused roadmapping that enables an asynchronous, collaborative, and iterative roadmap development process. We propose a simple unified syntax based on feature modeling language with a number of extensions to represent four heterogeneous levels of a roadmap, namely marketing, UX design, engineering design, and technology. Proposed DSL has flexible semantics (user can establish own taxonomy of concepts). It is also executable, i.e., changes made in one part of a model propagate to the other parts of a model, which reflects correctly and facilitates the process of cross-domain decision making.

In the third contribution, we complement our model-based approach to user experience design and roadmapping with various tools for user testing, which enables evidence-based strategy elaboration.

The proposed approach is embodied into a cloud-hosted interactive application, validated together with industrial partners, and illustrated by the case study of strategic planning in the automotive context.

Titre: Conception et roadmapping de l'expérience utilisateur basées sur les modèles. Application au cockpit intelligent du véhicule autonome

Mots clés: Planification stratégique, Roadmapping technologique, Ingénierie des systèmes basée sur les modèles, Ingénierie dirigée par les modèles, Conception centrée sur l'utilisateur

Résumé: Le concept d'expérience utilisateur (UX) comprend tous les aspects, les réponses et les conséquences de l'utilisation d'un produit ou d'un service. Le processus de conception de l'UX, le design UX, est l'une des initiatives les plus importantes de tout business orienté utilisateur, car un produit qui offre le meilleur UX du marché est voué au succès.

L'UX concerne de multiples niveaux de perception et de cognition humaines: un produit doit répondre aux objectifs de haut et de bas niveau d'un utilisateur, être désirable, utilisable et esthétiquement attrayant. La complexité du concept de l'UX augmente considérablement lorsque le contexte est pris en considération.

Pour développer un plan qui respecte la complexité et l'ambiguïté de l'UX, la méthodologie réussie de planification stratégique centrée sur les utilisateurs, doit marier deux pôles contradictoires: un haut niveau de flexibilité, et une planification minutieuse et exhaustive. Une des solutions proposées dans la littérature est le "design roadmapping".

Le principe du design roadmapping consiste à combiner les feuilles de route de la technologie et du produit avec le troisième niveau responsable de la planification du concept UX. La promesse du design roadmapping est d'harmoniser les efforts de diverses équipes en fournissant un plan robuste, mais flexible, qui fixe plutôt des objectifs et des priorités que des délais. Cependant, le design roadmapping ne donne que peu d'indications pour faire face à la complexité de l'UX.

Dans cette thèse, nous proposons une approche de conception et de roadmapping en combinant le design roadmapping avec le roadmapping à base de modèles. Cette dernière vise à numériser les feuilles de route en définissant une syntaxe et une sémantique formelles pour les concepts de planification.

L'utilisation de modèles dans le roadmapping apporte de la flexibilité et réduit la charge de travail des créateurs des roadmaps en automatisant les tâches de routine, en fournissant des interfaces visuelles avancées et en améliorant l'intégration avec le flux de données des tests utilisateurs.

Comme première contribution, ce travail de recherche contribue au domaine du roadmapping (feuille de route) technologique en proposant un méta-modèle de roadmapping. Ce dernier définit une feuille de route indépendamment du contexte applicatif et fournit un modèle structurel générique pour les outils supportant le roadmapping. Ce méta-modèle est conçu comme le résultat d'une analyse de travaux antérieurs publiés dans la littérature scientifique et validé par la représentation de trois roadmaps suffisamment différentes.

Dans la deuxième contribution, nous avons conçu un langage spécifique au domaine (Domain-specific Language) pour le roadmapping centré sur l'UX et un processus permettant un processus asynchrone, collaboratif et itératif de développement de roadmaps. Nous proposons une syntaxe simple et unifiée basée sur le langage de modélisation des caractéristiques avec un certain nombre d'extensions pour représenter quatre niveaux hétérogènes d'une feuille de route, à savoir le marketing, la conception UX, la conception technique et la technologie. Le DSL proposé a une sémantique flexible (l'utilisateur peut établir sa propre taxonomie de concepts). Il est également exécutable, c'est-à-dire que les modifications apportées à une partie du modèle se propagent aux autres parties du modèle, ce qui reflète correctement et facilite le processus de prise de décision inter-domaines.

L'approche proposée est intégrée dans une application interactive hébergée dans le cloud et appliquée à l'étude de cas de la planification stratégique dans une entreprise automobile.

To my beloved Grecia

“One key part of problem solving is the language we use.”

Alan Cooper, *The Inmates Are Running the Asylum:
Why High Tech Products Drive Us Crazy
and How to Restore the Sanity*

List of abbreviations

ADS — Automated Driving System
AHP — Analytic Hierarchy Process,
APF — Architecture of Product Family
AST — Abstract Syntax Tree
CA — Conjoint Analysis
CDL — Component Definition Language
CMI — Cockpit Multimodal Interactif
DS I — Descriptive Study I (stage of DRM)
DS II — Descriptive Study II (stage of DRM)
DSL — Domain-Specific Language
DT — Design Thinking
DFMC — Design For Mass Customization
DRM — Design Research Methodology
DSM — Design Structure Matrix
FODA — Feature-Oriented Domain Analysis
GPS — Generic Product Structure
HSI — Human Systems Integration
GRL — Goal-oriented Requirement Language
ICED — International Conference on Engineering Design
INCOSE — International Council on Systems Engineering
KE — Kansei Engineering
NFR — Non-Functional Requirements
NPD — New Product Development
MBSE — Model-Based Systems Engineering
MB-TRM — Model-Based Technology Roadmapping
MDA — Model-Driven Architecture
MDE — Model-Driven Engineering
MOF — Meta-Object Facility
MPS — Meta Programming System

TRM — Technology Roadmapping
TF&SC — Technological Forecasting and Social Change (journal)
OMG — Object Management Group
OPM — Object Process Methodology
PF — Product Family
PS — Prescriptive Study (stage of DRM)
PFA — Product Family Architecture
R&D — Research and Development
QFD — Quality Function Deployment
RC — Research Clarification (stage of DRM)
RE — Requirements Engineering
SysML — Systems Modeling Language
UML — Unified Modeling Language
UX — User Experience
UXD — User Experience Design

Contents

List of abbreviations	6
1 INTRODUCTION	11
1.1 Complexity in the New Product Planning	14
1.2 Functions and challenges of the technology roadmapping	15
1.3 Model-driven Engineering (MDE)	20
1.4 Contributions and previously published works	24
2 RESEARCH SCOPE	26
2.1 Industrial context	26
2.2 Objectives and success criteria	27
3 LITERATURE REVIEW	30
3.1 Technology roadmapping	30
3.1.1 Major research streams in the technology roadmapping	30
3.1.2 Model-based Technology Roadmapping	33
3.2 Conceptual models, taxonomies and definitions	35
3.2.1 Marketing	36
3.2.2 User experience design	38
3.2.3 Engineering design	40
3.2.4 Technologies and Resources	44
3.3 Model-driven Engineering	46
3.3.1 Metamodeling	46
3.3.2 Domain-specific languages	49
3.4 Agile approaches to product design and innovation	53
3.5 Synthesis and research gaps identification	56
4 RESEARCH METHODOLOGY	59
4.1 Design research as a science	59
4.2 Design Research Methodology	61
4.3 Research methods and validation	63
4.4 Design theory classification: process view versus product view	64
5 METAMODEL OF THE MODEL-BASED TECHNOLOGY ROADMAPPING	66
5.1 Introduction	66
5.2 Related literature	67
5.3 Research methodology	68
5.4 The metamodel	71
5.5 Validation	74
5.6 Discussion	81

6	MODEL-BASED USER EXPERIENCE DESIGN AND ROADMAPING FRAMEWORK	83
6.1	Introduction	83
6.2	Related literature	84
6.3	Model-based UXD and Roadmapping	84
6.4	The process	92
6.5	Case study	93
6.5.1	General description	93
6.5.2	Analysis	96
6.6	Discussion	97
7	DATA-DRIVEN DESIGN ROADMAPING	109
7.1	Introduction	109
7.2	Related literature	110
7.3	MB-TRM integrated with user testing	110
7.4	Case study: Cockpit Multimodal Interactive for semi-autonomous vehicle	111
7.4.1	Design of experiement	111
7.4.2	Results	114
7.5	Discussion	115
8	DISCUSSION	121
8.1	Contribution	121
8.2	Limitations and weak points	123
8.2.1	Major limitations	123
8.2.2	Issues of implementation in industrial settings	123
8.2.3	Relation to the commercially available tools	124
8.2.4	Directions for future work	125
9	CONCLUSIONS	126
	BIBLIOGRAPHY	146
	Appendices	147
A	SYNTHÈSE	148
B	METAMODEL IMPLEMENTATION	151
C	CASE STUDY 1.1: NASA ROADMAP: DSL (M1)	158
D	CASE STUDY 1.1: NASA ROADMAP: User Data (M0)	164
E	CASE STUDY 1.2: DESIGN ROADMAP: DSL (M1)	167
F	CASE STUDY 1.2: DESIGN: USER DATA (M0)	172
G	CASE STUDY 1.3: SCENARIO-BASED ROADMAP: DSL (M1)	174

H	CASE STUDY 1.3: SCENARIO-BASED ROADMAP: USER DATA (M0)	177
I	USER STORY MAP OF THE SUPPORT TOOL	179
J	CASE STUDY 3.1: CMI USER TESTING	180
K	MODEL FOR DISCRETE-EVENT SIMULATION FOR DRIVER PERFORMANCE ASSESSMENT	185
K.1	Introduction	185
K.2	Background	187
K.2.1	Mental workload modeling	187
K.2.2	Eyes-off-the-road time estimation	188
K.2.3	Cognitive simulation models	188
K.3	Discrete-event simulation model architecture	189
K.4	Results	193
K.5	Conclusions and future work	195

1 INTRODUCTION

When someone designs a product, he faces a problem belonging to two completely different realms. The most apparent one is the design of a product itself, which includes the creation of a product's shape (visual design, materials selection, etc.), the design of the product's architecture and functions (software and hardware engineering), and the design of a production process (selecting technologies and components, manufacturing planning). The second realm concerns not an artifact but people that will use this artifact. The attention of specialists in marketing, user experience design, customer development, program and product managers (from now on referred to as a product team) is directed not to their company's products, but outwards, to the external environment, i.e., to customers and trends, current and prospective users, partners and competitors.

Being overlooked in the past (a product manager's role dates back to the 1930s, the first mention of user-centered design — to 1970s), nowadays the importance of the product team is highly appreciated. The majority of products we are using presently succeed due to the superior product vision, proper positioning, and product innovations. Companies based solely on the technology innovations are at risk to be substituted by more usable and well-adapted alternatives or becoming a component supplier.

This dissertation is devoted to the problem of *new product development (NPD) strategic planning*. A *product* is something sold by an enterprise to its customers (Eppinger and Ulrich 2015). To be sold, a product needs to deliver some benefits and/or a pleasant experience to the end-users. *NPD* is "the set of activities beginning with the perception of a market opportunity and ending in the production, sale, and delivery of a product" (Eppinger and Ulrich 2015). In this definition, we would emphasize that *NPD* process focuses not only on the engineering of products but also on exploring users and their needs. Finally, *strategic planning* is "a systematic process of identifying opportunities and threats in the future environment and of formulating policies, based on organizational resources and goals for operations in that environment over a relatively long term" (Steiner 1972).

The most popular approach for a strategic planning in enterprises is *technology roadmapping (TRM)*. A technology roadmap is a "vision of the future in a specific field based

on the collective opinion of important variables" (Lu, Chen, and Yu 2019). The typical format for technology roadmap is a "multiple-layered chart including market, product and technology layers" (Geum et al. 2015a).

Accordingly, *TRM* is the process of creating the technology roadmap. TRM is an informationally intensive interdisciplinary process (in details the challenges of TRM will be discussed in sections 1.2 and 3.1). These properties justify the need for *methodologies and digital tools* supporting the TRM. This is the **first aspect** of the global problem we are addressing in this Ph.D.

The **second aspect** concerns *user experience (UX)*. The UX concept includes "all the aspects of how people use an interactive product: the way it feels in their hands, how well they understand how it works, how they feel about it while they're using it, how well it serves their purposes, and how well it fits into the entire context in which they are using it" (Alben 1996). UX is a core concept for planning in enterprises focusing on the end-user value (Kim, Beckman, and Agogino 2018). Compared to planning of systems and technologies, UX planning is much more challenging. As any human-related concept, UX is *indeterministic, subjective, quantitative, and ambiguous* (in details these issues are discussed in section 1.1). To give an example, the program manager of such a recent UX-focused undertaking as Metaverse needs to analyze the following aspects of UX to formulate product, marketing, and technology strategy (just to name a few):

- How do people with different religious and political beliefs perceive the concept of virtual reality (VR)?
- Which of the VR use cases are the most beneficial and appealing to different users?
- How could virtual reality change society?
- Which emotions/physiological responses do people experience in VR (joy, excitement, motion sickness, fear)?

Answering these questions is vital for prioritizing product features, planning the development of new human-machine interfaces and other related technologies, designing marketing campaigns. The problem is that these questions have no single determined and quantifiable answer. Also, these answers are changing in time.

Therefore, the *overarching problem* we are addressing is how to **manage and diminish the ambiguity and uncertainty in the strategic planning of complex user-centered NPD**.

Traditionally, the NPD processes and tools are analyzed within the design science domain, and the planning of NPD — in the technology management domain. In this research, we are trying to consider this problem from a new perspective, i.e., *model-based systems engineering (MBSE)* perspective. MBSE implies a description of the domain of interest using some standard or ad-hoc *Domain-specific Language (DSL)*.

Representing the UX in a DSL will not make the vague and subjective problems more determined and objective. However, modern informational systems can effectively store and process large amounts of data and represent it visually and interactively. We assume that this will *simplify decision-making and promote communication efficiency*.

We aim to develop our own DSL for roadmapping with desired properties. The aim determines our research methodology. To design the DSL for the target problem of the NPD planning, we will first propose *a metamodel that describes any DSL for roadmapping at the abstract level*. For that, we will carefully review roadmapping literature to define common elements: entities, links, attributes, formats. Then, we will at first review the literature studying each separate level of our roadmap, namely marketing, user experience design (UXD), engineering design, and technology. After that, we will *propose a DSL's concrete syntax and semantics*.

To validate our technology roadmap metamodel, we adapted the example-driven approach to metamodel validation (López-Fernández et al. 2015). We selected three publicly available roadmaps belonging to significantly different areas: a roadmap of a space agency, a roadmap of a small user-oriented startup, and a scenario-based roadmap of a company entering a highly competitive market, and formally proved their instantiability from our proposed metamodel.

For the validation of the DSL, we embodied our methodology into a cloud-based collaborative web application and implemented a case study of an automotive company designing a product family of vehicles with reduced CO2 emissions. We also conducted a validation of the approach with our industrial partner on their NPD case, but this information is rather sensitive and we do not have rights to publish it.

We further extended our approach by integrating user testing procedures into our proposed framework and tested this integration with real users.

In the next sections of the introduction, we will outline the set of global problems existing in the new product development, technology management, and model-based design. Section 1.1 presents the challenges typically encountered during NPD planning. Section 1.2 discusses advantages and problems of technology roadmapping practice. Section 1.3 introduces model-based design advantages and pitfalls.

This thesis manuscript can be of interest to product managers, product owners, UX designers, executives, strategists, regular participants of roadmapping sessions, or scholars in the fields of technology management and design science.

1.1 Complexity in the New Product Planning

Companies that provide products and services directly to customers exist in a complicated environment. In the literature it is known as VUCA-environment (Volatile, Uncertain, Complex, and Ambiguous) (Bennis and Nanus 1986). To survive and develop progressively, such companies should adapt their product line to diverse and constantly changing user needs, differentiate their offer from competing, take into account and anticipate policy changes, the evolution of old or the emergence of new trends and technologies.

In some cases, the nature of this type of business is connected to the design and maintenance of the complex systems, which lifts up the degree of complexity to the next level. Development, management, and planning associated with complex systems naturally contain risks intrinsic to all complex structures. This set of problems is traditionally studied in the systems engineering realm.

Automotive companies are prominent examples of such businesses. On the one hand, they deal with complex systems composed of many components and technologies. On the other, their domain is highly competitive, and their success links tightly to how well they serve their customers.

Typically these organizations are extremely large. Their structure joins thousands of people organized into teams, business units, legal entities, and partnerships. The structure that comprises all companies collaborating to achieve a common goal (e.g., the OEM, its partners, investors, policy-makers, and the supply chain members) is often entitled to an extended enterprise (Post, Preston, and Sachs 2002).

Figure 1.1 provides a classification of products based on external (environmental) and

internal (of a product itself) complexity affecting planning in companies.

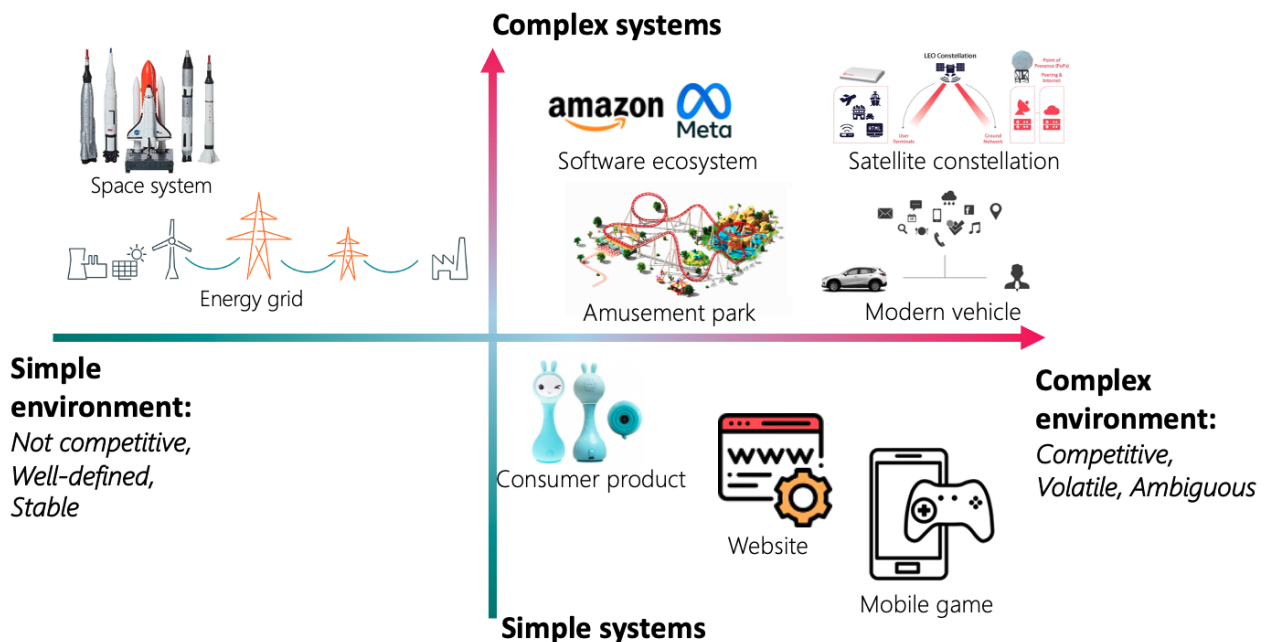


Figure 1.1: Complexity dimensions: internal (inherent to systems) and external (inherent to the VUCA-environment)

These two sources of complexity: inherent in systems, and inherent in the environment, result in the need for efficient tools for strategic communication and coordination. These are the functions of technology roadmapping that we will discuss in the next section.

1.2 Functions and challenges of the technology roadmapping

Technology roadmapping (TRM) is one of the most convenient approaches for developing and communicating long-term plans. TRM is a process producing a technology roadmap, which Phaal, Farrukh, and Probert (2004) define as a "time-based chart, comprising a number of layers that typically include both commercial and technological perspectives." In practice, roadmaps serve the internal and external communication of plans, decisions, and responsibilities and help to focus on long-term goals of high-priority (Albright and Kappel 2003).

A typical example of TRM is the *2020 NASA Technology Taxonomy (2019)*, dealing with incredibly complex, risky, and expensive projects, which, however, exist in a financially

stable, low-competitive environment (see Fig. 1.1). Under such conditions, planning is possible for years and even decades ahead.

Originating from large technological enterprise practices, TRM now encompasses various applications and organizational contexts. Phaal, Farrukh, and Probert (2004) distinguish eight classes of roadmaps by the object of planning, such as products, capabilities, strategies, long-range goals, knowledge assets, programs, processes, and operations of complex system integration. Kim et al. (2016) introduce design roadmapping and provide an example of roadmapping applied to a single-product user-experience-focused startup.

Figure 1.2 depicts structure and information flows in a design roadmap. In their design roadmapping proposal, Kim et al. (2016) and Kim, Beckman, and Agogino (2018) stressed the importance of building strategic plans around UX design (UXD), as the UX is the most influential factor explaining the success or failure of products. Design roadmapping differs from traditional TRM by the presence of the corresponding UXD layer, describing organization’s plans related to the UX concepts (e.g., use cases the future products will offer, personas the organization is planning to target, key UX-centered metrics it aims to achieve).

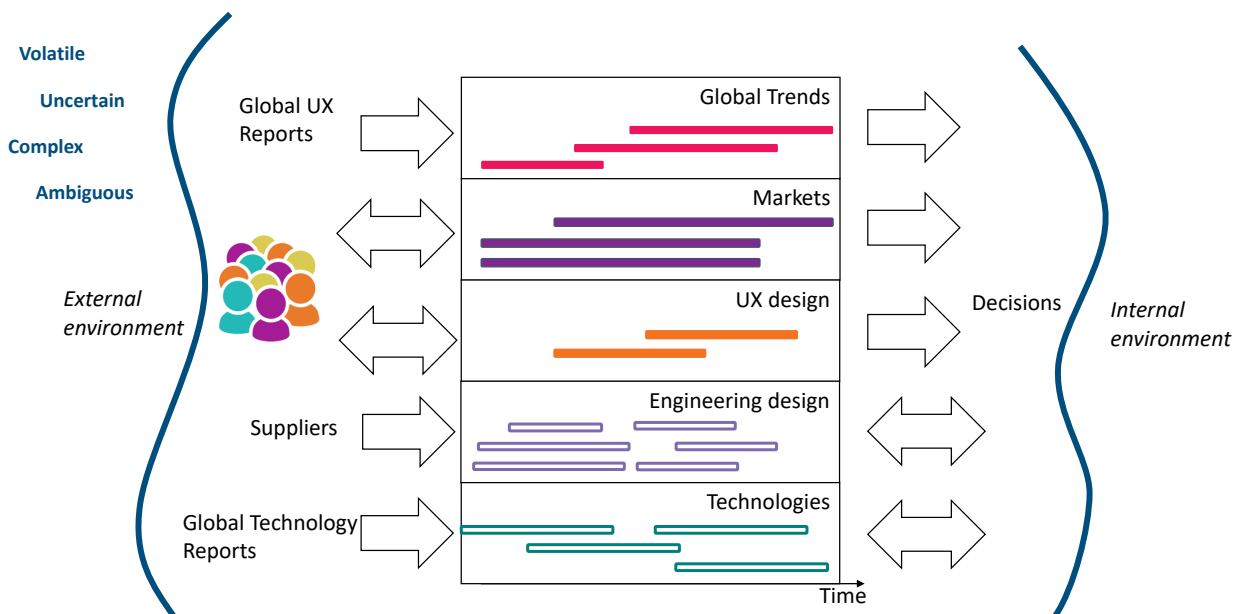


Figure 1.2: Structure and information flows in design roadmapping

Here it is worth articulating the difference between the marketing and UXD layers. Both marketers and UXD specialists are responsible for product success; however, the former usually think of people in large numbers (i.e., in terms of the total addressable market). They rather consider profiles than personas. The job of the latter is to study concrete individuals. For example, a particular woman Emma of 43 years old, living in a small town near Paris that a UX designer met and interviewed in person, for marketer falls into the category of french female 40-50 years old. The other significant difference is that UX designers study interaction with a product as an enduring scenario happening in a particular context (e.g., Emma just woke up at 7 am, it is 5 degrees outside, she prepares her morning coffee, etc.) and create so-called user journeys. At the same time, marketers focus more on marketing channels. Dove, Reinach, and Kwan (2016) described a representative case of how differently marketers and UX designers approach the same problem.

The representation of roadmap on figure 1.2 shares the metaphor proposed by Phaal and Muller (2009) that sees a technology roadmap as a strategic lens through which "a complex system (such as a business) can be viewed." We would add that this lens helps not only to see the strategy of a company but also helps to transform the information coming from the external business environment (from general trends, users, competitors, and suppliers) into the strategic decisions communicated internally.

In practice, roadmaps can be classified by the purpose for external and internal communication. Roadmaps for external communication can be shared with partners to orchestrate common activities, investors to attract resources, etc. Internal roadmaps communicate goals, priorities, timelines, and the vision. Figure 1.3 shows roadmap as a strategic prism, across the edges of which outsiders can see a bit of internal plans, insiders — the vision on the evolution of the external business environment, and executive managers see both reflections.

Certainly, the information visible to different parties is different. Companies prefer to hide sensitive data and negative scenarios from outsiders; engineers benefit more from seeing short-term but rather detailed information, while executives choose to see the complete picture plus alternative strategies, but with a higher level of granularity.

Traditionally, the roadmaps serving different purposes are built independently, often by various departments. Further, we will show that model-based technology roadmapping (MB-TRM) enables a holistic approach when all related information is consol-

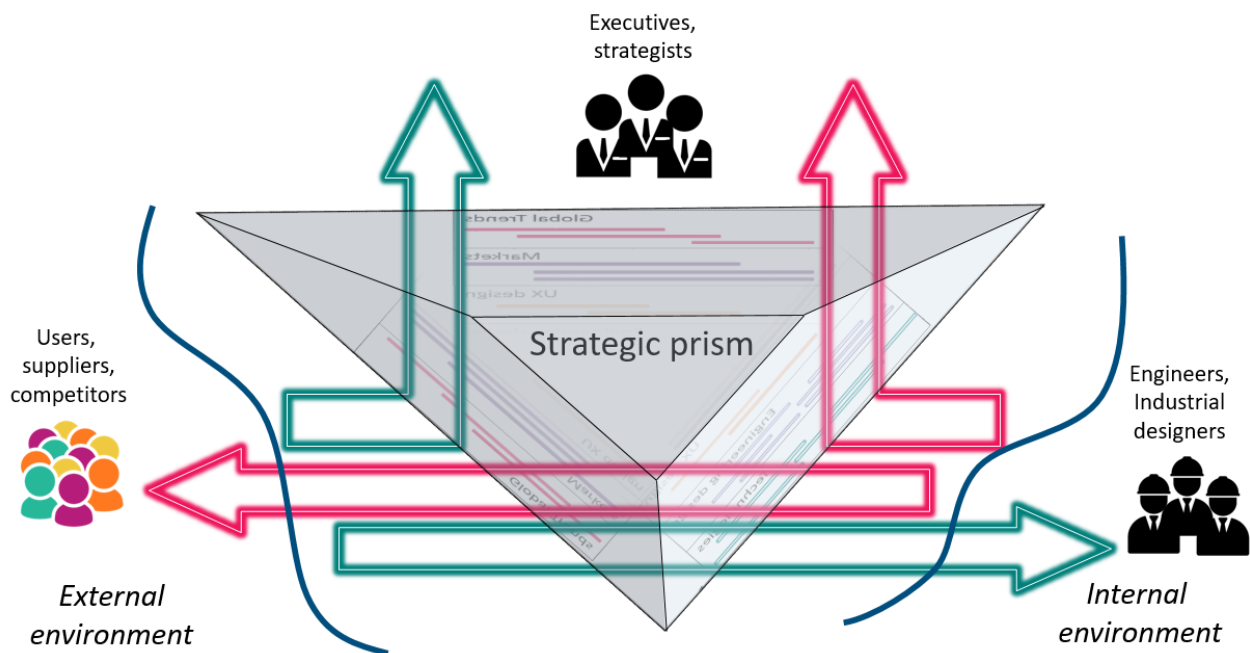


Figure 1.3: TRM as a strategic prism

dated but displayed to stakeholders depending on their goals and permissions.

TRM was practiced for over 40 years and extensively studied in academia. Several drawbacks were identified during this exploration.

Kim et al. (2016) and Kim, Beckman, and Agogino (2018) identified several challenges in current roadmapping practices for NPD through a series of semi-structured interviews with practitioners:

- Even though one of the main functions of roadmapping is the communication of plans, there is still a conflict, especially when the communication happens between multidisciplinary teams (if, for example, they use different definitions);
- TRM lacks a feedback loop from final users of a product, so the focus on creating the value for a customer is not followed;
- Lack of flexibility of a roadmap in the fast-changing environment; necessity to keep roadmap 'alive' leads to a request of an agile and iterative process;
- Inability to predict the future, hence often the "plan is not followed."

McMillan (2003) described their own experience and brought up several practical lessons

learned in Rockwell Automation:

- Senior management must demonstrate a commitment to the process;
- Common definitions should be established for elements of a roadmap, but the process should be flexible enough to re-define these elements;
- Business units should at some point "endorse and own the process" of roadmapping;
- Some business units perceive roadmapping as a "mandated exercise to be completed on the command of management" because this process's benefits were not well understood.

Gradini et al. (2019) introduced a notion of static document-based TRM (as opposed to the interactive model-based or computer-based TRM) and summarized a number of limitations of this traditional process:

- Document-based TRMs require "manual rework" in response to changes in the environment;
- Large roadmaps with many interdependencies "are prone to inconsistencies";
- TRM shall be concise (sometimes a single-page) document; however, the amount of underlying information is often enormous, which is overwhelming for a human analyst;
- TRMs "do not allow immediate answers to transversal questions, such as assessing the number of projects impacting a given technology";
- Due to the lack of automation, document-based TRMs are hard to maintain, especially in case of inconsistencies;
- The number of design alternatives and scenarios under consideration is limited due to the static nature of the documents.

In this research work, the two following problems which repeatedly appear in the literature are of particular interest. First, the future of socio-economic systems is hardly predictable. Hence, there is a need to keep a roadmap alive, or, as Gerdsri et al. (2019)

put it, 'get maintained.' This is especially true for the VUCA setting (Kim, Beckman, and Agogino 2018). The environment is too turbulent, so the future is simply impossible to foresee and forecast. So how would one plan something long-term if what will happen in one year is entirely unknown?

One answer to this seeming paradox is scenario planning: we should consider several alternative future variants. Scenario-based roadmapping is one of the most prominent research directions in the field (Saritas and Aylen 2010; Geum, Lee, and Park 2014; Cheng et al. 2016). The growing number of researchers (Phaal, Simonse, and Ouden 2008; Kim, Beckman, and Agogino 2018; Al-Ali and Phaal 2019) also highlight the other approach: roadmap should be developed in an agile manner to be quickly updated in response to new inputs. In this way, previous iterations can be obsolete, but the roadmap's newest version always represents an up-to-date vision. The open question here is how to enable agile roadmapping, as the traditional T-plan process (Phaal et al. 2003) implies time-consuming multidisciplinary workshops with key stakeholders, which are not so easy to gather together?

The other significant drawback is that roadmaps, done at the coarse-grained strategic level, are often detached from the fine-grained tactical level of the product and engineering design. Kim et al. (2016) and Kim, Beckman, and Agogino (2018) enriched TRM with UXD layer supplying decision-makers with the "ground truth" of various specialists doing user research. Indeed, design roadmapping helps to focus on the end-user value. However, traditional document-based roadmaps can capture only the most critical insights and user requirements, abstracting out the details due to space constraints (in the paper format, a roadmap should be concise). Also, the coherence between different requirements is left to the people's common sense and consensus among stakeholders.

Model-based technology roadmapping (MB-TRM) proposed by Knoll, Golkar, and Weck (2018) and further developed by Gradini et al. (2019), Golkar and Garzaniti (2020), and Breckel et al. (2021) seems to resolve at least partially some of the aforementioned difficulties.

1.3 Model-driven Engineering (MDE)

According to INCOSE (2007), MBSE is "the formalized application of modeling to support system requirements, design, analysis, verification and validation activities be-

ginning in the conceptual design phase and continuing throughout development and later life cycle phases". In simple words, MBSE is a practice of designing systems using models, i.e., some simplified representations of reality. Often the practice of model-based engineering is described as a major paradigm shift relative to traditional document-based engineering.

This section will first provide a historical overview of the evolution of MDE and MBSE, then discuss pitfalls of these paradigms and finally describe a vision of how MDE/MBSE can be applied to the TRM.

MBSE is developed for and applied in the complex hardware systems design (such as aerospace systems and automotive systems) but rooted in the other paradigm change that happened earlier in software systems design. This shift is known under the label of Model-Driven Engineering (MDE).

Historically, the significant shifts in computer systems design happened with the level of abstraction raise (Mellor 2004). First, the low-level programming languages (such as assembly language) abstracted out the machine code. Then, higher-level object-oriented programming languages (such as C and Java) abstracted out assembly code. After that, more specialized languages appeared and abstracted out essential functions of networking, data processing, and visual rendering (think of browsers written primarily on C++ but operating with JavaScript, HTML, and CSS). Modern software frameworks and libraries lift abstraction further, providing ready-to-use functions and classes for development, testing, cloud computing, statistical processing, visualization, machine learning, etc. It is easy to notice that languages evolve from general-purpose to domain-specific, and their elements — from basic and concrete to complex and abstract.

The next step that was envisioned in the 1980s but yet not wholly achieved is MDE. The final goal of MDE is to abstract out the concrete software implementation (Mellor 2004). Within this paradigm, an implementation can be automatically generated from a requirements specification, which is meant to be represented through blocks with standard semantics.

This movement has significant practical consequences: each time the abstraction level is raised, we expand the reuse of lower-level components (meaning that we do not re-implement them). Often, it means that systems will require more memory space and processing power due to redundant code import. However, we enormously gain in human productivity.

Apparently, the ultimate goal of this evolution is machines that understand natural language.

Model-driven Architecture (MDA) is one of the proposals for MDE initiated by Object Management Group (OMG). MDA relies on a concrete DSL called Unified Modeling Language (UML), which was later extended for hardware systems design under the name of Systems Modeling Language (SysML). SysML is now the first language of choice for MBSE.

The arguments for model-based development in the hardware world are different (one cannot expect to manufacture a rocket from SysML specification in the near future). Usually, MBSE is advocated as a more efficient approach compared to document-based engineering (Henderson and Salado 2021). MBSE helps specify complex systems on the early conceptual stage, perform some simulations (which relaxes the need for physical prototyping), help to generate specifications for verification and manufacturing.

SysML is not the only option for MBSE. Object Process Methodology (Dori 2002) is an example of alternative general-purpose systems modeling language featured by simpler syntax. SysML, OPM, and a couple of other options are adopting in industry and extensively studied in academia.

At the same time, in the world of software engineering, early enthusiasm turned to skepticism. The community embodied the initial promise of code generation only in a limited number of applications. Moreover, voices of concern started to appear, claiming that UML often reduces the productivity of software teams due to blind adoption, unrealistic expectations, and misuse (Bell 2004). Nowadays, UML syntax is widely understood in the community but is mainly used to document database models, while user requirements are usually modeled through user story maps, user journey maps and managed in Kanban boards.

Scholars also expressed number of concerns regarding the adoption of SysML (Chami et al. 2018), among which such challenges as 'Awareness and Change Resistance' and 'Tool Dependency and Integration' dealing with training and human factors. We, therefore, build our model-based approach aiming at minimizing learning costs and maximizing benefits for future users.

So, how the model-based approach can help in designing a roadmap? First, in MB-TRM all strategic requirements are stored in a centralized database, therefore differ-

ent representations of a roadmap (see fig. 1.3) for various stakeholders can be easily implemented through modern visual interfaces. This also simplifies the process of roadmap creation: a big portion of data shall not be displayed in a strategic roadmap, but is required for the informed strategic decision-making. Multiple views with interactive interfaces enable efficient display and processing of this underlying data.

Second, MB-TRM allows to work with the database in asynchronous manner. Stakeholders connect and edit their part of the model and, thanks to the formal DSL syntax, this changes can propagate to the other parts of the model, requiring an action from other stakeholders in their zone of responsibility. Some insights can even come directly from users (e.g., in the process of filling the survey). This enables truly agile roadmapping as a roadmap state always up-to-date (see fig. 1.4).

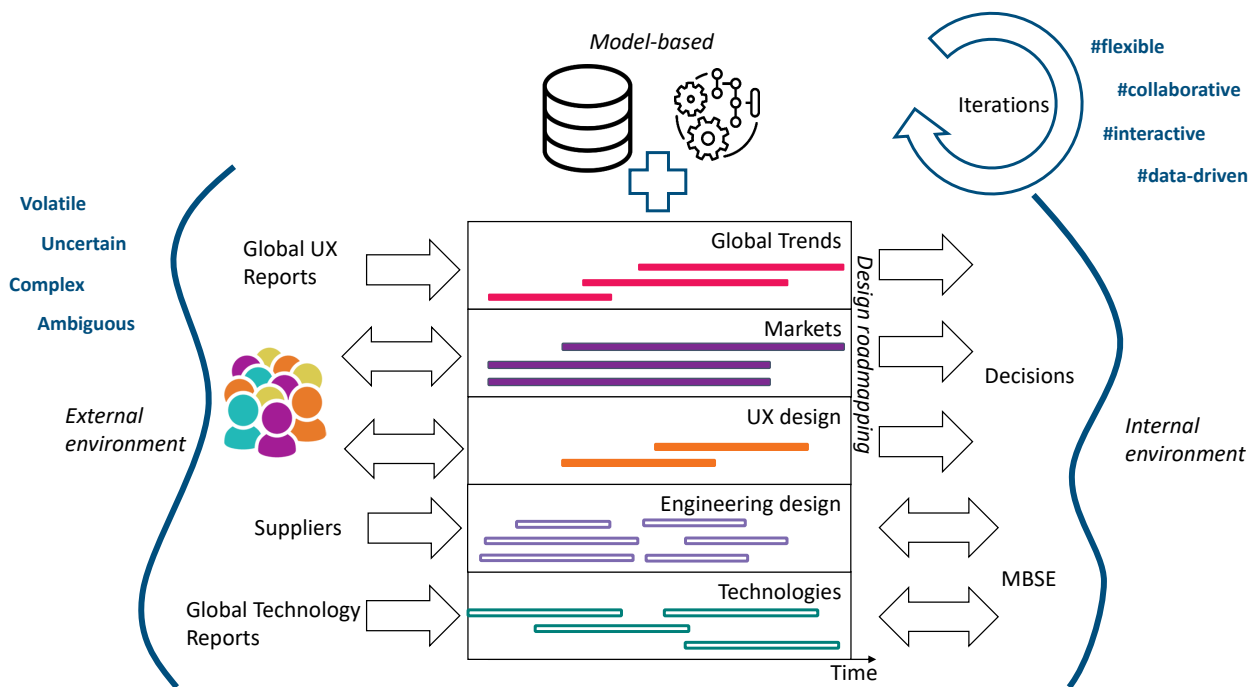


Figure 1.4: Model-based technology roadmap

We are not limiting ourselves by applying a general-purpose MBSE language (e.g., SysML) to the actual problem. We believe that strategic planning and UXD significantly differ from systems architecture, and therefore some new DSL tailored to the problem has to be developed. The requirements for the DSL are the following:

- The DSL should uniformly represent concepts on four heterogeneous levels: mar-

keting, UX design, engineering design, technology

- This DSL should respect the fact that UX is a composite concept determining the market success of a product (Kujala et al. 2011; Karapanos et al. 2009)
- UX is a process; therefore, the language should be suitable for modelling causal relationships
- UX happens in the infinite number of contexts and situations, with unique people (which can be nevertheless grouped), so the nature of the problem is qualitative and combinatorial
- Multi-disciplinary teams have their definitions and conceptual models; therefore, language should have flexible semantics
- It should be simple and easy to learn for the specialists outside MDE/MBSE domains.

1.4 Contributions and previously published works

This thesis contributes to technology roadmapping, technology management, design science, and model-driven engineering domains. All contributions described in the dissertation were peer-reviewed and published previously (except the second, which is at the moment of thesis manuscript submission is under the peer review). The relation to published materials will be indicated in this section and in the beginning of the corresponding chapter.

To keep this dissertation coherent, introduction chapters were rewritten. Literature review sections are moved to the chapter 3 of this dissertation and significantly extended. In the chapters describing contributions, there are sections called 'Related literature' that briefly discuss the most significant references related to this contribution. The non-edited materials can be found in the publisher's websites and in the French open archive HAL. Chapters describing contributions also contain additional materials not included in papers due to space constraints. Most of the figures were previously published and appear in this thesis with minor alterations (references will be indicated in captions).

In the following, we briefly describe three contributions that are reported in three chapters.

In chapter 5 we propose **the metamodel of a model-based technology roadmap**. This metamodel defines roadmaps independently from the application domain and describes common architectural elements and their relationships that appear in most, if not all, roadmaps. However, some of the entities are particular only to MB-TRM due to their digital nature. This metamodel was designed by abstracting and classifying the elements of roadmaps published in the literature. The metamodel was validated by representing three different roadmaps using modern language workbench JetBrains Meta Programming System (MPS). This contribution was published in Technological Forecasting and Social Change (Yuskevich et al. 2021a).

In chapter 6 we present **the framework for user experience design and roadmapping**. This framework is built upon the proposed metamodel and aims at creating a coherent, holistic modeling language for four heterogeneous strategy-related realms: marketing, UX design, engineering design, and technology. The proposed DSL is a feature modeling language with extensions. This contribution is built on a more concrete level and, therefore, in the semantic space, relies on concrete conceptual models proposed in four realms of interest. The framework was validated in an exemplary case study of the zero-emission vehicles transition. This contribution was submitted to Technological Forecasting and Social Change (Yuskevich et al. 2022).

In chapter 7 we **augment the framework with user testing to enable data-driven (evidence-based) new product planning**. We demonstrate how particular concepts of the UXD and engineering design layer (personas, product architectures) can be refined using Conjoint Analysis and Kansei engineering. This contribution was presented in the International Conference on Engineering Design (Iuskevich et al. 2021).

There is a decoupled contribution that we made during the PhD on **driver's cognitive and perceptive workload assessment**. We put this contribution in Appendix K in a non-edited form (Iuskevich et al. 2020).

2 RESEARCH SCOPE

2.1 Industrial context

Design research usually implies the improvement of current design practice. Blessing and Chakrabarti (2009) underlines the lack of use of the design research results in practice. Therefore, it is important to carefully review the industrial setting in which this Ph.D. was framed.

This research was set up in the l'Institut de Recherche Technologique (IRT) SystemX within a project called CMI (Cockpit Multimodal Interactif). The project also included a number of industrial partners — Saint-Gobain, Renault, Valeo, and Arkamys. The project's team created a prototype of a cockpit for a vehicle with a Dual-mode Automated Driving System (Dual-mode ADS, often referred to as semi-autonomous or partially automated). The cockpit is equipped with multimodal Human Machine Interfaces (light, ambient light, vocal and non-vocal sounds, haptic feedback).

CMI project aimed at exploring ergonomic consequences of integrating a multimodal cockpit into the semi-autonomous vehicle. Modern automated driving systems cannot work equally well in all conditions (highways, driving within towns, off-roads). Only partial automation is available in some conditions, and in others, a car shall be driven manually. Such a system behavior imposes additional requirements on the Human-Machine Interface (HMI) design, i.e., additional information about parameters of ADS should be communicated to a driver. Consequently, the goal of CMI was to explore how these design changes influence UX and driver performance in different driving situations, the efficiency of specific interfaces, and the reactions of users of different profiles. To achieve this goal, the cockpit prototype was designed and built at IRT SystemX. This cockpit was tested with real users of different profiles (more or less enthusiastic about autonomous driving).

In the academic side, this research project was supported and supervised in the Laboratoire Génie Industriel CentraleSupélec, a member of the University Paris-Saclay.

Despite we directly contributed to CMI (contribution 4) or used CMI as a case study (contribution 3), the thesis pivoted toward the needs of the automotive HMI design department of Saint-Gobain — a major producer of construction materials, focusing on glazing. Automotive division of Saint-Gobain produces windshields, sunroofs, side-

and backlites. Saint-Gobain invests in R&D of thin films, smart glasses, and integrated solutions.

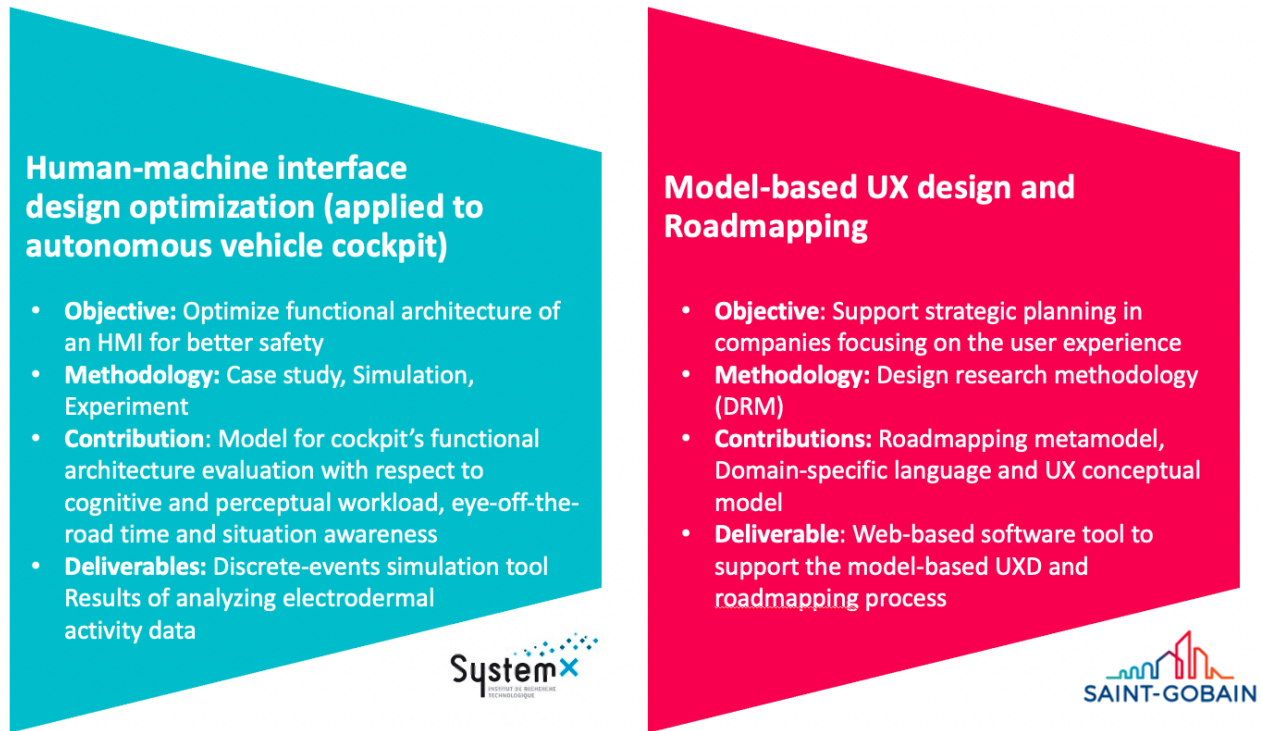


Figure 2.1: Industrial context

2.2 Objectives and success criteria

During unstructured interviews, we have identified company's needs, which we cannot fully disclose, but they generally coincide with pain points identified in Kim, Beckman, and Agogino (2018):

- UX trends change fast and in a hardly predictable direction
- target markets have a complex structure and exhibit diverse, often contradicting needs
- the feedback loop from users is lacking
- mapping user needs to technologies and components is a difficult task due to miscommunication and general complexity and ambiguity.

The reflections on these quite clearly expressed needs resulted in the initial research proposition of creating a flexible interactive model-based support tool for NPD strategic planning.

Ultimately, such a framework should improve the market fit of a company family of products, and enhance its competitive position in a long run. In terms of management, such a framework may potentially reduce the number of prototypes, help meet the deadlines and keep the focus on the end-user value. However, such criteria are not measurable within the scope of a single Ph.D. project due to time and financial limits. Therefore, case study was chosen as main validation approach.

Consequently, the main objective is to **build a methodology and a support tool for NPD strategic planning with desired properties: interactiveness, collaborativeness, flexibility. It should correctly represent aspects captured during the strategic plan formulation in a UX-focused company, namely marketing, UX design, engineering design, technologies.**

This main research objective is formulated according to our **assumptions**, for which we also found the support in the literature:

- UX is the key and the most holistic concept explaining product success and failure (Kujala et al. 2011; Karapanos et al. 2009)
- UXD should be executed in the iterative manner of constant prototyping and experimentation (Brenner and Uebernickel 2016; Meinel et al. 2020)
- Many problems in design and management arise from miscommunication, therefore common definitions, models, and vision need to be clearly defined and communicated among collaborators (McMillan 2003)
- Computer-assisted tools, if properly applied, provide a great productivity boost in companies, including those focusing on the end user (Gradini et al. 2019)

Beside the main research objective, we formulate **objectives** for each separate contribution.

- Contribution 1 objective:
 - Identify common elements among roadmaps and develop a metamodel that will define a roadmap independently from the application context and establish foundation for the support tool design

- Contribution 2 objective:
 - Propose a model-based approach for strategic planning of NPD (design syntax and semantics of a DSL)
- Contribution 3 objective:
 - Integrate the methodology and the support tool with user testing to create a feedback loop from customers (i.e., enable data-driven design roadmapping)

3 LITERATURE REVIEW

Figure 3.1 shows the structure of our contributions.

Our main contributions are to the technology roadmapping domain. Therefore, we will comprehensively review the technology roadmapping body of literature (see section 3.1). Besides, due to the multidisciplinary nature of TRM, we will need to review separately literature related to distinct layers of a roadmap for our case study in the section 3.2: market (subsection 3.2.1), UXD (subsection 3.2.2), engineering design (product family design, MBSE; subsection 3.2.3), and technology (subsection 3.2.4).

Except DRM, many elements of our research methodology fall into MDE domain. Therefore, we also review approaches and best practices in MDE (section 3.3). More specifically, we will be interested in metamodeling and in DSLs for representing product families and concepts of user-centered design.

Finally, we assume our framework to be agile and iterative from the procedural point of view. This assumption is based on many evidences from the literature. However, the strategy creation is rarely seen to be executed in the agile manner. Therefore, we need to review existing agile approaches to product design and innovation (see section 3.4).

3.1 Technology roadmapping

3.1.1 Major research streams in the technology roadmapping

Growing research interest in the TRM domain resulted in several recently published systematic literature reviews. Alcantara and Martens (2019) and Vinayavekhin et al. (2021) conducted bibliometric literature reviews, built citation clusters, and identified main themes and trends in this research. Park et al. (2020) adopted the other approach to the systematic literature review. They conducted a number of research activities (topic modeling, genealogical analysis, content analysis, and interviews) and presented a school of thought view on the TRM body of knowledge. Researchers belonging to seven schools of thought distinguished in this work adopted different research methods and came up with various propositions of desired TRM formats, purposes, and processes. Kerr and Phaal (2020) focused their literature review on the

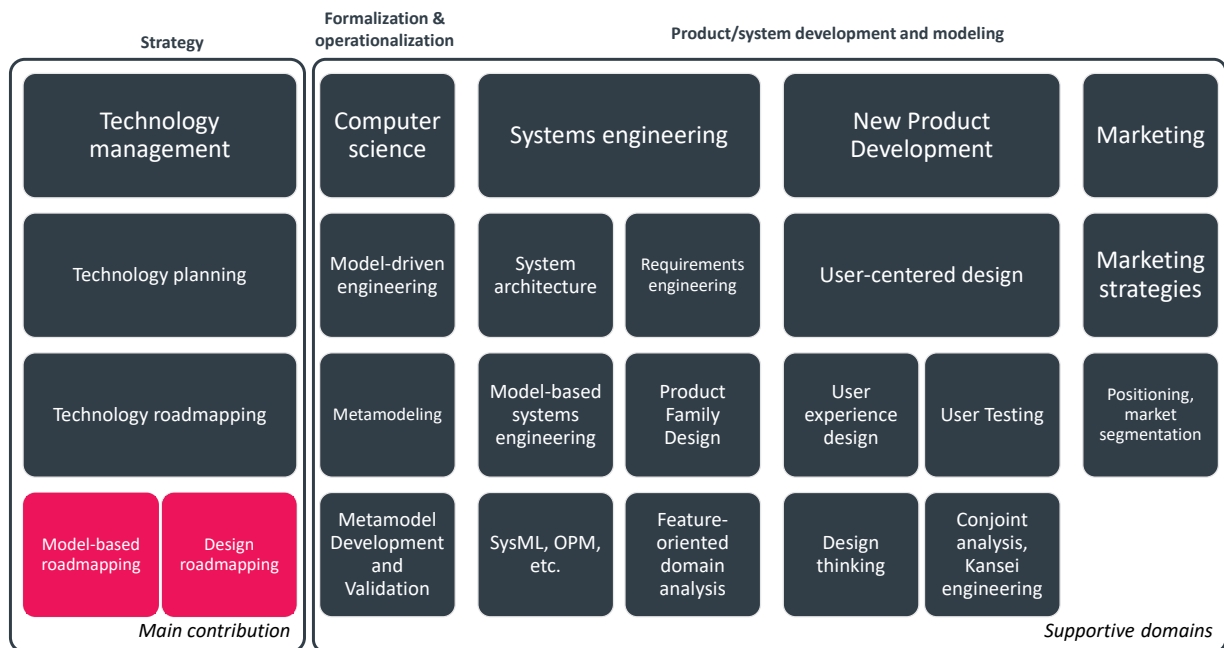


Figure 3.1: Scientific domains reviewed

TRM Industrial practice over the last three decades.

The early meta-analysis of Kostoff and Schaller (2001) classified roadmaps into expert-based, computer-based, and hybrid. They defined an expert-based roadmap as an approach where "a team(s) of experts is convened to identify and develop attributes for the nodes and links of the roadmap," and suggested that the computer-based roadmapping is a process where "large textual databases that describe science, technology, engineering, and end products are subject to computer analyses." They also acknowledged that fully automatized procedure is not achievable soon and identified the third class of roadmaps integrating computer-based analysis with expert-based decision-making.

Later Park et al. (2020) highlight the role of the contribution into expert-based roadmapping of the Department of Engineering at the University of Cambridge. This group explores TRM elicitation processes based on workshops, i.e., such endeavors where multidisciplinary teams gather together to achieve common understanding through real-time communication. Park et al. (2020) denote them as Cambridge practical school. Among many contributions of this school, the most significant are the following. They developed a so-called T-plan, a series comprising three workshops focusing on mar-

ket, product, technology, and ending with a final workshop devoted to the integration of plans (Phaal et al. 2003). This group also studied structural and visual aspects of roadmapping. Phaal, Farrukh, and Probert (2004) proposed a taxonomy of roadmaps by purpose and graphical format. They further elaborated the visual aspect of roadmaps in later works (Phaal, Farrukh, and Probert 2009; Kerr and Phaal 2015). Routley, Phaal, and Probert (2013) explored TRM in the context of industrial dynamics.

Phaal and Muller (2009) discussed architectural aspects of a roadmap. This paper is particularly interesting in the context of this dissertation, as their authors identified elements constituting a roadmap (layers and timeframes), discussed the role of taxonomies of concepts of interest that underpin roadmap architecture, questions of levels of abstraction and granularity.

The second school of thought identified by Park et al. (2020) and denoted as Seoul school embrace computer-based roadmapping. This school located in Asia is not by coincidence. In the process of interview, Park et al. (2020) have found a confirmation that people of some of the eastern cultures "find it very difficult to articulate their knowledge and capabilities in roadmapping workshops... When they are asked to make roadmaps in workshops they are likely to have a meltdown." Moreover, high-profile managers typically seek for numeric evidence-based arguments to support their decisions. In the foundational for this research stream paper, Lee and Park (2005) classified roadmaps by the information required and proposed a modular framework for the digital roadmaps' standardization and customization.

The most widespread instrument in computer-based roadmapping is text mining. Kajikawa, Takeda, and Matsushima (2010) employed citation networks approach to identify technology trends in the energy industry. Geum et al. (2015b) used the machine-learning-related association rule mining method to establish relations between keywords and build product/service evolution paths. Jeong and Yoon (2015) built roadmaps based on the patent text mining and then classified the inventions by morphological properties. Zhang et al. (2016) proposed a mixed method, which included text mining, bibliometrics, and expert-based activities for competitive technical intelligence. They also identified a basic syntactic pattern $O(L, I, T)$, (Object, Label, Implication, Time), and the relationship $R(O_i, O_j)$. This pattern helps to perform the text mining and later constitutes a roadmap. Similar syntactic pattern (Technology-Relationship-Technology) was presented by Miao et al. (2020). The other example of hybrid TRM can be found in Nazarenko et al. (2021).

Phaal, Farrukh, and Probert (2005) underlined the importance of TRM as an interface integrating other strategic approaches as a "common reference point for the ongoing discussion and a place to store information." Therefore, a lot of research efforts were devoted to complementing TRM with the decision-making and technology prioritization methods. Park et al. (2020) labels these streams as Portland and Bangkok schools, though their geography is broader. Daim and Oliver (2008) applied technology evaluation and prioritization to build energy efficiency roadmap. Gerdsri (2005) applied technology assessment, evaluation, hierarchical modeling, and forecasting to build so-called technology development envelopes (aggregated value vs. time).

The other fairly popular decision-making tools integrated with TRM are Quality Function Deployment (QFD) (An, Lee, and Park 2008; Lee, Phaal, and Lee 2013; Noh et al. 2021); Analytical Hierarchy Process (AHP) (Jeon, Lee, and Park 2011; Lee and Geum 2017); Design Structure Matrix (DSM) (Son et al. 2018); morphological analysis (Yoon, Phaal, and Probert 2007; Silveira Junior et al. 2018).

Kim et al. (2016) and Kim, Beckman, and Agogino (2018) presented a design roadmapping approach. Their contribution is two-fold. First, they underlined the importance of building roadmaps around UXD layer. Second, they found evidence and built argumentation around the need for agile iterative roadmapping in companies dealing with the VUCA-environment.

3.1.2 Model-based Technology Roadmapping

Knoll, Golkar, and Weck (2018) established a new research direction in TRM, called model-based technology roadmapping (MB-TRM). This stream is deeply rooted in the Model-based systems engineering (MBSE) and was conceptually created in response to the practical need of the aerospace industry for mapping and adapting MBSE tools and approaches to the strategic level. The shift from TRM to MB-TRM has the similar stakes to the shift from document-based to model-based systems engineering (Gradini et al. 2019). The essence of MB-TRM is in representing strategic information using formal syntax and common semantics. Once data expressed in a domain-specific language (DSL), — no matter by human or automatically — it can be more effectively stored, altered, or processed.

A number of works foreshadowed MB-TRM. Similar to MB-TRM, all the computer-based TRM approaches are featured by graphical user interface (GUI), and include quantitative tools that facilitate work of strategic teams. Lee and Park (2005) articulated advan-

tages of storing strategic data in a centralized database and interacting with it through GUI, namely simpler customization to the particular needs, collaborativeness (including access to strategic data by both insiders and, with certain reservations, outsiders), the ability of keeping a roadmap alive, and guarantee of the data consistency. However, scholars in computer-based TRM were rather focused on the automatic data retrieval using text-mining, than on the formal representation of strategic information using Domain-specific languages (DSLs).

Phaal and Muller in their architectural framework laid foundations for MB-TRM (Phaal and Muller 2009). They expressed the need in a "common language and structure" in which strategy should be expressed, identified elements, common for all roadmaps (levels, timeframes), and, most importantly, described a data structure underpinning the roadmap architecture — hierarchical taxonomies of concepts (markets, products, technologies, resources, etc.; see Fig. 3.2)

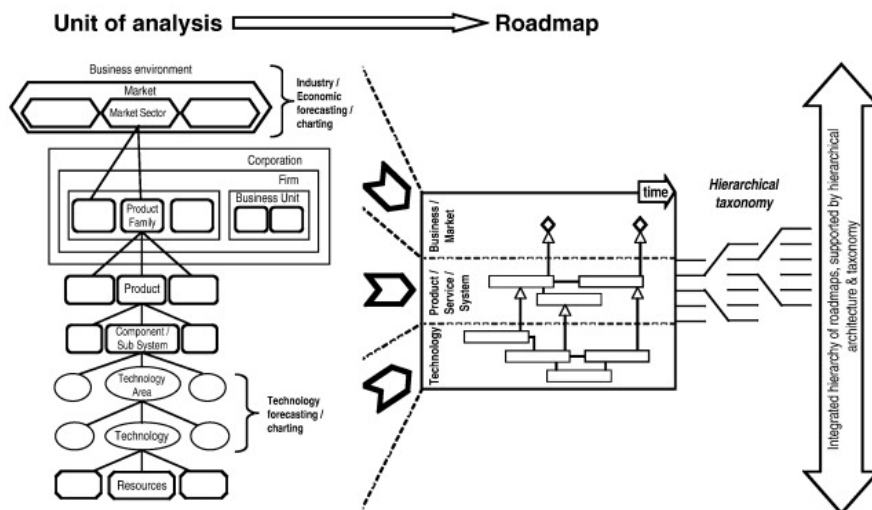


Figure 3.2: Roadmap architecture proposed by Cambridge practical school (Phaal and Muller 2009)

Geum, Lee, and Park (2014) put the information on expected industrial dynamics in the causal-loop diagram, and simulated different scenarios of the business landscape evolution using system dynamics.

Knoll, Golkar, and Weck (2018), the first proper MB-TRM, used Object-Process Methodology (OPM) language (Dori 2002) to express a family of practical aerospace system (see Fig. 3.3). Gradini et al. (2019) discussed the problems of the 'traditional' document-based TRM and suggested how MB-TRM can resolve these difficulties. Golkar and

Garzaniti (2020) proposed Model-Based Technology Roadmapping Architecture composed of key functions and processes enabling MB-TRM in the industrial setting (model structure, visualization, data analysis, knowledge management, and continuous improvement). In terms of process, Knoll, Golkar, and Weck (2018), Gradini et al. (2019), and Golkar and Garzaniti (2020) rely on a workshop-based procedures running real-time in concurrent design facilities.

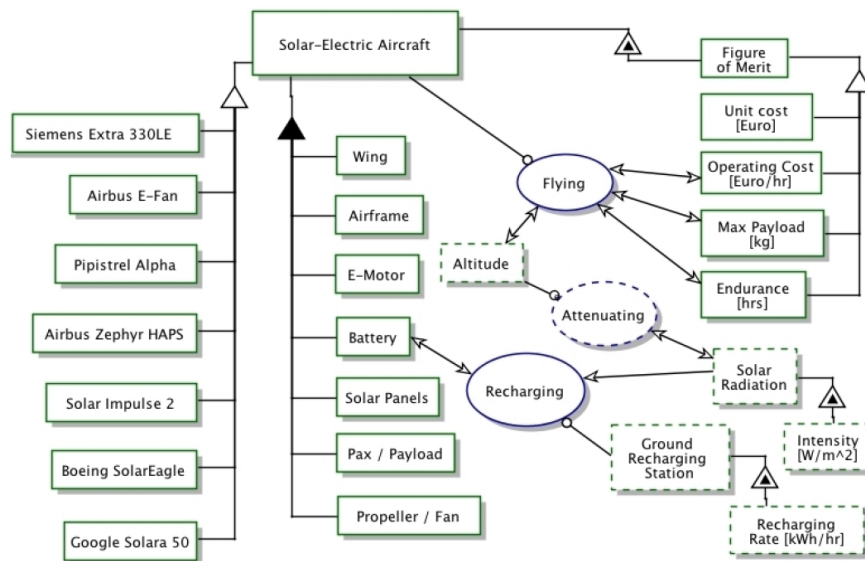


Figure 3.3: Family of solar-electric systems represented in OPM (Knoll, Golkar, and Weck 2018)

Breckel et al. (2021) proposed a MB-TRM approach with SysML-inspired DSL to manage automotive value chain. The distinctive feature of their proposed syntax is time-dependency, which is mandatory for strategic planning applications.

3.2 Conceptual models, taxonomies and definitions

As we aim to design a DSL for the UX-focused MB-TRM, we will need to propose the syntax and semantics of this language.

In order to develop the semantics, in this chapter we will review the conceptual models in the domains that are of interest for our roadmap level-by-level: marketing, user experience design, engineering design, technologies and resources.

To identify the appropriate syntax of our DSL, we review DSLs proposed for NPD and product family design in the subsection 3.3.2 of the following section.

During this investigation, we will specifically target two aspects: qualitative properties of the information representing decisions/design alternatives on each level, and the criteria, quantitatively characterizing these decisions. In other words, our goal is to build taxonomies and select sets of quantitative metrics on each level. It is worth mentioning here that for this part of the literature review, we are not pursuing to explore the newest contributions in marketing, design science, or engineering but to find established models and definitions to illustrate our approach.

3.2.1 Marketing

Market segmentation is one of the major research streams in marketing. Segmentation is one of the functions of the marketing strategy and an input for the product family design. Successful market segmentation fosters competitive advantage, positively influences financial performances and brand image (Schuiling and Kapferer 2004).

Here is a simple illustrative example of two alternative strategies a food company can choose. It can specifically target teenagers in all countries by proposing corresponding foods (e.g., chips and soda) or target Mexicans of all ages by selling products of national cuisine (e.g., jalapeño pepper).

Hassan and Craft (2005) proposed and analyzed the following factors related to market segmentation:

- macro-level segmentation:
 - macroeconomics (level of economic development, industrialization, a form of government, etc.);
 - geo-demographics (location, population demographics);
 - macro-cultural factors (religion, culture, language);
- micro-level segmentation:
 - demographics (age, income, gender, education, family, lifestyle);
 - attitude and usage (buyer needs and wants, product benefits, attitude toward product);
 - micro-culture (religion, ethnicity, regional identity, urbanization of dwellings, social class);

- brand loyalty (existing and potential loyalty, frequency of product use).

El-Ansary (2006) presented a variety of taxonomies and frameworks for marketing strategies and market metrics from different points of view.

Keshavarz Ghorabae et al. (2017) proposed a taxonomy of market segment assessment metrics based on Porter's five competitive forces (Porter 1989).

Junior and Almeida (2018) presented a review of marketing practice based on archetypal entities (personas). Also, they use widely understood indicators to assess market segments, namely total addressable market, served available market, and target market (Blank and Dorf 2012).

Except for structure and metrics, we are interested in the temporal dimension. The process that comprises market segmentation, targeting, differentiation, and positioning is called marketing strategy (El-Ansary 2006).

There were many strategies proposed in the literature (Nair and Boulton 2008). One example that is influential in the context of complex systems is Low-end (Christensen 1993) and High-end (Dyer and Furr 2015; Kilkki et al. 2018) disruption strategies. The idea of the former is that new entrants with innovative but not yet the most performant technology may choose first to serve the least demanding customers to then disrupt high-end markets by progressively improving technology performance (see Fig. 3.4). The latter happens when a company chooses to target the premium market by providing some unique value offer (usually connected not to the performance, but rather to the UX, e.g., some technology that is eco-friendly) and then disrupt a mass segment by lowering the cost of technology due to economy of scale (see Fig. 3.5).

We will use this straightforward model to demonstrate two alternative scenarios for our illustrative case study.

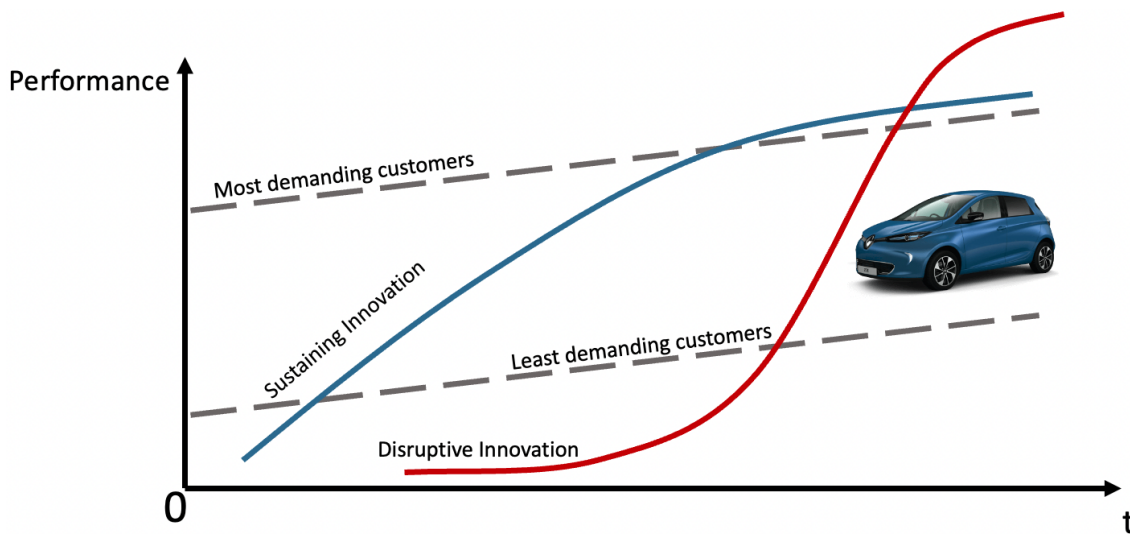


Figure 3.4: Low-end disruption strategy (Christensen 1993), illustration from Yuskevich et al. (2021b)

3.2.2 User experience design

A user experience design is a “wicked problem.” UX comprises so many ambiguous human-related concepts that even providing a unified definition is problematic (Lallemand, Gronier, and Koenig 2015; Law et al. 2009).

A definition that reflects this ambiguous nature of UX is proposed by Alben (1996):

“All the aspects of how people use an interactive product: the way it feels in their hands, how well they understand how it works, how they feel about it while they’re using it, how well it serves their purposes, and how well it fits into the entire context in which they are using it.”

Another definition provides a more specified view (Hassenzahl and Tractinsky 2006):

“A consequence of a user’s internal state (predispositions, expectations, needs, motivation, mood, etc.) the characteristics of the designed system (e.g. complexity, purpose, usability, functionality, etc.) and the context (or the environment) within which the interaction occurs (e.g. organizational/ social setting, meaningfulness of the activity, voluntariness of use, etc.).”

As can be inferred from these definitions, UX is a holistic concept embracing human feelings, product characteristics and usage context. UX is also enduring in time: it starts from the anticipation of the use, continues in the moment of use (which is a

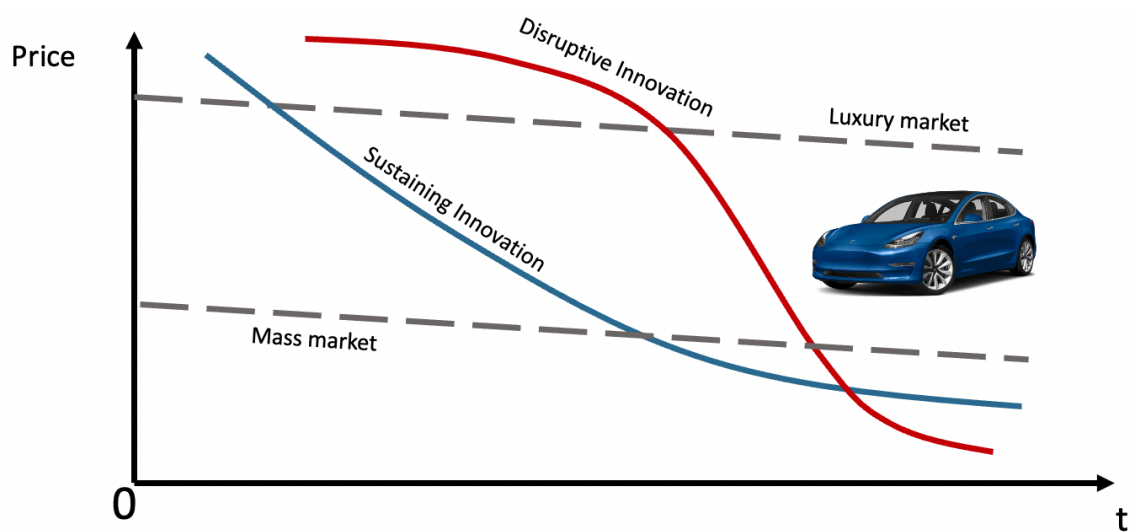


Figure 3.5: High-end disruption strategy (Nair and Boulton 2008), illustration from Yuskevich et al. (2021b)

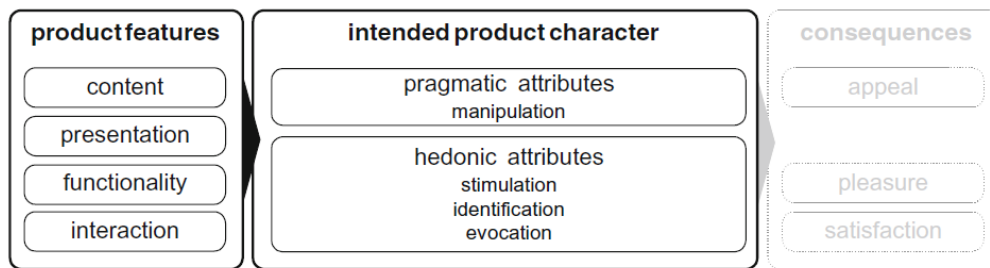
process itself), and remains in users' memory (Marti and Iacono 2016). In the process of scrutinizing this concept, researchers came up with several valuable models that we will utilize for the UXD layer of our roadmap.

Hassenzahl (2018) proposed a model that distinguishes between intended (how designers envision a product, Fig. 3.6a) and apparent (how people perceived it, Fig. 3.6b) product character. The other elements of this model can be often encountered in other UX conceptualizations (e.g., Robert and Lesage (2017)): product features, pragmatic and hedonic attributes, situation (context), and consequences (appraisal).

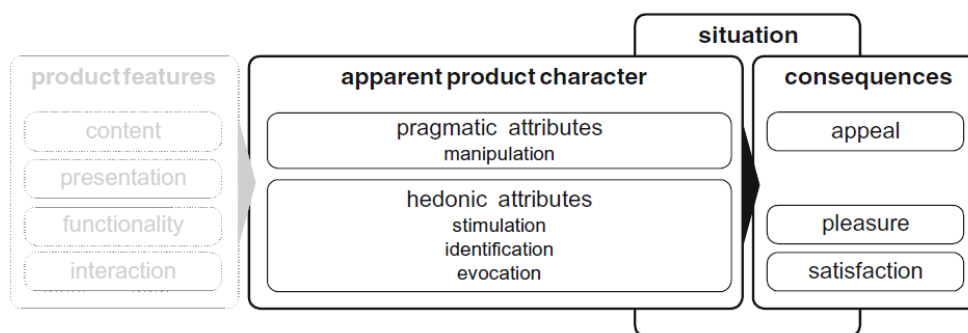
Hassenzahl and Carroll (2010) and Pucillo and Cascini (2014) came up also with the goal-based model of UX. They argue that UX can be described through a hierarchy of user goals (see Fig. 3.7). On top of hierarchy, there are high-level be-goals, e.g., "I want my kids to be safe." On the lower levels, there are do-goals (e.g., "I will ensure the safety of my kid in the car") and motor-goals (e.g., "I will double-check if the seatbelt is fastened").

More instrumental models isolate and attempt to measure consequences. Usually, the following elements are distinguished: utility, usability, desirability (ISO 9241-210:2019 2019), plus accessibility, creditability, findability, and the holistic measure called the value (Morville 2005).

User experience designers also actively utilize personas and user roles Cohn (2004).



(a) A designer perspective



(b) A user perspective

Figure 3.6: UX conceptual model of Hassenzahl (2018)

The former should be considered as a bridge between marketing and UXD domain, as in both areas, the concept of personas is commonly understood but used with a different purpose.

Finally, the concept of usage links UXD with engineering design. Just as goals, usage can be decomposed from abstract to concrete into a hierarchy of concepts. Alexander and Maiden (2004) define use cases “as a collection of scenarios,” which are the “alternative ways of achieving a goal.” Scenarios (not to be confused with strategic scenarios), in their turn, can be decomposed into scenario steps (often referred to as activities, see Fig. 3.8) and further to elementary tasks.

3.2.3 Engineering design

Generally speaking, in the engineering design, the conceptual models will heavily depend on the application domain.

In complex systems engineering, it is common to distinguish between logical (functional, behavioral, and temporal) and physical architecture (Walden et al. 2015). The logical architecture consists of functions and sub-functions, physical architecture con-

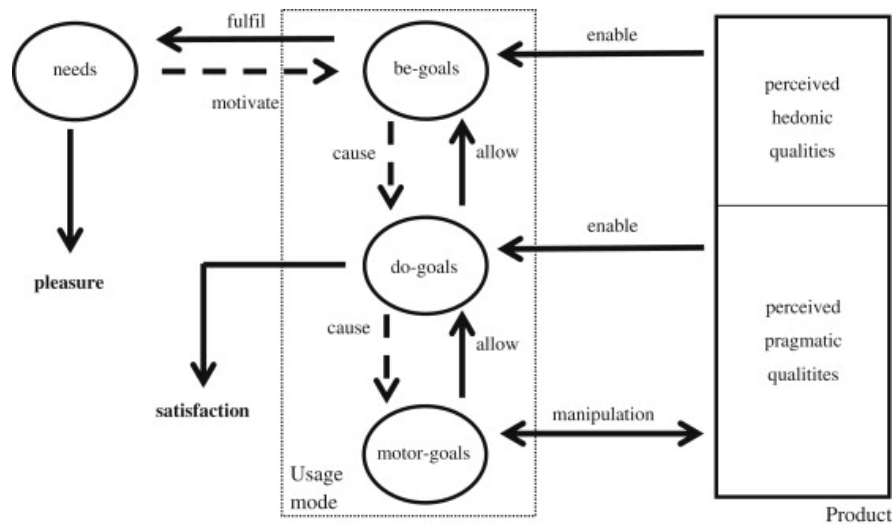


Figure 3.7: Goal-based UX model of Hassenzahl and Carroll (2010) interpreted by Pucillo and Cascini (2014)

sists of sub-systems and components. Elements of the logical architecture (functions) typically require one or a number of elements of the physical architecture. In our methodology, functions are linked to the UXD layer (functional architecture relates to the goals and behavioral to the usage). Components are connected to the technology layer.

In software systems, historically, a number of abstraction levels is distinguished. For example, in computer networks, the seven-layer OSI model is adopted (Zimmermann 1980). When designing software, however, lower infrastructural layers are typically abstracted out. The product itself is usually represented through epics and user stories (Cohn 2004; Choma, Zaina, and Beraldo 2016), domain, and feature models.

Garrett (2010) presented a layered model of a web-based software product consisting of strategy (purpose), scope, structure, skeleton, and surface. This model can also be applied to hardware products with the human-machine interface (HMI), one of the main concerns in the user-centered product design.

For the quantitative assessment, products and systems are usually evaluated by comparing their figures of merit (e.g., power, clock rate, capacity), -ilities (e.g., reliability, usability, flexibility), costs, and constraints (e.g., mass, cost, size).

This seeming simplicity should not be misleading. The problem is that the number of performance indicators is vast for most systems, their contribution to the end-

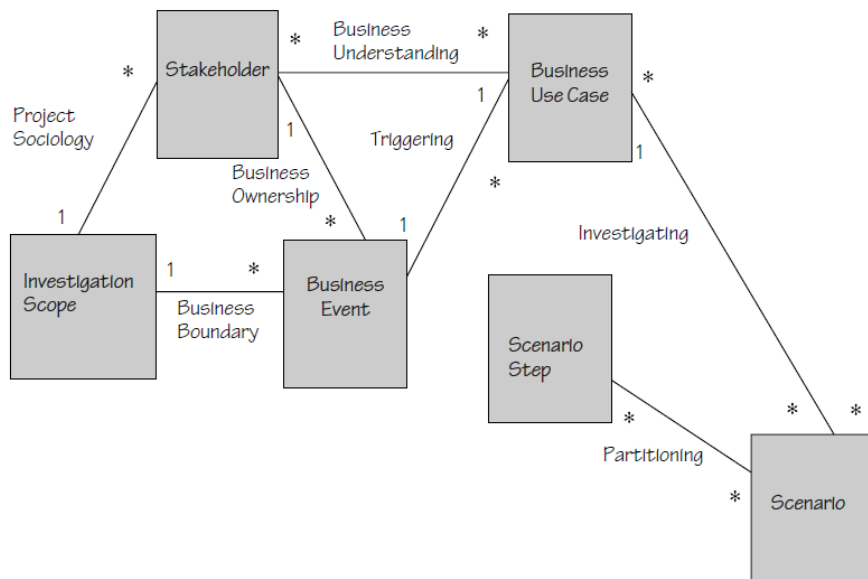


Figure 3.8: Usecase-based model of Alexander and Maiden (2004)

user value is uneven, and the satisfiability is not guaranteed. Therefore, particular approaches have been developed to help in such a multi-attribute decision-making. For example, in Cost-Benefit Analysis, net aggregated benefits are weighted against net aggregated costs (Nickel, Ross, and Rhodes 2009).

Multi-Attribute Tradespace Exploration further develops this idea by formulating system design as an optimization problem where aggregated utility and expense are functions of controlled design variables (Ross and Hastings 2005). For risk assessment, additional metrics have to be introduced. For example, risks of incorporation of new technologies into a parent system can be assessed through Technology Invasiveness Index (Smaling and Weck 2007).

The link between architectural decisions and quantitative metrics is studied by Selva, Cameron, and Crawley (2016). They introduced six patterns related to architectural decisions (i.e., combining, downselecting, assigning, partitioning, permuting, and connecting) and formulated them as the combinatorial optimization problems.

However, we are focusing not on complex systems but on products. In the context of product architecture, mass customization and product families are the key concepts to consider.

To win the price competition, companies should make the most of economies of scope

and scale. Economies of scale are "cost advantages reaped by companies when production becomes efficient" (*What Are Economies of Scale?* N.d.). Generally speaking, the bigger the organization, the more cost advantages it can get through specialization of labor, bulk orders, lower costs of capital, lower overhead costs per unit, etc. (*What Are Economies of Scale?* N.d.) The cost advantages attained while producing a variety of products together are called economies of scope (*Economies of Scope Definition* n.d.).

The intention of companies to produce the same products in large quantities confronts the fact that users and their needs are all unique. This contradiction is resolved by a method called mass customization. The goal of this approach is to keep prices close to mass production, and nevertheless, fulfill individual customer requirements. This is achieved by active use of the reusability/commonality of modular architectures. The production process is planned to delay the product differentiation step as far as possible.

Tseng, Jiao, and Merchant (1996) developed a design for mass customization (DFMC) approach. The idea of DFMC is to enable mass customization by focusing on product family (PF) architecture (PFA) in the early design stages. DFMC also expands the scope of early-stage design optimization with sales and service functions, as in many cases, product differentiation happens during these steps.

Du, Jiao, and Tseng (2001) introduced the concept of Architecture of Product Family (APF) composed of common base (shared modules produced in large quantities to foster economies of scale) and differentiation enablers (modules or accessories produced in smaller quantities but in a large variety to promote economies of scope). The latter is a source of variety in product families. Besides, Du, Jiao, and Tseng (2001) identified mechanisms of variety generation through attaching/removing, scaling, and swapping of differentiation enablers. Jiao, Zhang, and Wang (2007) proposed a Genetic Algorithm for early-stage PF design based on APF.

When PFA is defined, how do we know which instances of a PF (i.e., distinct products) are the most appealing for a company to sell and for the users to consume?

The most popular approach to classify the product features is the so-called Kano model (Kano 1984). According to this model, some attributes are referred to as 'must-be,' i.e., a customer expects them to be in the product and therefore is not much pleased when it is fulfilled. However, a customer will be dissatisfied if the product lacks these features. Some other characteristics are 'one-dimensional — the more, the better (e.g., the higher range in one charge in an electric vehicle, the more satis-

fied a customer is). Finally, 'attractive' attributes are the most expected, and, therefore, a customer gets excited about finding them out.

Kano model can be easily integrated with other quantitative approaches (e.g., QFD (Shen, Tan, and Xie 2000)). Xu et al. (2009) proposed an extension of Kano model called the analytical Kano model that focuses on customer needs analysis and promotes quantitative measures during this process.

Zhou et al. (2017) represented user-visible aspects of Amazon Kindle through feature model and performed sentiment analysis of user-generated reviews in order to determine which features have a positive or negative influence on overall UX.

Kim et al. (2017) elicited the link between user testing and the early-stage product design and developed a so-called scenario-based conjoint analysis (CA). CA is a type of user testing where a user is supposed to choose between two alternative product configurations. The idea of scenario-based conjoint analysis is to structure a survey in a way that it tells a story and alternate product configuration depending on choices made in the previous stages of a plot. Colombo et al. (2020) used (CA) to rank modular product configurations based on aggregated value.

Tucker and Kim (2011) studied changes of the preferences over time (they call it preference trend mining), which links well the PF domain to the TRM. Based on analysis of the preference dynamics, they proposed a classification of product attributes on standard, nonstandard, or obsolete. Jun, Park, and Yeom (2014) demonstrated the possibility of preference trend mining using the web search information.

A huge amount of work has been done to rank distinct product variants based on Kansei engineering (KE). KE was developed in Toyota. It aims to establish the link between product architecture and the most subtle and ambiguous part of the human perception — the emotions. Quan et al. (2019) used the results of a differential scale survey with KE word pairs (e.g., female-masculine, unique-ordinary, simple-refined, modern-traditional, etc.) to rank product options using AHP. Kett and Wartzack (2016) and Kett, Schmitt, and Wartzack (2017) used KE for market segmentation and product positioning.

3.2.4 Technologies and Resources

Just as with UX, the concept of technology is not easy to define. Wahab, Rose, and Osman (2011) conducted a comprehensive literature review and identified twenty def-

initions of technology. Most of the authors distinguish two significant aspects of technology (e.g., Kumar, Kumar, and Persaud (1999)):

- informational aspect: technology as a knowledge, a “know-how”;
- physical aspect: technology as a set of tools, blueprints, equipment, and processes.

Hein (2016) argues that artifacts produced with a given technology, its embodiments, are often considered as part of a technology (in a sense of artifact’s form, not a matter). For example, when we say the “rocket engine RD-180” along with production pipeline and know-hows, we mean any of the engines of this type. When technology is integrated into a system, it becomes a component.

For better readability and manageability of a roadmap, it is convenient to build a taxonomy of relevant technologies. Taxonomy of NASA’s technology roadmaps (2020 *NASA Technology Taxonomy 2019*) comprises four layers: technology area, technology subarea, technology type, and technology itself. For example:

- TX01 Propulsion systems
 - ◇ TX01.2 Electric space propulsion
 - ▽ 1.2.3 Electromagnetic
 - △ Pulsed inductive thruster.

Similar to the systems, technologies are characterized by figures of merit (e.g., power-to-weight ratio, transistor density, efficiency) and -ilities. Unlike systems, technology costs are typically assessed not per unit but as total capital investments needed to acquire given technological capability. Technology readiness level is usually used as a proxy metric of risks (Mankins 2009).

Technologies are tightly connected to the notion of resource — an "asset tied semipermanently to the firm that allow its managers to conceive and execute value-creating strategies" (Barney 1991). The resource-based theory of firm explains the performance and sustainability of companies by examining their resources (or core competencies). Prahalad and Hamel (1997) explain this concept through the following metaphor "the corporation, like a tree, grows from its roots. Core products are nourished by competencies and engender business units, whose fruit are end products".

Resource-based perspective is especially important in the view of strategic planning. In this process, an organization should pursue not envisioning the future product portfolio but to "determine the desirable competence portfolio for the future" (Probst 1998).

3.3 Model-driven Engineering

In our first contribution, we are going to develop a general metamodel that will define the structure of a model-based roadmap independently from the application context. To do this, we will review the practice of designing and validating the metamodels in the first subsection.

Then, in the second subsection, we will review DSLs that were developed to model product families and user experience.

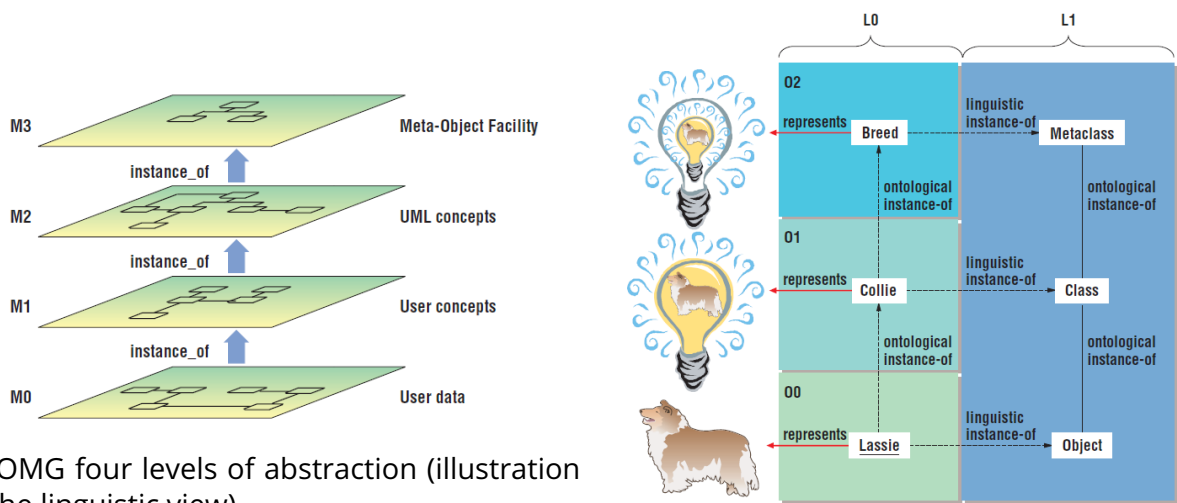
3.3.1 Metamodeling

The concept of a metamodel is far from being intuitive. Mellor (2004) defines meta-model as "a model of a modeling language. It defines the structure, semantics, and constraints for a family of models." At the same time, a model is "a simplification of something <..> It consists of sets of elements that describe some physical, abstract, or hypothetical reality." (Mellor 2004)

In simple words, a metamodel is a representation of highly abstract concepts of some language or ontology. Figure 3.9 shows two example types of metamodels — linguistic and ontological (Atkinson and Kuhne 2003).

On the left, a model of OMG abstraction levels is shown. Level M0 represents user data stored in the database (for example, concrete values of parameters and class instances). Level M1 defines these classes and parameters, for example, in UML. Level M2 is an actual metamodel; it determines elements of a language (e.g., for UML: a class has a name, attributes, and methods, can be instantiated and connected to other classes by various links). In the Level M3 located the meta-metamodel, a so-called Meta-Object Facility (MOF). It defines a type system used for compatibility with other MOF-based languages (e.g., the SysML).

On the right, an ontological view (Atkinson and Kuhne 2003) on a metamodel is presented. In this view, a metaclass (metamodel element) is simply a second level of



(a) OMG four levels of abstraction (illustration of the linguistic view)

(b) Illustration of the ontological view on meta-modeling

Figure 3.9: Illustrations of the concept of metamodel from Atkinson and Kuhne (2003)

abstraction on the real-world object. For example, Lassie is a name of a particular dog. Collie is the name of a breed of this dog. The breed here is the second level of abstraction — Lassie (Collie (breed)).

But what for do we need a metamodel? Metamodels provide numerous practical advantages in the MDE practice (Ober and Prinz 2006; Terrasse et al. 2006):

First, DSLs built upon the same metamodel are interoperable to some extent. Let's consider the following sentence of Henry Gleason Jr.: "The iggle squiggs trazed wombly in the harlish hoop". We do not understand the meaning of each particular word, but almost magically, the whole sentence makes sense. This is because this artificial language shares many traits (we may say a metamodel) with English. We understand where is the object, the subject, the verb, and which part of a sentence describes the context. Similarly, a UML environment can partially understand a SysML model so that the model can be imported into the environment. Both languages (UML and SysML) are MOF-based, and, therefore, with certain reservations, interoperable.

Second, even the brightest people cannot accommodate a complex body of knowledge entirely in their heads. In this case metamodels (or ontologies¹) are useful generalizations of this information. Scientific realms (e.g., biology) operate with more ab-

¹We will use terms ontology and metamodel interchangeably; for more details, see Terrasse et al. (2006)

stract concepts, or classes (e.g., bacteria), meanwhile scientific domains (e.g., microbiology) — with more concrete classes (e.g., *Mycobacterium tuberculosis*). Specialists from different domains but the same realm can adapt the level of abstraction and discuss essentials on the abstract level. Again, here we face some kind of language interoperability (in this case ontological).

Third, in many cases, metamodels are enlightening. It is easier to explain complex things on the abstract level. Adult people mostly learn new things from the general (a metamodel) to specific (a model).

Finally, from a practical perspective, metamodels provide a basis, 'an inspiration' for developers of concrete DSLs.

In the process of metamodeling, people use inductive reasoning. They observe some instances (real objects or concepts) and, in order to build an abstract representation, find the common characteristics and relationships between these instances.

Figure 3.10 shows a meta-ontology O2 proposed by Gangemi et al. (2006) and interpreted in our published work (Yuskevich et al. 2021a). This representation underlines the subjectivity of metamodels and ontologies, as they manifest rational agents' intended conceptualizations, influenced by agents' objectives and accepted definitions. Accordingly, a metamodel should exhibit a consensus in a community.

Therefore, to create a metamodel of the technology roadmapping domain, we need to review concrete instances (technology roadmaps) and develop a conceptualization that satisfies a set of criteria. Vrandečić (2009) did a meta-analysis and summarized a comprehensive set of criteria for ontology (and metamodel) validation: accuracy, adaptability, clarity, completeness, conciseness, consistency, and organizational fitness.

López-Fernández, Guerra, and Lara (2014) distinguish two groups of criteria for validating a metamodel. The first group of criteria answers the question "are we building the right metamodel?" (usefulness and generality), and the second group — "are we building the metamodel right?" (usability).

To prove MB-TRM metamodel's usefulness and generality, we need to demonstrate that it correctly describes the vast majority of roadmaps. To prove the usability, we need to ensure that our metamodel uses best practices (properly written in a widely understood language).

There are two alternatives for creating and validating a metamodel. One way is to

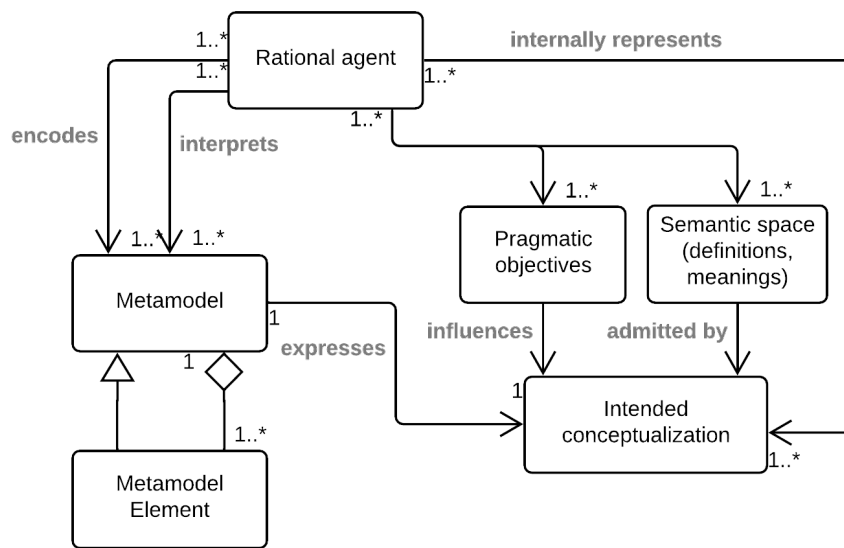


Figure 3.10: Conceptual model of metamodeling (Yuskevich et al. 2021a) (originated in Gangemi et al. (2006))

gather domain experts in a workshop aiming to reach a consensus over this metamodel. The other way is to, at first, build a metamodel inductively based on a significant number of published roadmaps, and then validate it by proving that some other roadmaps (that were not used in the process of metamodel creation) can be also instantiated from it.

López-Fernández et al. (2015) developed a so-called example-driven approach to metamodel elicitation and validation. Their algorithm takes several exemplary models as input and induces a metamodel as an output. For validation, this metamodel can then be tested against valid and invalid models.

3.3.2 Domain-specific languages

This section aims to explore the DSLs that can be used for roadmapping, user-centered design, and product family design.

The two most popular standard visual modeling languages in the MBSE field are SysML (OMG 2019) and OPM (Dori 2002). As was discussed above, both of these languages were already applied to MB-TRM (Knoll, Golkar, and Weck 2018; Breckel et al. 2021).

However, SysML and OPM were created for complex systems architecture specification, and therefore hardly applicable for product design, marketing, and UX modeling. More specifically, their syntax does not satisfy the criteria that we postulated in section 1.3. These DSLs do not offer a mechanism of general model instantiation, which is needed for PF representation and market segmentation. They are not expressive when it comes to the modeling of causal relationships. They do not have the capability of defining custom definitions (conceptual model) and require a lot of time to learn and master.

The majority of DSLs used for product modeling are built around the concept of a product family (see 3.2.3). Models produced with these languages are often referred to as variability models. Du, Jiao, and Tseng (2001) proposed a DSL for product family design supporting the description of a PF as a whole, called Generic Product Structure (GPS), and representation of individual product variants through the mechanism of GPS instantiation. In GPS, modules and functions are classified into common bases (present mandatory in all products of a PF) and optional differentiation enablers.

A pretty similar idea lies in the foundation of feature modeling language. Feature modeling is used for the software product line modeling within a framework called Feature-oriented domain analysis (FODA). Figure 3.11 shows a standard illustrative feature model of a (quite outdated) family of mobile phones. All mobile phones should have a screen and function of calling (mandatory features). Some mobile phones have a GPS and media capabilities. A screen can be basic, colour, or high-resolution (group of alternatives). The media capabilities can include a camera, an MP3, or both.

There is a logic defined on feature models. For instance, parent features affect child features: if media is disabled, so the camera and MP3. In turn, if MP3 or camera is enabled, the parent feature should be enabled too. Also, the feature tree has so-called cross-tree constraints that propagate changes in other branches. For example, a camera requires a high-resolution screen, and if the screen is basic, GPS is not possible and vice versa. Product variants are configured from PF feature tree by enabling/disabling optional features.

Here we must point out several important observations. First, FODA notation lacks any predefined semantics: components, functions, and attributes are represented by exact same rectangle. Second, the feature tree is a directed acyclic graph. This means that besides product families, this syntax can represent causal structures, version history, citation graphs, and other timed processes. Third, a feature tree has

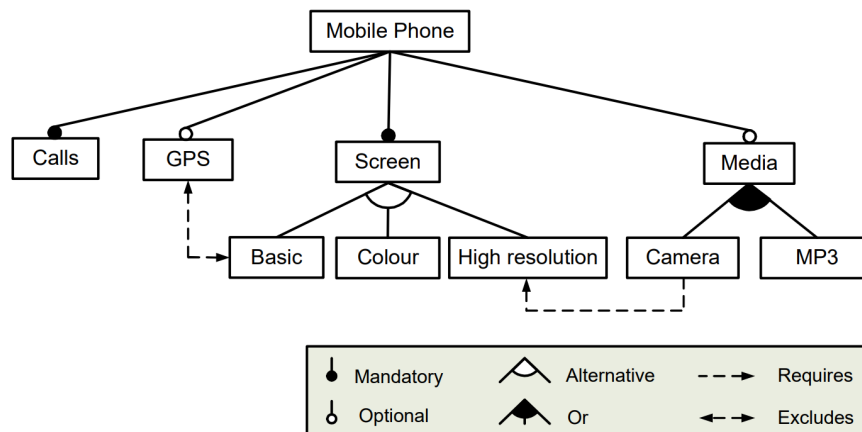


Figure 3.11: A sample feature model (Benavides, Segura, and Ruiz-Cortés 2010)

built-in capabilities for automation and consistency checks. For instance, Benavides, Segura, and Ruiz-Cortés (2010) and Ripon, Hossain, and Bhuiyan (2013) provided an overview of proposals of feature trees automated analysis (e.g., generation of all possible variants, generation with filtering, optimization detection of dead, false optional, redundant features, etc.)

A feature model can be specified through a three-valued propositional logic (yes, no, indetermined) (Saller et al. 2012). Höfner, Khedri, and Möller (2011) developed a product family algebra, a textual representation of a feature model. For example, this is how feature tree on Figure 3.11 can be represented with this proposed notation (multiply polynomials to get all product options in the form $\prod p_i$):

$$\begin{aligned} & \text{calls} \cdot \text{Opt}[\text{gps}] \cdot (\text{basic} + \text{colour} + \text{high_resolution}) \cdot (\text{camera} + \text{mp3} + \text{camera} \cdot \text{mp3}) \\ & \text{gps} \cdot \text{basic} = 0 \\ & \text{camera} \rightarrow \text{high_resolution} \end{aligned}$$

Such formalizations help to develop more efficient numeric algorithms to process feature models (e.g., for solving multi-view reconciliation problem as suggested by Höfner, Khedri, and Möller (2011)).

Casalánguida and Durán (2012) demonstrated how feature model can be generated from UML model. Interoperability between feature models and OPM is studied by Wang and Dagli (2013).

Several works contributed to linking feature models with user testing, which is especially important in the context of this thesis. Such methods would potentially communicate to the company the user opinions toward different product features through the partially automated feedback loop. Zhou et al. (2017) integrated feature model with customer opinion mining. Muller and Lillack (2011) augmented feature models with conjoint analysis.

Abrantes and Figueiredo (2014) applied feature models to the problem of NPD portfolio management and identified aspects that can be facilitated by feature models, namely, the analysis of portfolio interdependencies, artifact reuse, and component-to-function mapping.

Nešić et al. (2019) considered feature modeling from the methodological point of view and suggested 34 best practices to follow during in this process.

There are a number of contribution aiming at extending standard syntax of feature models. In the work of Casalánguida and Durán (2012), they used a feature tree syntax with cardinality (e.g., a vehicle mandatory has 4 wheels). Feature models were created to represent quantitative properties of products. However, there is a need of representing also quantitative properties of feature models (characteristics and non-functional requirements). Benavides, Trinidad, and Ruiz-Cortés (2005) extends feature models with quantitative attributes, which have a domain (possible values, e.g., $market_size = \{200e3...300e3\}$). Attributes can be derived from other attributes using algebraic operations ($revenue = units_sold \cdot price$).

Passos et al. (2011) study Component Definition Language (CDL). CDL is another textual variability modeling language that can be viewed as feature model extended with quantitative attributes, and some semantics; features are classified in four kinds: packages, components, options, and interfaces.

Amyot and Mussbacher (2002), Chung and Prado Leite (2009), and Bresciani and Donzelli (2003) are elaborating the Goal-oriented Requirement Language (GLR) within a NFR (Non-Functional Requirements) framework. GLR shares many elements with feature models (optional and mandatory requirements, OR, XOR groups) and is richer in some aspects 3.12. For example, they add a kind of quantitative links called positive and negative contributions (links marked with --, -, +, ++).

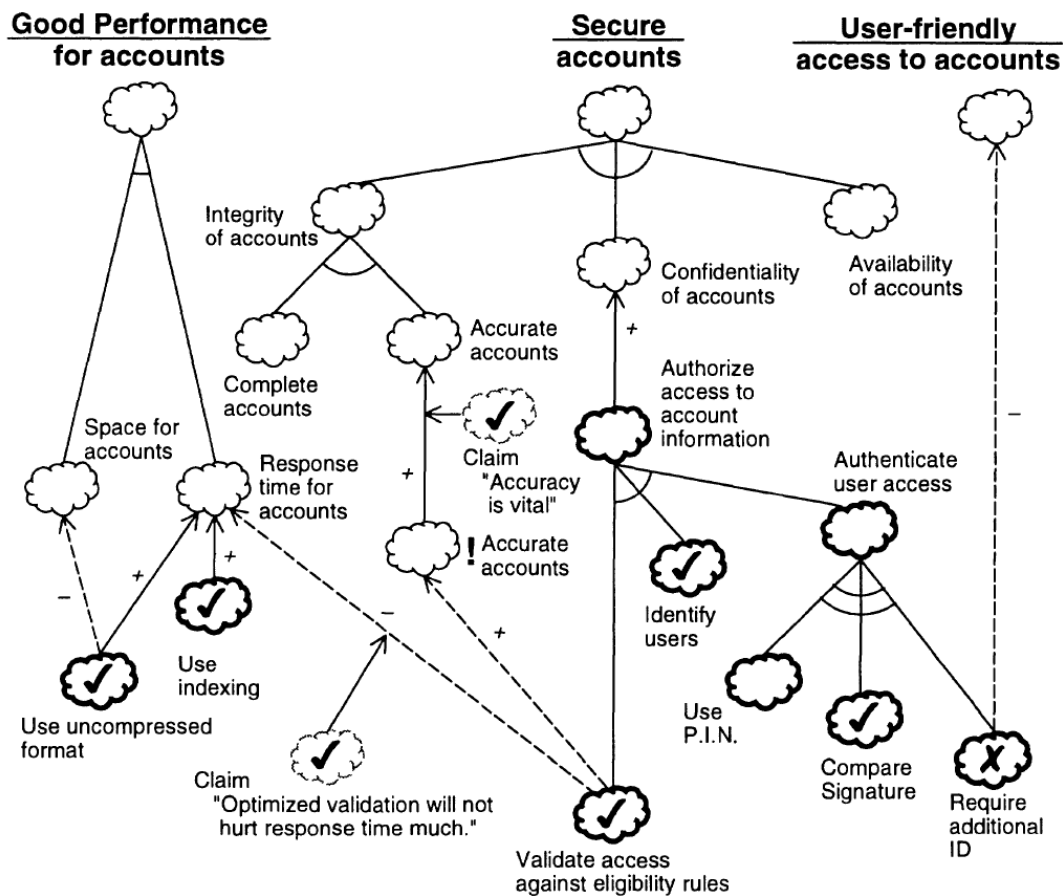


Figure 3.12: A sample GSN model (Chung et al. 2012)

3.4 Agile approaches to product design and innovation

The word agile is commonly associated with project management. In terms of administration, agile project management advocates a more egalitarian flat teams structure, in contrast to the hierarchical approach of traditional project management (Fernandez and Fernandez 2008). In terms of time management, those who practice agile prefer iterative and adaptive processes to traditional linear and incremental process (a 'waterfall'). Wysocki (2006) also distinguishes an extreme strategy, where even the ultimate goal of a project may be changed to the new, discovered in the course of the project. 'Lean startup' (Ries 2011) methodology that preaches 'fail fast, succeed faster' philosophy is an example of an extreme strategy.

We assume that the popularity of agile approaches is connected to the fact that the business environment becomes more complex, uncertain, and knowledge-intensive. Network structures and iterative processes are more adaptive to uncertainty as infor-

mation and responsibilities are distributed among people, which constantly learn and are ready to respond to changes.

We believe, however, that one should not fall into the fallacy of blind adoption and instead pursue taking the best of both approaches. The strategy of large organizations, for example, may be linear for some business verticals and agile for others. Mature projects with a lower degree of uncertainty may be managed in a phase-gate manner to assure quality and meet deadlines, while in less mature projects, a higher degree of flexibility is required.

In the product design realm, the word 'agile' has a very similar meaning but emphasizes the feedback loop from customers. Design is often viewed as a learning process. Beckman and Barry (2007) distinguish four learning styles exposed by different individuals: diverging (good at observing real people in an actual context), assimilating (good at analyzing and structuring the information), converging (good at ideating and problem solving), and accommodating (good at learning from hands-on experience). Beckman and Barry (2007) argue for combining these learning styles in one process by composing the product team of people with different mindsets.

Even though these perspectives are rather complementary than contradictory, researchers usually take one of the following perspectives. Systematic approaches to NPD take the positivist stance to data gathering, while agile approaches take the interpretivist stance. The former tries to cope with uncertainty by collecting and processing quantitative data, while the latter prefers ethnographic observations and interviews.

Meinel et al. (2020) proposes to classify NPD approaches on user preference-driven, e.g., Quality function deployment (Franceschini and Rossetto 1995), Kano model (Kano 1984), and user experience-driven, e.g., Design Thinking (Brenner and Uebernickel 2016). The former are quantitative and objective; technology-, calculation-, specification-driven. The latter are qualitative and subjective; empathy-, visualization-, iteration-driven.

DT is the most influential methodology to agile NPD. Brenner and Uebernickel (2016) distinguishes the following principles of DT as a *mindset*:

- *'Innovation is made by humans for humans.'* This principle puts the highest priority on user-centredness.
- *'Combining of divergent and convergent thinking.'* The double diamond (Fig. 3.13)

is a very popular illustration of DT process that reflects these sequential phases. Initially, a product team generates hypotheses on user needs (divergent stage I). Then, they assess these needs and select a single one or a few to address in a current iteration (convergent stage I). After that, a product team brainstorms to come up with a large number of possible solutions (divergent stage II). Finally, in the convergent stage II, they select the best product concept to prototype and test.

- *'Fail often and early', 'build prototypes that can be experienced, and 'test early with customers'* principles of DT acknowledge humans' complex and ambiguous nature, and therefore advocate acquiring new knowledge through experimentation.
- *'Design never ends.'* This principle manifests the interactivensess of the DT process. For instance, the process illustrated in fig. 3.13 is not a one-way street. As soon as the prototype is tested and user feedback is gathered, the team may and should reformulate the initial hypothesis in a new design cycle.
- *'Design Thinking needs a special place.'* This principle underlines the importance of special facilities and tools supporting the process.

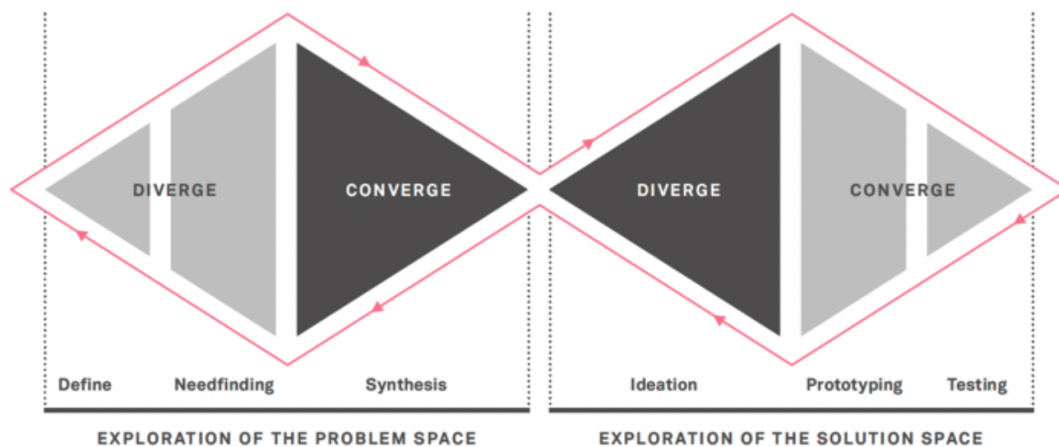


Figure 3.13: Convergent-divergent phases in DT (Hehn et al. 2020)

There are pieces of evidence that applying DT leads to better new product concepts. Meinel et al. (2020) has conducted a workshop where student teams were proposed to design a new product concept with QFD and DT. The study has found that groups

instructed to follow DT came up with concepts of higher feasibility, relevance, and specificity.

Similar to project management perspective, these two classes of NPD approaches are not necessarily mutually exclusive. They may and should be synthesized. Indeed, more flexible and iterative tools are needed in the early stages and/or in situations of high uncertainty, while more formal and structured approaches are suitable when company gathered significant amount of data in the field of their product expertise.

Hehn et al. (2020) note that artifacts generated during DT and requirements engineering (RE) are interoperable. They identified three strategies for composing RE and DT: run DT before RE, infuse RE process with DT tools, or integrate DT into RE routine. Bekhradi et al. (2018) assumes that high uncertainty associated with user-centered innovations can be limited by applying systematic approaches. They propose Investments in Technology and Markets (RITHM), a systematic approach to R&D planning aiming at reducing the number of iterations, prototypes, and pivots.

Based on the results of this part of the literature review, we conclude that our assumption of relying on the agile principles is well supported in the literature for situations of high uncertainty. Nevertheless, the main goal of this Ph.D. is not to design the NPD process, but a process for strategic planning of NPD. However, if we are agile at a tactical level, we should be imperatively agile at the strategic level, because the results of each individual iteration can change high-level goals. Therefore, the process of agile NPD planning should be flexible and iterative. This claim is also supported by Al-Ali and Phaal (2019), who concludes that the agile "roadmap will not have all the right answers from the beginning. Therefore, explicitly designing MVPs allows for systematic business experiments that aims to clarify the uncertainties and improve the DT roadmap iteratively." This idea applied to our proposed framework is explained in the section 6.4.

3.5 Synthesis and research gaps identification

The major research gap identified in the TRM body of literature is the lack of a metamodel. Phaal, Farrukh, and Probert (2004) and Phaal and Muller (2009) laid foundation for such a metamodel by classifying roadmaps and identifying common elements. However, a general metamodel has not been proposed and formally validated. The emergence of MB-TRM foregrounded this need. Golkar and Garzaniti (2020) in their

Model-Based Technology Roadmap Architecture presented a high-level structural and functional view on MB-TRM system. Nevertheless, the question of the structure of the roadmap itself remained open.

The minor research gap relates to the MB-TRM subdomain. Previously, authors addressed MB-TRM in the view of the aerospace industry. Our analysis took a broader perspective, which included automotive, science, product, policy, and service-systems roadmaps.

The specific research questions we are answering in the scope of contribution 1 are the following:

RQ1.1 Can a metamodel be proposed that is compatible with existing definitions of a technology roadmap?

RQ1.2 Can the insatiability of the proposed metamodel be demonstrated in various contexts?

RQ1.3 Is the proposed metamodel aligned with principles of MB-TRM?

RQ1.4 Is the proposed metamodel usable? To approach usability, it can be, for instance, assured that metamodel is internally consistent, concise, and built following best practices.

In the NPD planning and user-centered design fields, we are relying on the contribution of Kim, Beckman, and Agogino (2018). They convincingly supported a need and developed principles for agile user-experience-focused roadmapping. The research gap here is how to enable and facilitate such a process. Our approach is to integrate it with MB-TRM. Another essential gap we are addressing is on the MDD side. To our knowledge, there is no DSL reported in the literature that can uniformly represent such distinct domains as marketing, UXD, engineering design, and technologies. We propose both the syntax and semantics of such a DSL. The specific research questions in contribution 2 are the following:

RQ2.1 How to enable agile user-experience-focused roadmapping?

RQ2.2 Which syntax adequately represents marketing, UXD, engineering design, and technology domain?

RQ2.3 Which concepts, elements, indicators, and attributes shall be included in the user-centered MB-TRM framework?

The third contribution further validates the proposed framework and extends it with user testing. Kim, Beckman, and Agogino (2018) identified the lack of feedback loop from users in the current TRM practice. However, it was not articulated how this feedback loop can be organized, especially in the view of 'alive', constantly up-to-date road-map. In our opinion, the only choice here is automated user testing and opinion mining procedures integrated with UX-focused DSL. The possibilities for such an analysis were already elaborated in the literature (Zhou et al. 2017; Muller and Lillack 2011; Kett, Schmitt, and Wartzack 2017).

The research question of this contribution is rather practical:

RQ3.1 How to enable a user feedback loop allowing agile data-driven design road-mapping process?

During this Ph.D., we also contributed to Human Systems Integration (HSI) domain. This Ph.D. was a part of a project aiming to develop a novel multimodal cockpit for a semi-autonomous vehicle. We adapted a discrete-events simulation approach to the driver's cognitive and perceptive workload assessment. This research aimed to create a tool for the early-stage design optimization of a cockpit's functional architecture. This tool was developed, implemented in python, and tested successfully. However, this research is loosely coupled to the other contributions and, therefore, is not included in the manuscript's main body. We provide the conference paper full text reporting this research in the Appendix K.

4 RESEARCH METHODOLOGY

This chapter describes the research methodology: research framework, research methods, and validation approaches. This Ph.D. is in the design science domain. As it will be described in the following section 4.2, by nature, design research is subject to several pitfalls and methodological challenges. Scholars consistently report "lack of scientific rigor" (Blessing and Chakrabarti 2009) in the design science domain. Vermaas (2014) points that design research "is creating in a sense too many theories and models, which jeopardises the coherence of the discipline." Therefore, it is important to examine and discuss this aspect in details.

4.1 Design research as a science

In the context of this thesis, design should be understood as a set of "activities that actually generate and develop a product from a need, product idea or technology to the full documentation needed to realize the product and to fulfill the perceived needs of the user and other stakeholders" (Blessing and Chakrabarti 2009). Accordingly, design science is a discipline that studies design practices.

Vermaas (2014) classifies design theories into *descriptive*, *demarcating*, and *prescriptive*. *Descriptive* design theory aims at "describing design practices that are regularly taken as design." *Descriptive* design theory is scientific if it "binds together our knowledge of these regular design practices, and arrives at understanding, explanation and prediction of and about them" (Vermaas 2014). The major aim of *demarcating* design theories is in "fixing the borders of what is to be taken as design practices." Finally *prescriptive* design theory aims at "singling out particular types of design practices and positing favourable properties about these practices" (Vermaas 2014).

Many contributions in the field of design science (including this Ph.D. work) aim at creating support tools improving design practices, and therefore prescriptive. Prescriptiveness introduces some methodological difficulties. First, if a researcher sets favorable properties of a design practice, then results of the study may be prone to subjective biases (researchers' beliefs and assumptions influence the knowledge produced). Second, Vermaas (2014) note that prescriptive design theory may study design practices that are "not yet part of the world of experience." Simply put, these theories

propose new design practices. In this case, "the assessment whether it may be a scientific theory becomes more involved".

Vermaas (2014) argues that design theories usually combine *descriptive*, *demarcating*, and *prescriptive* properties. He concludes that mainly prescriptive design theory may be counted as scientific if it "minimally add understanding and explanation to the prediction that the singled out design practices have the posited properties."

Goldkuhl (2012) discusses three paradigms in information systems research: positivism aiming for explanation and prediction, interpretivism aiming for interpretation and understanding, and pragmatism aiming for intervention and change.

According to Goldkuhl (2012) "a pragmatist stance aiming for constructive knowledge that is appreciated for being useful in action." Coyne (1995) acknowledges the influence of rationalism on the design science and opposes to pragmatism, which in application to design methods means higher emphasis on human experience, subjective knowledge, and grants privilege to practice over theory. He claims that "the assumptions of rationalism are also evident in the design-methods movement, which sought to capture design expertise in process diagrams, to objectify the design process and make it explicit as an aid to collaboration and communication."

Epistemologically, design science is often associated with rationalism and with pragmatism (Coyne 1995; Hevner 2007). Hevner (2007) claims that "design science research is essentially pragmatic in nature due to its emphasis on relevance." Simply put, pragmatism evaluates knowledge by its applicability in practice. If some knowledge is useful, then it is valid. Such a perspective is quite legitimate in many cases, when the object of research is the complex socio-economic phenomenon. According to Farjoun, Ansell, and Boin (2015) "pragmatism is well suited to understanding the contemporary challenges of change and complexity especially as they play out across multiple levels of analysis."

However, validating a design theory taking the pragmatist stance may sometimes be problematic. Vermaas (2014) argues that such a validation of new prescriptive design theories "requires years-long experiments in which sufficiently large numbers of engineering firms abandon their established design practices in favour of the prescribed practices, and in which relevant contextual factors like international economic growth and the behaviour of competitor firms are kept controlled."

Therefore, case studies remain the most used instrument to validate the prescriptive

knowledge. Vermaas (2014) points out that as result "design theories and models are proposed, studied, developed and tested relatively independently from each other, which leads to a rich variety of such theories and models <...> and an associated fragmentation of design research."

To overcome this, Vermaas (2014) proposes one alternative, i.e., testing by sophisticated falsification, which is, however, has not yet been widely applied in the design community, where at least partial pragmatic validation (e.g., through case studies and interview with experts) is still very popular.

4.2 Design Research Methodology

Methodologically, we propose to base ourselves on a standard framework called Design Research Methodology (DRM) proposed by Blessing and Chakrabarti (2009). In their view, any design research to some extent should pursue two objectives: *understanding* and the *support* of the design practice:

"

- the formulation and validation of models and theories about the phenomenon of design with all its facets (people, product, knowledge/methods/tools, organization, micro-economy, and macroeconomy);
- the development and validation of support founded on these models and theories, in order to improve design practice, including education, and its outcomes." (Blessing and Chakrabarti 2009)

DRM framework of Blessing and Chakrabarti (2009) contains four stages: Research clarification, Descriptive Study I, Prescriptive Study, Descriptive Study II.

The objective of the Research clarification (RC) is to find a realistic research goal and success criteria (e.g., decrease time-to-market).

In Descriptive Study I (DS I) researcher carefully studies the existing situation. In this stage factors influencing the success criteria are identified and classified.

In terms of research methods, the first two stages most often compose literature reviews and interviews.

During the Prescriptive Study (PS), researchers propose their approach (a support) to

address the problem. Often, during this stage, some software prototype or an algorithm is created (Blessing and Chakrabarti 2009).

The Descriptive Study II (DS II) aims at measuring the impact of the proposed solution on the design process with respect to defined criteria.

Blessing and Chakrabarti (2009) specifies, that a given research project rarely completes all the stages of DRM. Depending on the scope, some may finish on the comprehensive elaboration of Descriptive Study I or a Prescriptive Study and be later resumed by colleagues (often from a different research group). If the scope is not too broad, all stages may be covered, and then possibly refined in-depth by colleagues.

Often, to achieve the global research objective, it should be decomposed to smaller facets with limited scope, and those sub-objectives can go through all the stages of DRM. Meanwhile in the global scope, only some phases of DRM are accomplished.

Application of the DRM to our research work and its individual contributions is schematically represented on figure 4.1.

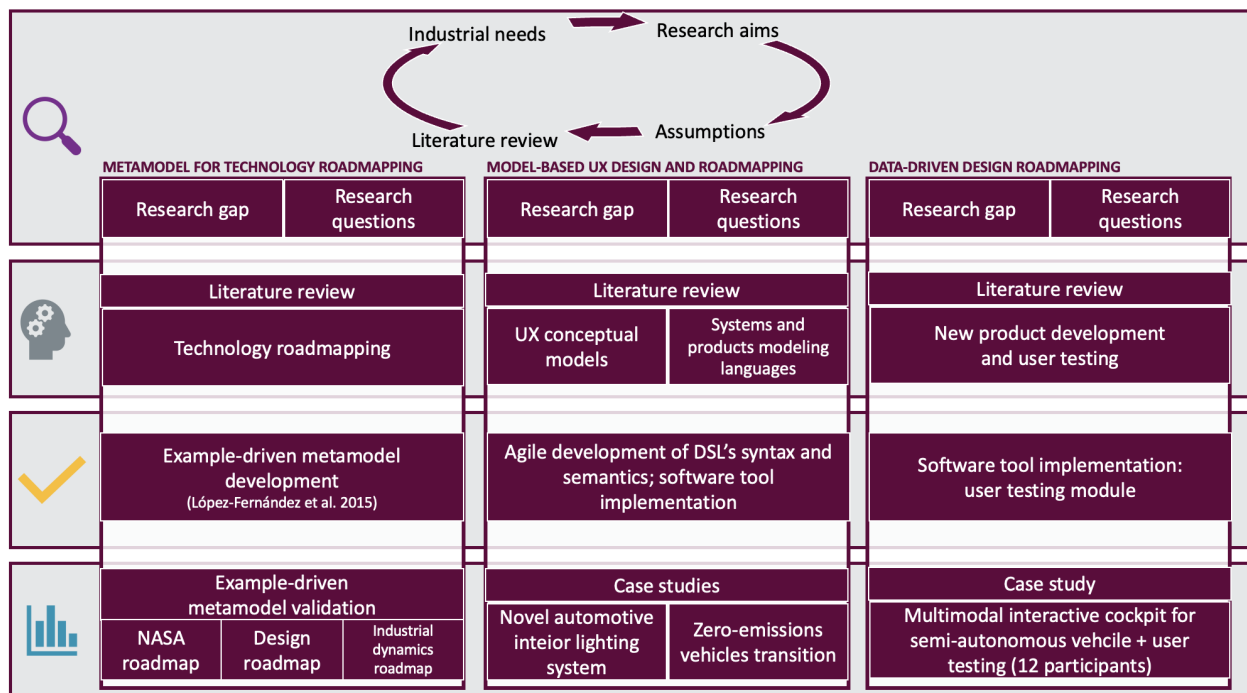


Figure 4.1: DRM applied to the thesis

To summarize, DRM does not oblige researchers to go through all the stages of a framework RC → DS I → PS → DS II, but it is important to define scope as soon as

possible. We applied such a reasoning to this Ph.D. thesis and it will be described in the following section.

4.3 Research methods and validation

In terms of the global research objective, the validity of an approach was assessed by a number of case studies. Case study is the most widely used qualitative research method in the information systems research (Darke, Shanks, and Broadbent 1998), defined as "an empirical enquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident" and it "relies on multiple sources of evidence" (Yin 2009). During case studies, our goal was to make sure that the real-world data are represented in our proposed DSL completely and adequately.

Additionally, our approach was embodied into functional cloud-based software. The software cannot be considered as a proof of validity by itself, but in case of its adoption in the company, future researchers can gather more pieces of evidence of the pragmatic validity of a proposed approach. Also, the fact of the implementation in a programming language, and that this code compiles and functions as envisioned, partially protects it from the logical flaws and inconsistencies.

Now, we briefly review validation methods of three contributions constituting this Ph.D.

Validation of the first contribution is based on the example-driven metamodel validation procedure (López-Fernández, Guerra, and Lara 2014), which exhibits properties of a formal proof. We use a modern language workbench (a specialized software for creating DSLs). Formality and strictness of this environment guarantees that the sequential instantiation of a metamodel to DSL, and of DSL to the user data were unambiguous and rigorous (see Appendixes B, C, D, E, F, G, H).

In the second contribution, we use unstructured interviews to establish the requirements and, after the support tool was completed, the initial assessment. The process of the methodology and the support tool design followed the agile software development paradigm (collect requirements, prototype, demonstrate, repeat). Also, for the construction of the DSL's syntax and semantics, we synthesized conceptual models and syntaxes published in the literature (i.e., review-based study).

We used two case studies to validate the second contribution. In the first case study, which has happened in Saint-Gobain, we ensured that our DSL represents correctly concepts of the UXD layer, which was elicited in the frame of the standard design thinking procedure. Unfortunately, results of this case study cannot be disclosed. Therefore, we created the second case study illustrating the approach entirely, at all levels: marketing, UXD, engineering, technology.

For the validation of the third contribution, we use CMI case study. The case study execution was again supported by our tool: 22 people was invited to give their feedback on a CMI cockpit, out of which 12 answered our questionnaire. The case study demonstrates close integration of the user testing procedure with our DSL, and identifies a number of insights that can be used in the process of product planning.

The research gaps, research questions, and contributions are summarized in figure 4.2.

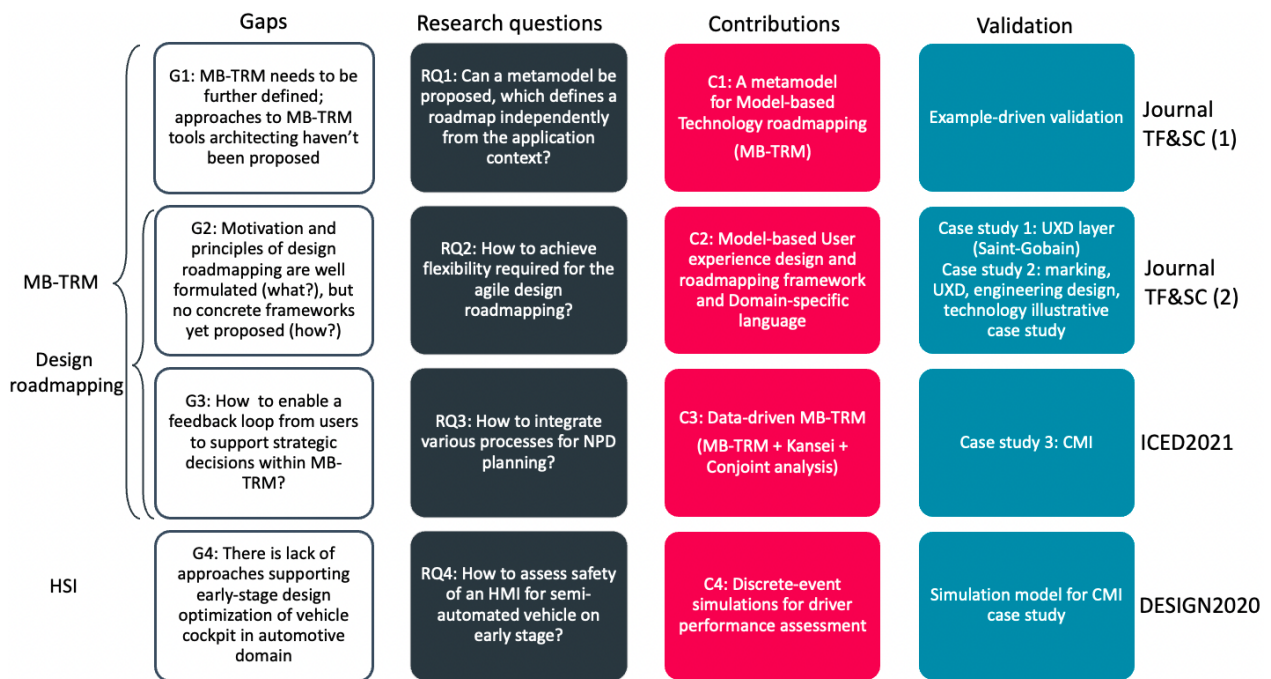


Figure 4.2: Structure of contributions

4.4 Design theory classification: process view versus product view

Blessing and Chakrabarti (2009) classifies design theories into process view and product view. In the process view, a process determines a product. To design better

products, one should improve a process in which a product was created (Blessing and Chakrabarti 2009). Respectively, in the product view, suitable processes are selected based on properties of a desired product.

Clearly, this Ph.D. takes the process view. Roadmapping and UXD are processes, and supporting them with a software will improve any products (product families) designed in their frame.

However, reader will not find in this dissertation any other process diagrams than the one on figure 6.9. The reason is, and this is one of our assumptions, design process (and even a strategic planning of NPD) cannot be conducted in a sequential phase-gate manner (see section 3.4). It shall be iterative and agile (experiment-refine) due to its uncertain and fast-changing nature.

Therefore, our focus, as in any agile framework, is not procedural (which processes to execute and in which sequence), but structural (which concepts should we clarify, which information and artifacts should we produce, what are the relations between concepts, etc.) When this defined, the elicitation of this information can happen in any unambiguous order, depending on a concrete task priority and contributors' availability. Moreover, some tasks tend to reiterate in response to new inputs. This view is supported by Fagan (2005), who highlights pragmatic orientation of agile developers opposing to rationalists' practices noting that "agile methods eschew the creation of design documents, except for those such as story and task cards that are temporary and only serve as a vehicle for planning and communication until they are translated into the actual code. "

5 METAMODEL OF THE MODEL-BASED TECHNOLOGY ROADMAPPING

Publications

The materials of this chapter were published in Ilya Yuskevich et al. (Dec. 2021a). "A metamodel of an informational structure for model-based technology roadmapping." In: *Technological Forecasting and Social Change* 173, p. 121103. issn: 00401625. doi: [10.1016/j.techfore.2021.121103](https://doi.org/10.1016/j.techfore.2021.121103).

Open access

Authors' version before review can also be found in <https://hal.archives-ouvertes.fr/hal-03330657/document>.

Abstract: Recent contributions in the field of technology roadmapping often aim to apply various numerical models and tools to facilitate the roadmapping process and enrich its outcomes. This trend resulted in the emergence of so-called model-based technology roadmapping. We consider it as the future development of the traditional document-based paradigm. One of the general approaches to support the model-based roadmapping is to develop a roadmap's metamodel that would define it independently from the application context and link it to the existing roadmapping literature. In this paper, we attempt to create such a metamodel by generalizing and formalizing existing document-based roadmaps. We validate our metamodel via reproducing three very different roadmaps from the literature, not included in the set of roadmaps from which the metamodel was created.

5.1 Introduction

Technology roadmapping (TRM) is used for strategic planning of technologies, products, services, systems, etc. in a large variety of industrial contexts, from technology startups to space agencies. TRM is a successful tool for elicitation and communication of long-term plans internally and externally, and it helps organizations to stay focused on the essential strategic goals over the long run (Albright and Kappel 2003).

Recently, it was proposed (Knoll, Golkar, and Weck 2018; Golkar and Garzaniti 2020) to consider TRM from Model-based systems engineering perspective, under the name of

Model-based technology roadmapping (MB-TRM). MB-TRM potentially enables a number of features beneficial for roadmapping practice: more efficient data management, asynchronous iterative process of roadmap development, access control, consideration of multiple alternatives, partial automation, various data consistency checks, etc.

MB-TRM proposal implies that roadmap data should be expressed in some standard or specially developed DSL. However, due to the extreme variety of TRM applications, it is very unlikely that a hypothetical all-purpose DSL for all possible roadmapping applications can satisfy all requirements. Therefore, the task of building a metamodel that will define a family of DSLs for roadmapping is of current interest.

This chapter addresses research gap G1: lack of the metamodel, that would define the structure of the model-based roadmap independently from the application context and may be used as a starting point for roadmapping software development. It contributes to the model-based technology roadmapping domain of technology management. In this research, we used, at first, the review of existing roadmapping literature to classify the roadmap's structural elements. For the roadmap construction, we used inductive reasoning and modeling in a modern language workbench. Then, for the validation, we applied case studies and an ad-hoc formal approach using a modern language workbench. Therefore, this contribution accomplishes all phases of DRM (section 4.2): RC (review-based) → DS I (review-based, identification of common elements) → PS (comprehensive, metamodeling) → DS II (comprehensive, metamodel validation).

5.2 Related literature

In this section only the most important contributions are listed. The comprehensive literature review is provided in the sections 3.1, 3.3.

This contribution is largely based on Phaal and Muller (2009), who laid foundation for formal architecting of technology roadmaps, Knoll, Golkar, and Weck (2018), Gradini et al. (2019), and Golkar and Garzaniti (2020), who established MB-TRM research direction, and López-Fernández et al. (2015), who developed a formal example-driven approach for the metamodels validation.

5.3 Research methodology

To address metamodel development and validation, we propose to adapt the approach of metaontology O2 of Gangemi et al. (2006) (Figure 3.10). On this level, a metamodel expresses the intended conceptualization of some domain by rational agent(s). The intended conceptualization is influenced by pragmatic objectives and semantic space, which, generally speaking, may differ between rational agents. Typically, metamodeling then becomes an exercise in finding a social consensus on the "right" metamodel.

Our objective here is not to find a metamodel for TRM based on social consensus between rational agents. Instead, our objective is to construct a sufficiently general metamodel for modeling a wide range of existing roadmaps.

More specifically, we will focus on the verification of the following statements:

- S1 DSLs instantiated from our metamodel correctly express several publicly available technology roadmaps.
- S2 The roadmaps that we selected for validation belong to significantly different contexts.
- S3 Our metamodel can be used as a prototype of an MB-TRM software system, supporting CRUD-operations and graph-traversal functions.

If S1 holds, then our metamodel is internally consistent (RQ1.4, see section 3.5) and compatible with definitions of a technology roadmap (RQ1.1). S2 asserts that our metamodel constitutes a useful generalization for a practice of TRM (RQ1.2). S3 is a formalized version of RQ1.3.

We propose to base ourselves on the example-driven metamodel validation procedure from (López-Fernández et al. 2015). López-Fernández, Guerra, and Lara (2014) highlight two broad classes of validation objectives: "are we building the right metamodel?" and "are we building the metamodel right?" The former aims to validate the correspondence of metamodel concepts to definitions and meanings accepted in the domain of interest. The latter's goal is to verify that the metamodel is consistent and meets the standards and best practices of MDE, ensuring that a metamodel is reusable.

López-Fernández, Guerra, and Lara (2014) and López-Fernández et al. (2015) proposed

an example-driven procedure of metamodel construction and validation. They also developed a tool that enables semi-automatic metamodel construction from several model fragments created by domain experts. The resulting metamodel can then be validated by testing it against valid and invalid models.

Since metamodels are built for practical purposes, they can also be validated by direct instantiation for a specific use case. As an example, García-Holgado and García-Peñalvo (2017) performed such validation for learning ecosystems.

The process of metamodel construction and validation is shown in Fig. 5.1. The process starts with a review of published document-based technology roadmaps. During the review, a metamodel creator (the author of this manuscript) identifies common elements among various documents, conceptualizes and classifies them. The first version of a metamodel (level M2 of MOF architecture) is a product of this step. In the second step, we select several roadmaps with significantly different structures and application domains. Then, for each of these roadmaps, the application domain-specific model (M1) is instantiated from a metamodel (M2). In the third step, each roadmap (user data M0) is instantiated from a corresponding application domain-specific model (M1). Produced data should be identical to the data contained in the selected documents. If we failed to go through all the steps, we can modify a metamodel and repeat the instantiation. The process needs to be repeated iteratively until a metamodel is mature enough.

The process of choosing roadmaps for this study prone to selection bias. There were no specific measures taken to mitigate this risk. This is one of the major limitations of our methodology. We picked published roadmaps based on the following criteria:

- To accurately conceptualize the domain of interest, the completeness of the information is an important consideration. However, in the most cases, the authors tend not to disclose the sensitive roadmap information fully, usually not going beyond schematic representation. Hence, the pool of available options is, in fact, quite limited;
- Generalizability is an essential property of any valid metamodel. We attempted to select roadmaps belonging to different industries, made in organizations of different sizes and positions, served to different purposes.

To support the procedure, we selected a tool that assists in the creation and composition of DSLs (a language workbench). We have chosen JetBrains MPS — a modern

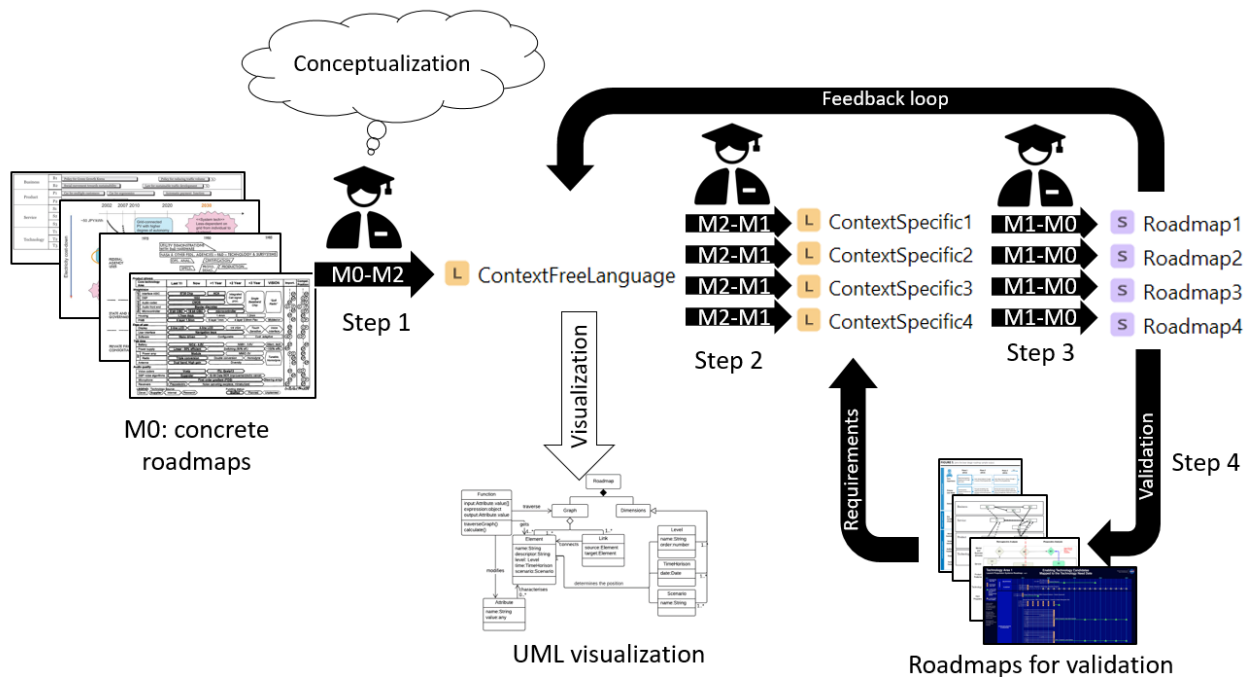


Figure 5.1: The iterative process of metamodel construction and validation (Yuskevich et al. 2021a)

language workbench¹ that allows us to define an abstract and concrete syntax (structure and editor in MPS-terms) of a DSL and write code in a new language (create models) with projectional editing. The projectional editor allows direct manipulation of an abstract syntax tree (AST) of a model. The latter is important for metamodel validation — any expression incompatible with the syntax defined for DSL cannot modify an AST of a model. In this way, we automatically ensure the internal consistency of a metamodel.

Language composition is presented in Fig. 5.2. Both the application domain-independent language (metamodel M2) and the application domain-specific language (model M1) are written in the Base Language — a Java-like language for DSL definition in MPS. The operation of metamodel instantiation is equivalent to language extension in JetBrains MPS. User data (in our case, a roadmap) is defined using the concrete and abstract syntax of application domain-specific language (M1).

We chose a language workbench in which an AST is represented in textual form by default. For the sake of clarity, we will present an abstract syntax of the DSL in the form of

¹Note: for more information on MPS, visit [Glossary - Help | MPS \(2021\)](#).

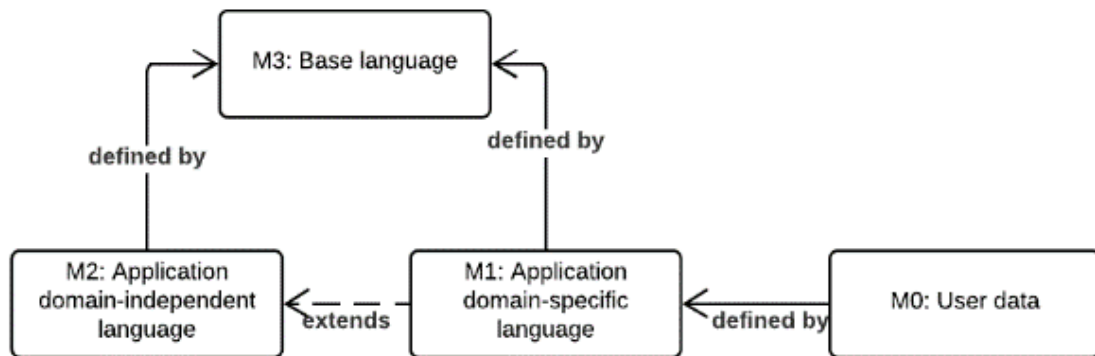


Figure 5.2: Language composition structure (Yuskevich et al. 2021a)

a UML class diagram (for concepts) augmented with OCL expressions (for constraints); and M0 roadmaps in a textual form aligned with concrete syntax of a corresponding DSL.

5.4 The metamodel

We conducted an extensive literature review to identify common elements, concepts and establish the classification (see Table 5.1).

This analysis supports the conceptual representation of a technology roadmap as a layered structure with a hierarchy of elements and attributes, connected with causal links and composing a directed graph. This view is well aligned with the architectural framework of Phaal and Muller (2009). Therefore, we took this structure as a foundation of our metamodel.

The identified concepts represent the basis of the proposed MB-TRM metamodel (see Figure 5.3, and Appendix B for details). The roadmap informational structure is a directed graph consisting of elements and links. Elements of a roadmap exist in the three-dimensional space defined by time, level, and scenario dimensions. Elements may be classified (modeled as "**Element** references to **ElementClass**"); classes can be organized into hierarchies to form taxonomies (each **ElementClass** can be a parent of other **ElementClass**). Each **Element** can have several attributes (**Attribute**) that take quantitative or qualitative values (**AttributeValue**). In some applications, a corresponding element class defines the set of attributes that **Element** can exhibit.

Table 5.1: Document-based technology roadmap commonalities (Yuskevich et al. 2021a)

Reference	Context	Levels	Elements	Attributes	Links
(Albers, Krämer, and Arslan 2017)	Strategic roadmap in automotive domain	Market, Product, Technology, Resources	Skill, Capability, Competency, Need, Product, Technology	-	Evolves to, Satisfies need, Requires resource
(Albright and Kappel 2003)	Product and technology roadmap in telecom. domain	Market, Product, Technology, Risk	Product/market driver, Market trend, Market segment, Product, Technology, Capability, Event, Risk	Market growth, Market share, Experience curve, Priority, Price, Risk consequence, Competitive position	Requires, Targets market niche, Evolves to
(Scalice et al. 2015)	TRM for modular products	Product, Market, Module	Product, Platform, Market, Technology, Module	Differentiation feature, Lifecycle stage, Lifestyle, Benefits, Driver importance weight, Ranking	-
(Toro-Jarrín, Ponce-Jaramillo, and Güemes-Castorena 2016)	Business Model Canvas and TRM integrated	Market, Product, Technology	Political strategy, Macroeconomic forces, Key trends, Market and Industry forces, New entrants, Substitutes, Technology, Regulatory, Cost trends	Economy growth, Employment rate, Market volume, Features, Product size	-
(Al-Ali and Phaal 2019)	TRM for digital transformation	Digital opportunity, Digital initiative, Resources	Disruptive/ Radical opportunities, Internal efficiencies, MVP, Legacy system, Resources, Digital technology	-	Replace, Interface, Migrate
(Zhang et al. 2016)	TRM for technical intelligence in photovoltaic industry	Material, Method, Device	Problem, Solution, Component, Implication	Efficiency, Stability, Losses, etc.	Solve, Relate, Improve, Upgrade

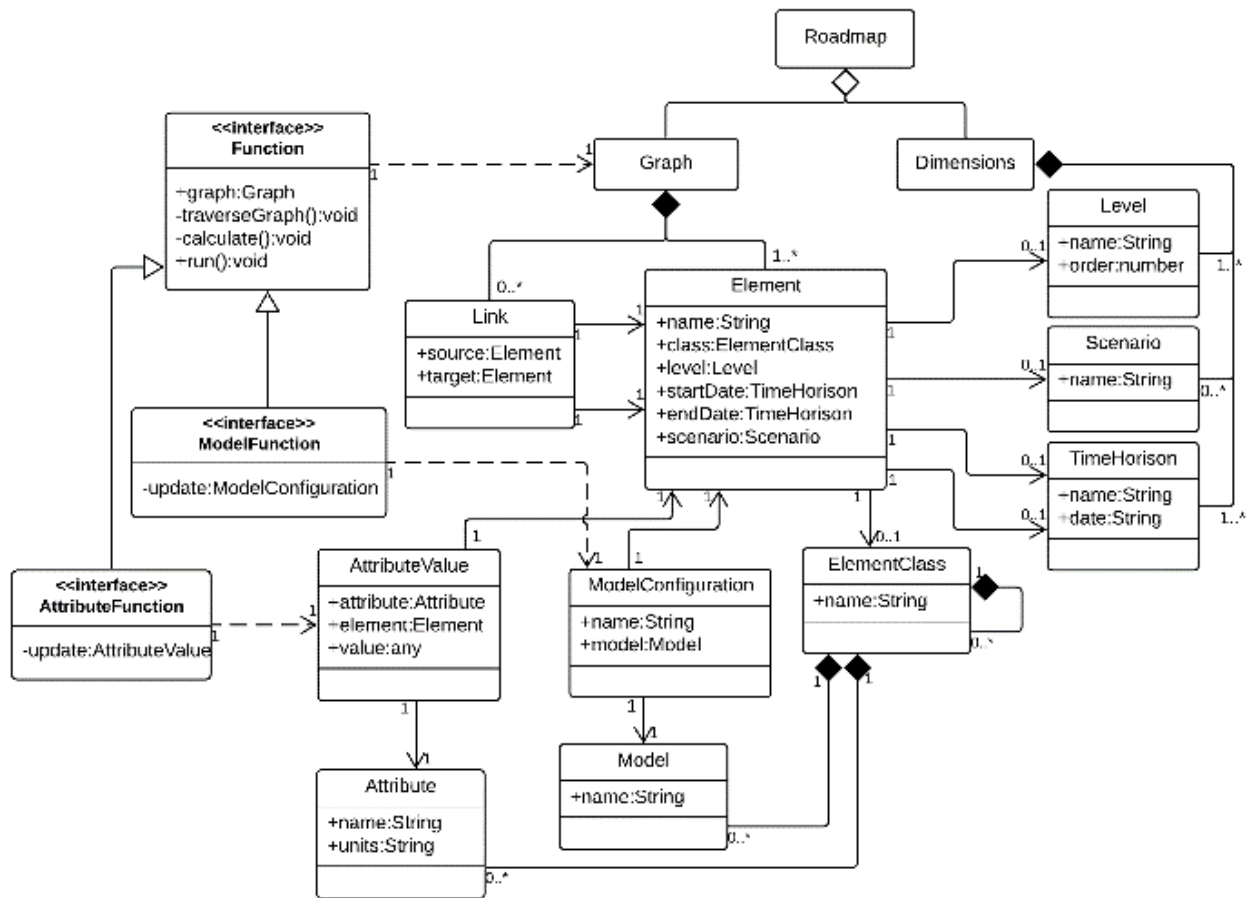


Figure 5.3: Metamodel for MB-TRM (Yuskevich et al. 2021a)

We believe that the majority of document-based TRM can be derived from this structure. To realize the full potential of the MB-TRM approach, we added functions and models. When the set of quantitative or qualitative attributes is not sufficient to characterize elements, we may employ models, e.g., structural, parametric, or stochastic. For their part, functions operate on the entire directed graph, execute models, perform calculations of aggregated metrics, and various checks. Functions and models are the exclusive features of MB-TRM and cannot be derived from document-based roadmaps.

The geometric interpretation of the metamodel is shown in Fig. 5.4 and 5.5. Points represent **Elements**, i.e., markets, systems, and technologies. Horizontal causal links represent the evolution of elements. Cross-level links model interdependencies.

Depending on the organization, the graph's topology illustrated by Fig. 5.4 and 5.5

may be fairly complex. Nodes at each level represent versions/configurations of just one system, technology, or market niche for different moments of time multiplied by the number of alternative realities under consideration (so, three scenarios for four moments of time produce twelve distinct states of the market). Each complex system is composed of hundreds of components and depends on a number of technologies. Technological giants and agencies typically manage tens of complex systems or products/services in the different stages of a life cycle, which are also connected either directly or through the use of the same pool of components/technologies. Connection types are numerous (requires, depends on, excludes, enhances, enables, integrates into, triggers, targets, etc). Moreover, so far, we were mainly talking about roadmap as a model, but each node (or **Element**) has a model itself (ranging from parametric model to a digital mockup). If we go further, such a system would naturally integrate with project management software. In this view, companies' strategic targets can be hierarchically decomposed to project's goals or even to individual tasks. One may argue on the last point that this is not what strategic tools currently doing and whether they should do it at all? However, our main idea here is to illustrate what this type of systems potentially will be able to do. The more data across an enterprise will be collected and aggregated, the more informed strategic decisions are.

5.5 Validation

To validate the metamodel, we additionally selected three roadmaps (see table 5.2) to check whether these roadmaps can be modeled based on the metamodel. We created an abstract and concrete syntax of a DSL and user data layer, precisely representing targeted roadmaps for each of them.

The instantiation of a metamodel for the NASA roadmap is shown in figure 5.6 (find more details in Appendix C). **TechnologyArea**, **TechnologyFamily**, **TechnologyClass**, **Technology**, and **Capability** are instances of **ElementClass**. They represent NASA's technology taxonomy. **Technology** and **Capability** define the set of attributes that concrete elements of a roadmap — **TechnologyCurrent**, **TechnologyTarget**, **CapabilityCurrent**, and **CapabilityTarget** — can take. Another specificity of NASA roadmap is that these elements do not refer to the time dimension directly (note cardinality 0..1 between **Element** and **TimeHorizon** in the metamodel). Instead, they are connected with multiple **MissionNeeds** by **Enables** and **Enhances** links. This is how different due dates for the same performance target are modeled.

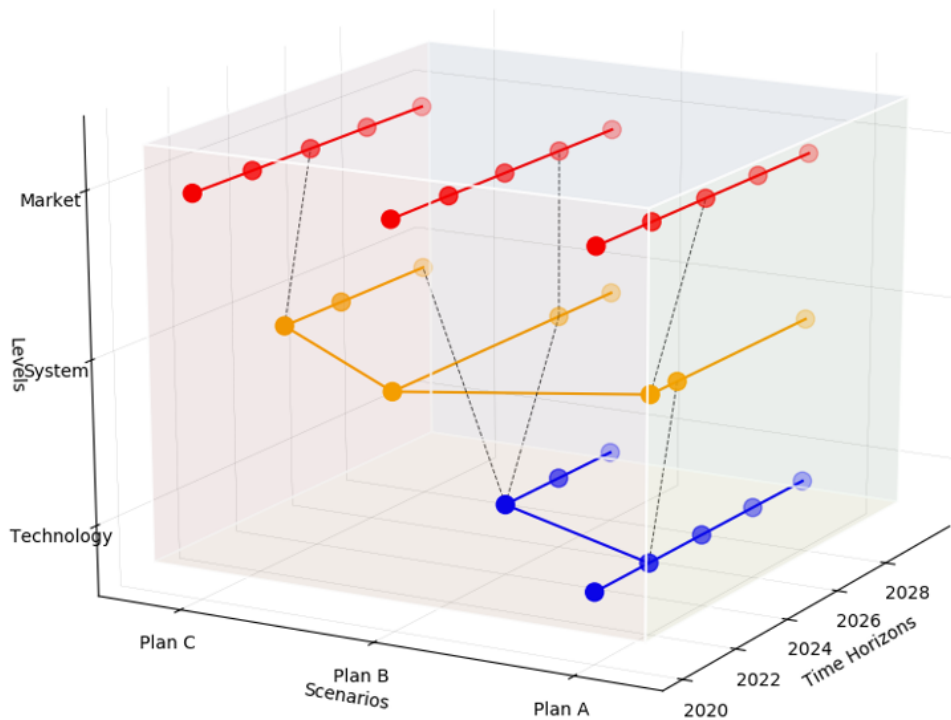


Figure 5.4: Geometric interpretation (Yuskevich et al. 2021a)

Table 5.2: Roadmaps selected for validation (Yuskevich et al. 2021a)

Reference	Context	Structural challenge
(2015 NASA Technology Roadmaps (Archive) 2015)	Technology roadmap of a space agency. Horizon of planning — 30 years.	Relatively complex taxonomy of elements. An enormous amount of data — more than 2000 pages.
(Kim, Beckman, and Agogino 2018)	UX-centered product, design, and technology roadmap of a single-product startup. Horizon of planning — 3 years	Simple structure, but data represent subtle and qualitative concepts — user needs and experiences.
(Geum, Lee, and Park 2014)	Emerging car-sharing business, heavily dependent on the turbulent external environment. Horizon of planning — 30 months.	Alternative realities (scenarios). The abundance of KPIs. A dynamic model of KPIs complements a roadmap..

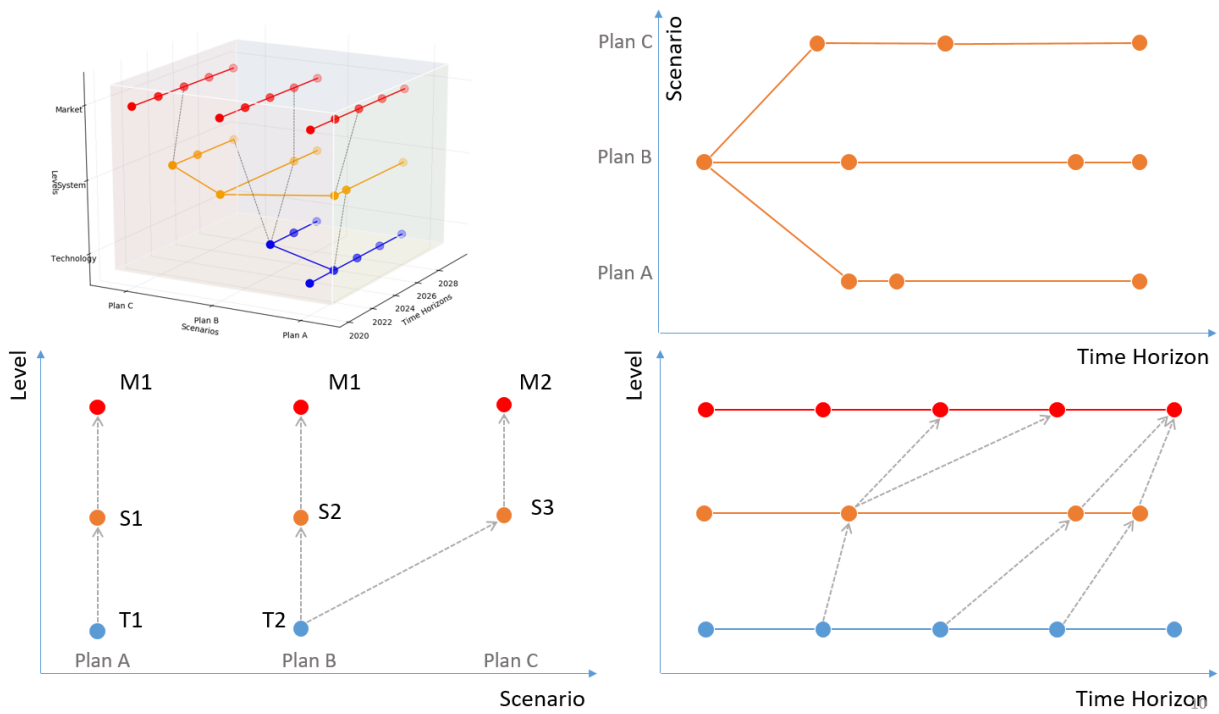


Figure 5.5: Geometric interpretation — cross-sections (Yuskevich et al. 2021a)

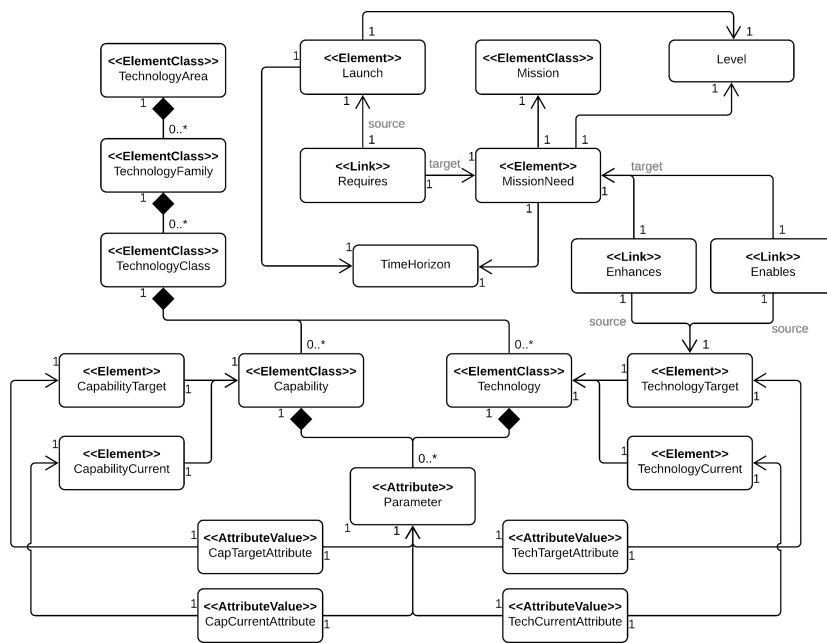


Figure 5.6: Instantiation of the metamodel for NASA TRM (Yuskevich et al. 2021a)

Concrete syntax created for this usecase is given in Figure 5.7-5.9 (detailed information also presented in Appendix D). For obvious reasons, we modeled only a small portion of the NASA roadmap. However, we can safely generalize our results on the entire document, thanks to the NASA roadmap’s formal structure.

```

Technology Area 1 Launch Propulsion Systems
  Technology Family 1.1 Solid Rocket Propulsion Systems
    Technology Class 1.1.1 Propellants
    Technology Class 1.1.2 Case Material
    Technology Class 1.1.3 Nozzle System
    Technology Class 1.1.4 Hybrid Rocket
    Technology Class 1.1.5 Fundamental Solid Propulsion Technologies
    Technology Class 1.1.6 Integrated Solid Motor Systems
    Technology Class 1.1.7 Liner and Insulation
  Technology Family 1.2 Liquid Rocket Propulsion Systems
  ...
  Technology Family 1.3 Air-Breathing Propulsion Systems
  ...
  Technology Family 1.4 Ancillary Propulsion Systems
  ...
  Technology Family 1.5 Unconventional and Other Propulsion Systems
  ...
  Technology Family 1.6 Balloon Systems
  ...
Technology Area 2 In-Space Propulsion Technologies

```

Figure 5.7: Concrete syntax for NASA TRM — classifier (Yuskevich et al. 2021a)

```

Insulating/Ablative Sprayable Liner is a Technology
with attributes:
  Current case material weigh units: %
Technology Description:
  This is a liner capable of insulative properties similar to fiber-filled nitrile butadiene rubi
Technology Challenge: Producing an acceptable sprayable liner is a technical challenge.
Technology Development Dependent Upon Basic Research: false
Technology Class: 1.1.7 Liner and Insulation
Classifier: 1.1.7.2
Technology State of the Art:
  This technology is under development in Space Launch System (SLS) Booster project. Current issues
  TRL: 5
define attribute values:
  [redacted]
Tech n Technology Parameter (AttributeValue in M2_AbstractLanguage) lation thickness
  TRL: 9
  define attribute values:
    Current case material weigh 25 %
A sprayable liner is a Capability
with attributes:
  Current case material weigh units: %
Capability Description:
  The sprayable liner can maintain the solid rocket booster motor case internal temperature below

```

Figure 5.8: Concrete syntax for NASA TRM — technology editor (Yuskevich et al. 2021a)

The instantiation of the metamodel for the design roadmap is shown on Figure 5.10 (for the details see Appendix E). The structure consists of 9 **Element** instances and 3

```

DRM5 Asteroid Redirect - Robotic spacecraft is a Mission

create Launch Launch_DRM5
  ( start date: 2018 level: Human exploration and Operations scenario: <no scenario> )

create Need for DRM5 Asteroid Redirect - Robotic spacecraft Need_DRM5
  ( epoch: 2015 level: Human exploration and Operations scenario: <no scenario> )

1.1.7.1 Performance Goal enables Need_DRM5
1.1.7.2 Performance Goal enables Need_DRM5

Press [Enter] to define enhancer links

Launch_DRM5 requires Need_DRM5

```

Figure 5.9: Concrete syntax for NASA TRM — roadmap editor (Yuskevich et al. 2021a)

Link instances. The meaning of each **Element** is characterized by a textual description.

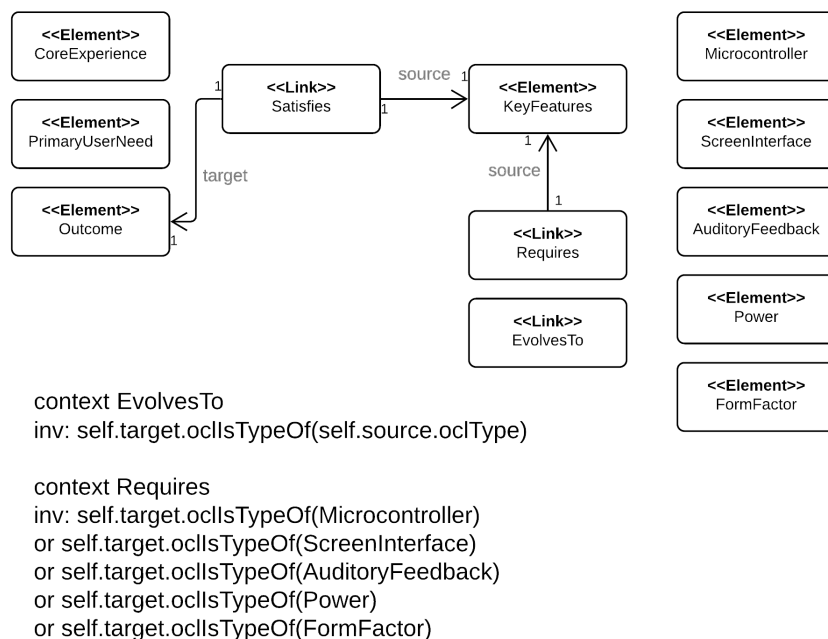


Figure 5.10: Instantiation of the metamodel for Design roadmap (Yuskevich et al. 2021a)

Some of the classes are not connected (Figure 5.10) to others by association links. It does not mean that they are unrelated to other parts of the model. The model of level M1 inherits the relationships defined by the metamodel (level M2). These general relationships may be additionally specified on the level M1 with OCL-expressions.

For instance, a target and a source of the link **EvolvesTo** should be of the same concept, as it signifies the evolution of needs/products/technologies. Furthermore, link **Requires** connects **KeyFeatures** and any **Element** that models technology. At the same time, link **Satisfies** can go strictly from **KeyFeatures** to **Outcomes**. Fragments of a concrete syntax for the design roadmap are given in Figures 5.11, 5.12.

The 'code completion' (e.g., Design, Product and Technology levels on Figure 5.11, fully this data is in Appendix F) and the 'intention actions' (e.g., 'Technology Parameter' on Figure 5.8) dialogs are shown on these and the following figures. This is done to illustrate that in MPS we do not write code to be parsed and compiled but directly manipulate AST, which is the major premise for the validation: strict compliance of every written expression to defined abstract syntax is ensured.

```

Core experiences:
create Core Experience Core experience 1
  ( start date: 2012 level: Design Roadmap scenario: <no scenario> )
  description: Make learning about diabetes fun through game play
create Core Experience Core experience 2
  ( start date: 2013 level: Design Roadmap scenario: <no scenario> )
  description: Learn about diabetes through a combo of free and guided play
create Core Experience Core experience 3
  ( start date: 2015 level: Design Roadmap scenario: <no scenario> )
  description: Learn about c
  Design Roadmap ^levels (M1_2_DesignRoadmap.null)
  Product Generation ^levels (M1_2_DesignRoadmap.null)
  Technology Development ^levels (M1_2_DesignRoadmap.null)
Primary User Needs:
create Primary User Need User need 1
  ( start date: 2012 level: Design Roadmap scenario: <no scenario> )
  description: Learning about diabetes should be fun and easy
create Primary User Need User need 2

```

Figure 5.11: Concrete syntax for Design roadmap — Elements (Yuskevich et al. 2021a)

```

Form-factor:
create Form Factor A Bear shell-1 (Teddy Bear)
  ( start date: 2012 level: Technology Development scenario: <no scenario> )
  description:
create Form Factor A Bear shell-2 (Teddy Bear)
  ( start date: 2013 level: Technology Development scenario: <no scenario> )
  description: easy to clean Asthma-friendly fabric

Links:
Generation 1 satisfies Outcome 1
Generation 1 requires Arduino
Generation 2 satisfies Outcome 2
Generation 3 requires Co
Generation 2 requires Co
Generation 1 evolves to
Generation 2 evolves to Generation 3
  Generation 1 ^key_features (M1_2_DesignRoadmap.null)
  Generation 2 ^key_features (M1_2_DesignRoadmap.null)
  Generation 3 ^key_features (M1_2_DesignRoadmap.null)

```

Figure 5.12: Concrete syntax for Design roadmap — Links (Yuskevich et al. 2021a)

Finally, the instantiation of a metamodel for the emerging car-sharing business (Geum, Lee, and Park 2014) is shown in figure 5.13 (also in Appendix G). This is an example of

a roadmap modeled in a three-dimensional space. **Casual** links connect **Business-Drivers, Products, Services, and Technologies** either within the same scenario or produce scenario branching by connecting two **Elements** placed in distinct scenario planes. In the original paper of Geum, Lee, and Park (2014), such a structure was used to launch a dynamic simulation to calculate business KPI's for pessimistic/-neutral/optimistic scenarios. The interdependencies between KPI's and elements of a roadmap were defined by 'causal and loop diagram.' Our metamodel is compatible with such a problem setting: a mutual influence between roadmap elements and KPI's is established by the Influence link, the transfer function of each KPI — in the **KpiModel**, and initial values — in the **KpiModelValues**. The simulation is executed by **DynamicSimulation** function that traverses graphs and iteratively updates **Kpi-ModelValues**.

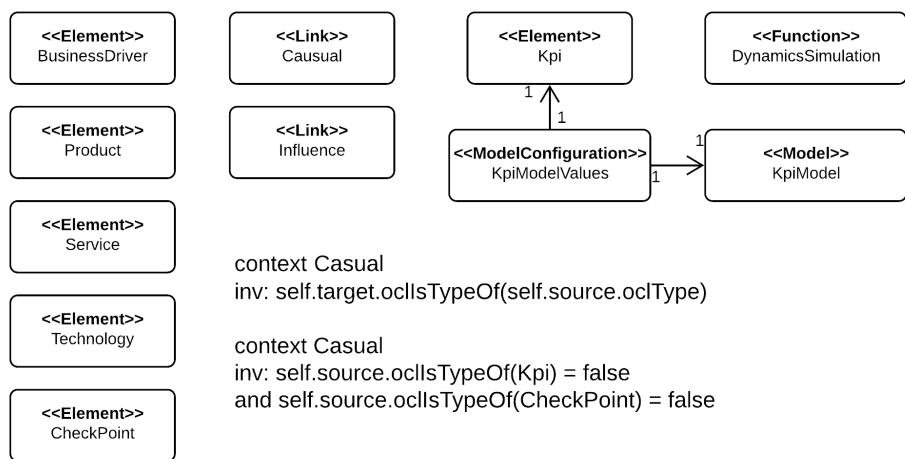


Figure 5.13: Instantiation of the metamodel for emerging car-sharing business roadmap (Yuskevich et al. 2021a)

The examples of a concrete syntax for this usecase are in figures 5.14-5.16 (see also Appendix H).

```

Add Product Delete
create Product Development of environmental-friendly car
  ( start date: Jan-2010 end date: Mar-2011 level: Product scenario: Neutral )
create Product Automatic payment function
  ( start date: Jul-2011 end date: Jan-2012 level: Product scenario: Neutral )
create Product Car for ergonomics
  ( start date: Jul-2010 end date: May-2011 level: Product scenario: Neutral )
create Product Car for multiple customers
  ( start date: Jan-2010 end date: Jul-2010 level: Product scenario: Neutral )
Add Service Delete
create Service e-Tracking service
  ( start date: Jul-2011 end date: Apr-2012 level: Service scenario: Neutral )
create Service Automatic return service
  ( start date: May-2011 end date: Nov-2011 level: Service scenario: Neutral )
create Service Public Transp. notification

```

Figure 5.14: Elements of a roadmap definition (Yuskevich et al. 2021a)

```

Policy for Green Growth Korea causes Law for sustainable traffic development [ Neutral -> Optimistic ]
Policy for Green Growth Korea [ Neutral ] causes Policy for reducing traffic volume [ Neutral ]
Social movement toward sustainability causes Law for sustainable traffic development [ Neutral -> Optimist
Social movement toward sustainability [ Neutral ] causes Policy for reducing traffic volume [ Neutral ]
Policy for reducing traffic volume [ Optimistic ] causes Policy for improving parking lot [ Optimistic ]
Law for sustainable traffic development [ Optimistic ] causes Law for tax deduction for car-sharing [ Opti
Law for tax deduction for car-sharing ^elements (M1_4_ScenarioPlanning.sandbox.Optimisti

```

Figure 5.15: Causal links definition (Yuskevich et al. 2021a)

```

create Key Performance Indicator Use of Car-sharing service model: F1
create Key Performance Indicator Total revenue model: F2
create Key Performance Indicator Energy Consumption model: F3
Social movement toward sustainability influence Use of Car-sharing service
Level Of Company Motivation influence Use of Car-sharing service
Use of Car-sharing service influence Improvement of traffic environment
Use of Car-sharing service influence Total revenue
Use of Car-sharing service
Use of Car-sharing service
Use of Car-sharing service

```

- Public Transp. notification
- Reduction of Environmental burden
- Seat customization service for multi-users
- Service for individuals
- Social movement toward sustainability
- Total revenue
- Use of Car-sharing service
- Weak Policy for Green Growth Korea

Figure 5.16: Dynamic model set up (Yuskevich et al. 2021a)

5.6 Discussion

During the validation, we were able to assert all three of our statements (S1-S3).

S1: we successfully created three DSLs that could precisely represent the information contained in the three publicly available roadmaps. By doing so, the internal consistency of our metamodel is validated. Second, we demonstrated the ability of the proposed abstract syntax to describe real roadmaps correctly.

S2: we deliberately selected extremely different roadmaps: the roadmap of the space agency versus the roadmaps of the startups; the descriptive roadmaps used for external and internal communication versus the executable roadmap used for simulation. By illustrating different case studies, we aimed to show that our metamodel applies to a wide range of contexts.

S3: our metamodel complies with MB-TRM principles and introduces concepts that extend conventional technology roadmaps: element models and graph traversal functions.

However, several research limitations can be identified and discussed. Being not a natural phenomenon but a kind of collective pragmatic knowledge, TRM's notion can be subject to different interpretations. Therefore, we have opted for the analysis of the scientific literature instead of seeking expert opinion. Based on that, we are convinced that our metamodel corresponds to what most academics mean by this notion. At the same time, we are aware of the possibility of alternative views.

There is a large variety of TRMs, which additionally integrate other methods of strategic management. We do not claim to have presented an all-encompassing metamodel. However, we do believe that the proposed metamodel is applicable to a broad set of TRMs. The other problem of our methodology is selection and confirmation biases: a set of roadmaps used for analysis was composed by the authors of this contribution. Therefore, it may be interesting to see the alternative variants of the metamodel and, consequently, the consensus version.

Finally, focusing on the informational properties, we have entirely ignored the part of the architecture responsible for the visualization of all this information. It is a notable omission as roadmapping is essentially a visual approach that needs to be addressed in the future.

6 MODEL-BASED USER EXPERIENCE DESIGN AND ROADMAPPING FRAMEWORK

Open access

The materials of this chapter will be in the future published in a scientific journal. Authors' version of the working paper can be found in <http://ssrn.com/abstract=4040650>.

Abstract: Technology roadmapping is a key but challenging strategic process that involves inter-disciplinary knowledge and requires the participation of key decision-makers. To improve and facilitate this activity, a concept of model-based technology roadmapping has been proposed in the literature. In this paper, we developed a Domain-specific language that uniformly represents key concepts related to the four levels essential for the strategic planning in the user-oriented business context: market, user experience design, engineering design, and technology. We implemented the proposed methodology in a collaborative interactive software and tested it in a case study connected to the relevant problem: sustainable transformation of the automotive sector.

6.1 Introduction

This chapter presents the core part of our approach to the Model-based UXD and roadmapping: DSL (its syntax and semantics) implemented in the form of a cloud-based collaborative support tool. The chapter provides argumentation for the choices made on DSL's syntax (extended feature modeling language) and semantics (that is composed of a number of conceptual models from marketing, UX design, engineering design, and technology).

The proposed approach is illustrated on the case study from the automotive domain: transition from internal combustion to electric engines. Also, we validated our approach with an industrial case study (automotive HMI), but these results are sensitive, so we do not publish them.

This chapter addresses research gap G2: practical integration of the model-based roadmapping and design roadmapping. Within our overall research agenda, this second contribution manifests the "Prescriptive study" stage of the DRM. If we consider

this contribution separately, with the objective to create a support tool with desired properties, it accomplishes the following stages of DRM: RC (review-based) → DS I (review-based, conceptual models on four levels of a roadmap) → PS (comprehensive, DSL syntax and semantics, implementation) → DS II (Comprehensive, illustrative case study, industrial case study, semi-structured interviews).

6.2 Related literature

In this section only the most important contributions are listed. The comprehensive literature review for this chapter is provided in the sections 3.1 and 3.3.

Our approach to Model-based User Experience Design and Roadmapping is essentially integrating Model-based Technology Roadmapping (Knoll, Golkar, and Weck 2018), and Design Roadmapping (Kim, Beckman, and Agogino 2018).

The idea of modeling NPD portfolios with feature tree syntax is traced to Abrantes and Figueiredo (2014).

In the view of the process, our framework can be classified as a support for the agile NPD, and hence refers to the Design Thinking literature (Brenner and Uebernickel 2016; Hehn et al. 2020).

6.3 Model-based UXD and Roadmapping

In our approach, we will use the extended syntax of FODA's feature modeling language. It will be shown in the following, this syntax conforms to all the objectives mentioned in the section 1.3. Figure 3.11 shows a simple example of a feature model representing a product (a mobile phone). All heterogeneous concepts (components, functions, systems, technologies, and attributes) are modeled with just one entity — a feature, that may be either optional, or mandatory.

The main value of feature tree syntax is in links. Links do not only have a descriptive purpose but also represent rules of the propagation of an optional feature's boolean state — enabled or disabled. The feature model can be “configured,” i.e., a special mode allows a user to decide which optional features to include in the particular configuration.

Feature tree (a **Model**) represents all possible configurations, a whole product fam-

ily. Each instance of a feature tree (a **ModelConfiguration**), e.g., phone with basic screen and MP3, represents a concrete product. Such a mechanism allows us to model product family evolution and market segmentation (Fig. 6.1).

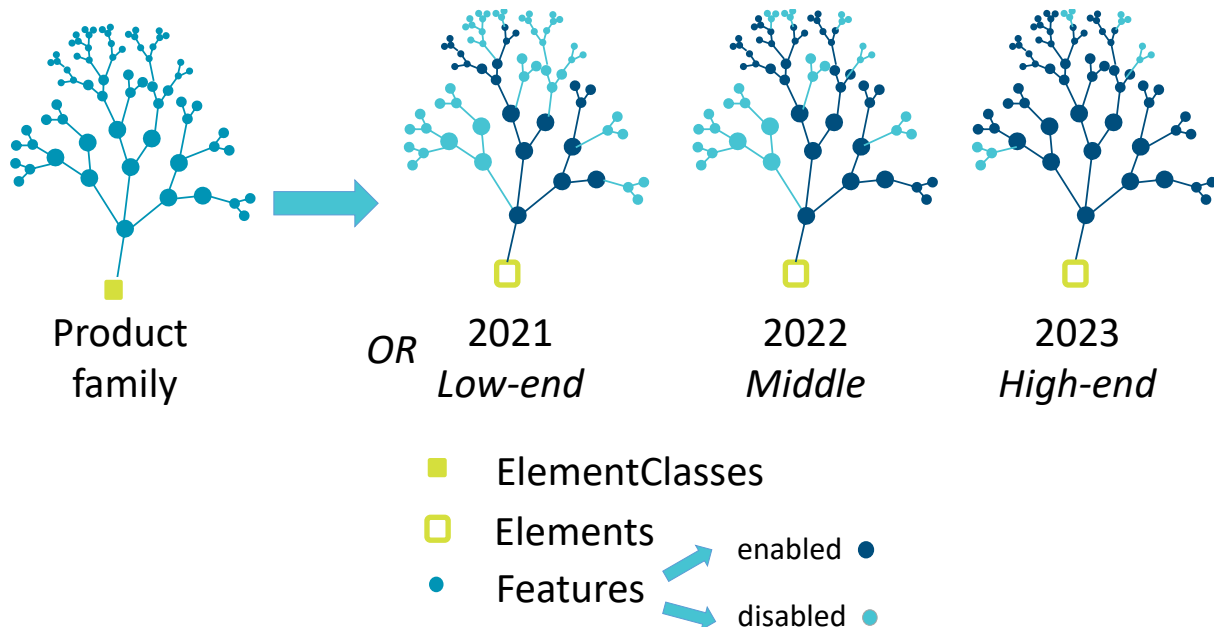


Figure 6.1: Product family evolution or segmentation with feature trees (Yuskevich et al. 2022)

The main value of feature tree syntax is in links. Links do not only have a descriptive purpose but also represent rules of the propagation of a boolean state — enabled or disabled. The feature model can be 'configured', i.e., a special mode allows a user to decide which optional features to include in the particular configuration.

Though, to tailor feature model syntax to MB-TRM needs, we were compelled to introduce several extensions.

First, we added a notion of a Subclass, a modifier that classifies features. Unlike SysML and OPM, entities of our DSL have customizable semantics (taxonomy for each aspect of a model can be defined in a separate view). Figure 6.2 illustrates the flexibility of our approach by expressing several concepts in SysML, OPM, and our DSL. In the UX design, activity, function, use case, as well as persona and user role have distinct meanings. This difference is lost in a plain OPM or SysML. Moreover, SysML possesses certain ambiguity (the same concept can be represented with different symbols depending on the diagram/view).

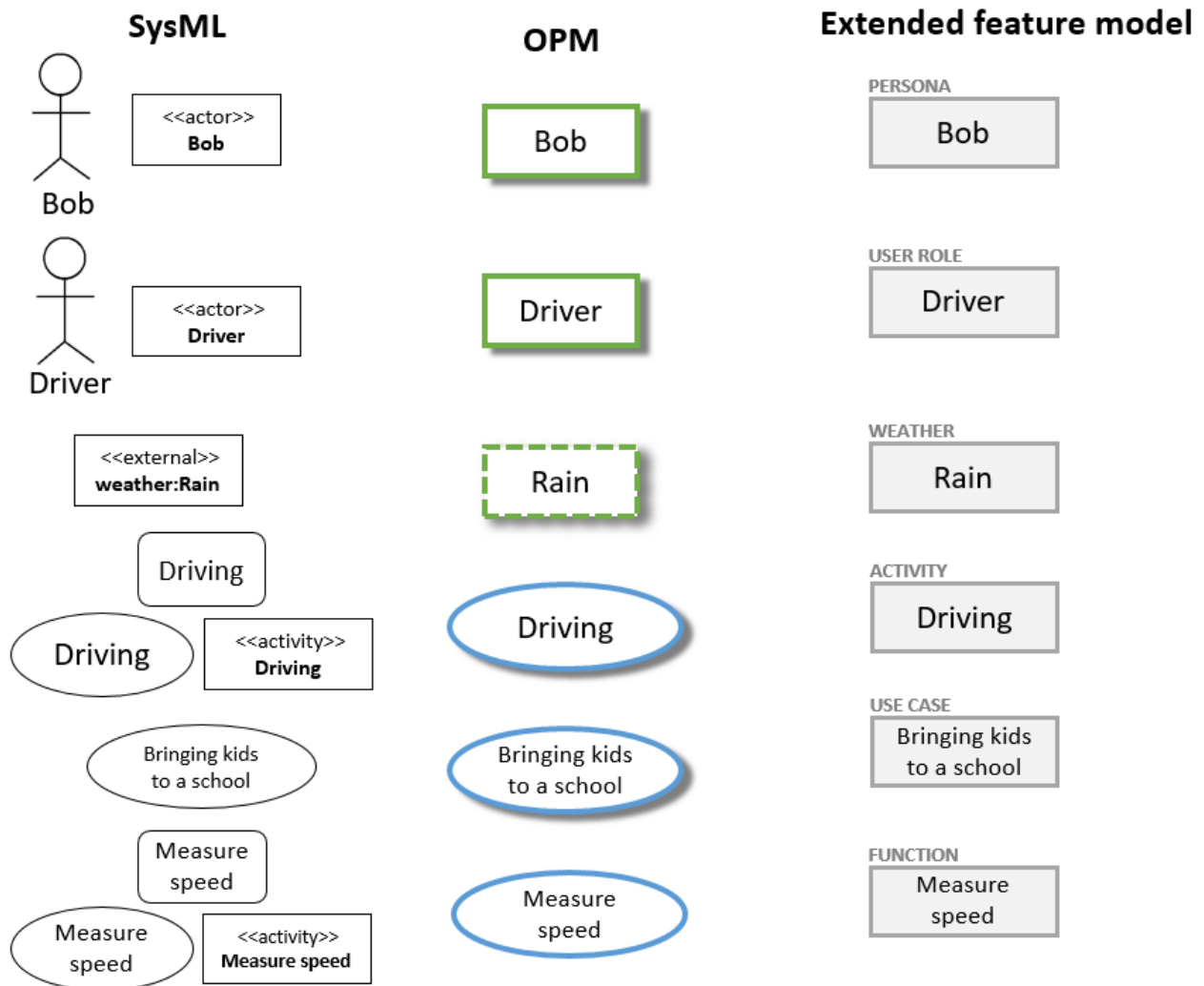


Figure 6.2: Comparison of the extended feature tree syntax with SysML and OPM (Yuskevich et al. 2022)

Subclasses can have parent-child relationships, so a user of a model can define custom taxonomies for each feature model (see Fig. 6.3 presenting an exemplary biological taxonomy of species defined with model Class → Family → Specie).

Second, we propose to extend the ontology of cross-tree links. Except standard requires and excludes, we introduced the causes and equals links. The logical operations that these links represent are defined in Figure 6.4, where ✓ means that feature is enabled, ✗ — disabled, ? — the state is unknown (both states possible). The left column stands for the cause, right — for effect.

Links of this type (causes, requires, equals, and excludes) connect features that have

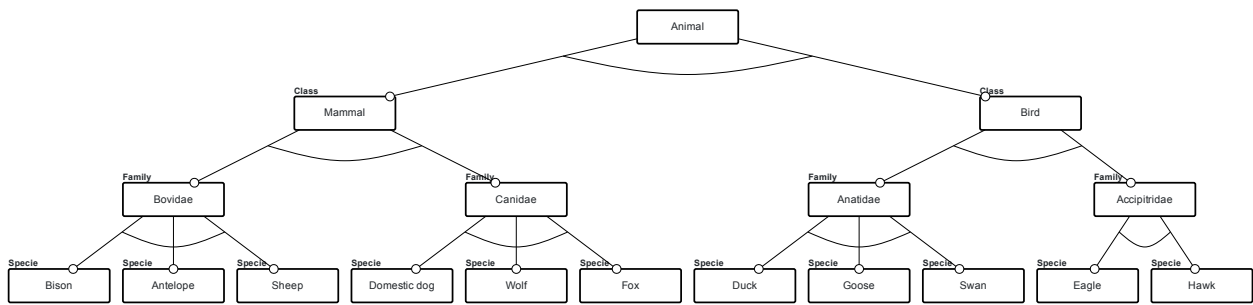


Figure 6.3: Biological classification is represented through extended feature tree syntax (Yuskevich et al. 2022)

qualitative meaning. For example, a goal ‘bring kids to school’ causes the use case ‘driving with kids’. An electric vehicle requires an electric traction motor. Having any car means that we also have (equals to) a goal ‘park a car’. Finally, living in a big city excludes free parking.

The other type of links that we have introduced by analogy with NFR framework (Chung et al. 2012) connects qualitative and quantitative features — enhances and worsens. If a qualitative feature is enabled (e.g., a climate control system), a numeric score of a link is added to a quantitative feature (e.g., a cost).

The last link type connects two quantitative features and represents mathematical operations. It modifies the numeric value of a quantitative target feature by adding to it (or multiplying it by) an aggregated value of a source quantitative feature multiplied by some constant (see fig. 6.5).

We estimate that a mature industrial roadmap should have thousands of features and links. The process of connecting two features directly will require manual work. Moreover, a ‘spaghetti’ of relationships quickly becomes not readable. Though, there is a way to simplify the model representation and modification with tables and matrices.

In the literature, several artifacts were proposed, which represent complex relationships in such a form. In the field of UXD, probably the most popular artifact is User journey map (Howard 2014); in the field of systems engineering — Design Structure Matrix (Eppinger and Browning 2012).

In our methodology and a support tool, these artifacts interchangeably complement the graph representation (see Fig. 6.6). The use of DSM is relatively straightforward. To model the fact that Function 3 requires Technology 5, one needs to place a symbol ‘requires’ in the cell 3-5 of the Functions-Technologies matrix. The advantage of DSM

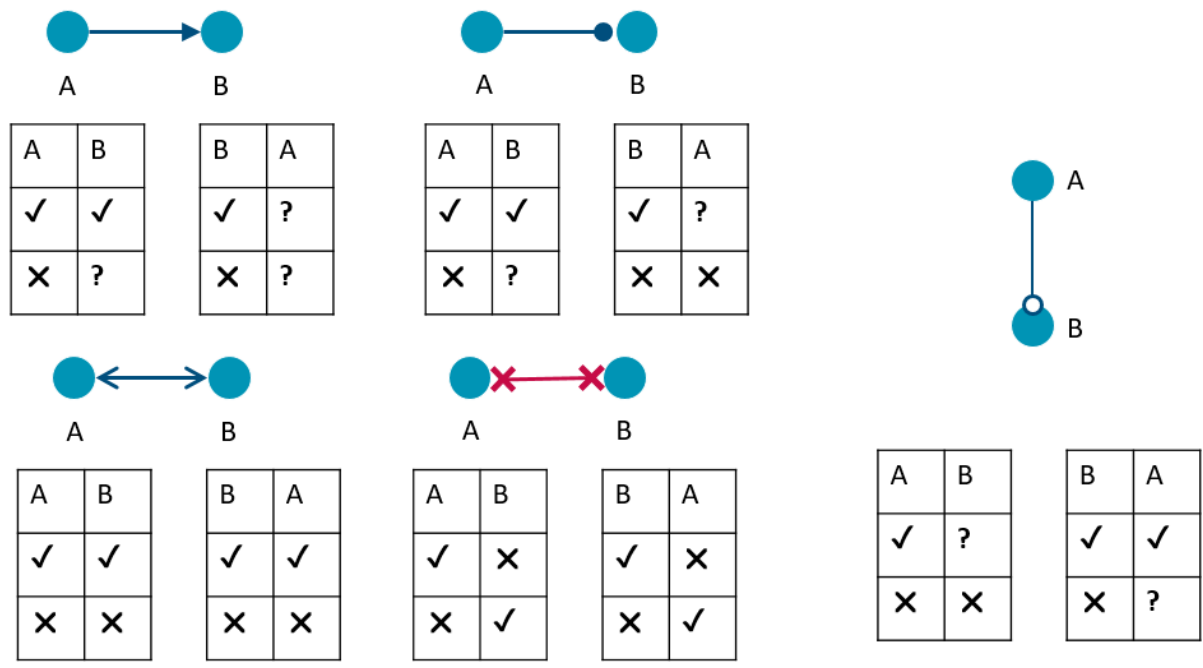


Figure 6.4: Cross-tree links syntax (Yuskevich et al. 2022)

is that it represents a large number of relationships in a compact form. The use of User journey maps requires a more detailed explanation.

User journey map represents a sequence of interactions between a user and a product. Typically, a user journey map is a table, columns of which denote situations (often called touch points, or steps, or activities), rows can capture various concepts — user goals, persona traits relevant to this situation, outcomes of the interaction, etc. Figure 6.7 shows an exemplary user journey map with a simple plug-in electric vehicle use case.

Now, let's as an example set up some rules for a user journey map from Figure 6.7. If features are located in the same column, then a certain link between them can be established if one of the following conditions on the rows is hold:

- **Persona trait** *causes* **User goal**
- **User goal** *requires* **Product feature**
- **User goal** *requires* **Product characteristic**

If these rules are defined, User journey map from Figure 6.7 can play a role of an in-

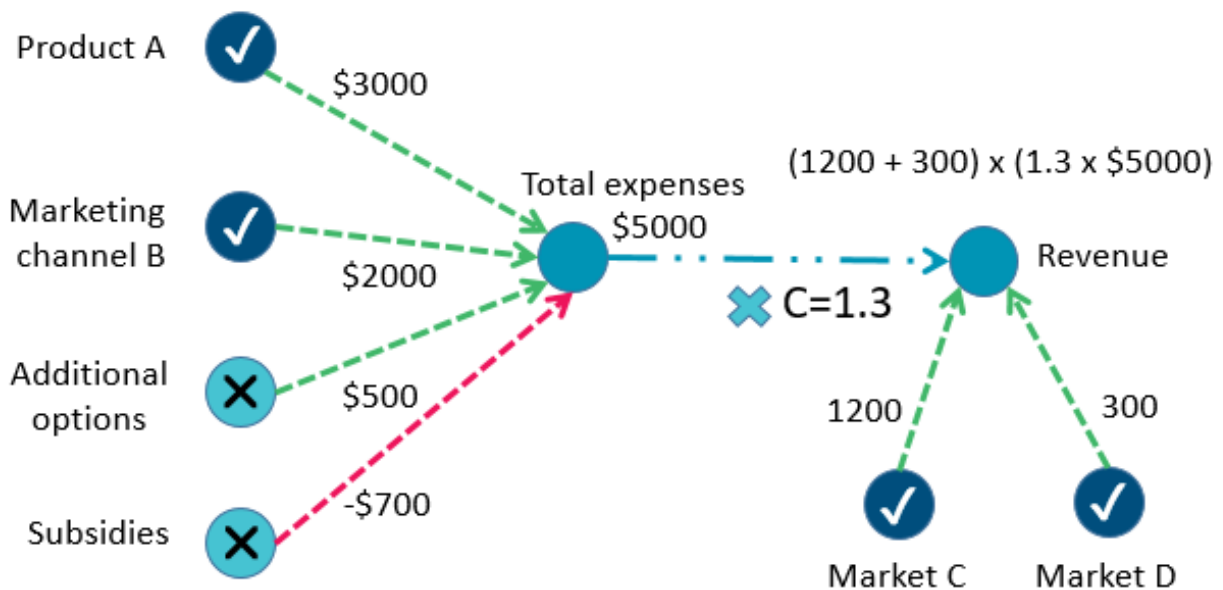


Figure 6.5: Example of how quantitative links represent a simple business model (Yuskevich et al. 2022)

terface for assigning links between features. Moreover, User journey maps are usually built for specific situations. This user journey is rather general, yet built for a specific Persona (Bob), riding specific car (plug-in electric). Other journeys can be created for more specific use case (e.g., plug-in electric in a heavy rain, plug-in electric in a traffic jam, etc). Therefore, links assigned by user journey are *conditional*.

Examples of conditional links generated from User journey map from Figure 6.7:

- **Persona trait** Working at Acme Corp. *causes* **User goal** Get the workplace comfortably and quickly *under conditions* {**Persona:** Bob, **Car type:** Plug-in electric, **Activity:** Morning ride}
- **User goal** Charge the battery *requires* **Product feature** EV charger cable *under conditions* {**Persona:** Bob, **Car type:** Plug-in electric, **Activity:** Parking near the workplace}
- **User goal** Getting back home comfortably and quickly *requires* **Product characteristic** Consumption: 20 kWh *under conditions* {**Persona:** Bob, **Car type:** Plug-in electric, **Activity:** Evening ride}

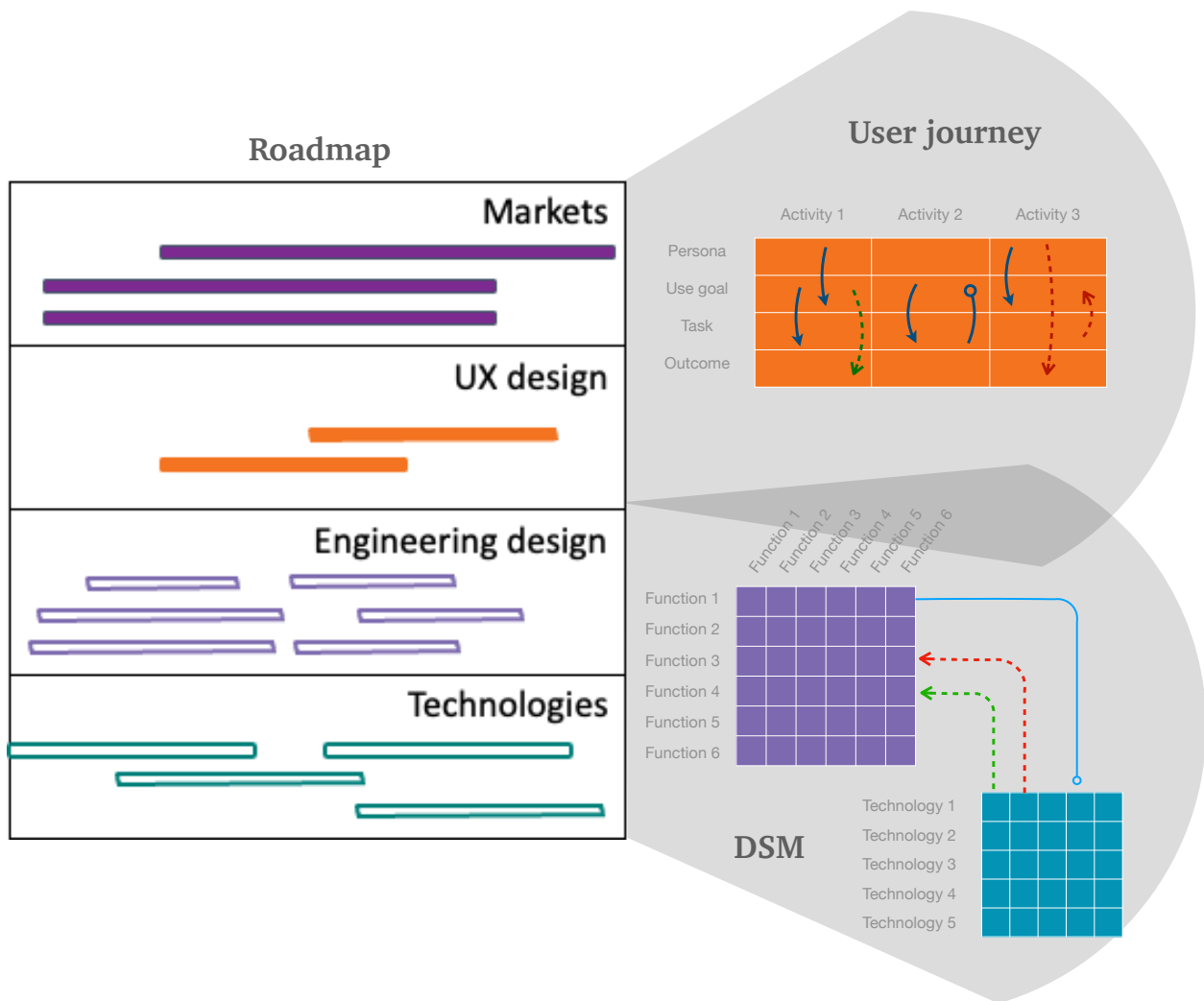


Figure 6.6: Using DSMs and User journeys for the cross-trees relationships representation

Figure 6.8 shows the roadmap's semantics for an automotive company that plans to enter a market with a product line of vehicles with reduced CO_2 emissions. This semantics is derived from a number of conceptual models we found in the literature. The big ellipses represent ElementClasses, smaller nested ellipses — Subclasses. At each level of a roadmap feature models in the left represent the qualitative aspect of a model, and in the right — quantitative.

Moreover, this figure represents typical links between Subclasses. It can be noted that the direction of these links also follows the decision-making flow. It goes from top to bottom in a qualitative axis and influences quantitative metrics at each level.

	Morning ride	Parking near the workplace	Evening ride	Parking near to home
User goals	<ul style="list-style-type: none"> Getting the workplace comfortably and quickly 	<ul style="list-style-type: none"> Keep the car for the evening ride Charge the battery Parking should be cheap 	<ul style="list-style-type: none"> Getting back home comfortably and quickly 	<ul style="list-style-type: none"> Keep the car for the next day ride
Persona traits	<ul style="list-style-type: none"> Working at Acme Corp. 			<ul style="list-style-type: none"> Lives in suburbs
Product features	<ul style="list-style-type: none"> Infotainment system <ul style="list-style-type: none"> Music player Navigation 	<ul style="list-style-type: none"> Automatic parking EV charger cable Anti-theft system <ul style="list-style-type: none"> Power door lock 	<ul style="list-style-type: none"> Infotainment system <ul style="list-style-type: none"> Music player Navigation 	<ul style="list-style-type: none"> Anti-theft system <ul style="list-style-type: none"> Power door lock
Product characteristics	<ul style="list-style-type: none"> Consumption: 20 kWh 	<ul style="list-style-type: none"> Length: max 4.5 m 	<ul style="list-style-type: none"> Consumption: 20 kWh 	

Persona: Bob, Car type: plug-in electric

Figure 6.7: Exemplary user journey map

Then effect propagates up in the quantitative axis (technology performances influence product characteristics, etc.)

As our approach implies flexible semantics, such a conceptual model may and should be designed for each roadmap separately, depending on specific need. For example, in this proposed model UX design level is rather simplified. Even when we build a roadmap for the same automotive industry, but for the different product family (e.g., Human-Machine Interfaces), we may want to consider more concepts, e.g., user role, external conditions (weather, time of the day), etc.

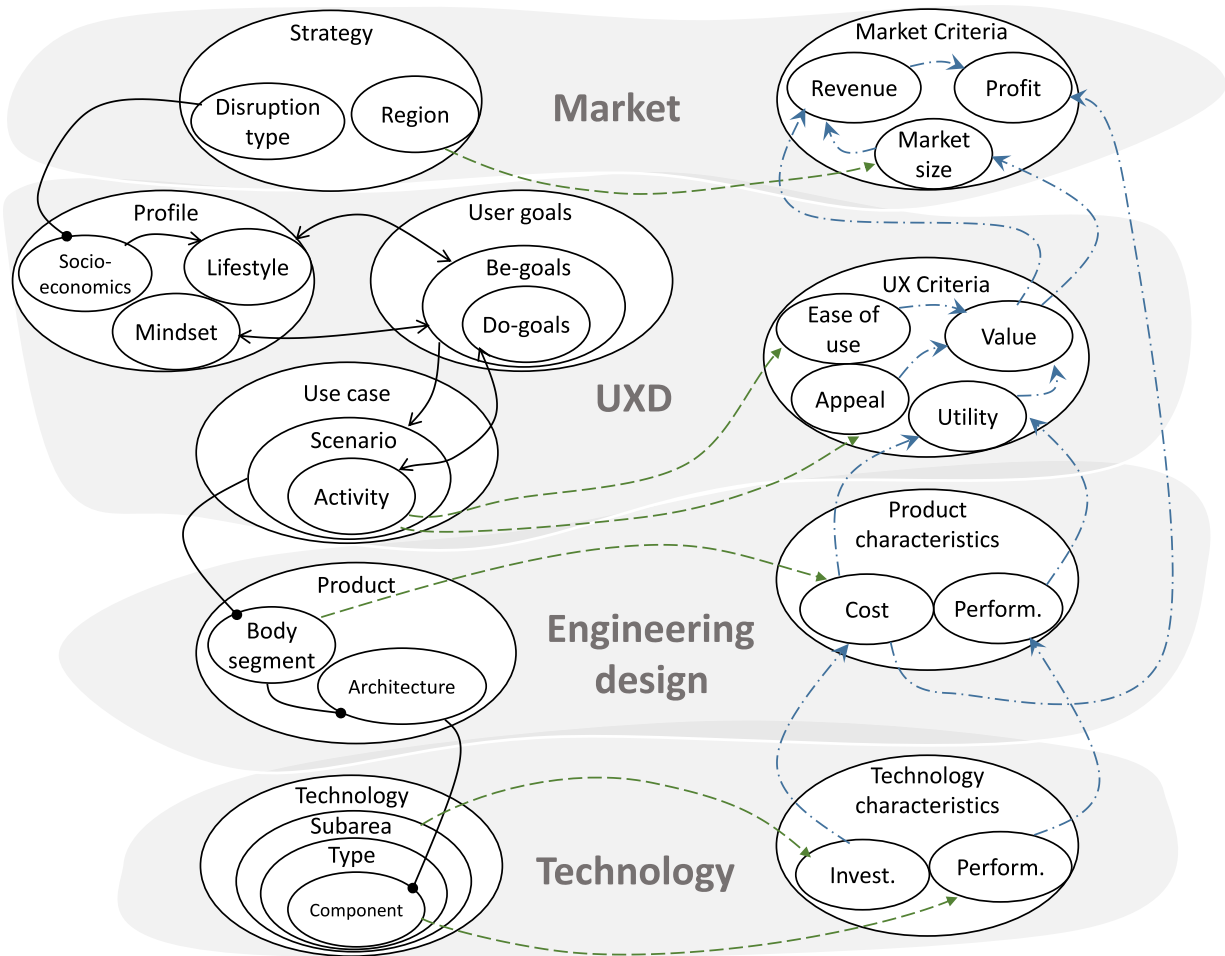


Figure 6.8: Semantics (Yuskevich et al. 2022)

6.4 The process

A model-based approach brings essential implications on the strategic planning process due to more efficient data creation, modification, and storage. This enables asynchronous, flexible, and iterative roadmap creation and maintenance procedure.

These properties conform very well with requirements imposed by the agile NPD. The computer system can easily incorporate new information coming from iterative prototyping and testing into the strategic plan (a roadmap). For instance, prototyping and user testing results may reveal that certain feature is not feasible or appealing for the target audience, which will require strategic changes. Moreover, these changes, due to the formal syntax of the DSL, are propagated and assessed against other objectives/contextual, engineering, or technological factors.

By the flexibility of the approach, we mean that the conceptual model can be relatively easily extended and redefined. For example, a product team may decide to assess user experience using some aggregated metrics such as customer satisfaction and later specify it as usability, utility, and desirability. Also, the flexibility means that we are not setting fixed boundaries between stages of a process (e.g., first, the conceptual model must be defined, then target markets and personas, etc.) Flexibility and iterativeness mean that we address the problem from a structural and not a procedural perspective. We define which concepts and artifacts have to be elaborated along the process, but the sequence of this work is not important.

Asynchronous roadmap creation helps to update roadmaps by different departments independently and remotely only when necessary (e.g., when a new strategic matter appears). Fig. 6.9 compares standard workshop-based T-plan process proposed by Phaal et al. (2003) and our proposed procedure.

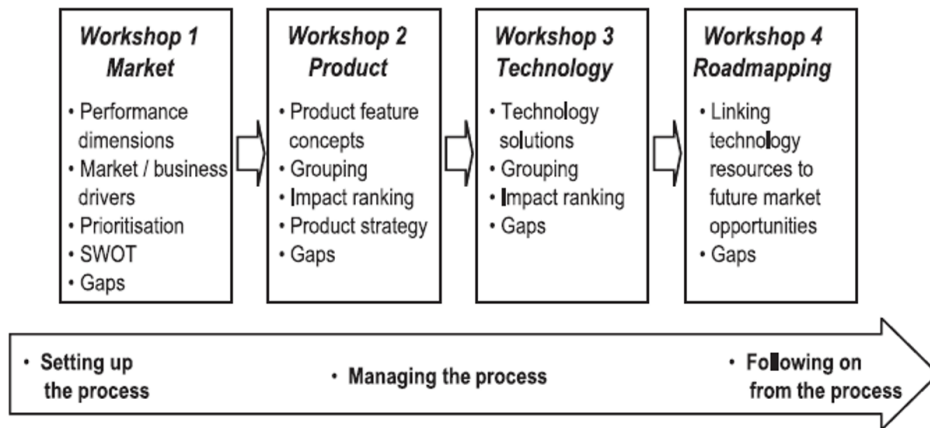
6.5 Case study

6.5.1 General description

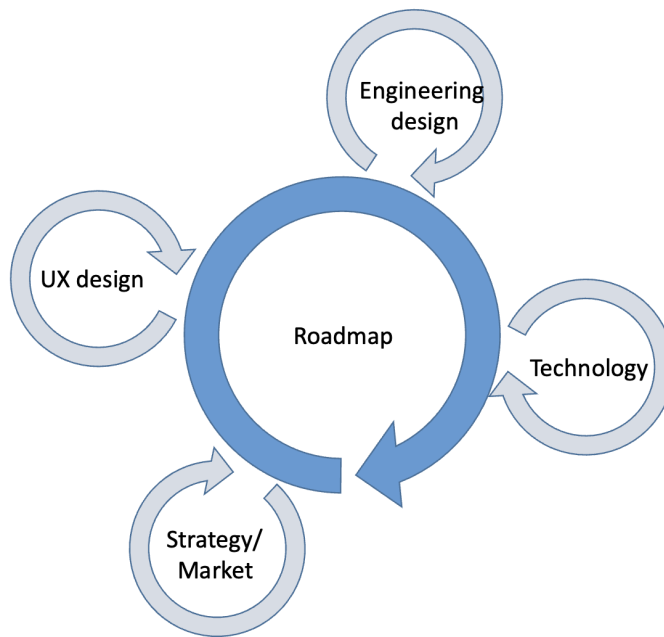
This case study was designed for illustrative purposes and, therefore, intentionally kept simple. In case there are automotive industry strategists among our readers, we want to make a disclaimer: we are aware that our assumptions may seem naïve or inaccurate. However, our purpose here is to demonstrate the approach's internal logic and consistency rather than build an industrial-grade roadmap. Besides, we have validated our approach in the industrial setting on the case study of human-machine interfaces NPD. Unfortunately, we cannot publish the results as they are commercially sensitive. Also, this case study is too detailed, and, therefore, hardly comprehensible

Having this in mind, the setting is the following. An established automotive company that currently produces cars with internal combustion engines tries to adapt to the recent market changes and therefore strives to propose a product line with reduced CO_2 emissions.

The company may choose two alternative strategies for entering these markets — low- and high-end. Also, markets are classified by geographical location — Europe, the Americas, and Asia, with different sizes in each segment.



(a) Workshop-based T-plan (Phaal et al. 2003)



(b) Agile (iterative and asynchronous) roadmapping process

Figure 6.9: TRM and agile MB-TRM processes compared

UX designers have defined four personas, two for low-end and two for high-end market segments. Roberto lives in suburbs, has kids, and possesses a traditional mindset. Alice is an eco-friendly person that lives in a big city. Kate has kids, possesses an eco-friendly mindset, and lives in the suburbs. Bob is a technology enthusiast who lives in a big city. Personas have different use cases depending on income, place of living, and family situation, e.g., daily commuting (with or without kids), long-distance trips, and occasional short trips. In the context of electric vehicles, the most important activity in each use case is parking, because depending on equipment available on a parking lot, a user can or cannot charge it.

Four product architecture types may satisfy diverse personas' goals: hybrid electric, plug-in hybrid electric, battery-electric, and fuel cell electric. Also, cars are classified by body size segment: mini, small, medium, large, luxury, sport, and sport utility vehicle (SUV).

Depending on chosen architecture, a company needs to invest in two technology areas: power train technologies (internal combustion engines, electric traction motors, fuel cells, and regenerative braking) and energy storage technologies (hydrogen and gasoline tanks, batteries, and supercapacitors).

The roadmap has two alternative scenarios depending on which strategy the company will choose. Low-end strategy is shown in figure 6.10, high-end strategy — in figure 6.11 (screenshots taken from our web-based software, all are zoomable). Alternative strategies determine not only the target market segment (high-end or low-end), but also personas belonging to these market segments (Roberto, Alice vs. Bob and Kate), use cases, particular to these personas, and, eventually, products satisfying personas' needs (family hybrid, mini plug-in vs. luxury electric sedan, fuel cell coupe).

The feature model of the market strategy and its configurations are shown in figure 6.12; personas — figure 6.13; use cases — figure 6.14; product family — figure 6.15; technology areas — figure 6.16 and 6.17. Each figure on top represents the model of an **ElementClass** (all possible design/decision options). Bottom figures show models representing **Elements**, which are the instances of corresponding **ElementClass** models¹.

When all roadmap elements are thoroughly defined, cross-tree links can be added (for example, Fuel cell electric vehicle requires hydrogen tank, having kids causes use

¹If some of these figures are not correctly displayed, try to open the pdf file of manuscript in Acrobat Reader or Google Chrome

case of bringing kids to school, etc.). The most convenient interface to link these multiple factors is through the DSM on the level of product-technologies and through the User journeys on the market-UXD levels (the latter is also implemented in the software). Feature model of an **ElementClass** (for instance, a persona) represents all variants in one model (high or low income, has kids or not, urban or suburban habitant). The roadmap user (e.g., participant of a TRM sessions) needs to instantiate the model to represent concrete persona (Roberto, Alice, Bob, Kate). These choices will not only specify the concrete instance, but also cause changes in other parts of the model through the cross-tree links.

6.5.2 Analysis

Because the proposed DSL is executable, roadmap contributors can analyze the key performance indicators, compare different scenarios and design options. The fact that such an analysis is cross-domain can be seen as a novelty of a proposed approach.

Figure 6.18 shows what we call a global view: a composite feature model representing all concepts and factors in all its combinatorial complexity, from markets to technologies.

Specifically, this view shows how high-level decisions (market and UXD) reduce the decision space on the lower levels (product and technology) and how lower-level performance indicators influence higher-level figures of merit. Moreover, the model identifies inconsistencies: the configurations in which lower-level decisions contradict higher-level decisions or simply highlights logical flaws (e.g., closed loops).

Especially, figure 6.18 illustrates a chain of causes and effects going from markets to technologies and back. Environmentalism causes changes in policy (tax on carbon emissions). A persona with a traditional mindset, who is not emotionally engaged in the green economy, can nevertheless be affected by this policy change through a be-goal "I want a car with cheap maintenance." The red dashed line signifies the negative effect of a tax on the UX criteria "Low gasoline spendings" (amplified by the necessity of bringing kids to school every day). Plug-in hybrid electric vehicle architecture requires "regenerative braking" technology. From this point, the quantitative links propagate back through criteria on multiple levels (regenerative braking → regeneration efficiency (%) → fuel consumption (l/100 km) → low spendings on gasoline). The direct chain can be executed due to propositional logic defined in the model (see Fig. 6.4); the backward chain is executed due to the extended syntax representing quantitative

links (Fig. 6.5).

Figure 6.18 also shows an exclusive link between the be-goal “Child-free” and the use case “bring kids to school every day”. This link opens an alternative instance of a model for persona with different motivations, use cases, UX criteria, etc. The benefit of the feature tree is that it represents all these possibilities in a single compact model.

Figure 6.19 a) shows an example of inconsistency (highlighted): a use case was selected that requires bringing kids to school for a persona with no kids. Figure 6.19 b) shows a given product’s calculated UX- and product-related metrics for a selected persona. Figure 6.19 c) shows the evolution of revenue and profit over time for one of the strategies.

Each bar on Fig 6.19 c) corresponds to the discrete-time (6 months each sequence). A single instance of a global model (Fig. 6.18) represents each time sequence. For example, figure 6.12 shows which specific markets the company targets at each time interval (for low-end strategy — first Europe, then Europe + Americas, then Europe + Americas + Asia, for high-end — first Americas, then Europe + Americas, then Europe + Americas + Asia). Additionally, product features, use cases, technologies evolve in time. Clearly, all these combinatorial variants will result in different technology and product characteristics, UX criteria, and, eventually, revenues and profits.

6.6 Discussion

To our knowledge, the proposed approach is the only model-based decision support tool that unifies and integrates such heterogeneous disciplines as marketing, UXD, and engineering. This is achieved by introducing simple yet powerful syntax based on feature models. Appendix I presents the User Story Map of a support tool we developed (each implemented functionality is displayed in the rectangle and classified by the category).

We have integrated the UXD layer into a roadmap which helps to focus on the end-user value and provides a better interface between markets and product concepts.

Our approach is agnostic to the process. A model-based roadmap can be designed during a series of workshops or, thanks to the consistency check mechanism, in an asynchronous manner. This improves the roadmap’s maintainability and enables agile roadmapping.

One significant drawback can be identified in this research. We proved our approach to be consistent internally but did not verify how well it works in the real industrial context. Even well-known model-based tools are prone to misuse. Therefore, utility and usability should be validated in the real industrial context. This might be considered as the direction for further research.

In the process of the research, we needed to base ourselves on the existing proven approaches and conceptual models. According to what we observed during case studies, it seems that an all-encompassing conceptual model is not possible even within the same domain. Therefore, we propose an approach with flexible semantics, such that taxonomies can be carefully designed for each specific class of products. The absence of a standard all-encompassing conceptual model may sound for some as a disadvantage. We see it more as a feature of our proposed approach that addresses the diversity of product families.

It is also interesting to note that the boundaries between layers are fuzzy; some concepts exist in both layers and create an interface between them. For example, both marketing and UXD share the concept of persona. User motor goals from the UXD layer are directly translated to the functional architecture of the engineering design layer. Components link product and technology layers as they are the elements of product architecture and, at the same time, the embodiment of a particular technology. This observation underlines the consistency of a proposed semantics.

Regarding the possible further extensions, first of all, boolean links can be generalized to Bayesian links as in Moullec et al. (2013). In this way, the effect will not necessarily follow the cause but appear with a certain probability (which reflects better the uncertain reality). Cardinality can be added as another possible extension to represent physical architecture better.

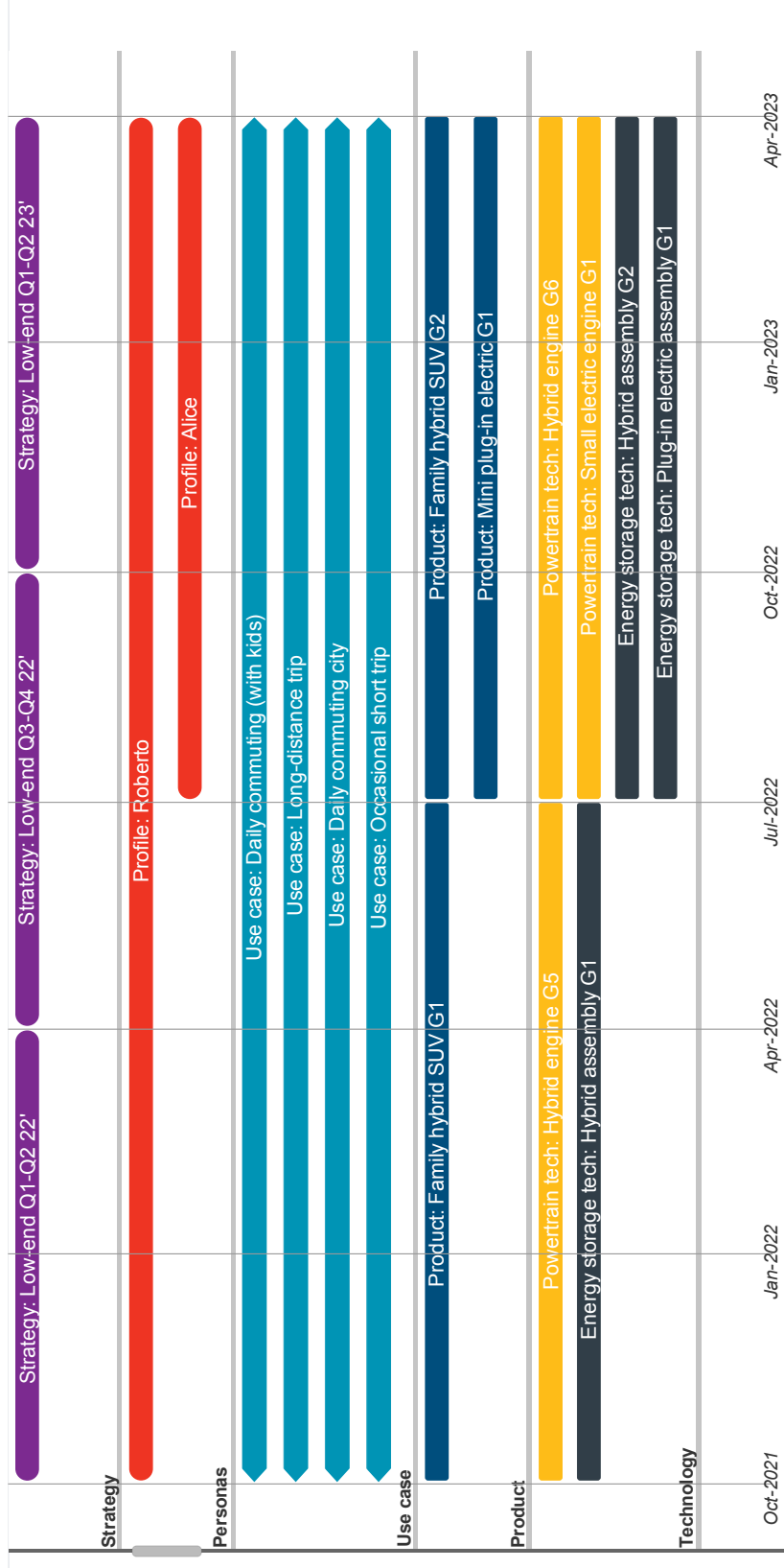


Figure 6.10: Low-end strategy roadmap (Yuskevich et al. 2022)

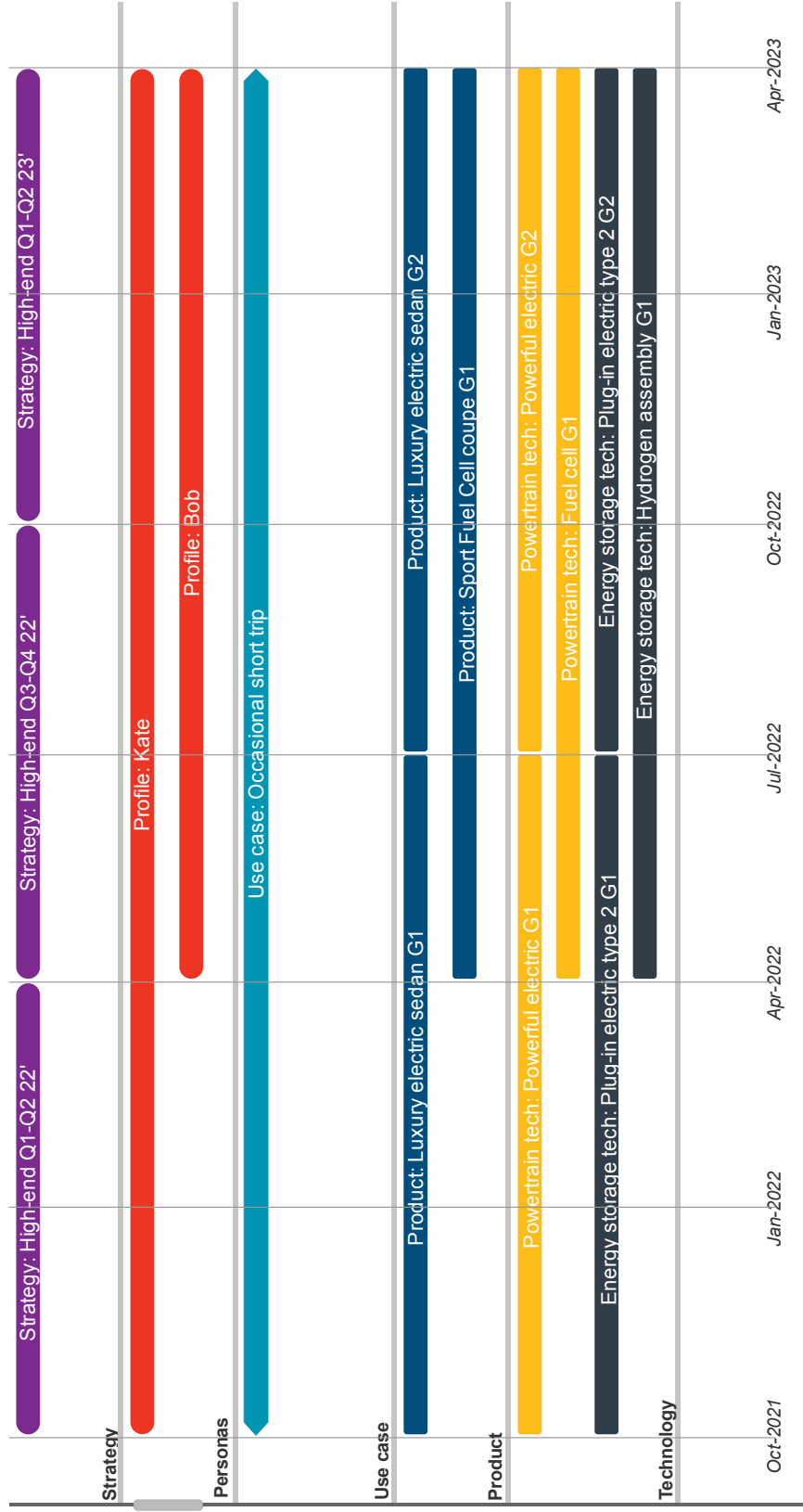
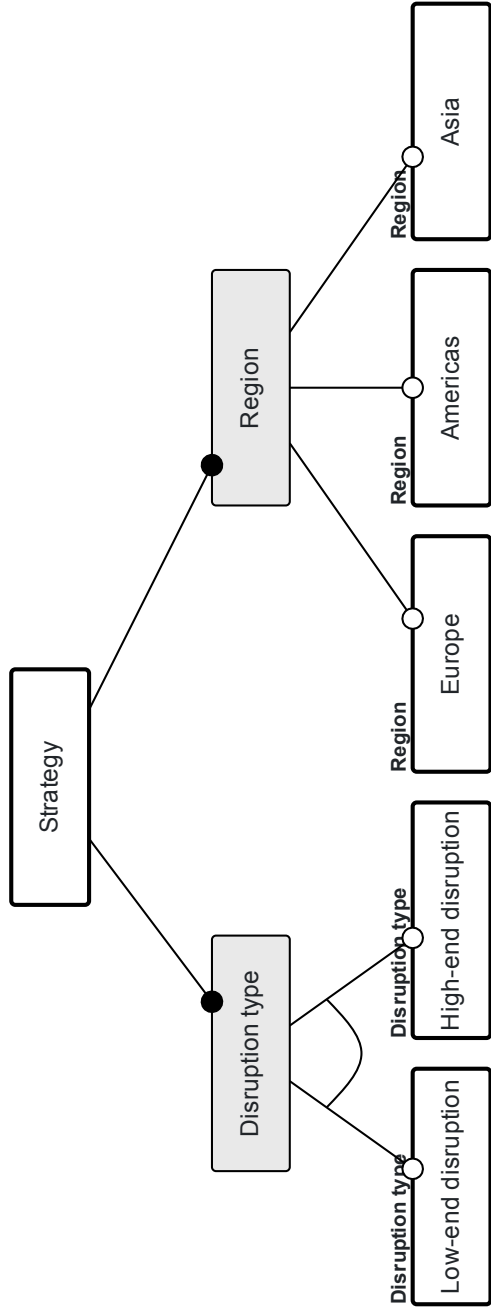
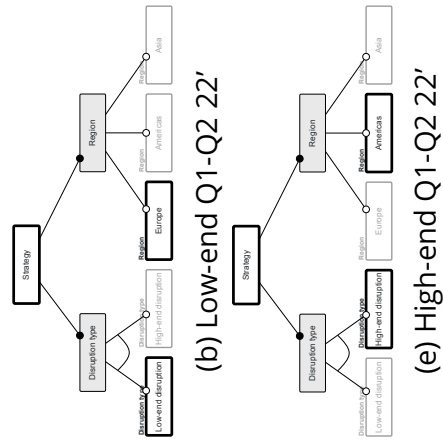


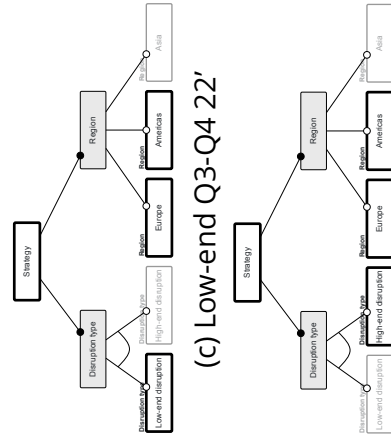
Figure 6.11: High-end strategy roadmap (Yuskevich et al. 2022)



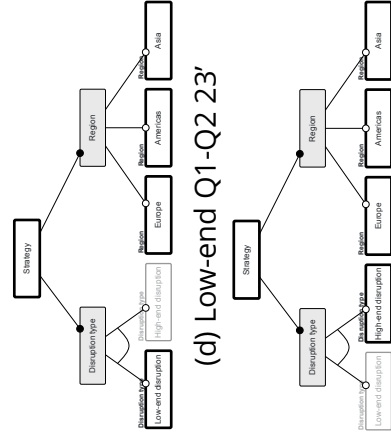
(a) The feature model of the market strategies



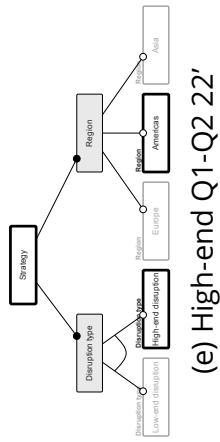
(b) Low-end Q1-Q2 22'



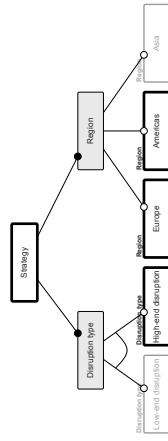
(c) Low-end Q3-Q4 22'



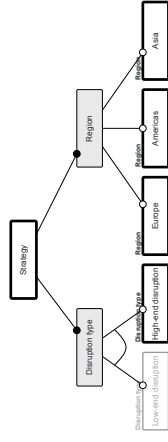
(d) Low-end Q1-Q2 23'



(e) High-end Q1-Q2 22'

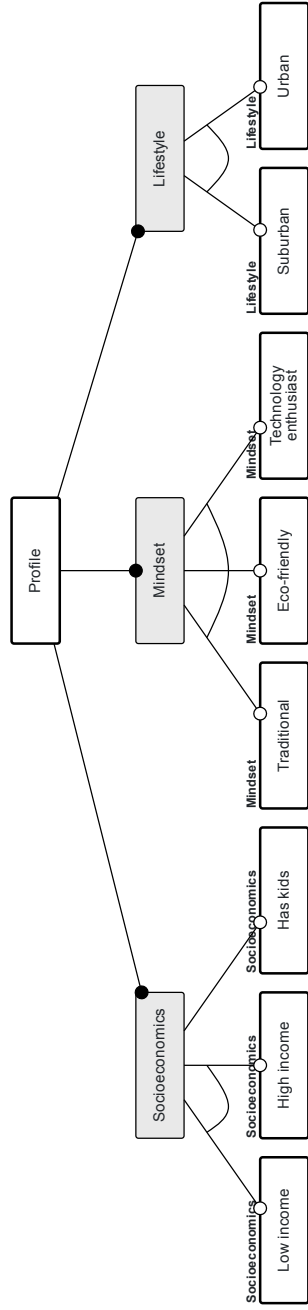


(f) High-end Q3-Q4 22'



(g) High-end Q1-Q2 23'

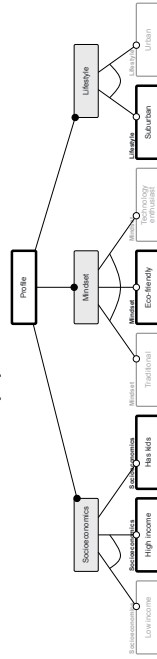
Figure 6.12: Market strategy models (Yuskevich et al. 2022)



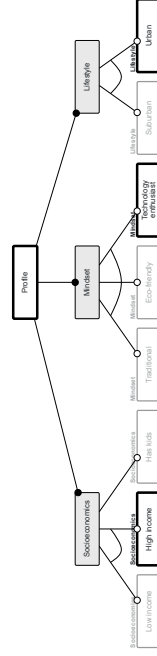
(a) The feature model of personas



(b) Roberto



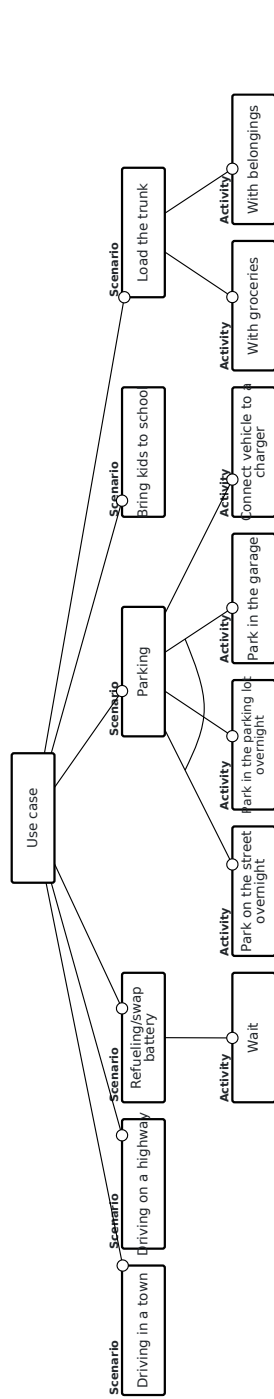
(c) Alice



(d) Kate

(e) Bob

Figure 6.13: Persona model (Yuskevich et al. 2022)



(a) The feature model of use cases



(b) Daily commuting suburbs (with kids)

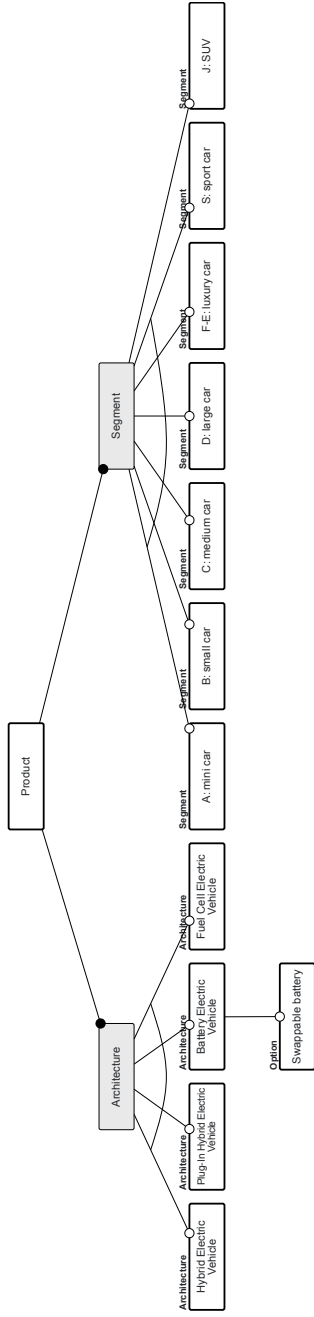
(c) Long-distance trip



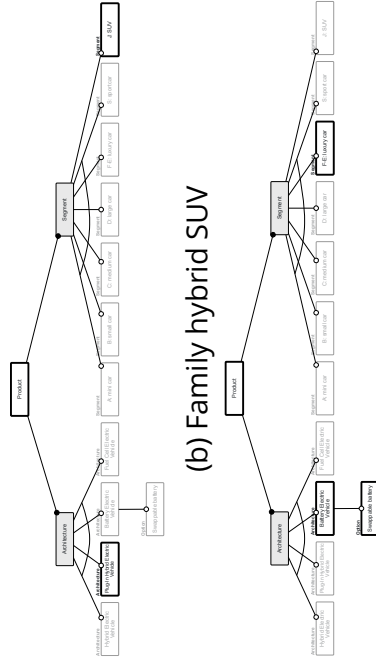
(d) Daily commuting city

(e) Occasional short trips

Figure 6.14: Use case model (Yuskevich et al. 2022)



(a) The feature model of product architectures



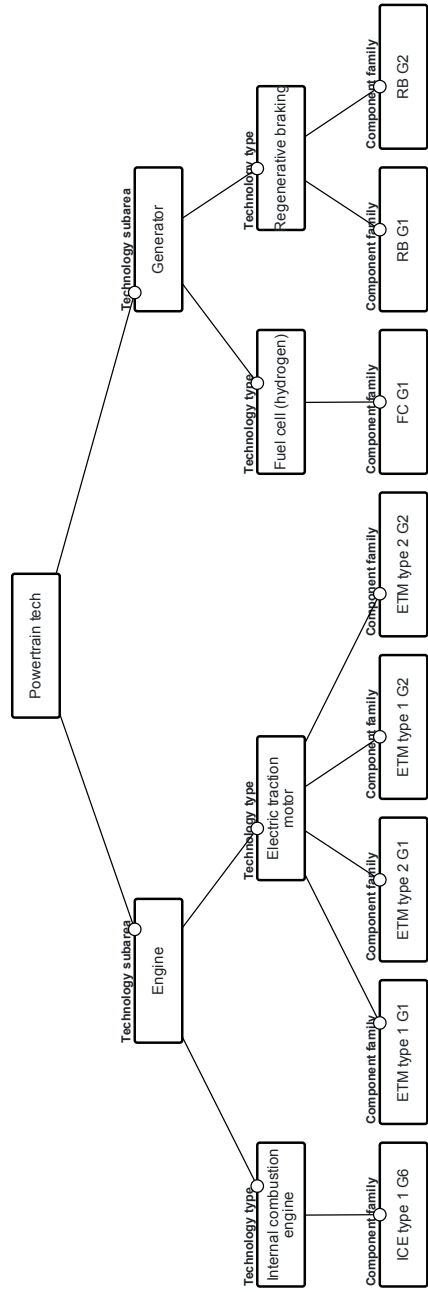
(b) Family hybrid SUV

(c) Mini plug-in electric

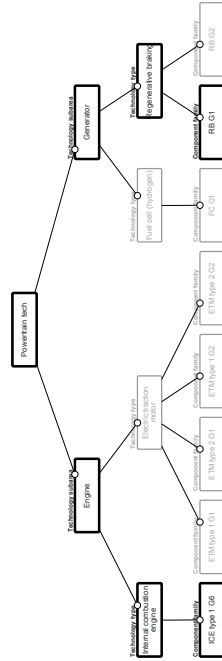
(d) Powerful electric sedan

(e) Sport Fuel-cell

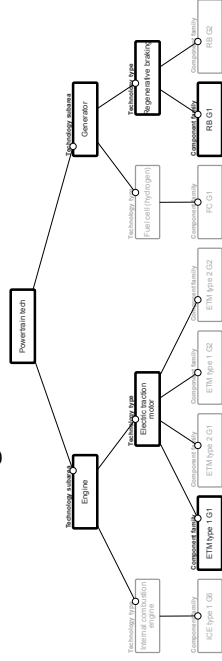
Figure 6.15: Product architecture model (Yuskevich et al. 2022)



(a) The feature model of powertrain technologies



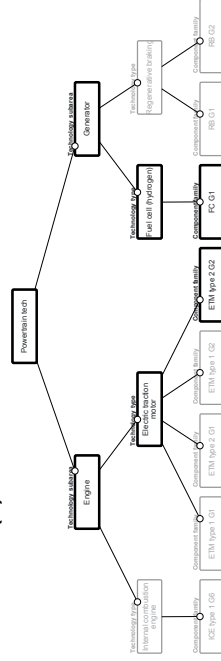
(b) Hybrid Generation 5



(c) Small electric Generation 1

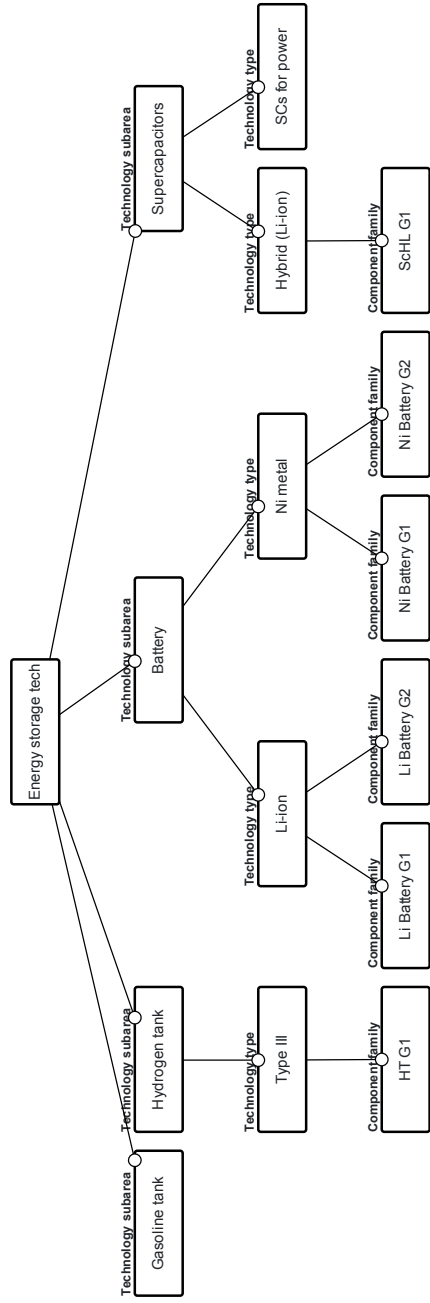


(d) Powerful electric Generation 1

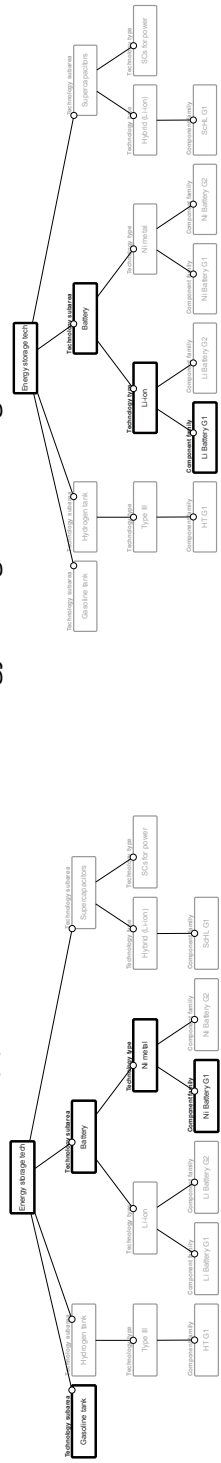


(e) Fuel cell Generation 1

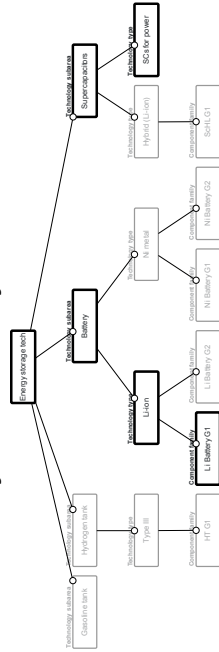
Figure 6.16: Energy storage technology area model (Yuskevich et al. 2022)



(a) The feature model of energy storage technologies

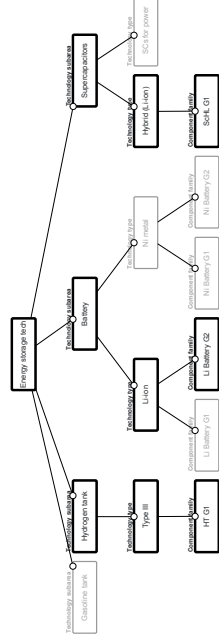


(b) Hybrid assembly Generation 1



(d) Plug-in electric type 2 Gen 1

(c) Plug-in electric type 1 Generation 1



(e) Hydrogen assembly Generation 1

Figure 6.17: Powertrain technology area model (Yuskevich et al. 2022)

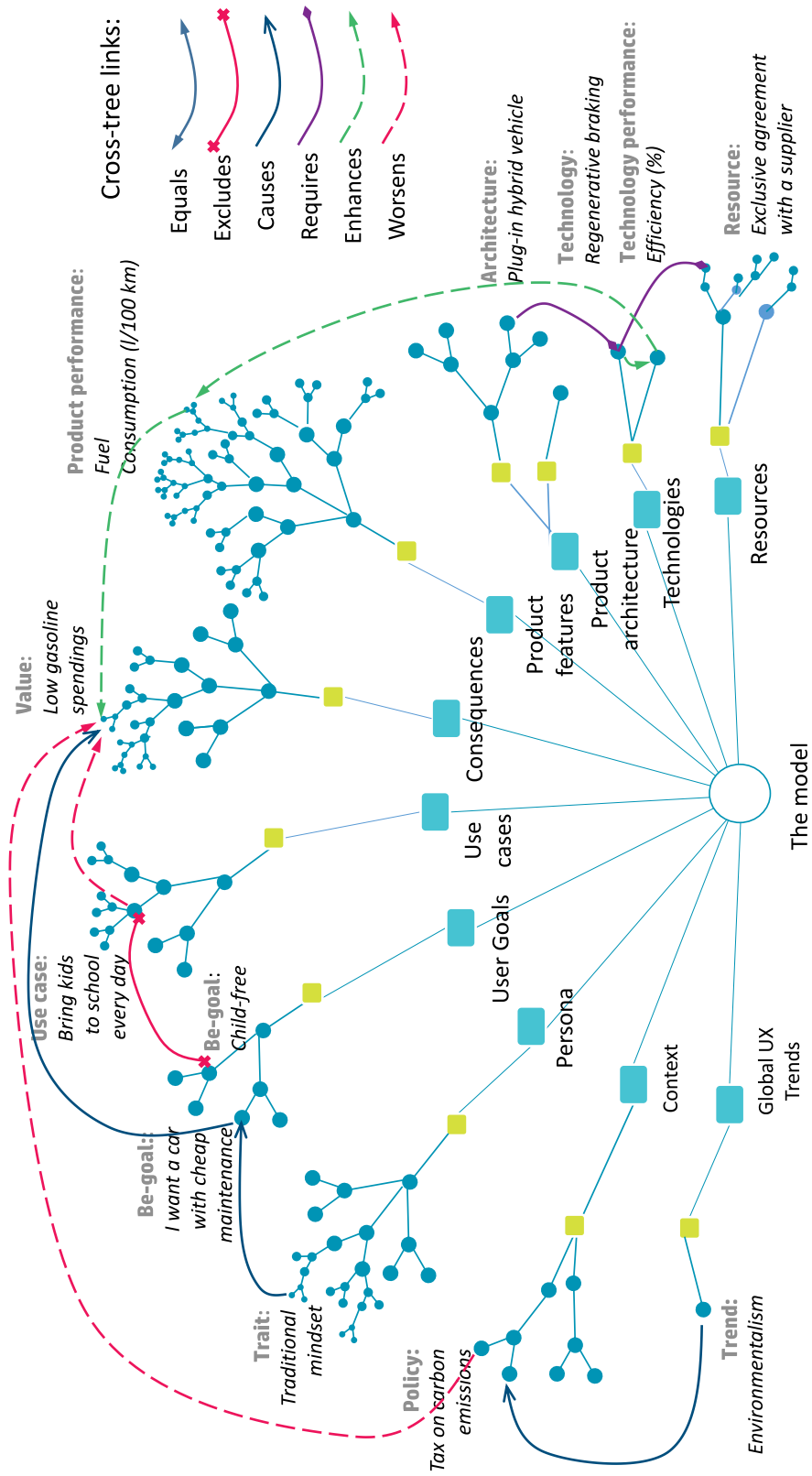
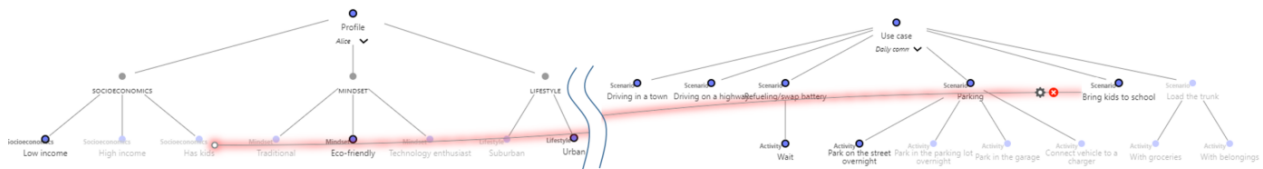


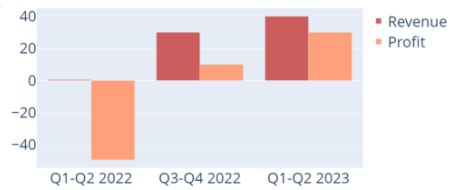
Figure 6.18: Global view of the case roadmap (Yuskevich et al. 2022)



a) Inconsistency highlighted

UX criteria	✕
Ease of use: 6 Appeal: 1 Value: 4	
Product characteristics	✕
Charging time: 5 Maintenance cost: 5 Acceleration: -6	

b) Numeric assessment



c) Evolution of business performance over time

Figure 6.19: Various analyses performed on a roadmap (Yuskevich et al. 2022)

7 DATA-DRIVEN DESIGN ROADMAPPING

Publications

The materials of this chapter were published in Ilia Iuskevich et al. (2021). "A Data-driven Approach To User-experience-focused Model-based Roadmapping for New Product Planning." In: *Proceedings of the Design Society 1*, pp. 61–70. doi: <https://doi.org/10.1017/pds.2021.7>.

Open access

Authors' version before review can also be found in <https://hal.archives-ouvertes.fr/hal-03330641>.

Abstract: User experience (UX) focused business needs to survive and plan its new product development (NPD) activities in a highly turbulent environment. The latter is a function of volatile UX and technology trends, competition, unpredictable events, and user needs uncertainty. To address this problem, the concept of design roadmapping has been proposed in the literature. It was argued that tools built on the idea of design roadmapping have to be very flexible and data-driven (i.e., be able to receive feedback from users in an iterative manner). At the same time, a model-based approach to roadmapping has emerged, promising to achieve such flexibility. In this work, we propose to incorporate design roadmapping to model-based roadmapping and integrate it with various user testing approaches into a single tool to support a flexible data-driven NPD planning process.

7.1 Introduction

This chapter presents the third contribution addressing the research gap G3: integrating UX-focused MB-TRM with other NPD processes, specifically with user testing. It complements PS stage of DRM, pursuing the major research objective of this Ph.D.

User-centered business needs the design roadmapping procedure to be data-driven and evidence-based. However, Kim, Beckman, and Agogino (2018) identified the lack of feedback loop from users in the current TRM practices. At the same time, several contributions established links between feature modeling language syntax and popular approaches to user testing — Kansei Engineering (KE) and Conjoint Analysis (CA)

(see 7.2). Accordingly, it seems an interesting research objective to test such integration and a capability of MB-TRM to provide an interface between diverse departments and processes in the real industrial case study. This chapter presents its setting and results.

7.2 Related literature

In this section only the most important contributions are listed. The comprehensive literature review for this section is provided in the subsection 3.2.3.

Phaal, Farrukh, and Probert (2005) discussed an integrative role of technology roadmapping, i.e., TRM, a convenient common place where multidisciplinary teams store the essential information. In this view, a roadmap should provide an interface to which different NPD and engineering approaches get linked and synchronized. Feature models play this integrative role in our UX-focused framework and should be easily linked to user testing.

Kett, Schmitt, and Wartzack (2017) joined market segmentation with product features through Kansei engineering. Muller and Lillack (2011) demonstrated the possibility of seamless integration of feature modeling and Conjoint analysis.

7.3 MB-TRM integrated with user testing

Figure 7.1 shows the data-intensive processes that were or may be potentially integrated into MB-TRM. Classical computer-aided roadmapping often includes data mining (patent and scientific literature bibliometrics and text mining). In NPD, opinion mining and sentiment analysis are popular and may be integrated for product assessment and trend forecasting. Yuskevich et al. (2021b) demonstrated possible integration of MB-TRM with trend extrapolation. Jiao, Zhang, and Wang (2007) demonstrated the possibility of product family optimization with a genetic algorithm. Geum, Lee, and Park (2014) demonstrated a combination of technology roadmapping and system dynamics for business modeling. A significant number of contributions complemented TRM with decision-making approaches (AHP, DSM, QFD, etc., see section 3.1.1).

All these approaches can bring added value in the context of TRM and even more so in the MB-TRM. We are especially interested in the UX-related processes within this contribution, specifically UX modeling, user testing, and user profiling.

UX heavily depends on the context in which interaction between a user and a product happens. For example, ambient light inside the vehicle will be perceived differently during night/day (in sunny weather ambient light may be not perceivable at all). Therefore, we are specifically interested in understanding how UX changes depend on user goals and use cases, and identify this change in our user testing. We introduced a concept of user stories from IT domain (Cohn 2004; Choma, Zaina, and Beraldo 2016) The format of a user story is the following: "I want <feature> in order to <goal> when/during <use case>". The expressions used in user stories are also linked to concepts of the corresponding feature trees (product features, use cases, goals).

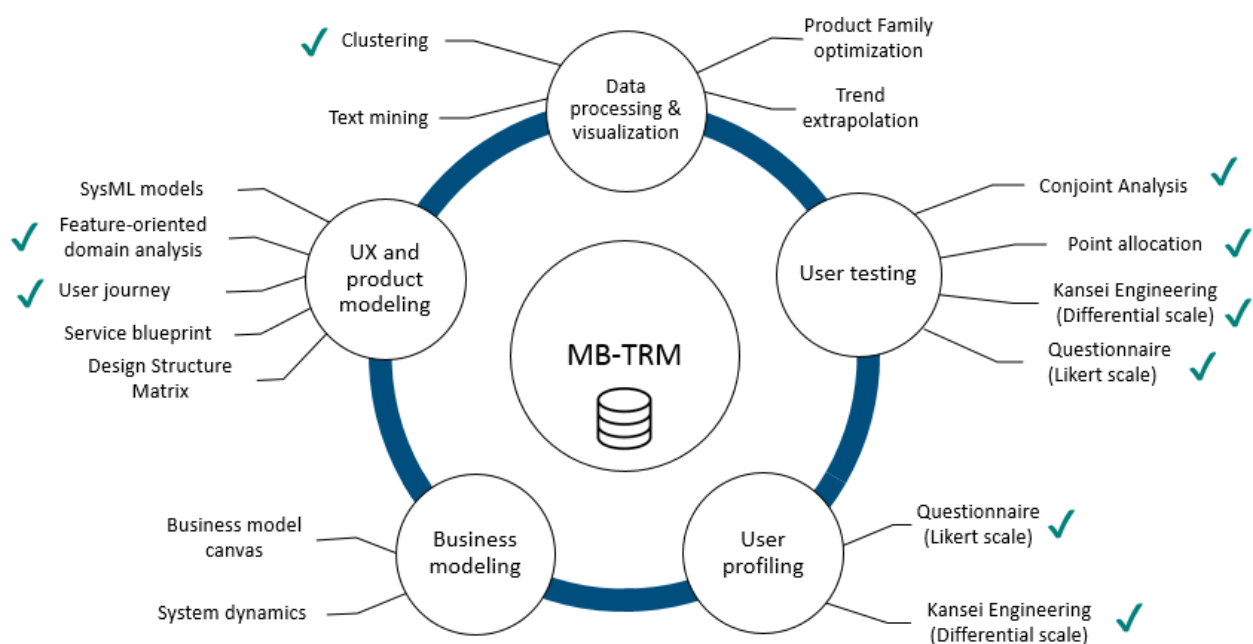


Figure 7.1: MB-TRM as a platform for integration NPD processes

7.4 Case study: Cockpit Multimodal Interactive for semi-autonomous vehicle

7.4.1 Design of experieiment

We implemented this case study usign the designed support tool. User tests were generated semi-automatically from four feature trees that model personas (Fig. 7.2), user goals (Fig. 7.3), use cases (Fig. 7.4), and product functions and interfaces (Fig. 7.5).

The team of the CMI project invited around 60 people to participate in the cockpit's prototype testing.

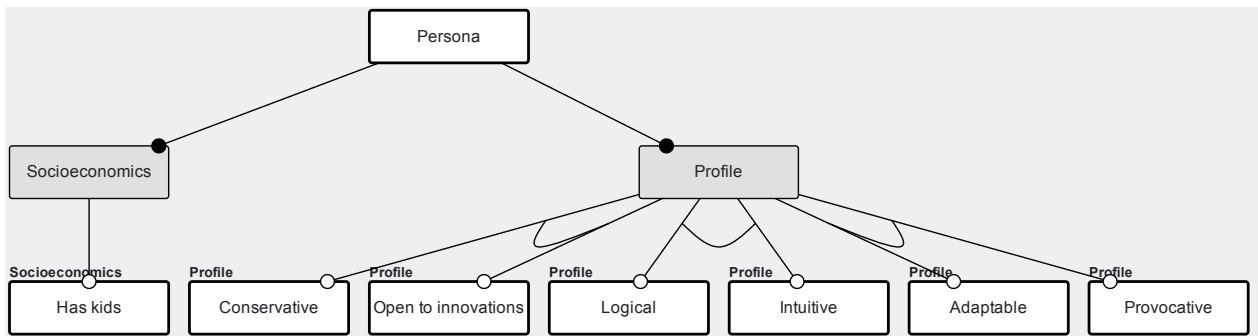


Figure 7.2: Feature model representing personas

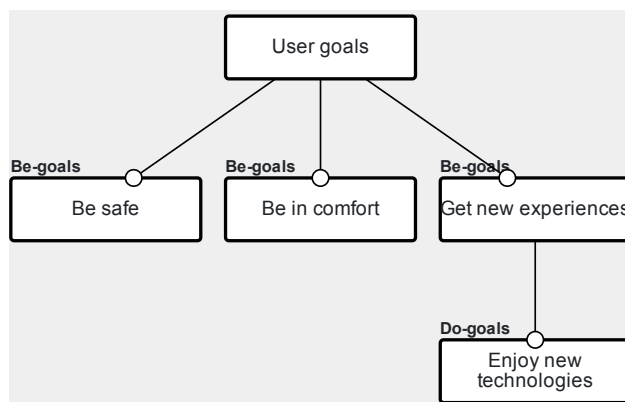


Figure 7.3: Feature model representing user goals

Participants went through a preselection process: they should be experienced drivers (more than five years), not working in the automotive industry, not feeling sickness in a simulated environment, etc. Then, to answer the questions of the CMI project, users were divided into two groups: early adopters and more conservative users. According to the objectives of the CMI project these two groups of people were related to two personas: Iris and Roberto respectively.

Participants went through a lengthy 3-hour procedure consisting of testing the cockpit prototype in our laboratory (5 different scenarios of driving in a simulated environment), collecting the real-time physiological measurements, going through a series of interviews and questionnaires.

The test described here was considered a side project; it was conducted online after the main session. Users might choose not to participate in this additional session. Only a preselected group of conservative users participated in our study. The response rate was 55%. Twelve users, who answered our online questionnaire, tested a basic

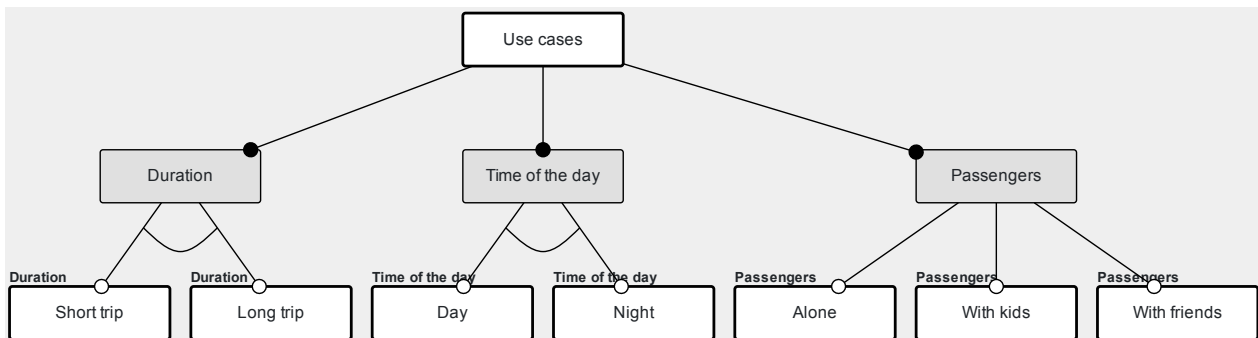


Figure 7.4: Feature model representing use cases

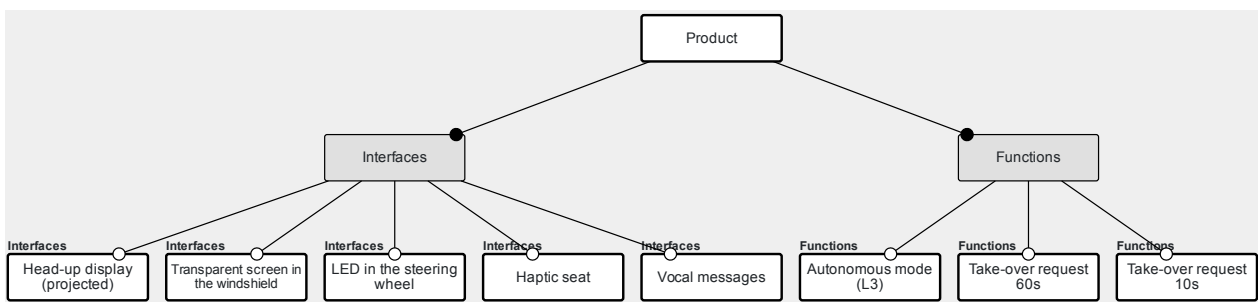


Figure 7.5: Feature model representing product functions and interfaces

version of a cockpit (two levels of automation — manual and fully autonomous (level 4), vocal and non-vocal signals, haptic seat, halo around windshield, HUD, illuminated steering wheel). The users classified as early adopters and users tested a version of a cockpit additionally equipped with a holographic virtual assistant did not receive the questionnaire.

The proposed process is described in the following. At the beginning of the test, a participant fills the KE questionnaire (self-assessment) to indicate their emotional/aesthetical profile (see fig. 7.6). Then, another KE questionnaire with different sets of adjectives is filled to assess the product's perceived aesthetics (see fig. 7.7). Then, the CA session is conducted (it is shown in figure 7.8). Following the concept of FODA-CA mapping (Muller and Lillack 2011), options were generated automatically from the feature tree that represents all possible product versions. The final screen is for the point allocation method applied to user stories (see fig. 7.9).

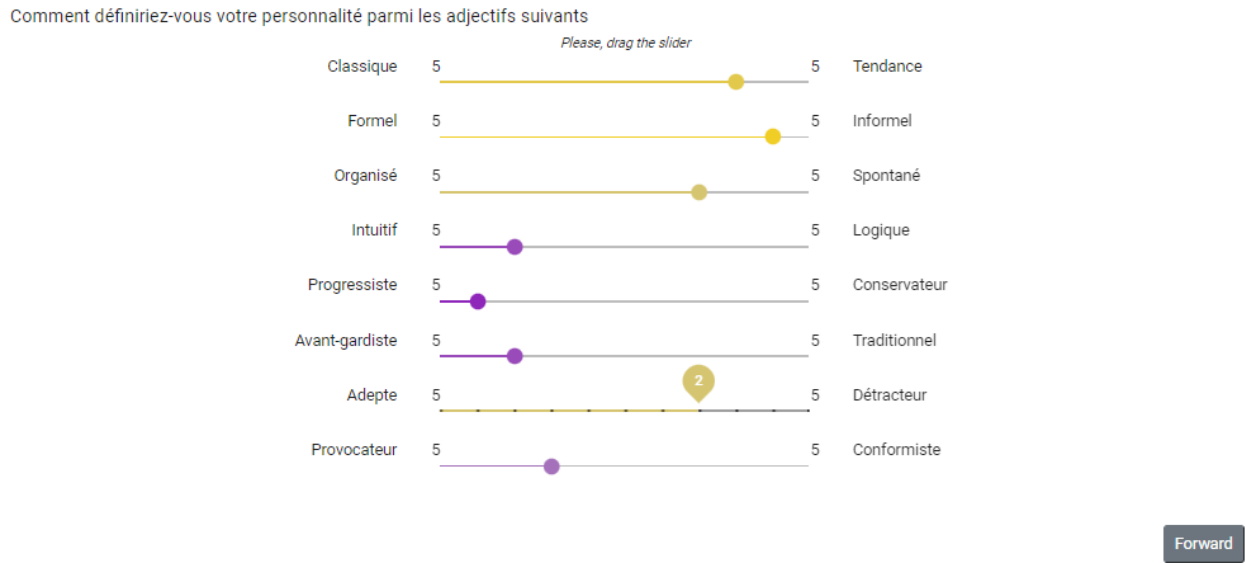


Figure 7.6: Kansei words self-assessment page (in French)

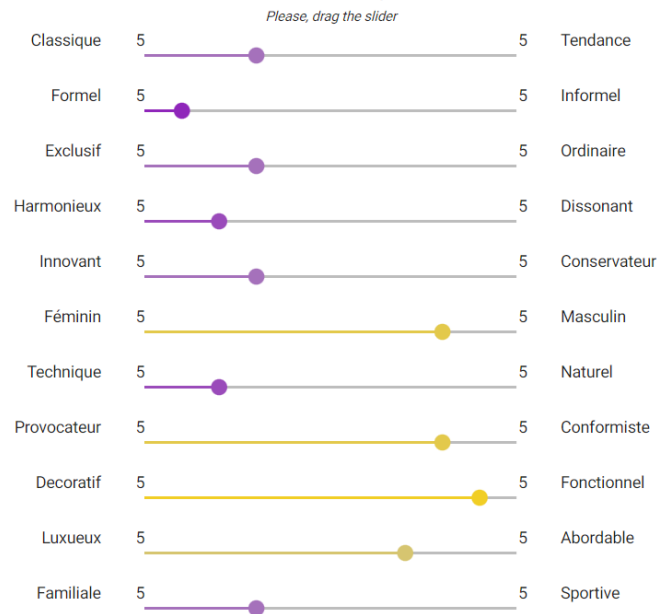
7.4.2 Results

We received answers from 12 users pre-classified as conservative users (which reduces the sensitivity of the following tests as the first stage of our procedure aims to classify users with KE differential adjectives). We processed the KE self-assessment results with k-means clustering. We identified three clusters (see Fig. 7.10, and Fig. 7.11), which we refer to as Conservative-Nonconformist (2 participants), Conservative-Logical (6 participants), Conservative-Intuitive (4 participants). The responses collected in the following stages were analysed for these three groups separately. Despite a small sample and the fact that users were already preselected, users of three subclusters demonstrated distinct opinions on the products. The overall opinions over the different features were also validated with respect to the conclusions made based on the interviews.

After clustering, users went through a stage of product assessment with differential Kansei adjectives (Fig. 7.7). The goal of this stage is to assess the aesthetic perception of a product (Fig. 7.12).

The next stage is conjoint analysis. It is an iterative test consisting of 5 steps where participants rate two alternative product options with a different set of features and price (Fig. 7.8). The results of conjoint analysis for each three clusters of participants is shown in figures 7.13 and 7.14.

Comment évalueriez-vous l'esthétique du cockpit que vous avez testé parmi les adjectifs suivants



Forward

Figure 7.7: Kansei words product aesthetics assessment page (in French)

Finally, participants were asked to allocate 20 points between alternative use cases (long and short trips, journey with kids) and goals concerning these use cases (safety, comfort). This test links three feature models together: feature models of personas, use cases, user goals. Results are presented in figures 7.15 and 7.16.

7.5 Discussion

The main goal of this chapter was to demonstrate the feature tree integration with various user testing procedures with an objective to enable data-driven roadmapping. To do this, user tests should be generated from the feature trees, sent to users, and processed semi-automatically. In this context, it is important to demonstrate the pipeline feature tree → user test → processing → strategic insight.

Despite the small sample, some of the product aspects were assessed by different user clusters differently with statistical significance. For example, the difference in comfort during driving (first statement in Fig. 7.16) between 'Conservative-Logical' and 'Conservative-Intuitive' groups is significant with $p=0.12$. 'Conservative-Logical' group

Imaginez-vous, dans 6 mois à partir de maintenant, dans une concession automobile en train d'acheter une nouvelle voiture. Évaluez entre 0 et 10 votre intérêt pour acheter l'option au prix indiqué (0=pas du tout intéressé, 10=tout à fait intéressé). Le prix de chaque option sera ajouté au prix de base de la voiture.

Les caractéristiques soulignées indiquent les différences entre les options, c'est à dire que les caractéristiques qui *ne sont pas* soulignées sont communes aux deux options.

Le mode Chauffeur figurera dans les options qui vous seront proposées.

The screenshot displays two product options for rating. Each option is presented in a white box with a vertical slider on the right side, ranging from 0 to 10. Option 1 is titled 'Option 1' and lists features: 'Ecran dans le pare-brise (se situe au centre du véhicule)', 'LED dans le volant' (highlighted in yellow), 'Messages vocaux dans appui-tête', 'Mode chauffeur', and 'Siege vibrant'. Its price is 4000 EUR, and the slider is set to 3. Option 2 is titled 'Option 2' and lists features: 'Indicateur tête haute (se situe au-dessus du tableau de bord)' (highlighted in yellow), 'Ecran dans le pare-brise (se situe au centre du véhicule)', 'Messages vocaux dans appui-tête', 'Mode chauffeur', and 'Siege vibrant'. Its price is 4000 EUR, and the slider is set to 10. Below the options are two buttons: '< Go back' (yellow) and 'Forward' (grey).

Figure 7.8: Conjoint analysis product assessment page (value)

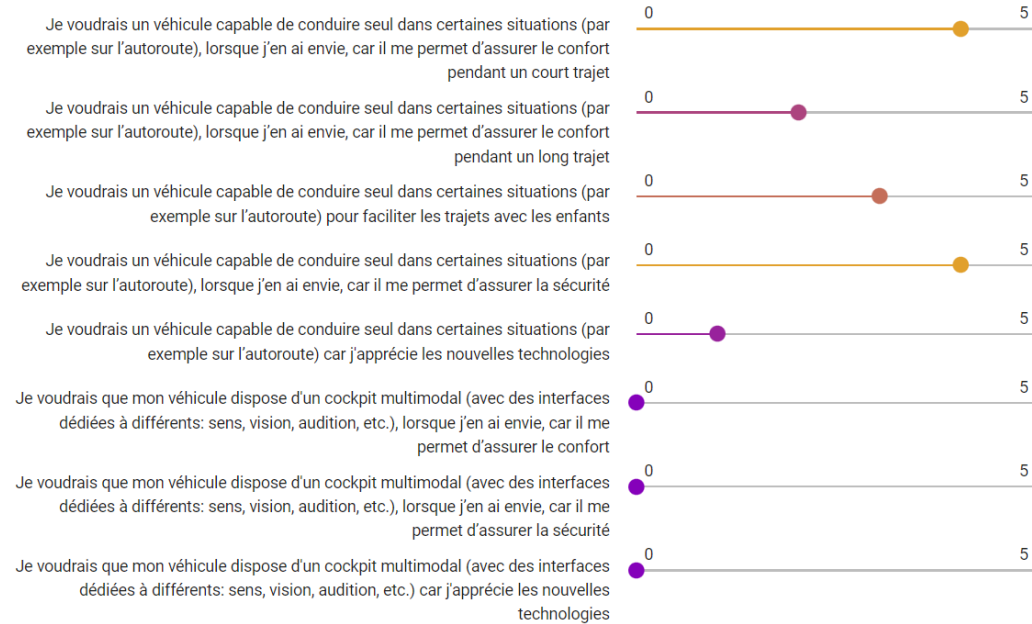
values safety more (fourth statement in Fig. 7.16) compared to 'Conservative-Intuitive' with $p=0.17$. The results of the conjoint analysis also exhibited different opinions between clusters (see, for example, results for transparent windshield screen, haptic seat, and LED in the steering wheel on the Fig. 7.8).

However, our main goal was not to validate Conjoint analysis or Kansei engineering (their validity is proven more than once in the literature) but to demonstrate the integration with MB-TRM. This integration is indeed tight: setting up the conjoint analysis from a product feature tree (Fig. 7.5) is not going beyond assigning prices for each feature and excluding some features from the testing if required. CA data processing is fully automated and recalculated each time a new response is received. In KE, only parameters of an algorithm should be tuned (e.g., number of clusters). In user stories rating, the designer needs to select features of interest from the set of available features (Fig. 7.4, 7.3) and refine statements to form a concise human-readable sentence.

Consequently, we demonstrated the almost effortless and efficient feedback loop from the end-users to strategic planning specialists is indeed feasible within our MB-TRM approach.

Indiquez si vous êtes d'accord avec les affirmations suivantes **en distribuant 20 points**

Please, drag the slider



6/20

Forward

Figure 7.9: User stories product assessment page (use cases)

P017	Conservative-Intuitive
P024	Conservative-Intuitive
P022	Conservative-Logical
P020	Conservative-Logical
P013	Conservative-Intuitive
P009	Conservative-Nonconformist
P025	Conservative-Intuitive
P012	Conservative-Intuitive
P015	Conservative-Logical
P010	Conservative-Logical
P001	Conservative-Nonconformist
P019	Conservative-Intuitive

Figure 7.10: Clusters built based on the KE self-assessment

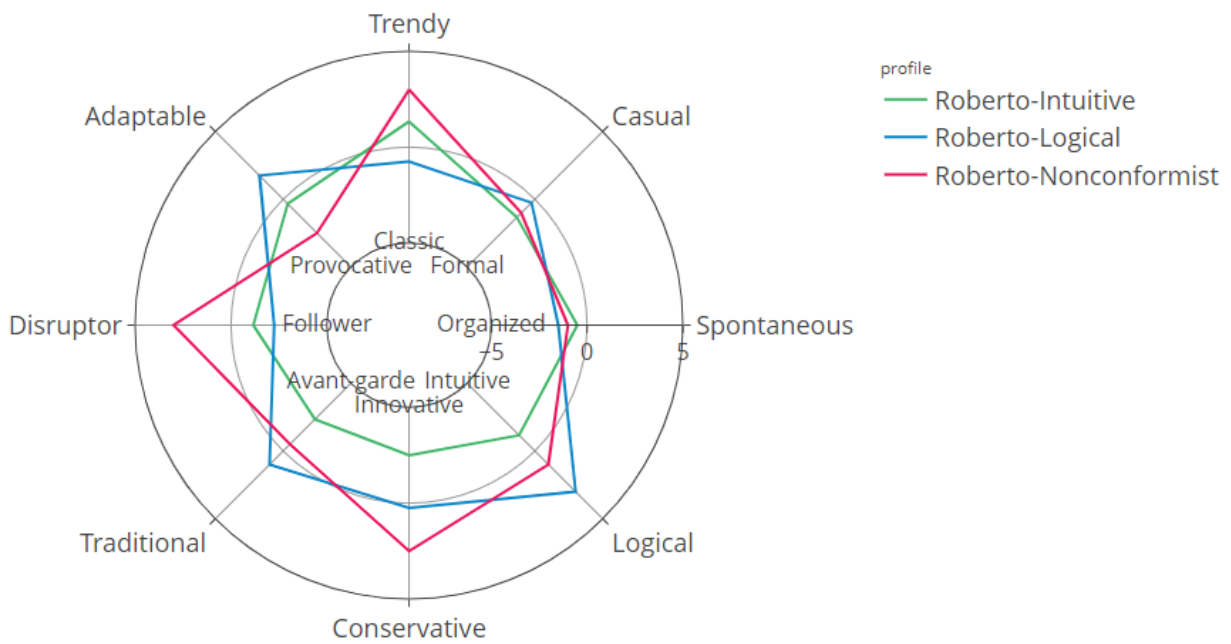


Figure 7.11: Spider chart representing properties of three identified clusters

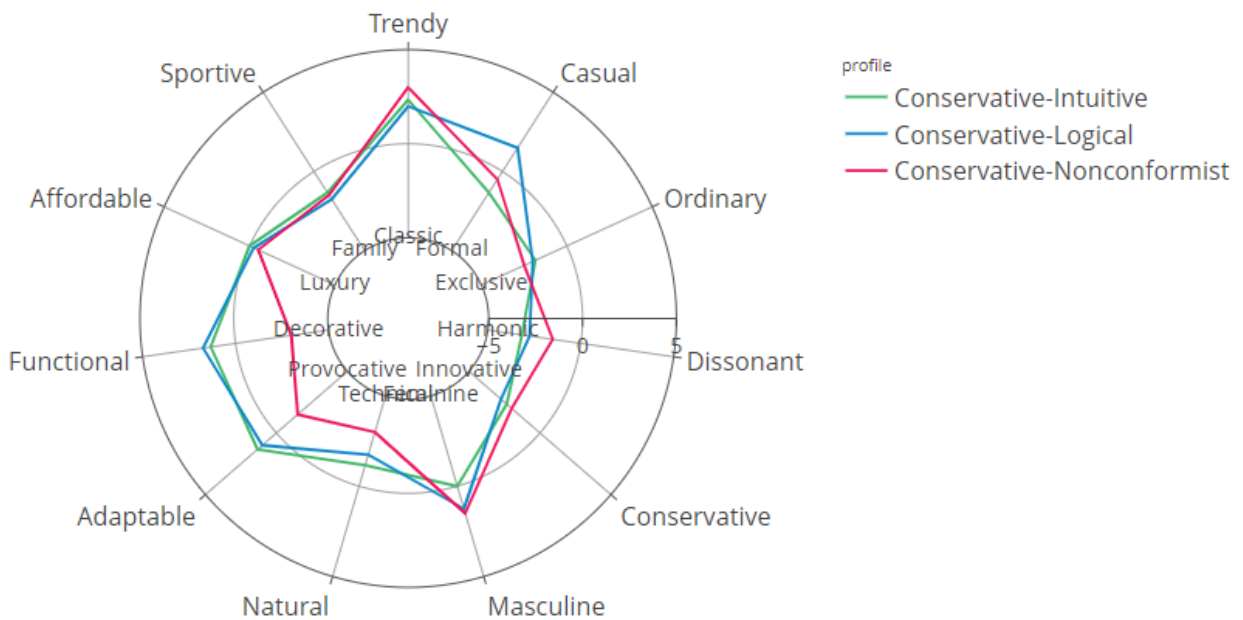


Figure 7.12: KE product assessment four three clusters of participants

	Conservative-Intuitive	Conservative-Logical	Conservative-Nonconformist
Indicateur tête haute (se situe au-dessus du tableau de bord)	0.145123	0.305351	-0.260728
Ecran dans le pare-brise (se situe au centre du véhicule)	1.644887	0.502765	0.673669
LED dans le volant	2.014202	2.546342	-0.478412
Messages vocaux dans appui-tête	1.291921	1.530620	1.828536
Mode chauffeur	1.344992	1.959382	2.299202
Siege vibrant	1.620617	0.848742	2.393614

Figure 7.13: Results of the CA data processing

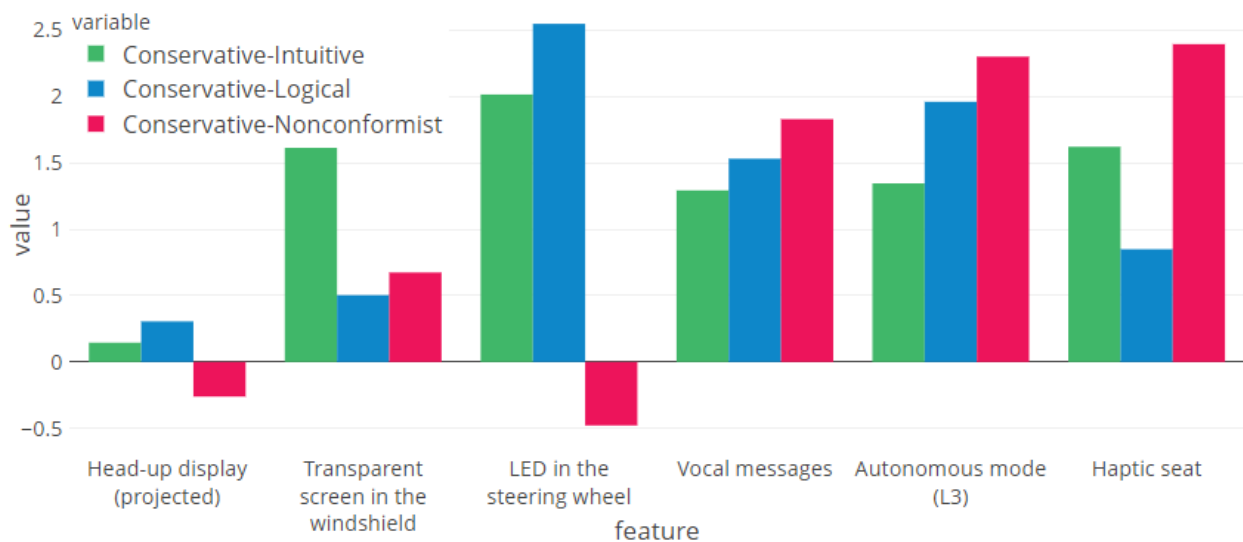


Figure 7.14: Chart representing the aggregated relative importance of product features depending on participant profile

	Je voudrais un véhicule capable de conduire seul dans certaines situations (par exemple sur l'autoroute), lorsque j'en ai envie, car il me permet d'assurer le confort pendant un court trajet	Je voudrais un véhicule capable de conduire seul dans certaines situations (par exemple sur l'autoroute), lorsque j'en ai envie, car il me permet d'assurer le confort pendant un long trajet	Je voudrais un véhicule capable de conduire seul dans certaines situations (par exemple sur l'autoroute) pour faciliter les trajets avec les enfants	Je voudrais un véhicule capable de conduire seul dans certaines situations (par exemple sur l'autoroute), lorsque j'en ai envie, car il me permet d'assurer la sécurité	Je voudrais un véhicule capable de conduire seul dans certaines situations (par exemple sur l'autoroute) car j'apprécie les nouvelles technologies	Je voudrais que mon véhicule dispose d'un cockpit multimodal (avec des interfaces dédiées à différents: sens, vision, audition, etc.), lorsque j'en ai envie, car il me permet d'assurer le confort	Je voudrais que mon véhicule dispose d'un cockpit multimodal (avec des interfaces dédiées à différents: sens, vision, audition, etc.), lorsque j'en ai envie, car il me permet d'assurer la sécurité	Je voudrais que mon véhicule dispose d'un cockpit multimodal (avec des interfaces dédiées à différents: sens, vision, audition, etc.) car j'apprécie les nouvelles technologies
0	1	4	5	1	2	2	2	3
1	2	5	3	1	3	1	2	3
2	5	5	5	5	0	0	0	0
3	4	5	3	0	3	3	0	2
4	2	4	4	3	3	2	2	0
5	0	0	0	0	0	0	0	0
6	5	5	4	3	0	0	3	0
7	0	0	0	0	0	0	0	0
8	5	0	0	3	2	3	5	2
9	1	5	1	4	5	1	1	2
10	1	4	3	3	3	3	3	0
11	0	0	3	0	3	3	2	3

Figure 7.15: Raw results of user stories point allocation

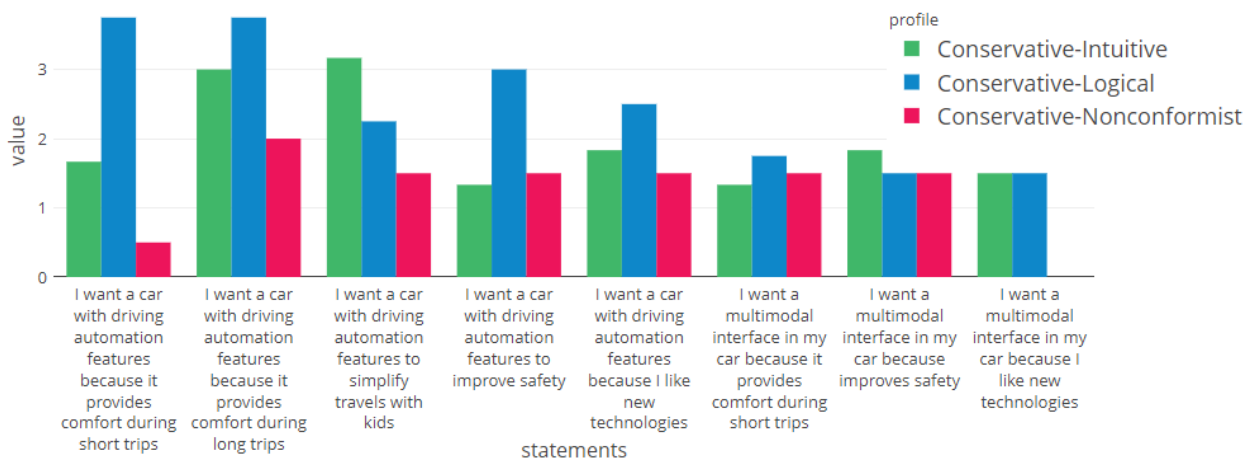


Figure 7.16: Aggregated results of point allocation method among user clusters

8 DISCUSSION

8.1 Contribution

The main research objective of this thesis is to explore ways and design a support tool for strategic planning in companies focusing on the User Experience. The actuality of the research is hard to overestimate: most of the products we use and like were designed with great emphasis on UX. The design of such a methodology and a support tool is not trivial: the UX is multifaceted, ambiguous, changing in time, and deeply individual.

Based on the literature, we have accepted the following properties of target support tool as assumptions: the approach should be flexible (i.e., adaptable to various industrial contexts), agile and iterative (i.e., constantly updated in response to new inputs), collaborative on the strategic level (between marketing, UX design, engineering, and technology teams), and have a constantly active feedback loop from end-users.

We proposed a novel approach to achieve this objective by integrating model-based roadmapping and design roadmapping, and developed a concise Domain-specific language for UX design, marketing strategies, product architectures, and families of technologies.

To our knowledge, such an approach is one of a kind. It is ideologically close to the lean startup methodology (Ries 2011) but applies to large organizations and not startups, or to design thinking methodology (Brenner and Uebernickel 2016; Hehn et al. 2020), but in application to product family and strategy rather than to a single product.

To achieve the main research goal, three distinct research gaps were identified and addressed: absence of generic MB-TRM metamodel, lack of the DSL, suitable for representing marketing/UXD/product/technology uniformly, and lack of literature elaborating link with the user testing procedures.

The first contribution was validated using a novel computer-assisted procedure. Metamodel, implemented in JetBrains MPS language workbench, was first instantiated to three DSLs to represent three selected roadmaps belonging to different domains. Then, the information presented in these selected papers is represented in a corresponding DSL. Consequently, the chain of operations (see Appendixes B, C, D, E, F, G, H) Base

Language (M3) → Metamodel (M2) → DSL (M1) → User data (M0, which is equivalent to data presented in the original papers) constitutes a formal proof that metamodel is valid with regard to three selected case studies. The fact that these case studies are sufficiently different (in terms of the organizational scale, industrial domain, and objectives) provides reasons to consider that our metamodel is also valid with respect to the broad majority of existing roadmaps.

The second contribution was internally validated with the case studies (one illustrative and one industrial not presented in the scope of this manuscript). We developed our desired support tool and constructed a meaningful case study with it. By doing so, we demonstrated that our approach is internally consistent and correctly models real-world concepts. For the external validation, we gathered the following opinions in the process of demonstration of tool's capabilities to our industrial partners:

- The tool will help to capitalize and reuse knowledge
- Methodology and DSL correctly represent our concepts of interest in a visual and compact form (marketing, UXD, and product departments)
- It will help to unify our cross-department definitions

In the third contribution we demonstrated seamless integration between DSL and user testing approaches. We conducted a case study based on CMI project with real users participation. We cluster users on three groups based on their self-reported identity, and these three groups demonstrated significantly different opinions on some features. This provides insights that can be directly used in the process of product family planning (e.g., not include HUD and LED in the steering wheel for a certain user profile/market niche).

Our contribution is largely *prescriptive*: we proposed a new design practice with favorable properties to solve an actual problem. At the same time, our approach is not a result of pure creativity but more of a prediction based on the comprehensive *descriptive* review-based study. In this view, this thesis provides an in-depth investigation of the modern design practices at the intersection of strategic planning, UX design, and MBSE. Hence, it adds understanding to the evolution of this field and, therefore, "may be counted as scientific" (according to Vermaas (2014)).

8.2 Limitations and weak points

8.2.1 Major limitations

The main limitation of this work is that we have not fully validated it in the real industrial context (due to the time and resource limitations of a single Ph.D. project), and, therefore, we are not able to answer the global overarching research question positively: does the agile model-based UXD and roadmapping help to design better products and survive in a competitive environment?

To answer this question, long and resource-intensive studies need to be conducted in a close collaboration with industrial partners. This drawback is significant, but not critical: DRM allows such a possibility. In the best case scenario (if our reasoning and conclusions will be supported in the community), other researches can continue gather evidences of pragmatic validity of this approach.

A considerable criticism may also be directed at the core idea of expressing vague and ambiguous UX-related concepts with formal syntax. Such concerns were raised by UX design specialists in course of the study. Despite we succeed in expressing UX concepts formally in a number of our case studies, the feasibility and usefulness of this view should be further studied in the future in various contexts.

8.2.2 Issues of implementation in industrial settings

The following difficulties may be encountered in the process of its implementation in the industrial setting:

- Perceived value. For the methodology to be adopted, a significant number of key stakeholders should be committed to it (and the rest should not have a strong opinion against it). For that, they should be convinced that it delivers soon and significant benefits.
- Learning curve. We estimate that our methodology is easy to learn. However, we did not evaluate this aspect quantitatively and, therefore, can not be wholly convinced that benefits overweight the costs of transition and training.
- Scaleability issues. Real industrial case studies are typically large and complex. One of the advantages of digital tools is that they scale well (compared to document-based processes). However, due to various reasons (from technical to usability

issues), the support tool and the methodology itself may fail under these conditions.

- Usability issues. Usability is one of the key aspects of software. Even if our support tool delivers required functionality, poor usability (bugs, outdated or not well-designed interfaces, low performance) can hamper its adoption.

8.2.3 Relation to the commercially available tools

A number of commercial tools implement a part of the envisioned functionality of our support tool. However, none repeats the functionality entirely.

Miro is a popular commercial software implementing online collaborative whiteboards. In Miro, various software and hardware development artifacts can be created: roadmaps, user journey maps, DSMs, and dozens of others. The main difference is that the concepts represented in Miro are purely visual. They are not interconnected, and there is no underlying syntax. Therefore, any features related to modeling, analytics, and automatization are not available in Miro.

Trello, Jira, Gitlab, YouTrack are so-called issue trackers used in agile teams to organize the working process. The flexible data model features JetBrains YouTrack, and, therefore, except standard Scrum-related concepts (issue, bug, user story), allows the creation of UX-related concepts (use case, persona, product feature). However, the only available artifact (as in all tools mentioned above) is the Kanban board. The reason is that these tools are designed for the specific purpose of supporting agile IT teams and cannot be used for UX modeling and strategic planning.

Aha! and ProductPlan are the tools designed for product management and provide rich roadmap development and visualization capabilities. Aha! includes user testing tools enabling feedback loop from customers. These tools are the closest analogs of the support tool and methodology we created at the strategic level. However, they do not include DSL on a concrete level and, therefore, are unable to analyze the domains of interest in detail, execute consistency checks, etc.

The clear interest to development of commercial tools underlines the importance of the identified problem. At the same time, our approach treats this task comprehensively and rigorously, and our support tool has several important differentiators.

8.2.4 Directions for future work

To address the major weak points of the present study, future research should be directed toward gathering more pieces of evidence of the pragmatic validity of the approach in a real industrial context.

The theory of MB-TRM may also be broadened by enriching the presented approach with other perspectives, e.g., the resource-based view of the firm, or quantitative road-mapping.

The resource-based view of the firm is inextricably linked to the strategic planning, as the strategic planning should at least take into account firm's core competencies (as it was mentioned in the definition provided in the introduction). At most, the long-term planning inside the organization should aim at development of the core competencies needed for firm's sustainability.

This research is focused on the qualitative factors and links, due to the mainly qualitative nature of UX. Further elaboration may be needed on the quantitative aspect of the proposed syntax to enable the integration of the proposed approach with various numerical methods of technology management, econometrics, and computer science.

9 CONCLUSIONS

This Ph.D. addresses the problem of strategic planning in companies building their products and services with an emphasis on the user experience. Society and business tend to become more complex. The more sophisticated the business environment will be, the more influential the concept of UX will become. Companies focusing on the UX are located at the end of the value chain (closer to the customer), and, therefore, their business is more profitable. Organizations that cannot address end-user needs directly are doomed to the dependent and vulnerable role of white-label technology providers.

Ultimately, our approach aims at helping companies that set a strategic goal of becoming more user-oriented. Our main hypothesis is that this objective can be achieved by integrating design roadmapping with model-based roadmapping. The resulting methodology and a support tool enable the agile, collaborative, and evidence-based process of UX-focused strategy elicitation.

In this Ph.D. we contributed to the entire MB-TRM domain by developing a metamodel that defines a model-based roadmap in abstract terms. Based on feature modeling language, we created a DSL that, in our view, adequately represents a strategic plan of a user-centered business. We implemented a support tool with the end-users' feedback loop, which helped validate our approach with a number of case studies.

We validated our contributions with three case studies and, where it was possible (contribution 1), with more formal example-driven validation.

We expect that the proposed approach will positively influence the following aspects of strategic planning.

First, adopting such a tool will enforce standardization of the UX-related concepts among different departments of a company (namely, design, engineering, and marketing), reduce conflicts, and improve communication efficiency.

Second, formal syntax offers a number of opportunities for consistency checks and work automation. For example, certain product features are useful for people with children (and this factor will have implications on use cases, user goals, and UX metrics). If a product feature requires having kids from a persona that has not, this contradiction will be highlighted.

Third, the capability of storing the standardized concepts in computer memory promotes information reuse (e.g., personas can be reused between several projects).

Fourth, the roadmap's hierarchical structure helps to keep separation of concerns, permitting for some to see the global picture and for others — more detailed information belonging to the domain of interest, while the consistency check happens behind the scene and notify interested parties when the logical structure of a model is not held.

Fifth, integration with user tests helps to organize an inexpensive iterative data-driven process of the new product family planning.

Bibliography

- 2015 NASA Technology Roadmaps (Archive) (Apr. 29, 2015). NASA. Library Catalog: www.nasa.gov. url: <http://www.nasa.gov/offices/oct/home/roadmaps/index.html> (visited on 07/24/2020).
- 2020 NASA Technology Taxonomy (Aug. 20, 2019). url: <http://www.nasa.gov/offices/oct/taxonomy/index.html> (visited on 11/02/2021).
- Abrantes, Rui and José Figueiredo (July 1, 2014). "Feature based process framework to manage scope in dynamic NPD portfolios." In: *International Journal of Project Management* 32.5, pp. 874–884. issn: 0263-7863. doi: [10.1016/j.ijproman.2013.10.014](https://doi.org/10.1016/j.ijproman.2013.10.014).
- Al-Ali, Ahmed Ghanim and Robert Phaal (June 2019). "Design Sprints for Roadmapping an Agile Digital Transformation." In: *2019 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*. 2019 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC). Valbonne Sophia-Antipolis, France: IEEE, pp. 1–6. isbn: 978-1-72813-401-7. doi: [10.1109/ICE.2019.8792597](https://doi.org/10.1109/ICE.2019.8792597).
- Alben, Lauralee (May 1996). "Quality of experience: defining the criteria for effective interaction design." In: *Interactions* 3.3, pp. 11–15. issn: 1072-5520, 1558-3449. doi: [10.1145/235008.235010](https://doi.org/10.1145/235008.235010).
- Albers, Albert, Lukas Krämer, and Masis Arslan (Jan. 1, 2017). "Roadmap-based Methodology for the Forecasting of Competences within Automotive Product Development." In: *Mid-Range Management Theory: Competence Perspectives on Modularity and Dynamic Capabilities*. Ed. by Ron Sanchez et al. Vol. 8. Research in Competence-Based Management. Emerald Publishing Limited, pp. 109–130. isbn: 978-1-78714-404-0 978-1-78714-403-3. doi: [10.1108/S1744-211720170000008005](https://doi.org/10.1108/S1744-211720170000008005).
- Albright, Richard E. and Thomas A. Kappel (Mar. 1, 2003). "Roadmapping In the Corporation." In: *Research-Technology Management* 46.2. Publisher: Routledge, pp. 31–40. issn: 0895-6308. doi: [10.1080/08956308.2003.11671552](https://doi.org/10.1080/08956308.2003.11671552).
- Alcantara, Douglas Pedro de and Mauro Luiz Martens (Jan. 1, 2019). "Technology Roadmapping (TRM): a systematic review of the literature focusing on models." In: *Technological Forecasting and Social Change* 138, pp. 127–138. issn: 0040-1625. doi: [10.1016/j.techfore.2018.08.014](https://doi.org/10.1016/j.techfore.2018.08.014).
- Aldrich, Theodore B., Sandra M. Szabo, and Carl R. Bierbaum (1989). "The Development and Application of Models to Predict Operator Workload During System Design." In:

- Applications of Human Performance Models to System Design*. Ed. by Grant R. McMillan et al. Boston, MA: Springer US, pp. 65–80. isbn: 978-1-4757-9244-7. doi: [10.1007/978-1-4757-9244-7_5](https://doi.org/10.1007/978-1-4757-9244-7_5).
- Alexander, Ian F. and Neil Maiden, eds. (Sept. 24, 2004). *Scenarios, Stories, Use Cases: Through the Systems Development Life-Cycle*. 1st edition. Chichester ; Hoboken, NJ: Wiley. 548 pp. isbn: 978-0-470-86194-3.
- Amyot, Daniel and Gunter Mussbacher (2002). "URN: Towards a new standard for the visual description of requirements." In: *International Workshop on System Analysis and Modeling*. Springer, pp. 21–37.
- An, Yoonjung, Sungjoo Lee, and Yongtae Park (Jan. 1, 2008). "Development of an integrated product-service roadmap with QFD: A case study on mobile communications." In: *International Journal of Service Industry Management* 19.5. Publisher: Emerald Group Publishing Limited, pp. 621–638. issn: 0956-4233. doi: [10.1108/09564230810903497](https://doi.org/10.1108/09564230810903497).
- Atkinson, C. and T. Kuhne (Sept. 2003). "Model-driven development: a metamodeling foundation." In: *IEEE Software* 20.5. Conference Name: IEEE Software, pp. 36–41. issn: 1937-4194. doi: [10.1109/MS.2003.1231149](https://doi.org/10.1109/MS.2003.1231149).
- Barney, Jay (1991). "Firm resources and sustained competitive advantage." In: *Journal of management* 17.1, pp. 99–120.
- Beckman, Sara L. and Michael Barry (Oct. 1, 2007). "Innovation as a Learning Process: Embedding Design Thinking:" in: *California Management Review*. Publisher: SAGE PublicationsSage CA: Los Angeles, CA. doi: [10.2307/41166415](https://doi.org/10.2307/41166415).
- Bekhradi, Alexandre et al. (Nov. 2, 2018). "Decision Support for R&D Activities of Innovative Technologies." In: ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers Digital Collection. doi: [10.1115/DETC2018-85657](https://doi.org/10.1115/DETC2018-85657).
- Bell, Alex E. (Mar. 2004). "Death by UML Fever: Self-diagnosis and early treatment are crucial in the fight against UML Fever." In: *Queue* 2.1, pp. 72–80. issn: 1542-7730, 1542-7749. doi: [10.1145/984458.984495](https://doi.org/10.1145/984458.984495).
- Bellet, Thierry et al. (2011). "Human Driver Modelling and Simulation into a Virtual Road Environment." In: *Human Modelling in Assisted Transportation*. Ed. by P. Carlo Cacciabue et al. Milano: Springer Milan, pp. 251–262. isbn: 978-88-470-1821-1. doi: [10.1007/978-88-470-1821-1_27](https://doi.org/10.1007/978-88-470-1821-1_27).

- Benavides, David, Sergio Segura, and Antonio Ruiz-Cortés (Sept. 2010). "Automated analysis of feature models 20 years later: A literature review." In: *Information Systems* 35.6, pp. 615–636. issn: 03064379. doi: [10.1016/j.is.2010.01.001](https://doi.org/10.1016/j.is.2010.01.001).
- Benavides, David, Pablo Trinidad, and Antonio Ruiz-Cortés (2005). "Automated reasoning on feature models." In: *International Conference on Advanced Information Systems Engineering*. Springer, pp. 491–503.
- Bennis, Warren and Burt Nanus (1986). *Leaders: The Strategies for Taking Charge*. Google-Books-ID: emy4zshg8jcC. Harper & Row. 260 pp. isbn: 978-0-06-091336-6.
- Blank, Steve and Bob Dorf (Mar. 1, 2012). *Garry, S: Startup Owner's Manual. Vol. 1*. Pescadero, Calif: K & S Ranch. 571 pp. isbn: 978-0-9849993-0-9.
- Blessing, Lucienne TM and Amaresh Chakrabarti (2009). *DRM: A design reseach methodology*. Springer.
- Boy, Guy A. (1998). "Cognitive function analysis for human-centered automation of safety-critical systems." In: *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '98*. the SIGCHI conference. Los Angeles, California, United States: ACM Press, pp. 265–272. isbn: 978-0-201-30987-4. doi: [10.1145/274644.274682](https://doi.org/10.1145/274644.274682).
- Breckel, Alexander et al. (Oct. 12, 2021). "A domain-specific language for modeling and analyzing solution spaces for technology roadmapping." In: *Journal of Systems and Software*, p. 111094. issn: 0164-1212. doi: [10.1016/j.jss.2021.111094](https://doi.org/10.1016/j.jss.2021.111094).
- Brenner, Walter and Falk Uebernickel, eds. (2016). *Design Thinking for Innovation: Research and Practice*. Springer International Publishing. isbn: 978-3-319-26098-3. doi: [10.1007/978-3-319-26100-3](https://doi.org/10.1007/978-3-319-26100-3).
- Bresciani, Paolo and Paolo Donzelli (2003). "REF: A Practical Agent-Based Requirement Engineering Framework." In: *Conceptual Modeling for Novel Application Domains*. Ed. by Manfred A. Jeusfeld and Óscar Pastor. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 217–228. isbn: 978-3-540-39597-3. doi: [10.1007/978-3-540-39597-3_21](https://doi.org/10.1007/978-3-540-39597-3_21).
- Casalánguida, Hernán and Juan Eduardo Durán (Sept. 2, 2012). "Automatic generation of feature models from UML requirement models." In: *Proceedings of the 16th International Software Product Line Conference - Volume 2*. SPLC '12. New York, NY, USA: Association for Computing Machinery, pp. 10–17. isbn: 978-1-4503-1095-6. doi: [10.1145/2364412.2364415](https://doi.org/10.1145/2364412.2364415).

- Chami, Mohammad et al. (2018). "Towards solving MBSE adoption challenges: the D3 MBSE adoption toolbox." In: *INCOSE International Symposium*. Vol. 28. 1. Wiley Online Library, pp. 1463–1477.
- Cheng, M. N. et al. (Oct. 1, 2016). "A scenario-based roadmapping method for strategic planning and forecasting: A case study in a testing, inspection and certification company." In: *Technological Forecasting and Social Change* 111, pp. 44–62. issn: 0040-1625. doi: [10.1016/j.techfore.2016.06.005](https://doi.org/10.1016/j.techfore.2016.06.005).
- Choma, Joelma, Luciana A. M. Zaina, and Daniela Beraldo (2016). "UserX Story: Incorporating UX Aspects into User Stories Elaboration." In: *Human-Computer Interaction. Theory, Design, Development and Practice*. Ed. by Masaaki Kurosu. Lecture Notes in Computer Science. Cham: Springer International Publishing, pp. 131–140. isbn: 978-3-319-39510-4. doi: [10.1007/978-3-319-39510-4_13](https://doi.org/10.1007/978-3-319-39510-4_13).
- Christensen, Clayton M. (1993). "The Rigid Disk Drive Industry: A History of Commercial and Technological Turbulence." In: *Business History Review* 67.4, pp. 531–588. issn: 2044-768X, 0007-6805. doi: [10.2307/3116804](https://doi.org/10.2307/3116804).
- Chung, Lawrence and Julio Cesar Sampaio do Prado Leite (2009). "On non-functional requirements in software engineering." In: *Conceptual modeling: Foundations and applications*. Springer, pp. 363–379.
- Chung, Lawrence et al. (Dec. 6, 2012). *Non-Functional Requirements in Software Engineering*. Google-Books-ID: MNrcBwAAQBAJ. Springer Science & Business Media. 458 pp. isbn: 978-1-4615-5269-7.
- Cohn, Mike (Mar. 1, 2004). *User Stories Applied: For Agile Software Development*. 1st edition. Boston: Addison-Wesley Professional. 304 pp. isbn: 978-0-321-20568-1.
- Colombo, E. F. et al. (Jan. 1, 2020). "Value analysis for customizable modular product platforms: theory and case study." In: *Research in Engineering Design* 31.1, pp. 123–140. issn: 1435-6066. doi: [10.1007/s00163-019-00326-4](https://doi.org/10.1007/s00163-019-00326-4).
- Coyne, Richard (1995). "Designing information technology in the postmodern age." In: Daim, Tugrul U. and Terry Oliver (2008). "Implementing technology roadmap process in the energy services sector: A case study of a government agency." In: *Technological Forecasting and Social Change* 75.5, pp. 687–720. issn: 0040-1625. doi: <https://doi.org/10.1016/j.techfore.2007.04.006>.
- Darke, Peta, Graeme Shanks, and Marianne Broadbent (1998). "Successfully completing case study research: combining rigour, relevance and pragmatism." In: *Information systems journal* 8.4, pp. 273–289.

- Dori, Dov (2002). *Object-Process Methodology*. Berlin, Heidelberg: Springer Berlin Heidelberg. isbn: 978-3-642-62989-1 978-3-642-56209-9. doi: [10.1007/978-3-642-56209-9](https://doi.org/10.1007/978-3-642-56209-9).
- Dove, Laura, Stephen Reinach, and Irwin Kwan (May 7, 2016). "Lightweight Journey Mapping: The Integration of Marketing and User Experience through Customer Driven Narratives." In: *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. CHI EA '16. New York, NY, USA: Association for Computing Machinery, pp. 880–888. isbn: 978-1-4503-4082-3. doi: [10.1145/2851581.2851608](https://doi.org/10.1145/2851581.2851608).
- Du, Xuehong, Jianxin Jiao, and Mitchell M. Tseng (2001). "Architecture of Product Family: Fundamentals and Methodology:" in: *Concurrent Engineering*. Publisher: Sage Publications Sage CA: Thousand Oaks, CA. doi: [10.1177/1063293X0100900407](https://doi.org/10.1177/1063293X0100900407).
- Dyer, Nathan and Jeff Furr (Aug. 20, 2015). *Tesla's High End Disruption Gamble*. Forbes. url: <https://www.forbes.com/sites/innovatorsdna/2015/08/20/teslas-high-end-disruption-gamble/> (visited on 02/09/2020).
- Economies of Scope Definition* (n.d.). en. url: <https://www.investopedia.com/terms/e/economiesofscope.asp> (visited on 12/03/2021).
- El-Ansary, Adel I. (Jan. 1, 2006). "Marketing strategy: taxonomy and frameworks." In: *European Business Review* 18.4. Ed. by Göran Svensson. Publisher: Emerald Group Publishing Limited, pp. 266–293. issn: 0955-534X. doi: [10.1108/09555340610677499](https://doi.org/10.1108/09555340610677499).
- Eppinger, Steven and Karl Ulrich (2015). *Product design and development*. McGraw-Hill Higher Education.
- Eppinger, Steven D and Tyson R Browning (2012). *Design structure matrix methods and applications*. MIT press.
- Fagan, Mary Helen (2005). "Comparing Traditional And Agile Development Approaches: The Case Of Extreme Programming." In: *Issues in Information Systems, VI 2*, pp. 23–28.
- Farjoun, Moshe, Christopher Ansell, and Arjen Boin (2015). "PERSPECTIVE—Pragmatism in organization studies: Meeting the challenges of a dynamic and complex world." In: *Organization Science* 26.6, pp. 1787–1804.
- Fernandez, Daniel J and John D Fernandez (2008). "Agile project management—agilism versus traditional approaches." In: *Journal of Computer Information Systems* 49.2, pp. 10–17.
- Feuerstack, Sebastian, Andreas Lüdtkke, and Jan-Patrick Osterloh (2007). "A Tool for Easing the Cognitive Analysis of Design Prototypes of Aircraft Cockpit Instruments:

- The Human Efficiency Evaluator." In: *Proceedings of the European Conference on Cognitive Ergonomics 2015*. ECCE '15. event-place: Warsaw, Poland. New York, NY, USA: ACM, 22:1–22:8. isbn: 978-1-4503-3612-3. doi: [10.1145/2788412.2788434](https://doi.org/10.1145/2788412.2788434).
- Flemisch, Frank Ole et al. (Mar. 4, 2014). "Towards cooperative guidance and control of highly automated vehicles: H-Mode and Conduct-by-Wire." In: *Ergonomics* 57.3, pp. 343–360. issn: 0014-0139. doi: [10.1080/00140139.2013.869355](https://doi.org/10.1080/00140139.2013.869355).
- Franceschini, F. and S. Rossetto (Dec. 1, 1995). "QFD: The problem of comparing technical/engineering design requirements." In: *Research in Engineering Design* 7.4, pp. 270–278. issn: 1435-6066. doi: [10.1007/BF01580463](https://doi.org/10.1007/BF01580463).
- Gangemi, Aldo et al. (2006). "Modelling Ontology Evaluation and Validation." In: *The Semantic Web: Research and Applications*. Ed. by York Sure and John Domingue. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 140–154. isbn: 978-3-540-34545-9. doi: [10.1007/11762256_13](https://doi.org/10.1007/11762256_13).
- García-Holgado, Alicia and Francisco J. García-Peñalvo (Oct. 18, 2017). "Preliminary validation of the metamodel for developing learning ecosystems." In: *Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality*. TEEM 2017. Cádiz, Spain: Association for Computing Machinery, pp. 1–9. isbn: 978-1-4503-5386-1. doi: [10.1145/3144826.3145439](https://doi.org/10.1145/3144826.3145439).
- Garrett, Jesse James (Dec. 16, 2010). *The Elements of User Experience: User-Centered Design for the Web and Beyond*. 2nd edition. New Riders. 194 pp.
- Gerdri, N. (2005). "An analytical approach to building a technology development envelope (TDE) for roadmapping of emerging technologies." In: *A Unifying Discipline for Melting the Boundaries Technology Management*: pp. 123–135. doi: [10.1109/PICMET.2005.1509682](https://doi.org/10.1109/PICMET.2005.1509682).
- Gerdri, Nathasit et al. (Apr. 2019). "Conceptual framework to assess the impacts of changes on the status of a roadmap." In: *Journal of Engineering and Technology Management* 52, pp. 16–31. issn: 09234748. doi: [10.1016/j.jengtecman.2017.12.001](https://doi.org/10.1016/j.jengtecman.2017.12.001).
- Geum, Youngjung, Sora Lee, and Yongtae Park (May 2014). "Combining technology roadmap and system dynamics simulation to support scenario-planning: A case of car-sharing service." In: *Computers & Industrial Engineering* 71, pp. 37–49. issn: 03608352. doi: [10.1016/j.cie.2014.02.007](https://doi.org/10.1016/j.cie.2014.02.007).
- Geum, Youngjung et al. (2015a). "Development of data-driven technology roadmap considering dependency: An ARM-based technology roadmapping." In: *Technological Forecasting and Social Change* 91, pp. 264–279.

- Geum, Youngjung et al. (Feb. 1, 2015b). "Development of data-driven technology road-map considering dependency: An ARM-based technology roadmapping." In: *Technological Forecasting and Social Change* 91, pp. 264–279. issn: 0040-1625. doi: [10.1016/j.techfore.2014.03.003](https://doi.org/10.1016/j.techfore.2014.03.003).
- Glossary - Help | MPS (Sept. 3, 2021). MPS 2020.1. url: <https://www.jetbrains.com/help/mps/glossary.html> (visited on 01/30/2022).
- Goldkuhl, Göran (2012). "Pragmatism vs interpretivism in qualitative information systems research." In: *European journal of information systems* 21.2, pp. 135–146.
- Golkar, Alessandro and Nicola Garzaniti (July 20, 2020). "Model based systems engineering approach to technology roadmapping." In: *Proceedings of the 2020 Summer Simulation Conference*. SummerSim '20. San Diego, CA, USA: Society for Computer Simulation International, pp. 1–12. isbn: 978-1-71381-429-0.
- Gradini, Raffaele et al. (July 22, 2019). "Model-based technology roadmapping: potential and challenges ahead." In: *Proceedings of the 2019 Summer Simulation Conference*. SummerSim '19. Berlin, Germany: Society for Computer Simulation International, pp. 1–11. doi: [10.5555/3374138.3374160](https://doi.org/10.5555/3374138.3374160).
- Green, Paul A. (2017). "Motor Vehicle Driver Interfaces." In: *The Wiley Handbook of Human Computer Interaction*. John Wiley & Sons, Ltd, pp. 843–888. isbn: 978-1-118-97600-5. doi: [10.1002/9781118976005.ch40](https://doi.org/10.1002/9781118976005.ch40).
- Hassan, Salah S. and Stephen H. Craft (Mar. 2005). "Linking global market segmentation decisions with strategic positioning options." In: *Journal of Consumer Marketing* 22.2, pp. 81–89. issn: 0736-3761. doi: [10.1108/07363760510589244](https://doi.org/10.1108/07363760510589244).
- Hassenzahl, Marc (2018). "The Thing and I: Understanding the Relationship Between User and Product." In: *Funology 2: From Usability to Enjoyment*. Ed. by Mark Blythe and Andrew Monk. Human-Computer Interaction Series. Cham: Springer International Publishing, pp. 301–313. isbn: 978-3-319-68213-6. doi: [10.1007/978-3-319-68213-6_19](https://doi.org/10.1007/978-3-319-68213-6_19).
- Hassenzahl, Marc and John Carroll (Mar. 10, 2010). *Experience Design: Technology for All the Right Reasons*. San Rafael, Calif.: Morgan and Claypool Publishers. 100 pp. isbn: 978-1-60845-047-3.
- Hassenzahl, Marc and Noam Tractinsky (Mar. 1, 2006). "User experience - a research agenda." In: *Behaviour & Information Technology* 25.2. Publisher: Taylor & Francis, pp. 91–97. issn: 0144-929X. doi: [10.1080/01449290500330331](https://doi.org/10.1080/01449290500330331).

- Hehn, Jennifer et al. (Mar. 2020). "On Integrating Design Thinking for Human-Centered Requirements Engineering." In: *IEEE Software* 37.2. Conference Name: IEEE Software, pp. 25–31. issn: 1937-4194. doi: [10.1109/MS.2019.2957715](https://doi.org/10.1109/MS.2019.2957715).
- Hein, Andreas Makoto (2016). "Heritage Technologies in Space Programs - Assessment Methodology and Statistical Analysis." PhD thesis. Technische Universität München.
- Henderson, Kaitlin and Alejandro Salado (2021). "Value and benefits of model-based systems engineering (MBSE): Evidence from the literature." In: *Systems Engineering* 24.1, pp. 51–66.
- Hevner, Alan R (2007). "A three cycle view of design science research." In: *Scandinavian journal of information systems* 19.2, p. 4.
- Höfner, Peter, Ridha Khedri, and Bernhard Möller (2011). "An algebra of product families." In: *Software & Systems Modeling* 10.2, pp. 161–182.
- Howard, Tharon (2014). "Journey Mapping: A Brief Overview." In: *Commun. Des. Q. Rev* 2.3, 10–13. doi: [10.1145/2644448.2644451](https://doi.org/10.1145/2644448.2644451).
- INCOSE (2007). *Systems Engineering Vision 2020*. Technical Operations International Council on Systems Engineering (INCOSE).
- ISO 9241-210:2019 (July 2019). *Human-centred design for interactive systems*. 9241-210:2019. ISO/TC 159/SC 4 Ergonomics of human-system interaction, p. 33. url: <https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/07/75/77520.html> (visited on 05/12/2020).
- Iuskevich, I. et al. (May 2020). "A Discrete-event Simulation Model For Driver Performance Assessment: Application To Autonomous Vehicle Cockpit Design Optimization." In: *Proceedings of the Design Society: DESIGN Conference* 1. Publisher: Cambridge University Press, pp. 2521–2530. issn: 2633-7762. doi: [10.1017/dsd.2020.157](https://doi.org/10.1017/dsd.2020.157).
- Iuskevich, Ilia et al. (2021). "A Data-driven Approach To User-experience-focused Model-based Roadmapping for New Product Planning." In: *Proceedings of the Design Society* 1, pp. 61–70. doi: <https://doi.org/10.1017/pds.2021.7>.
- Jeon, Jeonghwan, Hakyoon Lee, and Yongtae Park (Sept. 1, 2011). "Implementing technology roadmapping with supplier selection for semiconductor manufacturing companies." In: *Technology Analysis & Strategic Management* 23.8. Publisher: Routledge, pp. 899–918. issn: 0953-7325. doi: [10.1080/09537325.2011.604156](https://doi.org/10.1080/09537325.2011.604156).
- Jeong, Yujin and Byungun Yoon (May 1, 2015). "Development of patent roadmap based on technology roadmap by analyzing patterns of patent development." In: *Techno-*

- vation. Opportunity Recognition and Creation 39-40, pp. 37–52. issn: 0166-4972. doi: [10.1016/j.technovation.2014.03.001](https://doi.org/10.1016/j.technovation.2014.03.001).
- Jiao, Jianxin Roger, Yiyang Zhang, and Yi Wang (2007). "A generic genetic algorithm for product family design." In: *Journal of Intelligent Manufacturing* 18.2, pp. 233–247.
- Jun, Seung-Pyo, Do-Hyung Park, and Jaeho Yeom (July 2014). "The possibility of using search traffic information to explore consumer product attitudes and forecast consumer preference." en. In: *Technological Forecasting and Social Change* 86, pp. 237–253. issn: 00401625. doi: [10.1016/j.techfore.2013.10.021](https://doi.org/10.1016/j.techfore.2013.10.021).
- Junior, Moacyr Cortelette and J. P. Almeida (2018). "Toward an Ontological Analysis of Archetypal Entities in the Marketing Domain: Personas and Related Concepts." In: *ONTOBRAS*.
- Kajikawa, Yuya, Yoshiyuki Takeda, and Katsumori Matsushima (Apr. 13, 2010). "Computer-assisted roadmapping: a case study in energy research." In: *Foresight* 12.2, pp. 4–15. issn: 1463-6689. doi: [10.1108/14636681011035726](https://doi.org/10.1108/14636681011035726).
- Kano, Noriaki (1984). "Attractive quality and must-be quality." In: *Hinshitsu (Quality, The Journal of Japanese Society for Quality Control)* 14, pp. 39–48.
- Karapanos, Evangelos et al. (2009). "User experience over time: an initial framework." In: *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 729–738.
- Kerr, Clive and Robert Phaal (July 1, 2015). "Visualizing Roadmaps: A Design-Driven Approach." In: *Research-Technology Management* 58.4. Publisher: Routledge, pp. 45–54. issn: 0895-6308. doi: [10.5437/08956308X5804253](https://doi.org/10.5437/08956308X5804253).
- (June 1, 2020). "Technology roadmapping: Industrial roots, forgotten history and unknown origins." In: *Technological Forecasting and Social Change* 155, p. 119967. issn: 0040-1625. doi: [10.1016/j.techfore.2020.119967](https://doi.org/10.1016/j.techfore.2020.119967).
- Keshavarz Ghorabae, Mehdi et al. (Feb. 5, 2017). "Fuzzy extension of the CODAS method for multi-criteria market segment evaluation." In: *Journal of Business Economics and Management* 18.1, pp. 1–19. issn: 1611-1699, 2029-4433. doi: [10.3846/16111699.2016.1278559](https://doi.org/10.3846/16111699.2016.1278559).
- Kett, S. G. and S. Wartzack (2016). "Considering emotional impressions in product design: Quality of life theory and its impact on design strategy." en. In: *DS 84: Proceedings of the DESIGN 2016 14th International Design Conference*, pp. 1719–1728.
- Kett, Susan Gretchen, Benedikt Schmitt, and Sandro Wartzack (2017). "What the Statistics Tell Us—How to Use Empiric Data in Design for Emotional Impressions." In: *Research into Design for Communities, Volume 2*. Ed. by Amaresh Chakrabarti and

- Debkumar Chakrabarti. *Smart Innovation, Systems and Technologies*. Singapore: Springer, pp. 659–669. isbn: 978-981-10-3521-0. doi: [10.1007/978-981-10-3521-0_56](https://doi.org/10.1007/978-981-10-3521-0_56).
- Kilkki, Kalevi et al. (Apr. 2018). "A disruption framework." In: *Technological Forecasting and Social Change* 129, pp. 275–284. issn: 00401625. doi: [10.1016/j.techfore.2017.09.034](https://doi.org/10.1016/j.techfore.2017.09.034).
- Kim, Euiyoung, Sara L. Beckman, and Alice Agogino (Sept. 4, 2018). "Design Roadmapping in an Uncertain World: Implementing a Customer-Experience-Focused Strategy." in: *California Management Review*. Publisher: SAGE Publications Sage CA: Los Angeles, CA. doi: [10.1177/0008125618796489](https://doi.org/10.1177/0008125618796489).
- Kim, Euiyoung et al. (Oct. 1, 2016). "Design Roadmapping: A Framework and Case Study on Planning Development of High-Tech Products in Silicon Valley." In: *Journal of Mechanical Design* 138.10. Publisher: American Society of Mechanical Engineers Digital Collection. issn: 1050-0472. doi: [10.1115/1.4034221](https://doi.org/10.1115/1.4034221).
- Kim, Hyeji et al. (Nov. 3, 2017). "Scenario-Based Conjoint Analysis: Measuring Preferences for User Experiences in Early Stage Design." In: ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers Digital Collection. doi: [10.1115/DETC2017-67690](https://doi.org/10.1115/DETC2017-67690).
- Knoll, Dominik, Alessandro Golkar, and Olivier de Weck (Apr. 2018). "A concurrent design approach for model-based technology roadmapping." In: *2018 Annual IEEE International Systems Conference (SysCon)*. 2018 Annual IEEE International Systems Conference (SysCon). ISSN: 2472-9647, pp. 1–6. doi: [10.1109/SYSCON.2018.8369527](https://doi.org/10.1109/SYSCON.2018.8369527).
- Kontogiannis, Tom (Nov. 1, 2005). "Integration of task networks and cognitive user models using coloured Petri nets and its application to job design for safety and productivity." In: *Cognition, Technology & Work* 7.4, pp. 241–261. issn: 1435-5566. doi: [10.1007/s10111-005-0010-z](https://doi.org/10.1007/s10111-005-0010-z).
- Korosec, Kirsten (Oct. 10, 2019). *Waymo to customers: 'Completely driverless Waymo cars are on the way'*. TechCrunch. url: <http://social.techcrunch.com/2019/10/09/waymo-to-customers-completely-driverless-waymo-cars-are-on-the-way/> (visited on 11/15/2019).
- Kostoff, R.N. and R.R. Schaller (May 2001). "Science and technology roadmaps." In: *IEEE Transactions on Engineering Management* 48.2, pp. 132–143. issn: 1558-0040. doi: [10.1109/17.922473](https://doi.org/10.1109/17.922473).

- Kujala, Sari et al. (2011). "UX Curve: A method for evaluating long-term user experience." In: *Interacting with computers* 23.5, pp. 473–483.
- Kumar, Vinod, Uma Kumar, and Aditha Persaud (Apr. 1, 1999). "Building Technological Capability Through Importing Technology: The Case of Indonesian Manufacturing Industry." In: *The Journal of Technology Transfer* 24.1, pp. 81–96. issn: 1573-7047. doi: [10.1023/A:1007728921126](https://doi.org/10.1023/A:1007728921126).
- Lallemand, Carine, Guillaume Gronier, and Vincent Koenig (Feb. 1, 2015). "User experience: A concept without consensus? Exploring practitioners' perspectives through an international survey." In: *Computers in Human Behavior* 43, pp. 35–48. issn: 0747-5632. doi: [10.1016/j.chb.2014.10.048](https://doi.org/10.1016/j.chb.2014.10.048).
- Laughery, Ron et al. (July 2000). "Task Network Modeling and the Micro Saint Family of Tools." In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 44.6, pp. 721–724. issn: 1541-9312. doi: [10.1177/154193120004400649](https://doi.org/10.1177/154193120004400649).
- Law, Effie Lai-Chong et al. (2009). "Understanding, scoping and defining user experience: a survey approach." In: *Proceedings of the 27th international conference on Human factors in computing systems - CHI 09*. the SIGCHI Conference. Boston, MA, USA: ACM Press, p. 719. isbn: 978-1-60558-246-7. doi: [10.1145/1518701.1518813](https://doi.org/10.1145/1518701.1518813).
- Lee, Hakyeon and Youngjung Geum (Apr. 1, 2017). "Development of the scenario-based technology roadmap considering layer heterogeneity: An approach using CIA and AHP." In: *Technological Forecasting and Social Change* 117, pp. 12–24. issn: 0040-1625. doi: [10.1016/j.techfore.2017.01.016](https://doi.org/10.1016/j.techfore.2017.01.016).
- Lee, Jung Hoon, Robert Phaal, and Sang-Ho Lee (Feb. 1, 2013). "An integrated service-device-technology roadmap for smart city development." In: *Technological Forecasting and Social Change*. World problems, Emerging technologies and Creative Enterprise 80.2, pp. 286–306. issn: 0040-1625. doi: [10.1016/j.techfore.2012.09.020](https://doi.org/10.1016/j.techfore.2012.09.020).
- Lee, Sungjoo and Yongtae Park (June 1, 2005). "Customization of technology roadmaps according to roadmapping purposes: Overall process and detailed modules." In: *Technological Forecasting and Social Change* 72.5, pp. 567–583. issn: 0040-1625. doi: [10.1016/j.techfore.2004.11.006](https://doi.org/10.1016/j.techfore.2004.11.006).
- Lu, Hsi-Peng, Chiao-Shan Chen, and Hueiju Yu (2019). "Technology roadmap for building a smart city: An exploring study on methodology." In: *Future Generation Computer Systems* 97, pp. 727–742.
- López-Fernández, Jesús J., Esther Guerra, and Juan de Lara (Sept. 15, 2014). "Meta-Model validation and verification with MetaBest." In: *Proceedings of the 29th ACM/IEEE*

- international conference on Automated software engineering*. ASE '14. Vasteras, Sweden: Association for Computing Machinery, pp. 831–834. isbn: 978-1-4503-3013-8. doi: [10.1145/2642937.2648617](https://doi.org/10.1145/2642937.2648617).
- López-Fernández, Jesús J. et al. (Oct. 1, 2015). "Example-driven meta-model development." In: *Software & Systems Modeling* 14.4, pp. 1323–1347. issn: 1619-1374. doi: [10.1007/s10270-013-0392-y](https://doi.org/10.1007/s10270-013-0392-y).
- Mankins, John C. (Nov. 2009). "Technology readiness assessments: A retrospective." In: *Acta Astronautica* 65.9, pp. 1216–1223. issn: 00945765. doi: [10.1016/j.actaastro.2009.03.058](https://doi.org/10.1016/j.actaastro.2009.03.058).
- Marti, Patrizia and Iolanda Iacono (Sept. 2016). "Anticipated, momentary, episodic, remembered: the many facets of User eXperience." In: *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)*. 2016 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 1647–1655.
- McMillan, Alec (Mar. 1, 2003). "Roadmapping—Agent of Change." In: *Research-Technology Management* 46.2. Publisher: Routledge, pp. 40–47. issn: 0895-6308. doi: [10.1080/08956308.2003.11671553](https://doi.org/10.1080/08956308.2003.11671553).
- Meinel, Martin et al. (Feb. 10, 2020). "Does applying design thinking result in better new product concepts than a traditional innovation approach? An experimental comparison study." In: *European Management Journal*. issn: 0263-2373. doi: [10.1016/j.emj.2020.02.002](https://doi.org/10.1016/j.emj.2020.02.002).
- Mellor, Stephen J., ed. (2004). *MDA distilled: principles of model-driven architecture*. Addison-Wesley object technology series. Boston: Addison-Wesley. 150 pp. isbn: 978-0-201-78891-4.
- Miao, Hong et al. (2020). "Integrating Technology-Relationship-Technology Semantic Analysis and Technology Roadmapping Method: A Case of Elderly Smart Wear Technology." In: *IEEE Transactions on Engineering Management*. Conference Name: IEEE Transactions on Engineering Management, pp. 1–17. issn: 1558-0040. doi: [10.1109/TEM.2020.2970972](https://doi.org/10.1109/TEM.2020.2970972).
- Morville, Peter (2005). "Experience design unplugged." In: *ACM SIGGRAPH 2005 Web program on - SIGGRAPH '05*. ACM SIGGRAPH 2005 Web program. Los Angeles, California: ACM Press, p. 10. doi: [10.1145/1187335.1187347](https://doi.org/10.1145/1187335.1187347).
- Moullec, Marie-Lise et al. (Apr. 1, 2013). "Toward System Architecture Generation and Performances Assessment Under Uncertainty Using Bayesian Networks." In: *Journal of Mechanical Design* 135.4, p. 041002. issn: 1050-0472, 1528-9001. doi: [10.1115/1.4023514](https://doi.org/10.1115/1.4023514).

- Muller, Johannes and Max Lillack (Aug. 2011). "Conjoint Analysis of Software Product Lines: A Feature Based Approach." In: *2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications*. 2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications. ISSN: 2376-9505, pp. 374–377. doi: [10.1109/SEAA.2011.73](https://doi.org/10.1109/SEAA.2011.73).
- Nair, Anand and William R. Boulton (Jan. 1, 2008). "Innovation-oriented operations strategy typology and stage-based model." In: *International Journal of Operations & Production Management* 28.8. Publisher: Emerald Group Publishing Limited, pp. 748–771. issn: 0144-3577. doi: [10.1108/01443570810888599](https://doi.org/10.1108/01443570810888599).
- Nazarenko, Anastasia et al. (July 31, 2021). "Applying digital technologies in technology roadmapping to overcome individual biased assessments." In: *Technovation*, p. 102364. issn: 0166-4972. doi: [10.1016/j.technovation.2021.102364](https://doi.org/10.1016/j.technovation.2021.102364).
- Nešić, Damir et al. (Aug. 12, 2019). "Principles of feature modeling." In: *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. ESEC/FSE 2019. New York, NY, USA: Association for Computing Machinery, pp. 62–73. isbn: 978-1-4503-5572-8. doi: [10.1145/3338906.3338974](https://doi.org/10.1145/3338906.3338974).
- Nickel, Julia, Adam Ross, and Donna Rhodes (June 1, 2009). "Comparison of Project Evaluation Using Cost-Benefit Analysis and Multi-Attribute Tradespace Exploration in the Transportation Domain." In:
- Noh, Heeyong et al. (Feb. 1, 2021). "Opportunity-driven technology roadmapping: The case of 5G mobile services." In: *Technological Forecasting and Social Change* 163, p. 120452. issn: 0040-1625. doi: [10.1016/j.techfore.2020.120452](https://doi.org/10.1016/j.techfore.2020.120452).
- North, Robert A. and Victor A. Riley (1989). "W/INDEX: A Predictive Model of Operator Workload." In: *Applications of Human Performance Models to System Design*. Ed. by Grant R. McMillan et al. Boston, MA: Springer US, pp. 81–89. isbn: 978-1-4757-9244-7. doi: [10.1007/978-1-4757-9244-7_6](https://doi.org/10.1007/978-1-4757-9244-7_6).
- Ober, Ileana and Andreas Prinz (2006). "What do we need metamodels for?" In: 4th Nordic Workshop on UML and Software Modelling, pp. 8–28.
- OMG (2019). *System Modeling Language*.
- Parasuraman, Raja and Christopher D. Wickens (2008). "Humans: Still vital after all these years of automation." In: *Human factors* 50.3, pp. 511–520.
- Park, Hyunkyu et al. (May 1, 2020). "Twenty years of technology and strategic roadmapping research: A school of thought perspective." In: *Technological Forecasting*

- and Social Change* 154, p. 119965. issn: 0040-1625. doi: [10.1016/j.techfore.2020.119965](https://doi.org/10.1016/j.techfore.2020.119965).
- Passos, Leonardo et al. (2011). "A study of non-boolean constraints in variability models of an embedded operating system." In: *Proceedings of the 15th International Software Product Line Conference, Volume 2*, pp. 1–8.
- Pettitt, Michael, Gary E. Burnett, and Alan Stevens (2007). "An extended keystroke level model (KLM) for predicting the visual demand of in-vehicle information systems." In: *CHI*. doi: [10.1145/1240624.1240852](https://doi.org/10.1145/1240624.1240852).
- Phaal, R., C. J. P. Farrukh, and D. R. Probert (July 2005). "Developing a technology roadmapping system." In: *A Unifying Discipline for Melting the Boundaries Technology Management: ISSN: 2159-5100*, pp. 99–111. doi: [10.1109/PICMET.2005.1509680](https://doi.org/10.1109/PICMET.2005.1509680).
- Phaal, Robert, Clare J. P. Farrukh, and David R. Probert (Jan. 1, 2004). "Technology roadmapping—A planning framework for evolution and revolution." In: *Technological Forecasting and Social Change*. Roadmapping: From Sustainable to Disruptive Technologies 71.1, pp. 5–26. issn: 0040-1625. doi: [10.1016/S0040-1625\(03\)00072-6](https://doi.org/10.1016/S0040-1625(03)00072-6).
- Phaal, Robert, Clare J.P. Farrukh, and David R. Probert (Jan. 1, 2009). "Visualising strategy: a classification of graphical roadmap forms." In: *International Journal of Technology Management* 47.4, pp. 286–305. issn: 0267-5730. doi: [10.1504/IJTM.2009.024431](https://doi.org/10.1504/IJTM.2009.024431).
- Phaal, Robert and Gerrit Muller (Jan. 1, 2009). "An architectural framework for roadmapping: Towards visual strategy." In: *Technological Forecasting and Social Change*. Knowledge Driven Planning Tools for Emerging and Converging Technologies 76.1, pp. 39–49. issn: 0040-1625. doi: [10.1016/j.techfore.2008.03.018](https://doi.org/10.1016/j.techfore.2008.03.018).
- Phaal, Robert, Lianne Simonse, and Elke Den Ouden (2008). "Next generation roadmapping for innovation planning." In: *International Journal of Technology Intelligence and Planning* 4.2, p. 135. issn: 1740-2832, 1740-2840. doi: [10.1504/IJTIP.2008.018313](https://doi.org/10.1504/IJTIP.2008.018313).
- Phaal, Robert et al. (Mar. 1, 2003). "Starting-Up Roadmapping Fast." In: *Research-Technology Management* 46.2. Publisher: Routledge, pp. 52–59. issn: 0895-6308. doi: [10.1080/08956308.2003.11671555](https://doi.org/10.1080/08956308.2003.11671555).
- Porter, Michael E. (1989). "How Competitive Forces Shape Strategy." In: *Readings in Strategic Management*. Ed. by David Asch and Cliff Bowman. London: Macmillan Ed-

- ucation UK, pp. 133–143. isbn: 978-1-349-20317-8. doi: [10.1007/978-1-349-20317-8_10](https://doi.org/10.1007/978-1-349-20317-8_10).
- Post, James E., Lee E. Preston, and Sybille Sachs (Oct. 2002). "Managing the Extended Enterprise: The New Stakeholder View." In: *California Management Review* 45.1, pp. 6–28. issn: 0008-1256, 2162-8564. doi: [10.2307/41166151](https://doi.org/10.2307/41166151).
- Prahalad, Coimbatore K and Gary Hamel (1997). "The core competence of the corporation." In: *Strategische Unternehmensplanung/Strategische Unternehmensführung*. Springer, pp. 969–987.
- Probst, Gilbert JB (1998). "Practical knowledge management: A model that works." In: *PRISM-CAMBRIDGE MASSACHUSETTS-*, pp. 17–30.
- Pucillo, Francesco and Gaetano Cascini (Mar. 1, 2014). "A framework for user experience, needs and affordances." In: *Design Studies* 35.2, pp. 160–179. issn: 0142-694X. doi: [10.1016/j.destud.2013.10.001](https://doi.org/10.1016/j.destud.2013.10.001).
- Quan, Huafeng et al. (2019). "Personalized product evaluation based on GRA-TOPSIS and Kansei engineering." In: *Symmetry* 11.7, p. 867.
- Ries, Eric (Oct. 6, 2011). *The Lean Startup: How Constant Innovation Creates Radically Successful Businesses*. Trade Paperback edition. London: Portfolio Penguin. 336 pp. isbn: 978-0-670-92160-7.
- Ripon, Shamim, Sk Jahir Hossain, and Touhid Bhuiyan (Sept. 30, 2013). "Managing and Analysing Software Product Line Requirements." In: *International Journal of Software Engineering & Applications* 4.5, pp. 63–75. issn: 09762221, 09759018. doi: [10.5121/ijsea.2013.4505](https://doi.org/10.5121/ijsea.2013.4505). arXiv: [1310.0152](https://arxiv.org/abs/1310.0152).
- Robert, Jean-Marc and Annemarie Lesage (2017). "Designing and Evaluating User Experience." In: *The Handbook of Human-Machine Interaction*. Ed. by Guy A Boy. isbn: 978-1-138-07582-5.
- Ross, Adam M. and Daniel E. Hastings (2005). "11.4.3 The Tradespace Exploration Paradigm." In: *INCOSE International Symposium* 15.1. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/978111815837.2005.tb00783.x>, pp. 1706–1718. issn: 2334-5837. doi: [10.1002/j.2334-5837.2005.tb00783.x](https://doi.org/10.1002/j.2334-5837.2005.tb00783.x).
- Routley, Michèle, Robert Phaal, and David Probert (July 1, 2013). "Exploring industry dynamics and interactions." In: *Technological Forecasting and Social Change* 80.6, pp. 1147–1161. issn: 0040-1625. doi: [10.1016/j.techfore.2012.04.015](https://doi.org/10.1016/j.techfore.2012.04.015).
- Saller, Karsten et al. (2012). "Reducing feature models to improve runtime adaptivity on resource limited devices." In: *Proceedings of the 16th International Software Product Line Conference-Volume 2*, pp. 135–142.

- Saritas, Ozcan and Jonathan Aylen (Sept. 2010). "Using scenarios for roadmapping: The case of clean production." In: *Technological Forecasting and Social Change* 77.7, pp. 1061–1075. issn: 00401625. doi: [10.1016/j.techfore.2010.03.003](https://doi.org/10.1016/j.techfore.2010.03.003).
- Scalice, Régis Kovacs et al. (Mar. 1, 2015). "Modular deployment using TRM and function analysis." In: *Technological Forecasting and Social Change* 92, pp. 1–11. issn: 0040-1625. doi: [10.1016/j.techfore.2014.10.018](https://doi.org/10.1016/j.techfore.2014.10.018).
- Scerbo, Mark (2008). "Adaptive automation." In: *Neuroergonomics: The brain at work*. Ed. by Raja Parasuraman and Matthew Rizzo. Vol. 3. Oxford University Press, p. 239.
- Schuiling, Isabelle and Jean-Noël Kapferer (Dec. 2004). "Executive Insights: Real Differences between Local and International Brands: Strategic Implications for International Marketers." In: *Journal of International Marketing* 12.4, pp. 97–112. issn: 1069-031X, 1547-7215. doi: [10.1509/jimk.12.4.97.53217](https://doi.org/10.1509/jimk.12.4.97.53217).
- Selva, Daniel, Bruce Cameron, and Ed Crawley (2016). "Patterns in System Architecture Decisions." In: *Systems Engineering* 19.6. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/sys.21370> pp. 477–497. issn: 1520-6858. doi: [10.1002/sys.21370](https://doi.org/10.1002/sys.21370).
- Shen, X.X., K.C. Tan, and M. Xie (June 2000). "An integrated approach to innovative product development using Kano's model and QFD." In: *European Journal of Innovation Management* 3.2, pp. 91–99. issn: 1460-1060. doi: [10.1108/14601060010298435](https://doi.org/10.1108/14601060010298435).
- Silveira Junior, Luiz A. Bloem da et al. (Jan. 1, 2018). "Technology roadmapping: A methodological proposition to refine Delphi results." In: *Technological Forecasting and Social Change* 126, pp. 194–206. issn: 0040-1625. doi: [10.1016/j.techfore.2017.08.011](https://doi.org/10.1016/j.techfore.2017.08.011).
- Smaling, Rudolf and Olivier de Weck (2007). "Assessing risks and opportunities of technology infusion in system design." In: *Systems Engineering* 10.1, pp. 1–25. issn: 1520-6858. doi: [10.1002/sys.20061](https://doi.org/10.1002/sys.20061).
- Son, Heungwook et al. (Mar. 4, 2018). "Using a design structure matrix to support technology roadmapping for product–service systems." In: *Technology Analysis & Strategic Management* 30.3, pp. 337–350. issn: 0953-7325, 1465-3990. doi: [10.1080/09537325.2017.1310377](https://doi.org/10.1080/09537325.2017.1310377).
- Steiner, George Albert (1972). *Pitfalls in comprehensive long range planning*. Planning Executives Institute.
- Terrasse, Marie-Noëlle et al. (2006). "Do we need metamodels AND ontologies for engineering platforms?" In: *Proceedings of the 2006 international workshop on Global integrated model management - GaMMa '06*. the 2006 international workshop. Shang-

- hai, China: ACM Press, p. 21. isbn: 978-1-59593-410-9. doi: [10 . 1145 / 1138304 . 1138310](https://doi.org/10.1145/1138304.1138310).
- Toro-Jarrín, Miguel Angel, Idalia Estefania Ponce-Jaramillo, and David Güemes-Castorena (Sept. 1, 2016). "Methodology for the Building Process Integration of Business Model Canvas and Technological Roadmap." In: *Technological Forecasting and Social Change* 110, pp. 213–225. issn: 0040-1625. doi: [10 . 1016 / j . techfore . 2016 . 01 . 009](https://doi.org/10.1016/j.techfore.2016.01.009).
- Tseng, Mitchell M., Jianxin Jiao, and M. Eugene Merchant (1996). "Design for Mass Customization." In: *CIRP Annals* 45.1, pp. 153–156. issn: 0007-8506. doi: [https://doi.org/10.1016/S0007-8506\(07\)63036-4](https://doi.org/10.1016/S0007-8506(07)63036-4).
- Tucker, Conrad S. and Harrison M. Kim (Nov. 2011). "Trend Mining for Predictive Product Design." en. In: *Journal of Mechanical Design* 133.11, p. 111008. issn: 1050-0472, 1528-9001. doi: [10 . 1115 / 1 . 4004987](https://doi.org/10.1115/1.4004987).
- Vermaas, Pieter E (2014). "Design theories, models and their testing: on the scientific status of design research." In: *An anthology of theories and models of design*. Springer, pp. 47–66.
- Vinayavekhin, Sukrit et al. (Sept. 20, 2021). "Emerging trends in roadmapping research: A bibliometric literature review." In: *Technology Analysis & Strategic Management*, pp. 1–15. issn: 0953-7325, 1465-3990. doi: [10 . 1080 / 09537325 . 2021 . 1979210](https://doi.org/10.1080/09537325.2021.1979210).
- Vrandečić, Denny (2009). "Ontology Evaluation." In: *Handbook on Ontologies*. Ed. by Steffen Staab and Rudi Studer. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 293–313. isbn: 978-3-540-70999-2 978-3-540-92673-3. doi: [10 . 1007 / 978 - 3 - 540 - 92673 - 3 _ 13](https://doi.org/10.1007/978-3-540-92673-3_13).
- Wahab, Sazali Abdul, Raduan Che Rose, and Suzana Idayu Wati Osman (Dec. 25, 2011). "Defining the Concepts of Technology and Technology Transfer: A Literature Analysis." In: *International Business Research* 5.1, p61. issn: 1913-9012, 1913-9004. doi: [10 . 5539 / ibr . v5n1p61](https://doi.org/10.5539/ibr.v5n1p61).
- Walden, David D. et al., eds. (2015). *Systems engineering handbook: a guide for system life cycle processes and activities*. 4th edition. Hoboken, New Jersey: Wiley. 1 p. isbn: 978-1-118-99941-7 978-1-119-01512-3.
- Wang, Jing-Shiarn, Ronald R. Knipling, and Michael J. Goodman (1996). "The role of driver inattention in crashes: New statistics from the 1995 Crashworthiness Data System." In: *40th annual proceedings of the Association for the Advancement of Automotive Medicine*. Vol. 377, p. 392.
- Wang, Renzhong and C.H. Dagli (Mar. 19, 2013). *Developing a Holistic Modeling Approach for Search-based System Architecting*. Vol. 16. Journal Abbreviation: Procedia Com-

- puter Science Publication Title: Procedia Computer Science. doi: [10.1016/j.procs.2013.01.022](https://doi.org/10.1016/j.procs.2013.01.022).
- What Are Economies of Scale?* (N.d.). en. url: <https://www.investopedia.com/terms/e/economiesofscale.asp> (visited on 12/03/2021).
- Wickens, Christopher D. (Jan. 2002). "Multiple resources and performance prediction." In: *Theoretical Issues in Ergonomics Science* 3.2, pp. 159–177. issn: 1463-922X, 1464-536X. doi: [10.1080/14639220210123806](https://doi.org/10.1080/14639220210123806).
- Wittmann, Marc et al. (Mar. 1, 2006). "Effects of display position of a visual in-vehicle task on simulated driving." In: *Applied Ergonomics* 37.2, pp. 187–199. issn: 0003-6870. doi: [10.1016/j.apergo.2005.06.002](https://doi.org/10.1016/j.apergo.2005.06.002).
- Wortelen, Bertram et al. (Oct. 2016). "Towards the integration and evaluation of on-line workload measures in a cognitive architecture." In: *2016 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, pp. 11–16. doi: [10.1109/CogInfoCom.2016.7804516](https://doi.org/10.1109/CogInfoCom.2016.7804516).
- Wysocki, Robert K. (2006). *Effective Software Project Management*. USA: John Wiley & Sons, Inc. isbn: 0764596365.
- Xu, Qianli et al. (Jan. 1, 2009). "An analytical Kano model for customer need analysis." In: *Design Studies* 30.1, pp. 87–110. issn: 0142-694X. doi: [10.1016/j.destud.2008.07.001](https://doi.org/10.1016/j.destud.2008.07.001).
- Yin, Robert K (2009). *Case study research: Design and methods*. Vol. 5. sage.
- Yoon, Byungun, Rob Phaal, and David Probert (Dec. 2007). "Morphology analysis for technology roadmapping: application of text mining: Morphology analysis for technology roadmapping." en. In: *R&D Management* 38.1, pp. 51–68. issn: 00336807. doi: [10.1111/j.1467-9310.2007.00493.x](https://doi.org/10.1111/j.1467-9310.2007.00493.x).
- Yuskevich, Ilya et al. (Dec. 2021a). "A metamodel of an informational structure for model-based technology roadmapping." In: *Technological Forecasting and Social Change* 173, p. 121103. issn: 00401625. doi: [10.1016/j.techfore.2021.121103](https://doi.org/10.1016/j.techfore.2021.121103).
- Yuskevich, Ilya et al. (July 2021b). "Model-based approaches for technology planning and roadmapping: Technology forecasting and game-theoretic modeling." In: *Technological Forecasting and Social Change* 168, p. 120761. issn: 00401625. doi: [10.1016/j.techfore.2021.120761](https://doi.org/10.1016/j.techfore.2021.120761).
- Yuskevich, Ilya et al. (2022). "Model-based User Experience Design and Roadmapping." In: unpublished.

- Zhang, Yi et al. (Sept. 1, 2016). "Technology roadmapping for competitive technical intelligence." In: *Technological Forecasting and Social Change* 110, pp. 175–186. issn: 0040-1625. doi: [10.1016/j.techfore.2015.11.029](https://doi.org/10.1016/j.techfore.2015.11.029).
- Zhou, Feng et al. (Dec. 15, 2017). "Augmenting feature model through customer preference mining by hybrid sentiment analysis." In: *Expert Systems with Applications* 89, pp. 306–317. issn: 0957-4174. doi: [10.1016/j.eswa.2017.07.021](https://doi.org/10.1016/j.eswa.2017.07.021).
- Zimmermann, H. (Apr. 1980). "OSI Reference Model - The ISO Model of Architecture for Open Systems Interconnection." In: *IEEE Transactions on Communications* 28.4. Conference Name: IEEE Transactions on Communications, pp. 425–432. issn: 1558-0857. doi: [10.1109/TCOM.1980.1094702](https://doi.org/10.1109/TCOM.1980.1094702).

Appendices

A SYNTHÈSE

Lorsque quelqu'un conçoit un produit, il est confronté à un problème appartenant à deux domaines complètement différents. La plus évidente est la conception d'un produit lui-même, qui comprend la création de la forme d'un produit (conception visuelle, sélection des matériaux, etc.), la conception de l'architecture et des fonctions du produit (ingénierie logicielle et matérielle), et la conception de un processus de production (choix des technologies et des composants, planification de la fabrication). Le deuxième domaine ne concerne pas un artefact mais les personnes qui utiliseront cet artefact. L'attention des spécialistes du marketing, de la conception de l'expérience utilisateur, du développement client, des chefs de programme et de produit (désormais appelée équipe produit) est dirigée non pas vers les produits de leur entreprise, mais vers l'extérieur, vers l'environnement externe, c'est-à-dire vers les clients et tendances, utilisateurs actuels et potentiels, partenaires et concurrents.

Autrefois négligée (le rôle d'un chef de produit remonte aux années 1930, la première mention de la conception centrée sur l'utilisateur — aux années 1970), l'importance de l'équipe produit est aujourd'hui très appréciée. La majorité des produits que nous utilisons actuellement réussissent en raison de la vision supérieure du produit, du positionnement approprié et des innovations de produit. Les entreprises basées uniquement sur les innovations technologiques risquent d'être remplacées par des alternatives plus utilisables et mieux adaptées ou de devenir un fournisseur de composants.

Cette thèse est consacrée au problème de la planification stratégique du développement de nouveaux produits (DNP). Un produit est quelque chose vendu par une entreprise à ses clients (Eppinger and Ulrich 2015). Pour être vendu, un produit doit offrir certains avantages et/ou une expérience agréable aux utilisateurs finaux. Le DNP est "l'ensemble des activités commençant par la perception d'une opportunité de marché et se terminant par la production, la vente et la livraison d'un produit" (Eppinger and Ulrich 2015). Dans cette définition, nous soulignons que le processus DNP se concentre non seulement sur l'ingénierie des produits, mais également sur l'exploration des utilisateurs et de leurs besoins. Enfin, la planification stratégique est "un processus systématique d'identification des opportunités et des menaces dans l'environnement futur et de formulation de politiques, basées sur les ressources organisationnelles et les objectifs des opérations dans cet environnement sur un relativement long terme"

(Steiner 1972).

L'approche la plus populaire pour une planification stratégique dans les entreprises est le roadmapping technologique (TRM). Une roadmap technologique est une "vision du futur dans un domaine précis basée sur l'opinion collective de variables importantes" (Lu, Chen, and Yu 2019). Le format typique d'une roadmap technologique est un "graphique à plusieurs couches comprenant des couches de marché, de produit et de technologie" (Geum et al. 2015a).

En conséquence, TRM est le processus de création de la roadmap technologique. Le TRM est un processus interdisciplinaire intensif en information. Ces propriétés justifient le besoin de méthodologies et d'outils numériques supportant la TRM. C'est le premier aspect du problème global que nous abordons dans cette thèse.

Le deuxième aspect concerne l'expérience utilisateur (EU). Le concept EU comprend "tous les aspects de la façon dont les gens utilisent un produit interactif : la façon dont il se sent dans leurs mains, à quel point ils comprennent comment cela fonctionne, ce qu'ils en pensent pendant qu'ils l'utilisent, à quel point il sert leurs objectifs, et à quel point il s'intègre dans l'ensemble du contexte dans lequel ils l'utilisent" (Alben 1996). EU est un concept de base pour la planification dans les entreprises axées sur la valeur de l'utilisateur final (Kim, Beckman, and Agogino 2018). Par rapport à la planification des systèmes et des technologies, la planification EU est beaucoup plus difficile. Comme tout concept lié à l'humain, l'EU est indéterministe, subjectif, quantitatif et ambigu.

Par conséquent, le problème primordial que nous abordons est de savoir comment gérer et réduire l'ambiguïté et l'incertitude dans la planification stratégique d'un DNP complexe centré sur l'utilisateur.

Traditionnellement, les processus et outils DNP sont analysés dans le domaine de la science de la conception, et la planification du DNP — dans le domaine de la gestion de la technologie. Dans cette recherche, nous essayons de considérer ce problème sous une nouvelle perspective, c'est-à-dire la perspective de l'ingénierie des systèmes basée sur les modèles (ISBM). L'ISBM implique une description du domaine d'intérêt à l'aide d'un langage spécifique au domaine (DSL) standard ou ad hoc.

Représenter l'EU dans un DSL ne rendra pas les problèmes vagues et subjectifs plus déterminés et objectifs. Cependant, les systèmes d'information modernes peuvent stocker et traiter efficacement de grandes quantités de données et les représenter

visuellement et de manière interactive. Nous supposons que cela simplifiera la prise de décision et favorisera l'efficacité de la communication.

L'objectif de la thèse est de développer notre propre DSL pour le TRM avec les propriétés souhaitées. L'objectif détermine notre méthodologie de recherche. Pour concevoir le DSL pour le problème cible de la planification NPD, il a d'abord été proposé un métamodèle qui décrit tout DSL pour la roadmap au niveau abstrait. Pour cela, il a été soigneusement passé en revue la littérature roadmap afin de définir des éléments communs : entités, liens, attributs, formats. Ensuite, nous avons d'abord passé en revue la littérature en étudiant chaque niveau distinct de notre roadmap à savoir le marketing, la conception de l'expérience utilisateur (CEU), la conception technique et la technologie. Ensuite, nous avons proposé la syntaxe et la sémantique concrètes d'un DSL.

Pour valider notre métamodèle de roadmap technologique, nous avons adapté l'approche basée sur l'exemple à la validation du métamodèle (López-Fernández et al. 2015). Nous avons sélectionné trois feuilles de route accessibles au public appartenant à des domaines très différents : une roadmap d'une agence spatiale, une roadmap d'une petite startup orientée utilisateur et une roadmap basée sur des scénarios d'une entreprise entrant sur un marché hautement concurrentiel, et avons formellement prouvé leur instanciabilité à partir de notre métamodèle proposé.

Pour la validation du DSL, nous avons incorporé notre méthodologie dans une application Web collaborative basée sur le cloud et mis en œuvre une étude de cas d'une entreprise automobile concevant une famille de produits de véhicules à émissions de CO2 réduites. Nous avons également procédé à une validation de la démarche avec notre partenaire industriel sur leur cas DNP, mais cette information est assez sensible et nous n'avons pas le droit de la publier.

Nous avons encore étendu notre approche en intégrant des procédures de test utilisateur dans notre cadre proposé et testé cette intégration avec de vrais utilisateurs.

Ce manuscrit de thèse peut intéresser les chefs de produit, les propriétaires de produit, les concepteurs EU, les cadres, les stratèges, les participants réguliers aux sessions de roadmapping ou les universitaires dans les domaines de la gestion de la technologie et de la science du design.

B METAMODEL IMPLEMENTATION

This appendix documents metamodel validation procedure. The syntax of JetBrains MPS Base Language is similar to Java. Code written in this language defines the abstract syntax of the metamodel and derived DSLs. Each code snippet represents single concept of a metamodel or a DSL. Keyword `extends` signifies abstraction relationship. Keywords `children` and `references` define aggregation and association relationships. Lines under keyword `properties` specify fields of a concept. `alias` defines the concrete syntax keyword that creates an instance of this concept.

Concepts in the **LightYellow** belong to the highest abstraction level (JetBrains MPS Base Language, M3). Concepts in the **Purple** are the concepts of the metamodel (M2). **LightBlue** concepts are the elements of DSLs (M1). Finally, user data (M0) is not highlighted. It is defined in a concrete syntax and preceded by corresponding alias, e.g., **Technology Area** Propulsion Systems.

Metamodel.Roadmap

```
1 concept Roadmap extends BaseConcept
2     implements INamedConcept
3
4 instance can be root: false
5 alias: <no alias>
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 dimensions: Dimensions[1]
16 graph: Graph[1]
```

Metamodel.Graph

```
1 concept Graph extends BaseConcept
2     implements <none>
3
4 instance can be root: false
5 alias: <no alias>
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 links: Link[0..n]
13 elements: Element[0..n]
14
15 references:
16 << ... >>
```

Metamodel.Dimensions

```
1 concept Dimensions extends BaseConcept
2     implements <none>
3
4 instance can be root: true
5 alias: dimensions
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 epochs: Epoch[1..n]
13 levels: Level[1..n]
14 scenarios: Scenario[0..n]
15
16 references:
17 << ... >>
18
19
20 <default> editor for concept Dimensions
21   node cell layout:
22
23   # alias #
24   epochs:
25     (- % epochs % /empty cell: <default> -)
26   levels:
27     (- % levels % /empty cell: <default> -)
28   scenarios:
29     (- % scenarios % /empty cell: <default> -)
30
31
32 inspected cell layout:
33 <choose cell model>
```

Metamodel.Epoch

```
1 concept Epoch extends BaseConcept
2           implements INamedConcept
3
4 instance can be root: false
5 alias: epoch
6 short description: <no short description>
7
8 properties:
9 date: string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

Metamodel.Level

```
1 concept Level extends BaseConcept
2           implements INamedConcept
3
4 instance can be root: false
5 alias: level
6 short description: <no short description>
7
8 properties:
9 order: integer
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

Metamodel.Scenario

```
1 concept Scenario extends BaseConcept
2           implements INamedConcept
3
4 instance can be root: false
5 alias: scenario
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

Metamodel.Element

```
1 concept Element extends BaseConcept
2           implements INamedConcept
3
4 instance can be root: false
5 alias: Element
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 refClass: ElementClass[0..1]
16 startDate: Epoch[0..1]
17 endDate: Epoch[0..1]
18 level: Level[0..1]
19 scenario: Scenario[0..1]
```

Metamodel.Link

```
1 abstract concept Link extends BaseConcept
2           implements <none>
3
4 instance can be root: false
5 alias: <no alias>
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 source: Element[1]
16 target: Element[1]
```


Metamodel.ElementClass

```
1 abstract concept ElementClass extends BaseConcept
2                               implements INamedConcept
3
4 instance can be root: false
5 alias: <no alias>
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 attributes: Attribute[0..n]
13 element_model: ElementModel[0..1]
14 class: ElementClass[0..n]
15
16 references:
17 << ... >>
```

Metamodel.Attribute

```
1 concept Attribute extends BaseConcept
2                               implements INamedConcept
3
4 instance can be root: false
5 alias: <no alias>
6 short description: <no short description>
7
8 properties:
9 units: string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

Metamodel.ElementModel

```
1 concept ElementModel extends BaseConcept
2                               implements INamedConcept
3
4 instance can be root: false
5 alias: <no alias>
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

Metamodel.ModelConfiguration

```
1 concept ModelConfiguration extends BaseConcept
2                               implements INamedConcept
3
4 instance can be root: false
5 alias: <no alias>
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 model: ElementModel[0..1]
16 element: Element[0..1]
```

Metamodel.Function

```
1 interface concept Function extends <none>
2
3
4 properties:
5 << ... >>
6
7 children:
8 << ... >>
9
10 references:
11 graph: Graph[1]
```

Metamodel.ModelFunction

```
1 interface concept ModelFunction extends Function
2
3
4 properties:
5 << ... >>
6
7 children:
8 << ... >>
9
10 references:
11 update: ModelConfiguration[1]
```

Metamodel.AttributeFunction

```
1 interface concept AttributeFunction extends Function
2
3
4 properties:
5 << ... >>
6
7 children:
8 << ... >>
9
10 references:
11 update : AttributeValue[1]
```

Metamodel.AttributeValue

```
1 concept AttributeValue extends BaseConcept
2           implements <none>
3
4 instance can be root: false
5 alias: <no alias>
6 short description: <no short description>
7
8 properties:
9 value : string
10
11 children:
12 << ... >>
13
14 references:
15 attribute : Attribute[1]
16 element : Element[1]
```

C CASE STUDY 1.1: NASA ROADMAP: DSL (M1)

DSL.NASA.TechnologyArea

```
1 concept TechnologyArea extends ElementClass
2           implements <none>
3
4 instance can be root: false
5 alias: Technology Area
6 short description: <no short description>
7
8 properties:
9 classifier : string
10
11 children:
12 technologyFamilies: TechnologyFamily[0..n]
13
14 references:
15 << ... >>
```

DSL.NASA.TechnologyFamily

```
1 concept TechnologyFamily extends ElementClass
2           implements <none>
3
4 instance can be root: false
5 alias: Technology Family
6 short description: <no short description>
7
8 properties:
9 classifier : string
10
11 children:
12 technologyClasses: TechnologyClass[0..n]
13
14 references:
15 << ... >>
```

DSL.NASA.TechnologyClass

```
1 concept TechnologyClass extends ElementClass
2           implements <none>
3
4 instance can be root: false
5 alias: Technology Class
6 short description: <no short description>
7
8 properties:
9 classifier : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.NASA.Mission

```
1 concept Mission extends ElementClass
2           implements <none>
3
4 instance can be root: false
5 alias: Mission
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.NASA.Launch

```
1 concept Launch extends Element
2           implements <none>
3
4 instance can be root: false
5 alias: Launch
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.NASA.Need

```
1 concept Need extends Element
2           implements <none>
3
4 instance can be root: false
5 alias: Need
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.NASA.Requires

```
1 concept Requires extends Link
2           implements <none>
3
4 instance can be root: false
5 alias: requires
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.NASA.Enables

```
1 concept Enables extends Link
2           implements <none>
3
4 instance can be root: false
5 alias: enables
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.NASA.Technology

```
1 concept Technology extends ElementClass
2           implements <none>
3
4 instance can be root: false
5 alias: Technology
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 class_ref : TechnologyClass[1]
16 challenge : string
17 dependency : boolean
18 classifier : string
```

DSL.NASA.Capability

```
1 concept Capability extends ElementClass
2           implements <none>
3
4 instance can be root: false
5 alias: Capability
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.NASA.CapabilityCurrent

```
1 concept CapabilityCurrent extends Element
2                               implements <none>
3
4 instance can be root: false
5 alias: Capability State-of-the-art:
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.NASA.CapabilityTarget

```
1 concept CapabilityTarget extends Element
2                               implements <none>
3
4 instance can be root: false
5 alias: Capability Performance Goal:
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```


DSL.NASA.TechnologyCurrent

```
1 concept TechnologyCurrent extends Element
2                               implements <none>
3
4 instance can be root: false
5 alias: Technology State-of-the-art:
6 short description: <no short description>
7
8 properties:
9   trl : integer
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
16 description : string
```

DSL.NASA.TechnologyTarget

```
1 concept TechnologyTarget extends Element
2                               implements <none>
3
4 instance can be root: false
5 alias: Technology Performance Goal
6 short description: <no short description>
7
8 properties:
9   trl : integer
10  description : string
11
12 children:
13 << ... >>
14
15 references:
16 << ... >>
```

D CASE STUDY 1.1: NASA ROADMAP: User Data (M0)

Data.NASA.Classifier

```
1 Technology Area 1 Launch Propulsion Systems
2   Technology Family 1.1 Solid Rocket Propulsion Systems
3     Technology Class 1.1.1 Propellants
4     Technology Class 1.1.2 Case Material
5     Technology Class 1.1.3 Nozzle System
6     Technology Class 1.1.4 Hybrid Rocket
7     Technology Class 1.1.5 Fundamental Solid Propulsion Technology
8     Technology Class 1.1.6 Integrated Solid Motor Systems
9     Technology Class 1.1.7 Liner and Insulation
10  Technology Family 1.2 Liquid Rocket Propulsion Systems
11    Technology Class 1.2.1 LH2/LOX Based
12    Technology Class 1.2.2 RP/LOX Based
13    Technology Class 1.2.3 CH4/LOX Based
14    Technology Class 1.2.4 Detonation Wave Engines - Closed Cycle
15    Technology Class 1.2.5 Propellant
16    Technology Class 1.2.6 Fundamental Liquid Propulsion Technology
17  Technology Family 1.3 Air-Breathing Propulsion Systems
18    << ... >>
19  Technology Family 1.4 Ancillary Propulsion Systems
20    << ... >>
21  Technology Family 1.5 Unconventional and Other Propulsion Systems
22    << ... >>
23  Technology Family 1.6 Balloon Systems
24    << ... >>
25 Technology Area 2 In-Space Propulsion Technologies
26   << ... >>
27 Technology Area 3 Space Power and Energy Storage
28   << ... >>
29 Technology Area 4 Robotics and Autonomous Systems
30   << ... >>
31 Technology Area 5 Communications, Navigation, and Orbital Debris Tracking and
    Characterization Systems
32   << ... >>
33 Technology Area 6 Human Health, Life Support, and Habitation Systems
34   << ... >>
35 Technology Area 7 Human Exploration Destination Systems
36   << ... >>
37 Technology Area 8 Science Instruments, Observatories, and Sensor Systems
38   << ... >>
39 Technology Area 9 Entry, Descent, and Landing Systems
40   << ... >>
```

Data.NASA.Technology.1-1-7-1

```
1 Polybenzimidazole Acrylonitrile Butadiene Rubber (PBI NBR) Based Asbestos-Free Liner and
  Insulation is a Technology
2 with attributes:
3   Insulation voids units: per motor segment
4   Propellant/liner/insulation voids units: per motor segment
5 Technology Description: Reformulation of insulation using an alternative to asbestos
6 Technology Challenge: Currently there are process issues, creating significant rework
  or scraping of large cast motor segments
7 Technology Development Dependent Upon Basic Research: false
8 Technology Class: 1.1.7 Liner and Insulation
9 Classifier: 1.1.7.1
10 Technology State-of-the-art: This technology is under development in the Space Launch
  System (SLS) Booster project. However, certain issues remain that may require further
  technology assessment
11 TRL: 7
12 define attribute values:
13   Insulation voids 10 per motor segment
14   Propellant/liner/insulation voids 1 per motor segment
15 Technology Performance Goal: Achieve the same levels of internal temperature as existing
  solid rocket motors, but without the use of asbestos.
16 TRL: 9
17 define attribute values:
18   Insulation voids 0 per motor segment
19   Propellant/liner/insulation voids 0 per motor segment
20 Asbestos-free liner and insulation is a Capability
21 with attributes:
22   Case material temperature limits units: C
23 Capability Description: Solid motor casings <..>
24 Capability State-of-the-art: Shuttle reusable rocket motor <..>
25 define attribute values:
26   Case material temperature limits N/A C
27 Capability Performance Goal: Achieve the same levels of internal temperature <..>
28 define attribute values:
29   Case material temperature limits N/A C
```

Data.NASA.Technology.1-1-7-2

```
1 Insulating/Ablative Sprayable Liner is a Technology
2   with attributes:
3     Current case material weigh units: %
4   Technology Description: This is a liner capable of insulative properties similar to
   fiber-filled nitrile butadiene rubber (NBR) and ethylene propylene diene monomer (EPDM)
   insulations. Includes significant weight reduction and elimination of process issues
   addressed tod
5   Technology Challenge: Producing an acceptable sprayable liner is a technical challenge.
6   Technology Development Dependent Upon Basic Research: false
7   Technology Class: 1.1.7 Liner and Insulation
8   Classifier: 1.1.7.2
9   Technology State-of-the-art: This technology is under development in Space Launch System
   (SLS) Booster project. Current issues remain that may require further technology
   development.
10  TRL: 5
11  define attribute values:
12    << ... >>
13  Technology Performance Goal: Reduce insulation weight by reducing insulation thickness
14  TRL: 9
15  define attribute values:
16    Current case material weigh 25 %
17 A sprayable liner is a Capability
18   with attributes:
19     Current case material weigh units: %
20   Capability Description: The sprayable liner can maintain the solid rocket booster motor
   case internal temperature below thermal limits without use of asbestos as an
   insulating material. Previous systems operated with a waver; new systems cannot have a
   waver, thus new asbestos-free options must be developed
21  Capability State-of-the-art: Reusable Solid Rocket Motor (RSRM) Carbon Black Based Liner.
   The Shuttles RSRM required a waver to operate the last few years.
22  define attribute values:
23    Current case material weigh 0 %
24  Capability Performance Goal: Need to achieve same levels of internal temperature without
   use of asbestos
25  define attribute values:
26    Current case material weigh 25 %
```

Data.NASA.Mission.DRM5

```
1 DRM5 Asteroid Redirect - Robotic spacecraft is a Mission
2
3 create Launch Launch_DRM5
4   ( epoch: 2018 level: Human exploration and Operations scenario: <no scenario> )
5
6 create Need for DRM5 Asteroid Redirect - Robotic spacecraft Need_DRM5
7   ( epoch: 2015 level: Human exploration and Operations scenario: <no scenario> )
8
9 1.1.7.1 Performance Goal enables Need_DRM5
10 1.1.7.2 Performance Goal enables Need_DRM5
11
12 <Press [Enter] to define enhancer links>
13
14 Launch_DRM5 requires Need_DRM5
```

E CASE STUDY 1.2: DESIGN ROADMAP: DSL (M1)

DSL.Design.CoreExperience

```
1 concept CoreExperience extends Element
2                               implements <none>
3
4 instance can be root: false
5 alias: Core Experience
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.Design.PrimaryUserNee

```
1 concept PrimaryUserNeed extends Element
2                               implements <none>
3
4 instance can be root: false
5 alias: Primary User Need
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.Design.Outcome

```
1 concept Outcome extends Element
2     implements <none>
3
4 instance can be root: false
5 alias: Outcome
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.Design.KeyFeature

```
1 concept KeyFeature extends Element
2     implements <none>
3
4 instance can be root: false
5 alias: Key Function/Feature
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.Design.FormFactor

```
1 concept FormFactor extends Element
2     implements <none>
3
4 instance can be root: false
5 alias: Form Factor
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.Design.AudFeedback

```
1 concept AudFeedback extends Element
2           implements <none>
3
4 instance can be root: false
5 alias: Auditory Feedback
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.Design.Microcontroller

```
1 concept Microcontroller extends Element
2           implements <none>
3
4 instance can be root: false
5 alias: Micro-Controller
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.Design.Power

```
1 concept Power extends Element
2           implements <none>
3
4 instance can be root: false
5 alias: Power
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.Design.ScreenInterface

```
1 concept ScreenInterface extends Element
2           implements <none>
3
4 instance can be root: false
5 alias: Screen Interface
6 short description: <no short description>
7
8 properties:
9 description : string
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.Design.Satisfies

```
1 concept Satisfies extends Link
2           implements <none>
3
4 instance can be root: false
5 alias: satisfies
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.Design.Requires

```
1 concept Requires extends Link
2           implements <none>
3
4 instance can be root: false
5 alias: requires
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```


DSL.Design.Evolve

```
1 concept Evolve extends Link
2     implements <none>
3
4 instance can be root: false
5 alias: evolves to
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

F CASE STUDY 1.2: DESIGN: USER DATA (M0)

Data.DesignRoadmap

```
1 Core experiences:
2   create Core Experience Core experience 1
3   ( start date: 2012 level: Design Roadmap scenario: <no scenario> )
4   description: Make learning about diabetes fun through game play
5   create Core Experience Core experience 2
6   ( start date: 2013 level: Design Roadmap scenario: <no scenario> )
7   description: Learn about diabetes through a combo of free and guided play
8   create Core Experience Core experience 3
9   ( start date: 2015 level: Design Roadmap scenario: <no scenario> )
10  description: Learn about chronic disease through a combo of free and guided play
11 Primary User Needs:
12  create Primary User Need User need 1
13  ( start date: 2012 level: Design Roadmap scenario: <no scenario> )
14  description: Learning about diabetes should be fun and easy
15  create Primary User Need User need 2
16  ( start date: 2013 level: Design Roadmap scenario: <no scenario> )
17  description: Through storytelling kids relate to Jerry and cope with emotions while
18  learning about diabetes
19  create Primary User Need User need 3
20  ( start date: 2015 level: Design Roadmap scenario: <no scenario> )
21  description: All kids with chronic diseases want to cope their emotions and learn about
22  their disease through play
23 Outcome:
24  create Outcome Outcome 1
25  ( start date: 2012 level: Design Roadmap scenario: <no scenario> )
26  description: Newly diagnosed kids love Jerry but too easy for those who had it for a
27  longer period
28  create Outcome Outcome 2
29  ( start date: 2013 level: Design Roadmap scenario: <no scenario> )
30  description: Able to articulate symptoms, increase in confidence, able to master carb
31  counting
32  create Outcome Outcome 3
33  ( start date: 2015 level: Design Roadmap scenario: <no scenario> )
34  description: Emphasis on wellness as a main curriculum paired with disease-specific
35  curriculum
36 Key Functions/Features:
37  create Key Function/Feature Generation 1
38  ( start date: 2012 level: Product Generation scenario: <no scenario> )
39  description: BGL check, insulin dosing, feeding foods, 6 injection sites, light sensor
40  color detection
41  create Key Function/Feature Generation 2
42  ( start date: 2013 level: Product Generation scenario: <no scenario> )
43  description: BGL check, insulin dosing, feeding foods, 21 interactive storybooks, 6
44  injection sites, 4 tickle spots, RFID feeding create
45  create Key Function/Feature Generation 3
46  ( start date: 2015 level: Product Generation scenario: <no scenario> )
47  description: NFC feeding base, Tablet App different short stories and diagnostic tools,
48  collect data via software
```

Data.DesignRoadmap (continued)

```
44 Micro-controller:
45   create Micro-Controller Arduino
46   ( start date: 2012 level: Technology Development scenario: <no scenario> )
47   description:
48   create Micro-Controller Core Processor
49   ( start date: 2015 level: Technology Development scenario: <no scenario> )
50   description:
51
52 Screen Interface:
53   create Screen Interface Mono-color
54   ( start date: 2012 level: Technology Development scenario: <no scenario> )
55   description: with RFID
56   create Screen Interface Color Touch Screen-1
57   ( start date: 2013 level: Technology Development scenario: <no scenario> )
58   description: Nokia
59   create Screen Interface Color Touch Screen-2
60   ( start date: 2015 level: Technology Development scenario: <no scenario> )
61   description: with RFID, NFC Tags, Light sensor
62
63 Auditory Feedbacks:
64   create Auditory Feedback Speaker
65   ( start date: 2012 level: Technology Development scenario: <no scenario> )
66   description:
67
68 Power:
69   create Power AA batteries
70   ( start date: 2012 level: Technology Development scenario: <no scenario> )
71   description:
72   create Power Lithium-Ion Rechargeable Battery
73   ( start date: 2013 level: Technology Development scenario: <no scenario> )
74
75 Form-factor:
76   create Form Factor A Bear shell-1 (Teddy Bear)
77   ( start date: 2012 level: Technology Development scenario: <no scenario> )
78   description:
79   create Form Factor A Bear shell-2 (Teddy Bear)
80   ( start date: 2013 level: Technology Development scenario: <no scenario> )
81   description: easy to clean Asthma-friendly fabric
82
83 Links:
84   Generation 1 satisfies Outcome 1
85   Generation 2 satisfies Outcome 2
86   Generation 3 requires Core Processor
87   Generation 2 requires Color Touch Screen-1
88   Generation 1 evolves to Generation 2
89   Generation 2 evolves to Generation 3
90   Generation 1 requires Arduino
```

G CASE STUDY 1.3: SCENARIO-BASED ROADMAP: DSL (M1)

DSL.ScenarioBased.BusinessDriver

```
1 concept BusinessDriver extends Element
2           implements <none>
3
4 instance can be root: false
5 alias: Business Driver
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.ScenarioBased.Product

```
1 concept Product extends Element
2           implements <none>
3
4 instance can be root: false
5 alias: Product
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.ScenarioBased.Service

```
1 concept Service extends Element
2           implements <none>
3
4 instance can be root: false
5 alias: Service
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.ScenarioBased.Causes

```
1 concept Causal extends Link
2           implements <none>
3
4 instance can be root: false
5 alias: causes
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.ScenarioBased.Influence

```
1 concept Influence extends Link
2           implements <none>
3
4 instance can be root: false
5 alias: influence
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 << ... >>
13
14 references:
15 << ... >>
```

DSL.ScenarioBased.Kpi

```
1 concept KPI extends Element
2     implements <none>
3
4 instance can be root: false
5 alias: Key Performance Indicator
6 short description: <no short description>
7
8 properties:
9 << ... >>
10
11 children:
12 kpiModel : KpiModel[0..1]
13
14 references:
15 << ... >>
```

H CASE STUDY 1.3: SCENARIO-BASED ROADMAP: USER DATA (M0)

Data.ScenarioBasedRoadmap.Neutral

```
1 create Business Driver Policy for Green Growth Korea
2   ( start date: Jan-2010 end date: Jan-2011 level: Market scenario: Neutral )
3 create Business Driver Social movement toward sustainability
4   ( start date: Jan-2010 end date: Jan-2011 level: Market scenario: Neutral )
5 create Business Driver Policy for reducing traffic volume
6   ( start date: May-2011 end date: Dec-2011 level: Market scenario: Neutral )
7 create Business Driver Law for sustainable traffic development
8   ( start date: Jan-2011 end date: Sep-2011 level: Market scenario: Neutral )
9
10 create Product Development of environmental-friendly car
11   ( start date: Jan-2010 end date: Mar-2011 level: Product scenario: Neutral )
12 create Product Automatic payment function
13   ( start date: Jul-2011 end date: Jan-2012 level: Product scenario: Neutral )
14 create Product Car for ergonomics
15   ( start date: Jul-2010 end date: May-2011 level: Product scenario: Neutral )
16 create Product Car for multiple customers
17   ( start date: Jan-2010 end date: Jul-2010 level: Product scenario: Neutral )
18
19 create Service e-Tracking Service
20   ( start date: Jul-2011 end date: Apr-2012 level: Service scenario: Neutral )
21 create Service Automatic return Service
22   ( start date: May-2011 end date: Nov-2011 level: Service scenario: Neutral )
23 create Service Public Transp. notification
24   ( start date: Jul-2010 end date: Mar-2011 level: Service scenario: Neutral )
25 create Service Seat customization Service for multi-users
26   ( start date: Aug-2010 end date: Sep-2011 level: Service scenario: Neutral )
27 create Service Mileage Service
28   ( start date: Jan-2010 end date: Mar-2011 level: Service scenario: Neutral )
29 create Service Car port notification Service
30   ( start date: Jan-2010 end date: Jul-2010 level: Service scenario: Neutral )
31 create Service Service for individuals
32   ( start date: Jan-2010 end date: Aug-2010 level: Service scenario: Neutral )
```

Data.ScenarioBasedRoadmap.Pessimistic

```
1 create Business Driver Weak Social movement towards sustainability
2   ( start date: Jan-2010 end date: Jul-2010 level: Market scenario: Pessimistic )
3 create Business Driver Weak Policy for Green Growth Korea
4   ( start date: Jan-2010 end date: Sep-2011 level: Market scenario: Pessimistic )
```

Data.ScenarioBasedRoadmap.Optimistic

```
1 create Business Driver Law for tax deduction for car-sharing
2   ( start date: Sep-2011 end date: Jun-2012 level: Market scenario: Optimistic )
3 create Business Driver Law for sustainable traffic development
4   ( start date: Jan-2011 end date: Sep-2011 level: Market scenario: Optimistic )
5 create Business Driver Policy for improving parking lot
6   ( start date: Jul-2011 end date: Apr-2012 level: Market scenario: Optimistic )
7 create Business Driver Policy for reducing traffic volume
8   ( start date: Jan-2011 end date: Jul-2011 level: Market scenario: Optimistic )
9
10 create Product Automatic payment function
11   ( start date: May-2011 end date: Feb-2012 level: Product scenario: Optimistic )
12 create Product Car for ergonomics
13   ( start date: Jul-2010 end date: May-2011 level: Product scenario: Optimistic )
```

Data.ScenarioBasedRoadmap.Events

```
1 Policy for Green Growth Korea causes Law for sustainable traffic development [ Neutral ->
  Optimistic ]
2 Policy for Green Growth Korea [ Neutral ] causes Policy for reducing traffic volume [
  Neutral ]
3 Social movement toward sustainability causes Law for sustainable traffic development [
  Neutral -> Optimistic ]
4 Social movement toward sustainability [ Neutral ] causes Policy for reducing traffic volume
  [ Neutral ]
5 Policy for reducing traffic volume [ Optimistic ] causes Policy for improving parking lot [
  Optimistic ]
6 Law for sustainable traffic development [ Optimistic ] causes Law for tax deduction for
  car-sharing [ Optimistic ]
```

Data.ScenarioBasedRoadmap.KpiModel

```
1 create Key Performance Indicator Improvement of traffic environment model: M1
2 create Key Performance Indicator Level Of Company Motivation <no kpiModel>
3 create Key Performance Indicator Reduction of Environmental burden <no kpiModel>
4 create Key Performance Indicator Use of Car-sharing service model: F1
5 create Key Performance Indicator Total revenue model: F2
6 create Key Performance Indicator Energy Consumption model: F3
7 Social movement toward sustainability influence Use of Car-sharing service
8 Level of Company Motivation influence Use of Car-sharing service
9 Use of Car-sharing service influence Improvement of traffic environment
10 Use of Car-sharing service influence Total revenue
11 Use of Car-sharing service influence Energy Consumption
12 Use of Car-sharing service influence Reduction of Environmental burden
```


I USER STORY MAP OF THE SUPPORT TOOL

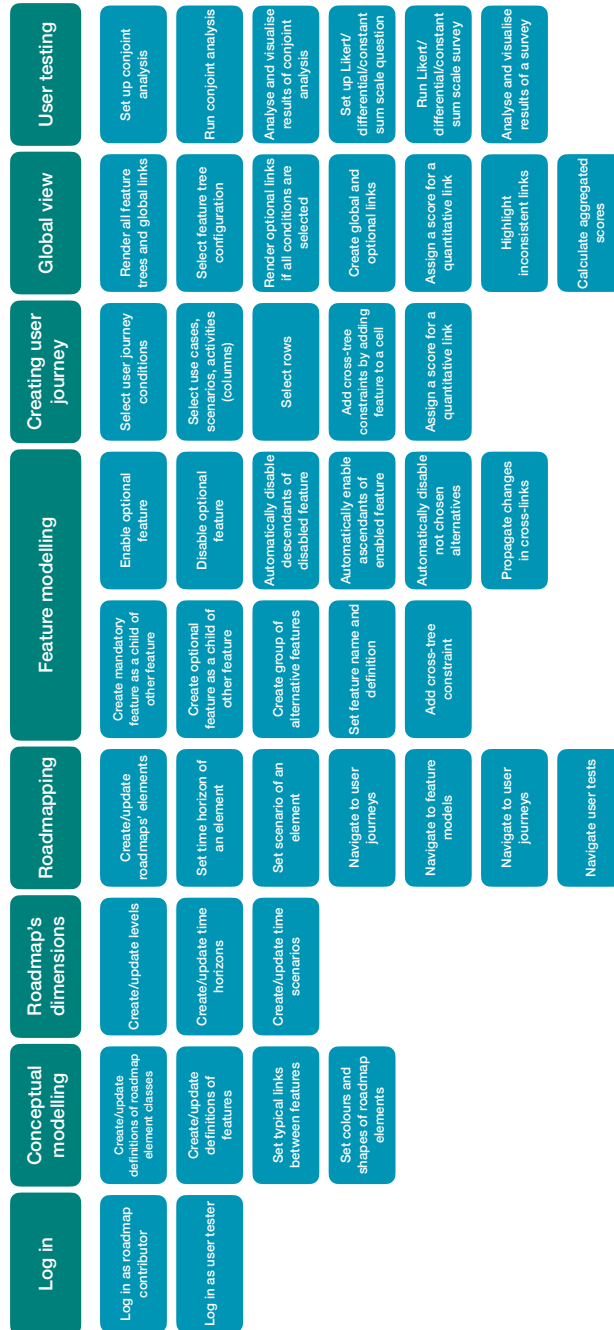
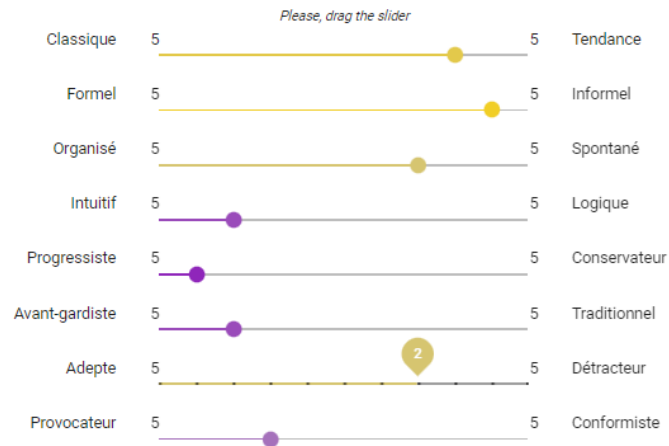


Figure I.1: User Story Map (zoomable)

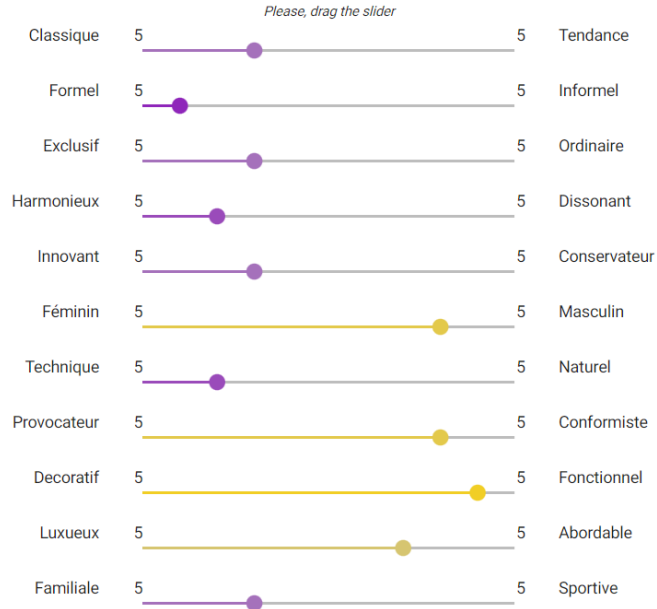
J CASE STUDY 3.1: CMI USER TESTING

Comment définiriez-vous votre personnalité parmi les adjectifs suivants



Forward

Comment évalueriez-vous l'esthétique du cockpit que vous avez testé parmi les adjectifs suivants



Forward

Imaginez-vous, dans 6 mois à partir de maintenant, dans une concession automobile en train d'acheter une nouvelle voiture. Évaluez entre 0 et 10 votre intérêt pour acheter l'option au prix indiqué (0=pas du tout intéressé, 10=tout à fait intéressé). Le prix de chaque option sera ajouté au prix de base de la voiture.

Les caractéristiques soulignées indiquent les différences entre les options, c'est à dire que les caractéristiques qui *ne sont pas* soulignées sont communes aux deux options.

Le mode Chauffeur figurera dans les options qui vous seront proposées.

<p>Option 1</p> <p>Indicateur tête haute (se situe au-dessus du tableau de bord)</p> <p>Ecran dans le pare-brise (se situe au centre du véhicule)</p> <p>LED dans le volant</p> <p>Mode chauffeur</p> <p>Siege vibrant</p> <p>Price: 3900 EUR</p> <p><i>Please, drag the slider to rate this option</i></p>	<p>Option 2</p> <p>Ecran dans le pare-brise (se situe au centre du véhicule)</p> <p>Mode chauffeur</p> <p>Siege vibrant</p> <p>Price: 3500 EUR</p> <p><i>Please, drag the slider to rate this option</i></p>
--	--

< Go back

Forward

Imaginez-vous, dans 6 mois à partir de maintenant, dans une concession automobile en train d'acheter une nouvelle voiture. Évaluez entre 0 et 10 votre intérêt pour acheter l'option au prix indiqué (0=pas du tout intéressé, 10=tout à fait intéressé). Le prix de chaque option sera ajouté au prix de base de la voiture.

Les caractéristiques soulignées indiquent les différences entre les options, c'est à dire que les caractéristiques qui *ne sont pas* soulignées sont communes aux deux options.

Le mode Chauffeur figurera dans les options qui vous seront proposées.

Option 1

Ecran dans le pare-brise (se situe au centre du véhicule)

LED dans le volant

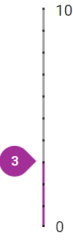
Messages vocaux dans appui-tête

Mode chauffeur

Siege vibrant

Price: 4000 EUR

Please, drag the slider to rate this option



Option 2

Indicateur tête haute (se situe au-dessus du tableau de bord)

Ecran dans le pare-brise (se situe au centre du véhicule)


Messages vocaux dans appui-tête

Mode chauffeur

Siege vibrant

Price: 4000 EUR

Please, drag the slider to rate this option



< Go back Forward

Imaginez-vous, dans 6 mois à partir de maintenant, dans une concession automobile en train d'acheter une nouvelle voiture. Évaluez entre 0 et 10 votre intérêt pour acheter l'option au prix indiqué (0=pas du tout intéressé, 10=tout à fait intéressé). Le prix de chaque option sera ajouté au prix de base de la voiture.

Les caractéristiques soulignées indiquent les différences entre les options, c'est à dire que les caractéristiques qui *ne sont pas* soulignées sont communes aux deux options.

Le mode Chauffeur figurera dans les options qui vous seront proposées.

Option 1

Indicateur tête haute (se situe au-dessus du tableau de bord)


Ecran dans le pare-brise (se situe au centre du véhicule)

Messages vocaux dans appui-tête

Siege vibrant

Price: 1000 EUR

Please, drag the slider to rate this option




Option 2

Ecran dans le pare-brise (se situe au centre du véhicule)

Price: 400 EUR

Please, drag the slider to rate this option



< Go back Forward

Imaginez-vous, dans 6 mois à partir de maintenant, dans une concession automobile en train d'acheter une nouvelle voiture. Évaluez entre 0 et 10 votre intérêt pour acheter l'option au prix indiqué (0=pas du tout intéressé, 10=tout à fait intéressé). Le prix de chaque option sera ajouté au prix de base de la voiture.

Les caractéristiques soulignées indiquent les différences entre les options, c'est à dire que les caractéristiques qui *ne sont pas* soulignées sont communes aux deux options.

Le mode Chauffeur figurera dans les options qui vous seront proposées.

Option 1

- Indicateur tête haute (se situe au-dessus du tableau de bord)
- Messages vocaux dans appui-tête
- Mode chauffeur**
- Siege vibrant

Price: 3600 EUR
Please, drag the slider to rate this option

Option 2

- Indicateur tête haute (se situe au-dessus du tableau de bord)
- LED dans le volant**
- Messages vocaux dans appui-tête
- Siege vibrant

Price: 800 EUR
Please, drag the slider to rate this option

< Go back Forward

Imaginez-vous, dans 6 mois à partir de maintenant, dans une concession automobile en train d'acheter une nouvelle voiture. Évaluez entre 0 et 10 votre intérêt pour acheter l'option au prix indiqué (0=pas du tout intéressé, 10=tout à fait intéressé). Le prix de chaque option sera ajouté au prix de base de la voiture.

Les caractéristiques soulignées indiquent les différences entre les options, c'est à dire que les caractéristiques qui *ne sont pas* soulignées sont communes aux deux options.

Le mode Chauffeur figurera dans les options qui vous seront proposées.

Option 1

- Indicateur tête haute (se situe au-dessus du tableau de bord)
- Ecran dans le pare-brise (se situe au centre du véhicule)
- LED dans le volant
- Messages vocaux dans appui-tête**
- Mode chauffeur**
- Siege vibrant

Price: 4200 EUR
Please, drag the slider to rate this option

Option 2

- Indicateur tête haute (se situe au-dessus du tableau de bord)
- Ecran dans le pare-brise (se situe au centre du véhicule)
- LED dans le volant

Price: 800 EUR
Please, drag the slider to rate this option

< Go back Forward

Imaginez-vous, dans 6 mois à partir de maintenant, dans une concession automobile en train d'acheter une nouvelle voiture. Évaluez entre 0 et 10 votre intérêt pour acheter l'option au prix indiqué (0=pas du tout intéressé, 10=tout à fait intéressé). Le prix de chaque option sera ajouté au prix de base de la voiture.

Les caractéristiques soulignées indiquent les différences entre les options, c'est à dire que les caractéristiques qui *ne sont pas* soulignées sont communes aux deux options.

Le mode Chauffeur figurera dans les options qui vous seront proposées.

Option 1

Indicateur tête haute (se situe au-dessus du tableau de bord)

Ecran dans le pare-brise (se situe au centre du véhicule)

Mode chauffeur

Price: 3600 EUR

Please, drag the slider to rate this option

Option 2

Mode chauffeur

Siege vibrant

Price: 3100 EUR

Please, drag the slider to rate this option

< Go back
Forward

Indiquez si vous êtes d'accord avec les affirmations suivantes **en distribuant 20 points**

Please, drag the slider

Je voudrais un véhicule capable de conduire seul dans certaines situations (par exemple sur l'autoroute), lorsque j'en ai envie, car il me permet d'assurer le confort pendant un court trajet	
Je voudrais un véhicule capable de conduire seul dans certaines situations (par exemple sur l'autoroute), lorsque j'en ai envie, car il me permet d'assurer le confort pendant un long trajet	
Je voudrais un véhicule capable de conduire seul dans certaines situations (par exemple sur l'autoroute) pour faciliter les trajets avec les enfants	
Je voudrais un véhicule capable de conduire seul dans certaines situations (par exemple sur l'autoroute), lorsque j'en ai envie, car il me permet d'assurer la sécurité	
Je voudrais un véhicule capable de conduire seul dans certaines situations (par exemple sur l'autoroute) car j'apprécie les nouvelles technologies	
Je voudrais que mon véhicule dispose d'un cockpit multimodal (avec des interfaces dédiées à différents: sens, vision, audition, etc.), lorsque j'en ai envie, car il me permet d'assurer le confort	
Je voudrais que mon véhicule dispose d'un cockpit multimodal (avec des interfaces dédiées à différents: sens, vision, audition, etc.), lorsque j'en ai envie, car il me permet d'assurer la sécurité	
Je voudrais que mon véhicule dispose d'un cockpit multimodal (avec des interfaces dédiées à différents: sens, vision, audition, etc.) car j'apprécie les nouvelles technologies	

6/20

Forward

K MODEL FOR DISCRETE-EVENT SIMULATION FOR DRIVER PERFORMANCE ASSESSMENT

Publications

The materials of this chapter were published in I. Iuskevich et al. (May 2020). "A Discrete-event Simulation Model For Driver Performance Assessment: Application To Autonomous Vehicle Cockpit Design Optimization." In: *Proceedings of the Design Society: DESIGN Conference 1*. Publisher: Cambridge University Press, pp. 2521–2530. issn: 2633-7762. doi: [10.1017/dsd.2020.157](https://doi.org/10.1017/dsd.2020.157). url: <https://doi.org/10.1017/dsd.2020.157> (visited on 12/03/2020).

Open access

Authors' version before review can also be found in <https://hal.archives-ouvertes.fr/hal-02866691>.

Abstract: The latest advances in the design of vehicles with the adaptive level of automation pose new challenges in the vehicle-driver interaction. Safety requirements underline the need to explore optimal cockpit architectures with regard to driver cognitive and perceptual workload, eyes-off-the-road time and situation awareness. We propose to integrate existing task analysis approaches into system architecture evaluation for the early-stage design optimization. We built the discrete-event simulation tool and applied it within the multi-sensory (sight, sound, touch) cockpit design industrial project.

K.1 Introduction

The accelerating technological progress rapidly changes the landscape of available design options in the automotive domain. Constant innovations in the design and functionality of an increasingly autonomous vehicle drive the evolution of the design of a cockpit. This happens because the increasing autonomy alters the status of the driver (authority of control is shared between driver and car). For example, in December 2017 Waymo LLC (the company is owned by Alphabet) has launched the commercial self-driving car service in Arizona. In their car most of the time driving tasks are

performed by machine, though trained driver is still required to take over in case of problems (Korosec 2019).

An automotive cockpit is an example of a human-machine interface (HMI). HMI can be defined as a technical system that allows human operators to monitor and control the state and behavior of the machine.

In the manual mode the driver is included into the control loop. In other words, he has full control over the car's course (i.e. steering) and acceleration. Accordingly, HMI of such car includes elements of direct control and stabilization – steering wheel, acceleration and brake pedal, tachometer, speedometer, etc.

At the highest level of autonomy the driver is fully excluded from the control loop, i.e. he becomes a passenger. Accordingly, the elements of control over car's trajectory are eliminated. Between these two extreme points there are countless amount of interface design options for so-called semi-automated cars for which authority of control is shared between human and machine (see spectrum of assistance and automation in Flemisch et al. (2014). Notable examples of such interfaces are the maneuver-based approach (Conduct-by-Wire), and haptic-multimodal approach (Horse-metaphor) (Flemisch et al. 2014).

Another practical semi-automated car design option is the vehicle with the adaptive level of automation. As defined in (Scerbo 2008) the automation is adaptive when “the level of automation or the number of systems operating under automation can be modified in real time. In addition, changes in the state of automation can be initiated by either the human or the system”. Up to date, infrastructure and technologies are not mature enough to enable the operation of fully autonomous vehicles. Under these conditions, adaptive automation is a good solution compared to fixed task allocation (between machine and human), since it allows to avoid cognitive overloads of a driver, boost situation awareness, or reduce complacency (Parasuraman and Wickens 2008) depending on the state of a driver or road conditions.

The major shortcoming of the adaptive automation is added complexity to the user interface. Traditional cars' interfaces have only elements of the direct control. Future fully autonomous cars will have only interfaces that communicate to the machine the coordinates of the final destination. At the same time, an interface of a car with the adaptive level of automation maintains elements of direct control, includes some features of self-driving cars and, additionally, is augmented by elements of control over the level of automation (switchers, indicators of level of automation, take-over

requests, etc).

Accordingly, interface for a car with the adaptive level of automation may increase workload and eyes-off-the road time, especially at the moments of transition between automation levels. These negative effects can be mitigated by optimizing of the HMI's functional architecture.

This work is intended to review existing methods of workload and eyes-off-the-road time estimation, integrate them into system architecture evaluation process, build the supporting tool and apply it for a new type of systems.

The remainder of the paper is structured as follows. In the next section we present the results of the literature review in the field of workload and eyes-off-the-road time modelling. Then, our proposed discrete-event simulation model is described. After that, we present the results of the case study. In the final section we discuss how the model can be extended and validated in future work.

K.2 Background

K.2.1 Mental workload modeling

Since mid-80s, a lot of research efforts have been devoted to the development of the human operator mental workload modeling tools. Most of these approaches are based on the idea of task analysis, or, more specifically, task network. Task network is a functional decomposition of a human operator's activities down to elementary tasks (Laughery et al. 2000). Then these elementary tasks are annotated with number of attributes (descriptors), e.g. required workload, task duration, triggering event, etc. The total cognitive workload is then calculated either with matrix-based approach (W/INDEX) (North and Riley 1989) or as the result of a simulation in a discrete-event environment (Aldrich, Szabo, and Bierbaum 1989). More recent works in this field propose to employ Petri nets to model human operator strategies, adaptive to the changing environment (Kontogiannis 2005).

(Boy 1998) generalized task network approaches under the name of Cognitive Function Analysis (CFA). This approach responds to the emergence of highly automated and cooperative systems. In CFA, complex cognitive functions may be allocated not only to a human, but also to a machine, which in turn is interacting with other automated machines. Accordingly, CFA is more suitable for the modern context where

authority of control is shared between humans and machines.

In the past, all these approaches were driven mainly by increasing automation in military, aerospace or power plants domains. Nowadays, research focus is shifting to the automotive systems due to the progress in the domain of car automation.

K.2.2 Eyes-off-the-road time estimation

Compared to mental workload modeling, eyes-off-the-road time prediction approaches received less attention in the literature. However, this metric is important from safety perspective in the automotive domain. Driver's distraction is the leading defined cause of road accidents according to (Wang, Knippling, and Goodman 1996). Up to 20% of crashes due to driver distraction are caused by the interaction with interior equipment (car interfaces or cellphone) (Green 2017). These facts underline the importance of efforts to minimize the potential visual distraction caused by elements of user interface during the design optimization.

In (Wittmann et al. 2006) it was experimentally proven that the location of the onboard display greatly influences the safety of driving, e.g. perception of the information on the head-up display causes shorter distraction compared to instrument cluster or central panel.

The integration of the Keystroke Level Model and occlusion technique was presented in (Pettitt, Burnett, and Stevens 2007). The goal of this approach is to predict eyes-off-the-road time having a list of driving tasks as an input. The validation of this approach have shown high accuracy with experimental results.

K.2.3 Cognitive simulation models

Approaches based on task analysis treat driver's cognition as a black-box (tasks play role of inputs and outputs). In contrast, cognitive simulation methods approach the modelling of the human internal mental processes. Without the model of the human cognition system it is impossible to evaluate another important safety-related metric – situation awareness.

The most comprehensive model of the human driver performances up to date is COgnitive Simulation MOdel of the DRIVER (COSMODRIVE) (Bellet et al. 2011). COSMODRIVE is composed of three modules – perception, cognition and action. It models in details all main human driver mental activities. It can be connected to SiVIC virtual

road environment platform to provide very detailed input to a perceptual model of the virtual driver.

A workload prediction method based on cognitive architecture for safety critical task simulation (CASCaS) (Feuerstack, Lüdtke, and Osterloh 2007) was used to build real-time assessment of driver's workload in order to enable adaptive automation (Wortelen et al. 2016).

K.3 Discrete-event simulation model architecture

In this section we describe the architecture of our proposed discrete-event simulation model, task data structure and workload scales.

From systems engineering point of view, the design of the autonomous vehicle cockpit is a challenging task even though the complexity of the physical architecture of this kind of a system is relatively low (compared to, for example, to aerospace systems). The complexity of the design of a cockpit rests on the ambiguity of the functional architecture which builds upon the ambiguity and complexity of the human driver behavior. The latter is complex due to the fact that the number of possible human reactions is large and hardly formalizable, and ambiguous because there are a number of known unknowns such as level of experience, personal attitude, tiredness, mood, etc.

However, we can claim without proof that two functional architectures of a cockpit may be compared in terms of safety without an ambition to predict absolute values of the cockpit's key performances. For example, if one variant of a cockpit constantly transmit a lot of unnecessary information to a driver, the critical information may be eventually missed, which means that other, less distracting variant of a cabin is generally safer. Our system of interest is a cabin of a car with adaptive levels of automation (Level 0-4). The Level 4 we will call further Automotive Driving (AD) mode. A driver can switch to a higher level of automation at any moment if this level is available due to road conditions (e.g. AD mode may not be available due to the absence of road markings). A higher to a lower level of automation switch is activated by the driver at any moment or by the machine if this higher level of automation is no longer available due to the road conditions.

Special mode called take-over request (TOR60) is activated 60 seconds before the moment when AD is no longer available. During this mode interface sends signals to a driver in order to advise him to put hands on a steering wheel and switch to a lower

level of automation. The state machine of the car of interest is shown on figure K.1.

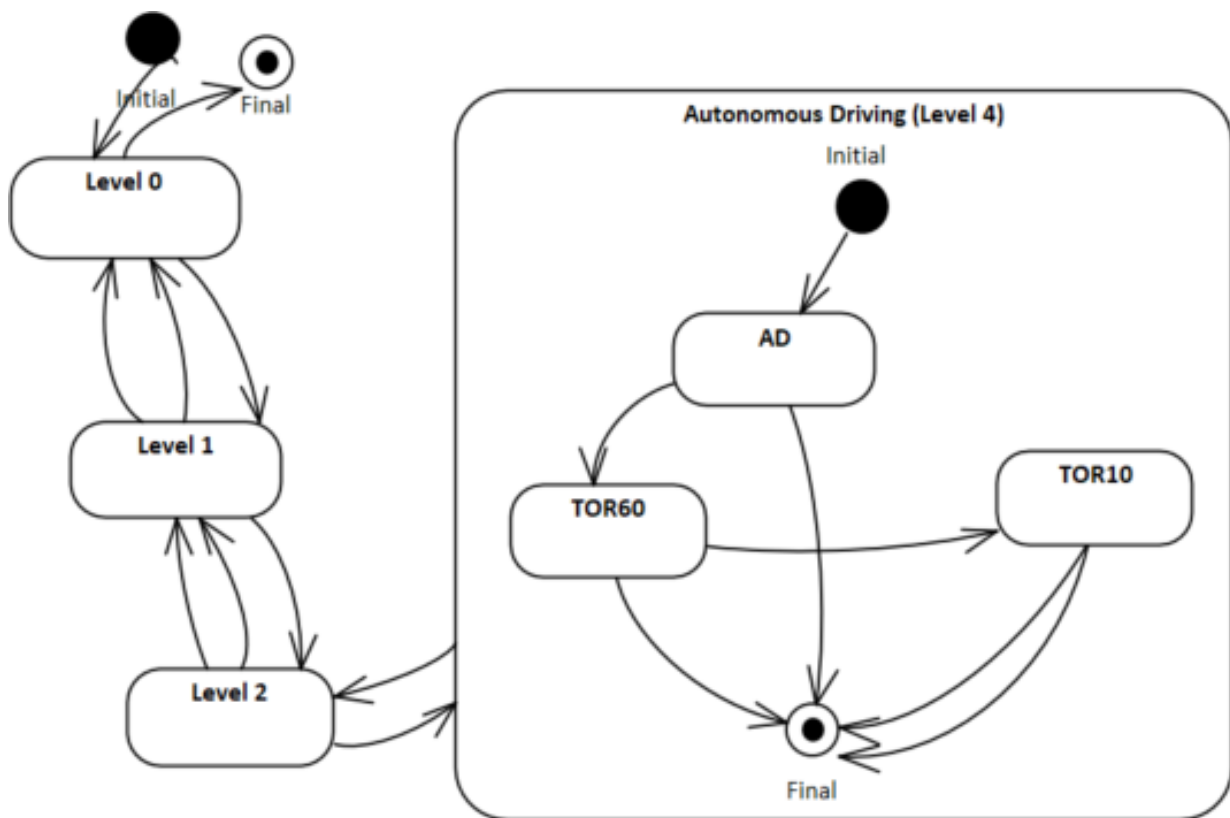


Figure K.1: The state-machine of the car with adaptive automation

The architecture of the human-machine interaction model is shown on figure K.2. It consists of 7 major elements: road conditions, vehicle's state machine and cognitive functions, driver's memory, tasks' schedule and cognitive functions, and task list. Elements are communicating to each other by means of events. For the modeling we use Python discrete-event simulation library SimPy.

Event is triggered in one element and influences the processes or state of another element. For example, an event of road conditions change may trigger the transition of the vehicle state (automation level).

Cognitive functions are modelled as a random processes generating events in time. For example, "check the speed on average every 20 seconds" is a cognitive function. It triggers events normally distributed in time with mean of 20 seconds.

Tasks are modelled by complex objects which in SimPy referred to as processes. Process is triggered by some event and active within the limited interval of time. When

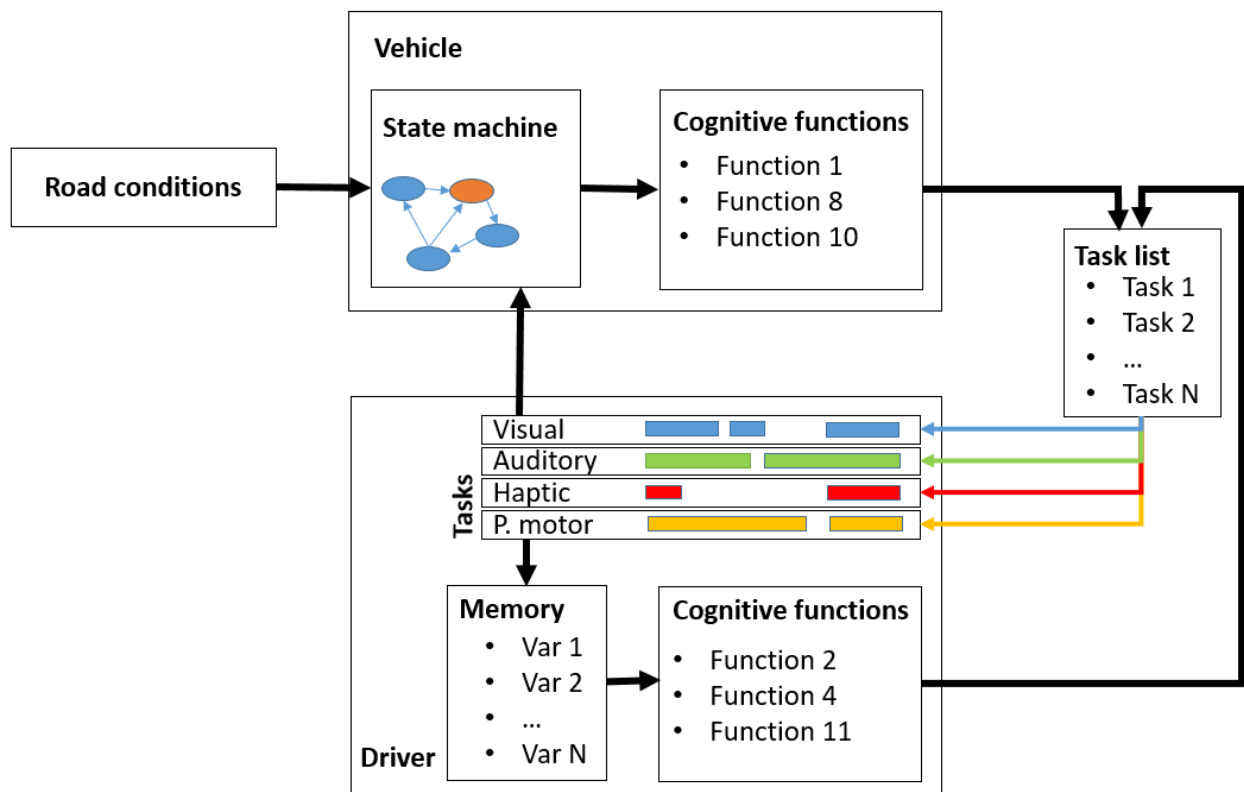


Figure K.2: Architecture of the discrete-event simulation model of HMI

process is active it requires some definite amount of limited resource. If the process is triggered but the resource is taken entirely by other processes, this process is put on the waiting list according to its priority. Processes from the waiting list with higher priority are executed first. After the process execution another event is generated, which triggers another processes, etc.

SimPy object "resource" models the limited capacity of user cognition and perception. According to multiple-resource theory (Wickens 2002), the human brain can perceive information from two sources efficiently if they require separate attentional resources. There are seven separate attentional resources used in our multi-sensory interface: visual (displays), visual peripheral (ambient lighting), auditory vocal and auditory non-vocal (speakers), haptic hands (steering wheel haptic actuator), haptic seat (seat haptic actuator) and psychomotor. We will use the simplest model of the resource conflict matrix, i.e. two tasks cannot use the same resource concurrently. The sum of active tasks' workloads in our model shall not exceed 10.

We describe each task as an excel row with fixed list of properties (Table K.1).

Table K.1: Data structure of a task

Name of property	Description
Name	Name of a task
Description	Textual description of a task
Location	Name of an interface element with which the driver should interact to accomplish a task (e.g. steering wheel, cluster, central console, etc), or, in other words, task-component allocation parameter
Cognitive workload descriptor	Textual description of cognitive task complexity
Perceptual workload descriptor	Textual description of perceptual task complexity
Perception type	Sensorial mode (Visual, Visual peripheral, Auditory Vocal, Auditory non-Vocal, Haptic hands, Haptic seat, Psychomotor)
Perceptual workload	Amount of perceptual workload required to accomplish the task (in a relative scale)
Cognitive workload	Amount of cognitive workload required to accomplish the task (in a relative scale)
Duration	Amount of time needed for task execution
Gaze Time	Time needed to change the visual focus from the road to the interface element (only for visual tasks)
Total time	Task duration plus gaze time multiplied by two
Cognitive function trigger	The name of CPF that triggers the task
Awareness parameter	The parameter in user memory that is updated after task execution
Triggers	The name of the task that shall be executed right after (if any)
Priority	Task relative importance (ordinal scale)

For example, cognitive function “check the current speed approximately every 10 seconds” generates event that triggers task “check out the speedometer” with “Inspect/Check (numerical)” perceptive descriptor and “Evaluate single aspect” cognitive descriptor. This task requires 0.2 seconds on gaze change from the road to instrument and 1 second to accomplish task. It takes 4.6 points of cognitive resources and 4.0 of visual perception resource. If there are no other active visual tasks, total cognitive workload less than 10 and total perceptual workload less than 10, task is executed during 1.4 seconds. Accomplished task generates the event which triggers update of variable “current speed” in the object representing driver’s memory.

In contrast to tasks initiated by the user (e.g. “check values in the display”, “change automation level”), tasks initiated by the machine (e.g. “send vocal message to the user”) cannot be put to the waiting list. If such a task is triggered and cognitive or perceptual resources of the driver are taken, then the task is aborted immediately.

We are using workload component scale presented in (Aldrich, Szabo, and Bierbaum 1989), to define workload values depending on task complexity (Fig. K.3).

Situation awareness is modeled as a correspondence of the values in the driver’s memory to the actual state of the car and road conditions. They may not correspond to each other if the task, which is responsible for the driver’s memory update is delayed or interrupted due to the overload. This trivial model of situation awareness is far from fidelity of the modern comprehensive models like COSMODRIVE but sufficient for our purposes.

The HMI’s functional architecture optimization problem is formulated as follows: minimize the number of perceptual and cognitive overloads and eyes-off-the road duration in a given interval of time, by optimizing the list of tasks and changing task-component allocation, wherein the situation awareness shall not go below certain limit.

K.4 Results

The sample of the discrete-event simulation output is shown on figure K.4. The top chart shows the road conditions and machine state, the bottom – the full span of the simulation sample (the highlighted area marks the zoomed fragment). The chart in the middle represents cognitive and perceptual workloads of the executed tasks. The pointer between 108 and 110 seconds highlights current task (19_04_26) and shows

Table K.2: Examples of design decisions

Change textual message “AD mode is available” on icon “AD” and reallocate it to Head-up display
Change textual message “L1 is activated” on icon “L1” on the Instrument cluster
Remove vocal message “Push on button to activate AD mode ”
Put vocal message “Drive Now” after haptic signal “TOR10” (avoid these signals to appear simultaneously)
Put non-vocal message “TOR10” after vocal message “Drive Now” (avoid these signals to appear simultaneously)

Table K.3: Averaged results of two design options

Indicator	Base	Optimized
Median eyes-off-the-road time, % of total time	11.7	10.3
Median cognitive overload, % of total time	0.87	0.57
Median perceptual overload, % of total time	0.19	0.13
Median situation awareness, average %	92.1	92.3

that at the moment the user has wrong understanding of the road conditions state and autonomous driving mode availability.

We will compare two configurations of the functional architecture – the first is proposed by the company’s internal experts in ergonomics (referred further to as base design option) and another one is manually optimized version of this base design option. The values of multiobjective function during this optimization were calculated with our proposed tool.

The examples of the design decisions recommended for the functional architecture optimization are given in table [K.2](#).

Discrete-event simulation environment uses pseudo-random generators to trigger events. Hence, to compare two functional architecture configurations we should execute a number of trials. The simulation time of each trial is 1000 minutes of virtual driving (computational time is around 30 seconds for each trial). Figures [K.5](#) and [K.6](#) show the statistics of the task execution in random trials for two architectures.

On figure [K.7](#) the results of 40 simulations (20 for the base and 20 for the optimized configuration) are shown in two-dimensional space.

K.5 Conclusions and future work

In this paper we have shown how existing task analysis approaches can be applied to the design of a modern vehicles with the adaptive level of automation. Our model designed to enable fast early-stage functional architecture optimization with respect to widely used safety metrics: cognitive and perceptual overload, eyes-off-the-road time and situation awareness.

We applied our model to the real industrial project and obtained a list of recommendations to improve the current conceptual design. Still, we can point out several directions for future work in order to improve accuracy and validity of our model:

- validation of the obtained workload and eyes-off-the-road results on the simulator with real humans;
- integration to our model more sophisticated and accurate conflict matrix (measure of a conflict between pair of perceptual resources);
- verification of the models of human driver's cognitive functions (e.g. how often driver checks the state of the machine?);
- functional architecture optimization subject to different human profiles (experienced/novice driver, open-minded/conservative);
- integration of models of human performance degradation (fatigue and drowsiness) and improvement (learning-curve) with time.

Cognitive

Descriptor	Workload value
Simple association	1.0
Select alternative	1.2
Sign/signal recognition	3.7
Evaluate single aspect	4.6
Encoding/Decoding/Recall	5.3
Evaluate several aspects	6.8

Auditory

Descriptor	Workload value
Non-vocal signal recognition	6.6
Vocal signal recognition	4.9

Haptic

Descriptor	Workload value
Detect simple signal	1.0

Visual

Descriptor	Workload value
Detect simple signal	1.0
Discriminate (Sign)	3.7
Inspect/Check (numerical)	4.0
Read (text)	5.9
Scan/Search/Monitor	7.0

Psychomotor

Descriptor	Workload value
Push the button	2.2
Switch toggle	2.2
Continuous adjustive controller	2.6
Discrete adjustive controller	5.8

Figure K.3: Workload component scale derived from (Aldrich, Szabo, and Bierbaum 1989)

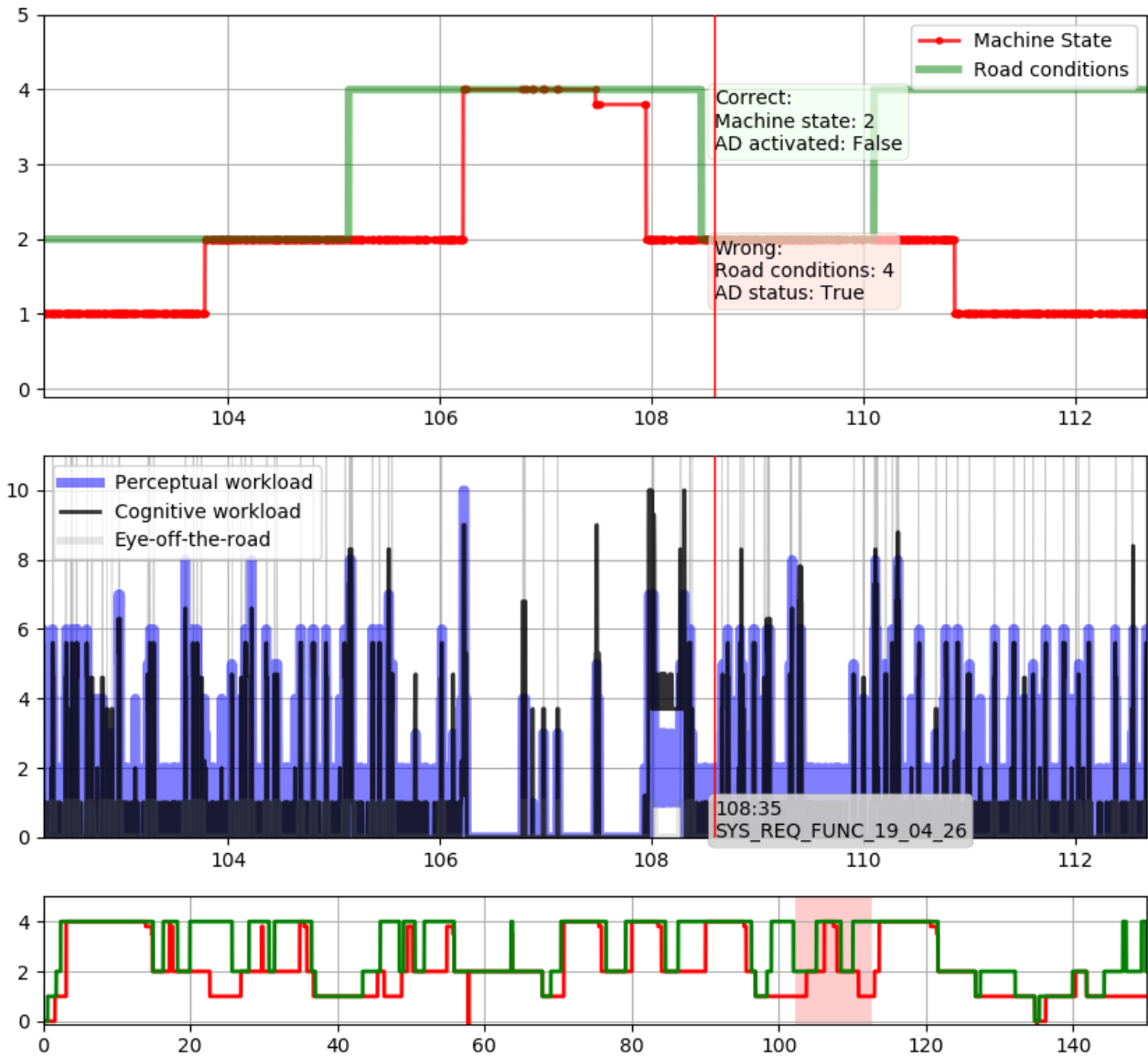


Figure K.4: Timescale output of the discrete-event simulation

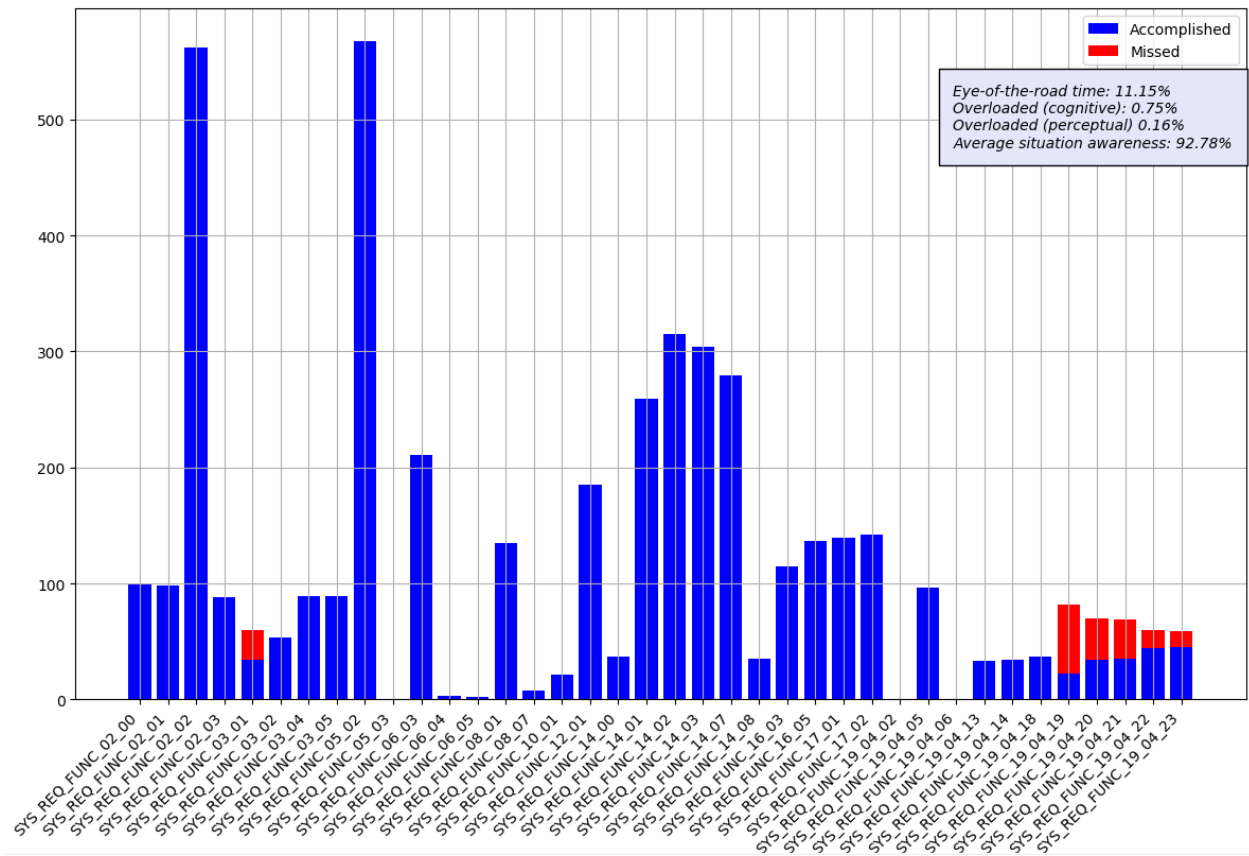


Figure K.5: Base design option simulation statistics

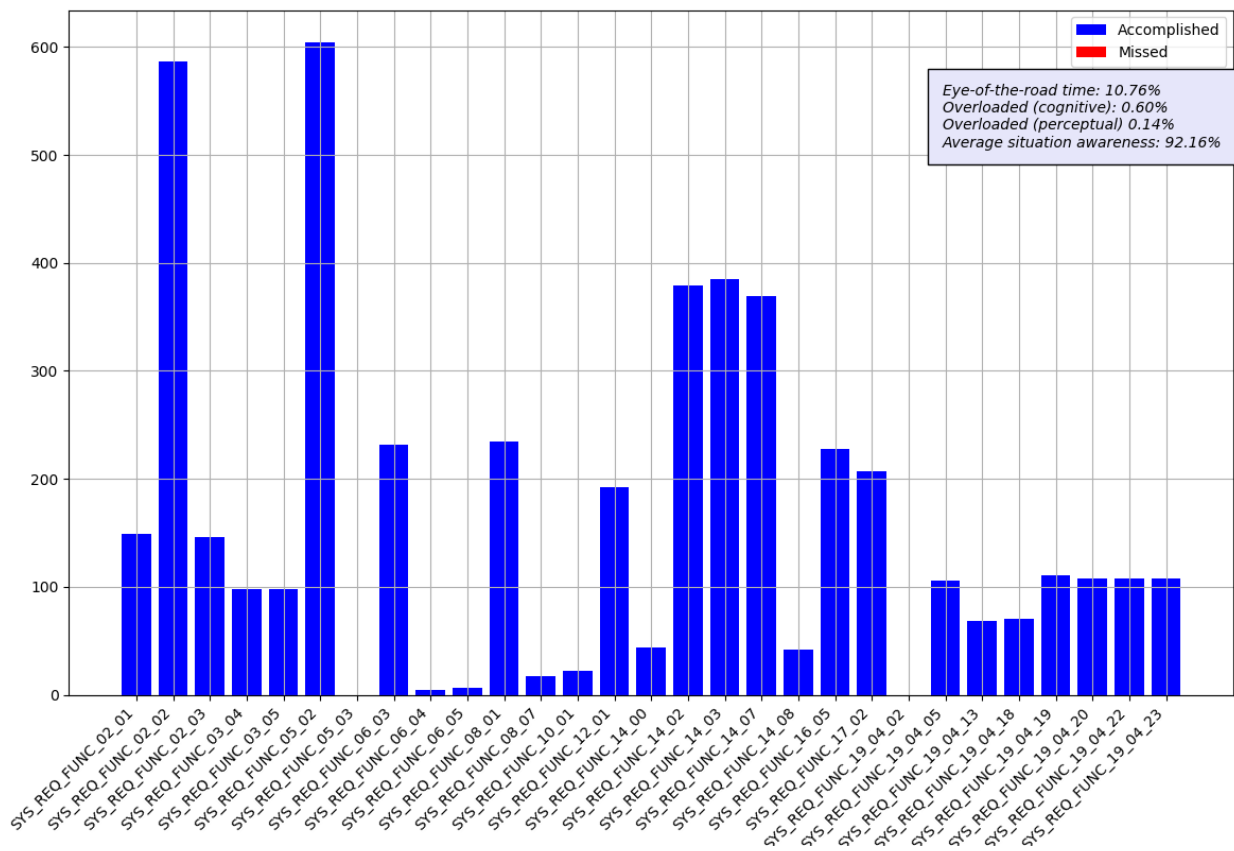


Figure K.6: Optimized design option simulation statistics

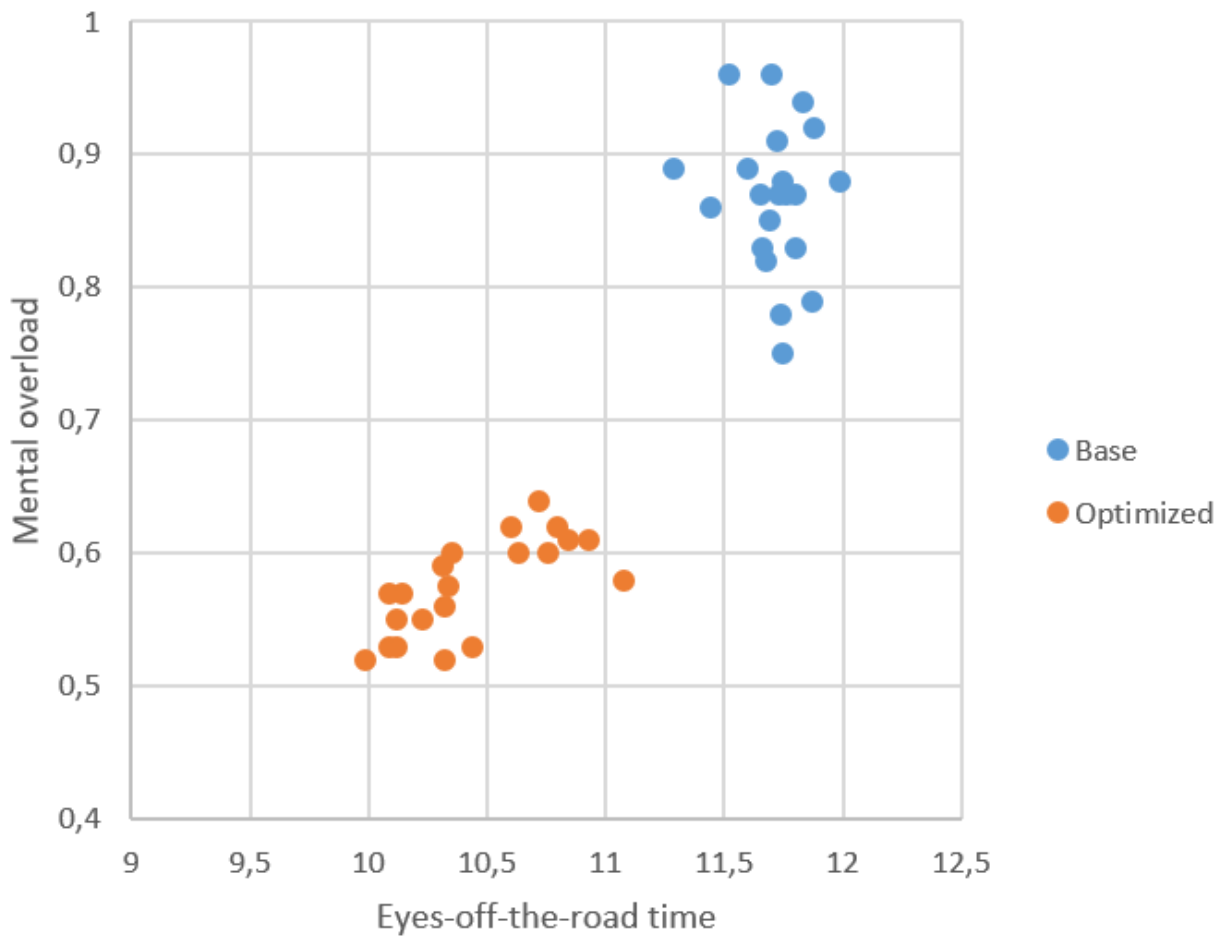


Figure K.7: The results of discrete-event simulations of the base and optimized design option