FAKULTÄT FÜR INFORMATIK

DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

SCOUP: A Framework for Sustainable Constraint-based Urban Planning

Mariana Dias Avezum

FAKULTÄT FÜR INFORMATIK
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Forschungs- und Lehreinheit 1
Angewandte Softwaretechnik

SCOUP: A Framework for Sustainable Constraint-based Urban Planning

Mariana Dias Avezum

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzende:	Prof. Gudrun J. Klinker, PhD
Prüfer der Dissertation:	1. Prof. Dr. Bernd Brügge
	2. Prof. Vivian Loftness

Die Dissertation wurde am 24.06.2020 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 19.08.2020 angenommen.

Abstract

The progress toward the Sustainable Development Goals (SDG) as defined by the United Nations has been steadily increasing, as both awareness about climate change as well as the digitalization of various cities and communities worldwide raises. While each region adapts their own initiatives and projects to achieve the goals, it becomes relevant to automatically consider to what extent these are being successful. This dissertation describes SCOUP, a framework for sustainable constraint-based urban planning to address the problem of measuring progress toward the SDGs in a comparable and automatic manner. The goal is to make it possible to compare different urban initiatives.

We investigated the problem from a cyber physical system perspective, as well as from an urban planning perspective. A cyber-physical system models cities as complex systems that integrate different data sources. With respect to urban planning, it is important to remain compliant with existing sustainability standards.

We describe a formalization that allows for the constraint-based calculation of ISO defined digitalization and sustainability standards. The formalization is based on OCL and a meta-model. The meta-model allows for development of projects that adhere to these standards. The tailorability of the meta-model was demonstrated in two prototypical systems, a water monitoring system and a search and rescue support system.

SCOUP was developed in an iterative process and validated in several systems, in particular two urban mobility initiatives. A multi-modal urban router allows users to calculate personal routes based on individual constraints. The goal was to analyze the impact of personal preferences in urban scenarios. The smart mobility initiative—implemented in the municipality of Kirchheim—allows urban planners to correlate real time sensor data with traffic simulations. The development of these systems demonstrates that SCOUP can be successfully applied to heterogeneous urban initiatives.

Zusammenfassung

Die von Vereinten Nationen definierten Sustainable Development Goals (SDG) haben das Bewusstsein für den Klimawandel als auch die Digitalisierung von Städten und Gemeinden geschärft. Zur Erreichung der SDGs implementiert jede Stadt ihre eigenen Maßnahmen und Projekte, was aber die Vergleichbarkeit zwischen Lösungen erschwert. SCOUP ist ein Framework für eine nachhaltige, regelbasierte Stadtplanung, dass die automatische Messung städtischer Indikatoren unterstützt. Ziel des Frameworks ist es, den Vergleich verschiedener städtischer Initiativen zu realisieren.

Wir untersuchen das Problem sowohl aus der Perspektive von cyber-physikalischen Systemen (CPS) als auch aus der Perspektive der Stadtplanung. Ein CPS modelliert Städte als komplexe Systeme, die unterschiedliche Datenquellen miteinander verbinden. Für die Stadtplanung ist es relevant, dass bestehende Nachhaltigkeitsstandards eingehalten werden.

In dieser Dissertation beschreiben wir eine Formalisierung, die eine regelbasierte Berechnung von ISO-definierte Digitalisierungs- und Nachhaltigkeitsstandards ermöglicht. Die Formalisierung basiert auf OCL und einem erweiterbaren Metamodell. Das Metamodell ermöglicht die Entwicklung von Projekten, die diese Standards implementieren. Die Erweiterbarkeit des Metamodells wurde in zwei prototypischen Systemen zur Wasserüberwachung sowie zur Such- und Rettungsunterstützung demonstriert.

SCOUP wurde in einem iterativen Prozess entwickelt und in mehreren Systemen validiert, insbesondere in zwei städtischen Mobilitätsprojekten. Im ersten Projekt wurde ein multimodaler städtischer Routenplaner entwickelt, der es Benutzern ermöglicht, persönliche Routen auf der Grundlage individueller Vorgaben zu berechnen. Ziel war es, die Auswirkungen persönlicher Präferenzen in städtischen Szenarien zu analysieren. Das zweite Projekt zur intelligenten Mobilität—umgesetzt in der Gemeinde Kirchheim—ermöglicht es Stadtplanern, Echtzeit-Sensordaten mit Verkehrssimulationen zu korrelieren. Die Entwicklung dieser Systeme deutet darauf hin, dass SCOUP erfolgreich auf heterogene städtische Initiativen angewendet werden kann.

Acknowledgments

First and foremost I would like to thank Prof. Bernd Brügge for the years (even before my researcher position) of constant supervision and inspiration. Thank you for teaching me project management in the iPraktikum. Thank you for recognizing value in my hyperloop master thesis, and a big thank you for the opportunity to join the Chair for Applied Software Engineering. I always thought it was the perfect name for doing interesting and crazy projects as research. I would also like to thank Prof. Vivian Loftness, one of the most inspirational women I have had the pleasure to know. Thank you for the research stay at CMU, the invaluable feedback, for making me see the value of my own research, and for teaching me more than architectural principles.

For making me fall in love with teaching, research, and innovation projects, I would like to thank the entire staff of the Chair for Applied Software Engineering. Special thanks to Dr. Jan Ole Johanßen for the detailed review, patience, and academic coaching. Special thanks are also necessary for Monika Markl and Helma Schneider for the incredible support with any and all organizational levels, or when I took a bit too long to understand german processes. To the extended family of LS1, thank you to the entire Digital Health Team for the inspirations, laughs, and coffee breaks.

For seeing the value in this research and supporting it in all the necessary steps, special thanks to Ninah Wambugu, Jens Klinker, Jakob Smretschnig and all students involved in all the thesis projects that supported this dissertation. This dissertation would also not have been possible without the extensive support of the Kirchheim Smart Mobility Initiative. Thank you Tobias Schock, René Fassbender, Martin Glas, Stephanie Oettl and the entire group not only for the data made available but also for making me appreciate the practical and legal difficulties of so many urban initiatives.

For making me fall in love with the modeling of transportation and urban systems, I would like to thank Dr. Daniel Zehe and the entire research group at TUM Create. For the academic, entrepreneurial and personal mentoring, thank you Prof. Hana Milanov.

For making me fall and forever stay in love with Munich, a huge thank you to all of my friends. I would never have done it this far without you. Special thanks to Tijana Todorovska, Stephan Müller, André Malcher and the entire MSN group, as well to Ravi Kiran Khareedi, and Frederico Mercer. Some of you I have known for over 8 years, and growing with you has kept me happy and sane throughout all this time.

For keeping me true to my roots even 10 years after moving away, thank you to my entire family in Brazil. It is not easy to get through all of this far from you, and yet we go through everything together, as if the 9,800 km between us almost didn't matter. Thank you for the eternal support.

Contents

1 Introduction **1**

1.1 Visionary Scenario . . . 2

1.2 Contributions . . . 4

1.2.1 Applications . . . 5

1.3 Research Process . . . 6

1.4 Outline . . . 8

2 Technical Foundations **9**

2.1 Systems Engineering for Urban Systems . . . 10

2.1.1 Cities as an Example for Systems . . . 10

2.1.2 Application vs Domain Engineering . . . 12

2.1.3 Agile Processes in Urban Systems . . . 14

2.2 Ubiquitous Computing . . . 15

2.2.1 Fog Computing . . . 18

2.2.2 Virtualization Architectures . . . 20

3 Investigation of Sustainable Cities **23**

3.1 Smart Sustainable Cities . . . 24

3.2 SSC System Architectures . . . 26

3.2.1 ITU Smart Sustainable City . . . 26

3.2.2 iSapiens . . . 26

3.2.3 Multi-Level Smart City . . . 27

3.2.4 Discussion . . . 27

3.3 Sustainability Standards . . . 28

3.3.1 ISO 37120: City Services Indicator . . . 28

3.3.2 ISO 37122: Smart Cities Indicators . . . 30

3.3.3 Smart Sustainable City Indicators . . . 31

3.3.4 Sustainable Urban Mobility . . . 32

3.3.5 Taxonomy of Indicators . . . 33

3.4 Urban Planning Softwares 34
3.4.1 Urban & Traffic Simulations 34
3.4.2 Geographic Information Systems 35

4 SCOUP Framework **37**
4.1 Requirements Elicitation 38
4.2 SCOUP Concepts 41
4.3 Formalizing Sustainability Indicators 43
4.4 Meta-model 50
4.5 City Input Process 53
4.6 SCOUP Overview 54

5 Urban Routing **57**
5.1 Problem Investigation 58
5.1.1 Functional Requirements 59
5.1.2 Related Work 60
5.2 Reference Application: Multi-Modal BR Router 61
5.2.1 Route Cost Function 64
5.2.2 Personal Constraints 66
5.2.3 Mode of Transportation Detection 68
5.3 SCOUP Extension 70

6 Smart Mobility **73**
6.1 Problem Investigation 74
6.2 Kirchheim Stakeholders 78
6.3 Commutify Initiative 79
6.4 Reference Application: SmartHeim2 82
6.4.1 Data Availability 83
6.4.2 Simulation 84
6.4.3 Traffic Flow Sensors 88
6.4.4 Air Quality Sensor 90
6.4.5 Indicator Analysis 92
6.5 SCOUP Extension 94

7 Validation **97**
7.1 Comparison of Urban Initiatives across Regions 98
7.2 Representation of Physical Entities 100
7.3 Data Source Integration 102
7.4 Multiple Stakeholder Support 104

7.5 Discussion 105

8 Conclusion **107**

8.1 Summary 108

8.2 Future Work 109

8.3 Epilogue 110

Appendices **113**

A Urban Initiatives for the Sustainable Development Goals **115**

B Kirchheim Indicators **119**

C IEEE License **121**

List of Abbreviations **123**

List of Figures **125**

List of Tables **127**

Bibliography **129**

Chapter 1

Introduction

1.1 Visionary Scenario 2
1.2 Contributions 4
1.2.1 Applications 5
1.3 Research Process 6
1.4 Outline 8

The worldwide urban population is increasing at a higher rate than metropolitan infrastructure can support. Across the globe, only 53% of urban residents have access to convenient public transport, while 90% of the urban population breathe polluted air. Furthermore, the United Nations (UN) Department of Economic and Social Affairs estimates that these numbers should only worsen, as 68% of the world population is expected to live in urban centers by 2050 [UNDESA18]. In order to address the problems caused by rapid urbanization and the effects of climate change, the UN ratified 17 Sustainable Development Goals (SDGs) as "the blueprint to achieve a better and more sustainable future for all" in 2015[1].

As the goals were developed to be extensive in scope, rapid urbanization has a direct or indirect impact on over half of the 17 SDGs. The goal most relevant in terms of urbanization however is SDG number 11: "Sustainable Cities and Communities" which defines the guiding principals on how to ensure that the cities and human settlements become inclusive, safe, resilient, and sustainable. Among the targets of SDG 11 are the access to safe, affordable, accessible and sustainable transportation for the entire world population by 2030, with a focus on the expansion of public transportation. Increasing access to public transportation networks would also help achieve a second target of SDG 11: reducing the adverse environmental impact of cities, especially as far as air quality and waste management are concerned.

[1] Source: `https://www.un.org/sustainabledevelopment/sustainable-development-goals/`

1.1 Visionary Scenario

One approach on how to move toward the SDG targets, while consistently measuring factual outcomes, is through the digitalization of components of the urban infrastructures.

Definition 1

Infrastructure: "The physical components of interrelated systems providing commodities and services essential to enable, sustain, or enhance societal living conditions" [Ful09].

When looking at the urban context, infrastructure encapsulates systems that manage and provide water supply, transportation, waste management, communication, electric power, as well as oil and gas systems. While some of these urban infrastructure systems have existed for as long as cities themselves, the partial digitalization of some of these systems has led to the term "Smart Cities". The term first appeared in the 1990s, but has since taken a wide range of meanings, depending on each specific project implementation [ABD15]. According to Albino *et al.*, the concept of smart cities usually encapsulates the usage of information and communication technologies to optimize its urban infrastructure. The definition of Smart City is analyzed in detail in Chapter 3.

Digital optimization can come in different shapes and forms, from the introduction of locations based traffic taxation through camera monitoring systems, to the automatic handling and separation of solid waste. To integrate different information and communication technologies into the urban infrastructure, several approaches have been presented, in particular cyber-physical systems, cloud architectures, and digital twins. Digital twins are simulations of a "comprehensive physical and functional description of a component, product or system, which includes more or less all information which could be useful in all – the current and subsequent – lifecycle phases" [BR16]. While Alam *et al.*describe a system architecture for digital twins in smart cities, the adoption of these for smart cities is not yet widespread [AE17]. One type of urban digital twins that is more widely used are traffic simulations.

Example 1: Traffic Simulations as Digital Twins

Traffic simulations allow for different scenarios evaluation and comparison of urban traffic, such as the effects of lane or intersections modifications, as well as bottleneck predictions. For example, the german state of North Rhine-Westphalia used a combination of vehicle detectors and traffic simulations to predict the traffic state in regions not covered by the detectors [Haf+04].

Digitalization of the urban infrastructure through simulations can also be used to better understand the consequences of a decision in one urban system into a different one. For example, how changes in the transportation network and behavior affect pollution in different areas. The transportation industry currently is responsible for 30% of the European Union's (EU) total CO_2 emissions, and while electric vehicles are in the rise, they still represent a minority of road vehicles [EU19]. To mitigate the effects of traffic while assuring the efficiency of the urban mobility network, governments worldwide have started partially digitizing the transportation infrastructure, which led to the term Intelligent Transportation Systems (ITS).

Definition 2

Intelligent Transportation System: a "service designed to make the movement of people or goods more efficient, safer, more economical, and less polluting. Such systems work by applying advanced and emerging technologies in information processing, communications, and electronics to surface transportation needs." [Mil12]

ITS are one source of data, that along with environmental sensors and actuators allow to better track sustainability progress. On one hand, there is a global push toward sustainable resource consumption, motivated by a renewed awareness of the consequences of climate change. At an intergovernmental organization level, this has led to the development of the SDGs, as well as international standards and indicators that aim to track the progress in this direction. Simply setting goals and tracking their progress, however, does not ultimately guarantee an improvement toward the development goals. While several projects and initiatives worldwide have contributed to social advances in these areas, their integration at a system level still lacks a standardized approach. On a technical level, modern system architectures such as layered architectures and fog computing (presented in detail in Section 2.2), allow for leveraging rapid data collection sensors with cloud computing capabilities. These characteristics are essential for any smart city initiative, and thus standardized research in this direction is starting to emerge. Furthermore, one common aspect among smart city projects, is their requirement to integrate different information sources. While this can become challenging when looking at it on a per-system-basis, the problem is easier to handle when abstracting it to the infrastructure. The following observation is derived:

Observation 1

A computational framework is needed and can be developed to strategically use growing sensor and other urban data sources to modularly and comparatively benchmark and evaluate progress in the UN SDG's.

To combine the disciplines of decentralized systems and smart cities, while ensuring cities' compliance to sustainability indicators, the following dissertation introduces **SCOUP**: A Framework for **S**ustainable **Co**nstraint-based **U**rban **P**lanning. SCOUP aims to improve city intelligence by visualizing constraint-compliance in order to promote integration between sustainability indicators and decentralized architectures, including reference implementations of how these can be realized in real life.

1.2 Contributions

This dissertations presents a set of modeling abstractions that can be used to monitor and control urban information flow toward sustainable smart cities under different scenarios. These should be developed as broad as possible and include aspects from both technical and urban infrastructure to address the different domains that a smart city must cover.

Based on the urban planning and decentralized systems foundations, the SCOUP framework is derived, which provides a common technical language for urban planners and system engineers. This framework is applied to two specific projects, where different reference applications were implemented. These reference applications help evaluate to what extent they addressed the challenges of urban planners and SCOUP's requirements. To further address some of the difficulties encountered by urban planners, a set of overarching tools is developed, to facilitate integration of different urban infrastructure areas. The core contributions of this dissertation are:

1. **Indicator Formalization** Development of a taxonomy for sustainability indicators using Object Constraint Language (OCL). A representation for these allows for a formal technical definition as to how sustainability indicators can be converted into technical system constraints. Examples are presented of how to map these into sustainability indicators, and the importance of having a consistent indicator calculation methodology.

2. **Meta-Model** developed according to the Meta Object Foundation (MOF) [MOF 2.5.1] for urban planning projects, which supports data and constraint integration, as well as tracking of urban initiatives and citizen's preferences over time. Major city attributes such as population and climate conditions are integrated into the model, to facilitate the relation between these and the results of urban initiatives. When completely applied, this meta-model allows for the continuous monitoring of sustainability indicators, as well as how urban initiatives impact these.

3. **Urban Development Software Components** that support decision making for city planning. These tools include extensions to traffic simulations that allow for street modifications, correlation of traffic and air pollution, tools to integrate unknown data sources, and how to consider citizens preferences in urban decisions. Visualization dashboards for data and indicator display are also provided throughout the reference applications. It is relevant to note that several of these tools were developed by collaborations with bachelor and master thesis, as well as some industry partners, and are thus referenced in the dissertation. While several of these components were developed specifically for the the reference applications described below, their development was done generically to be applicable to other projects. Several of these are made available open-source.

Furthermore, two reference applications are presented in detail, and reference systems for them are described. The reference implementations of the framework are used to demonstrate the applicability of SCOUP to different urban domains by each focusing on relevant SCOUP components. Furthermore, the extent the presented framework reduces the challenges discovered during the literature review and primary data collection phase is evaluated. The contributions of this dissertation is to make urban management projects more comparable worldwide, and to keep track of important sustainability indicators, in an attempt to facilitate integration and prioritization of sustainability efforts.

1.2.1 Applications

The following is an overview of the reference applications developed, how they relate to the functionalities of SCOUP, and which indicator is relevant for each.

1. **Multi-modal Transportation BR Router:** The system and correspondent application provides traveler and commuters with alternative routing options, based on multi-modal and personal constraints. The personalization of the routes allow for relating how constraints can be interpreted in a user level as preferences. By integrating different modes of transportation, the per capita usage of sustainable modes of transportation should increase, thus reducing the percentage of travelers dependent on cars.

2. **Traffic flow and air quality optimization in Kirchheim:** The small community of 13000 citizens east of Munich has seen a big population growth in the last years [BLS18]. This lead to an increased usage of the urban and road infrastructure, which now should be optimized. Through continuous monitoring of traffic flows with several installed sensors, and real time road recommendations,

the system allows for traffic flow and air quality pollution data to be gathered and analyzed in real-time. This data is further aggregated to static municipal sources and geographic data in order to measure several of the indicators proposed by the International Organization for Standardization (ISO), especially as far as transportation and environment are concerned.

1.3 Research Process

Software Engineering has long been recognized as an iterative process [BD10], and it's integration with design science is slowly starting to be researched [Eng+20]. Design Science is an approach for the conduction of applied research that allows for the incremental experimentation and evaluation of investigated problems and possible treatments for these. Wieringa formalizes this process as a design cycle composed of four phases: (1) problem investigation, (2) treatment design, (3) treatment validation and (4) treatment implementation [Wie14]. At the end of each cycle a new treatment can then start by evaluating the previous implementation and investigating further problems related to this. It is clear to see how these phases relate to the software engineering development process. Problem investigation is the state of the art analysis which describe the problems with the current systems. Treatment design relates to system design in the sense that it is the transition between general requirements and specific solution system considerations. Agile methodologies in software engineering also separate the development process into cycles, where the beginning of the each phase is a review of to what extent the previously defined requirements were achieved.

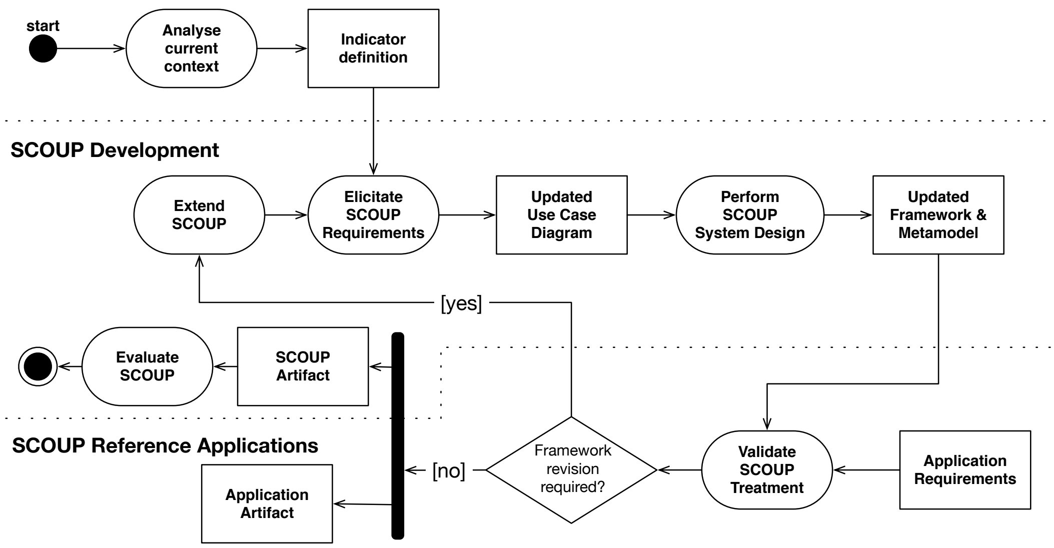


Figure 1.1: Research process used in this dissertation – Unified Modeling Language (UML) Activity Diagram.

The research process of this dissertation can be visualized in Figure 1.1, and its iterative process is based on the work of Wieringa [Wie14]. As mentioned, the main goal of this dissertation is to develop a computational framework to strategically use growing sensor and other urban data sources to modularly and comparatively benchmark and evaluate progress in the UN SDGs. To aid in the investigation of the effectiveness of the SCOUP framework, reference applications are presented to help access each of the aspects of the framework.

As the main contribution artifact is the SCOUP framework itself, an iteration of its development status is done through the implementation of two reference applications. Each application allows for SCOUP to be continuously extended, and then serve as a basis for the following work. Since SCOUP aims to be applicable to several of the UN's SDGs and thus must manage the relation between these.

Observation 2

This framework must anticipate conflicts and synergies between SDGs and how initiatives taken toward this direction may effect other areas of urban planning.

It is important to note that each of these applications were developed in cooperation with external stakeholders, and thus have requirements and limitations of their own. These requirements are then taken into consideration when going through the smaller instrumentation phase of the SCOUP development cycle. The first of these references application was developed through the iPraktikum [BKA15] and allows for commuters to insert their personal constraints into their routes. This supports the analysis of constraint aspect of SCOUP, which are then integrated into the framework. A second iteration of the SCOUP design cycle is then initiated with the Kirchheim project, which regards constraints from the air quality context and their impact. While the complexity of the Kirchheim project allows SCOUP to be analyzed against several stakeholders expectations and viewpoints to the system, the fact that these different stakeholders all have different data sources in different formats – that partially represent similar characteristics – helps validate both the applicability of the meta-model, as well as the indicator formalization. The meta-model treatment applied to the Kirchheim project also allows for the separation of simulated aspects of the community compared against information gathered in real time.

To validate the results achieved in both projects and in SCOUP, the calculated indicators are finally compared against an urban initiative that took place in Pittsburgh, independently from our research. The fact that the calculation of the same indicator is possible although there was no involvement in the system's design helps analyze and test the inter-city comparability of SCOUP through the formalized indicators.

1.4 Outline

After describing the motivation and core contributions in the Introduction, this dissertation follows by first analyzing the technical foundations of systems engineering and ubiquitous computing on Chapter 2. Development processes meant for usage with various software systems are described, as well as the main architecture styles that enable real time connection between several data sources.

Chapter 3 focuses on the problem investigation from a smart city perspective. For this, Smart Sustainable Cities are initially defined, and then relevant sustainability standards developed for the UN's SDGs are presented. These indicators are formalized into a generic taxonomy and a classification system for these is presented.

Based on these foundations, Chapter 4 formalizes SCOUP. The requirements derived from the previous analysis are presented, and the initial meta-model is derived from it. A first iteration of SCOUP's treatment design is initialized, and an outlook for the entire framework is described.

Chapter 5 presents the first instrumentation of SCOUP, and a multi modal routing system is used for this. The requirements for this system are presented, and the reference application demonstrates how SCOUP was used to facilitate this development. Based on a discussion of how well the developed application addresses it's initial requirements, SCOUP is extended to include some of the aspects in the system.

A next iteration of SCOUP's development cycle is described in Chapter 6 through the Kirchheim project. As with the previous chapter, the requirements for this system are initially presented. An overview is provided of all the involved stakeholders, and the reference application is presented, which contains some of SCOUP's urban development software components. SCOUP is then again extended based on how well the presented system addressed the initial requirements.

Chapter 7 analyzes the final version of SCOUP against its initial requirements and attempts to validate the entire framework. Specifically the comparison of urban initiatives, tailorability of the meta-model, data source integration and multiple stakeholder support are analyzed critically.

The conclusion in Chapter 8 summarizes the core contributions of this dissertation and considers their threats to validity. An outlook is presented for future work in the research, and a discussion is given about the usage and relevance of SCOUP.

This dissertation also includes three appendices. Appendix A presents a list of possible urban initiatives that could be implemented to promote the targets of the SDGs. Appendix B presents the list of indicators and data sources that were used for the Kirchheim mobility initiative. Last but not least, Appendix C displays the permission grant from IEEE of a publication this dissertation is based on.

Chapter 2

Technical Foundations

2.1 Systems Engineering for Urban Systems 10
  2.1.1 Cities as an Example for Systems 10
  2.1.2 Application vs Domain Engineering 12
  2.1.3 Agile Processes in Urban Systems 14
2.2 Ubiquitous Computing 15
  2.2.1 Fog Computing 18
  2.2.2 Virtualization Architectures 20

This dissertation models cities as a complex system, with several components, including both hardware, software, natural, man-made, modern and older elements to it. This chapter focuses on the technical aspects this work is built on, as far as both systems engineering and ubiquitous computing are concerned. We start by analyzing the components of System Engineering in Section 2.1. To organize future analysis, a discussion is presented as how urban planning has similarities with systems engineering in Section 2.1.1. We then consider how software product line engineering allows for the development of systems with a high variability in Section 2.1.2 and how agile methodologies have been applied to areas outside of software systems. From the systems engineering research field, one specific area that becomes relevant is ubiquitous computing, which is discussed in Section 2.2. System architectures common to these systems are presented and a discussion is given as to what extent they can be applied to the urban design domain. Specifically, a focus is given to fog computing and virtualization techniques.

2.1 Systems Engineering for Urban Systems

Systems engineering is is the study of how to develop and organize systems. De Cañeta *et al.* define four axis to classify a system [DCGGM11]: (1) Static vs Dynamic, depending on how the output of the current state correlates to the previous output; (2) Stochastic vs Deterministic based on depending on how random events influence or not the performance of the system; (3) Continuous vs Discrete vs Event based depending on how the processes inside of the system are computed; (4) Closed or Open-loop systems based on wether their control systems can use reference values to calculate and consider errors. The authors define the term Systems Engineering as the following:

Definition 3

Systems Engineering is a field of study "that aims to model, analyze, describe, and simulate physical systems." It is the practice of developing or engineering a "collection of interrelated elements pursing a particular objective." [DCGGM11]

Section 2.1.1 maps cities as complex systems using the framework provided by De Cañeta *et al.* [DCGGM11]. In the following Section 2.1.2 a further aspect of system engineering which can be applied to the urban context is described, the differentiation between application and domain engineering. Furthermore, in Section 2.1.3 a discussion is presented of the challenges of applying software development processes to the development of urban initiatives in cities.

2.1.1 Cities as an Example for Systems

When a city is considered a system, it becomes clear that the concept of systems engineering can be applied to the urban environment, in the sense that a city has a considerable collection of interrelated infrastructure components such as education, transport, health, recreation, etc. In an ideal scenario the improvement of it's citizens quality of life, and achieving the goals set forward by the SDGs can be seen as the cities' objective, which should be managed by a central government. The relation between the systems engineering and urban planning disciplines is made clear when analyzing the definition of systems engineering.

While De Cañete *et al.* recognize political systems as an example for the systems concept, they argue that the lack of physical and mathematical laws controlling the behavior of political systems means that these should not be governed by the systems engineering domain [DCGGM11]. When adding the lens of sustainability and the constraint-based design that communication technology allow, these limitations can be partially overcome.

De Cañete *et al.* go on to define that a system contains inputs and that these get regulated by controllers, to define and analyze outputs, which should ideally conform to the system's requirements [DCGGM11]. This can be applied to the context of a city in the sense that the required inputs can be seen as the economical, social and environmental context where it is located. These form the sustainability pillars, as described by Brundtland [WCED87]. These pillars are revisited later in Section 3.2, but for now it suffices to note that these pillars help us understand the variables that affect a city's system. As far as control mechanism for the urban development context and their required adjustment goes, cities rely on government policy making to regulate and control all aspects of urban life, which should ideally be reflected in some metric output by the system.

According to the definitions presented by De Cañete *et al.* in [DCGGM11] cities can be furthermore seen as dynamic systems, as their internal state and control mechanism depend on previous outputs and the existing registry of these. Cities also present stochastic behaviors as random variables (such as an election result) that affect the control mechanisms, and although several of the variables and control mechanisms related to a city present continuous magnitudes, measuring the performance of any government policy is often only feasible in a discrete time frame, such as reporting on results per year. The last item on the system classification framework presented in [DCGGM11] is the closed-loop-configuration, which also applies to cities as the result of a performance indicator can directly feedback and influence future events. This mapping of cities to systems leads us to the following observation.

Observation 3

As with systems, no two cities in the world are the same, and yet they generally suffer from comparable problems while aiming to progress toward the SDGs.

2.1.2 Application vs Domain Engineering

In the software engineering world, a common approach to the challenge of having to replicate several similar software systems is presented by Pohl in [PBDL05], where a product line process is presented for requirement engineering. Software engineering is a subtype of systems engineering, that focuses on the development of systems mostly or entirely dominated by software components.

Pohl presents a framework that defines two different processes when handling software engineering that present partially common requirements, one for domain engineering, and one for application engineering [PBDL05].

Definition 4

The **Software Product Line Engineering** process is composed of domain engineering and application engineering. Domain engineering identifies the commonalities and variabilities required in different systems. Application engineering, on the other hand, describes the specific applications or projects that implement the necessary variability of the product line [PBDL05].

We adapt the usage of Software Product Line Engineering by using cities as the common domain for the products developed with the framework presented in this dissertation. The differentiation between domain engineering and application engineering is relevant as we plan to make use of the common domain knowledge in the field of urban planning and design, yet the reference applications – presented later in the thesis – greatly varies according to each specific application. In order to handle this variability, Pohl introduces variability in all domain engineering artifacts, ranging from requirements, to architecture, components and test cases [PBDL05]. While scope limitations keep us from formally following every aspect described in Pohl's framework, the concept of dividing the development into domain and application engineering processes are nonetheless applied, and thus we describe each in more detail.

Domain engineering focuses on identifying and realizing the commonality and variability aspects of software product line engineering [PBDL05]. Critically, it does not only identify which components of a product line are similar, but also derives a procedure to identify, realize and track variable traits. Domain engineering is divided into requirements engineering, domain design, domain realization, and finally testing [PBDL05]. Each of these phases deals with the variability aspects individually and makes a clear distinction what part of the software products are common or different in each product derived from the software line. Domain engineering describes the basis for the product line framework, and what are the aspects that each product must conform to, in order to be a product from the product line.

The application engineering process goes through the same steps of requirements, design, realization and testing, but focuses on specific products, and not in the generalization. At the end of the application engineering process, a new product is developed that conforms both to the general guidelines laid forward in the domain engineering, and also to the specific requirements that apply only to that one variable product [PBDL05]. Application engineering can almost be seen as the customizable product, or rather realizing the variations, of the software product line.

Meta Modeling

A further approach of how to derive similar components inside the same domain is presented by the the Meta Object Facility (MOF). In it's Core Specification document from the Object Management Group, the relation between classifier and instance is explained, and how the ability to navigate from a specific instance to a more generic classifier can be applied throughout any number of layers [MOF 2.5.1]. The group proposes a structure of four meta layers for classifiers: The MOF meta-meta-model M3, the domain specific meta-model M2 with domain relevant classifiers known as stereotypes, the system model that defines the classes inside of a system in M1, and last but not least the concrete instances in run-time M0 [Sch17]. Table 2.1 describes this hierarchical relation. By defining the relevant domain as urban planning, SCOUP investigates and determines the classifiers common to this domain, which should be made available for all M1 systems implementing the framework.

M3	Meta-meta-model	Classifier (MOF 2.5.1)
M2	Meta-model	Stereotype (SCOUP extension of UML)
M1	System model	Class of urban domain
M0	Concrete instance	Object in urban domain

Table 2.1: Urban domain application to the MOF abstraction layers through SCOUP.

The idea behind the usage of the MOF and UML specifications is to allow for the stereotypes defined in SCOUP to serve as a domain engineering guideline for future applications interested in applying its concepts. SCOUP uses the stereotype concept described in UML 2.5.1 and extends the UML language through this usage. Chapter 4 defines these stereotypes.

Definition 5

Stereotypes is a type of modeling element that extends the semantics of a model. Stereotypes must be based on existing types or classes in the meta-model and are one of the extensibility mechanisms in UML 2.5.1, including constraints and tagged values [MOF 1.4].

2.1.3 Agile Processes in Urban Systems

While the Software Product Line Engineering process was designed for software processes, the context of urban planning presents some similarities with hardware development, in the sense that changes can be complicated to implement and require a process to track their effects throughout the development process. We now discuss the challenges of hardware software co-design, and how these apply to an urban context.

Agile methodologies have long been in use for the development of software systems [BD10]. The fact that hardware systems are harder to change, however, means that the applicability of agile methodologies to hardware-software hybrid systems is limited [Ave17]. A hybrid system is one composed of both software and non-software components. Thus, when imagining a city as a system, it is useful to map urban environments as hybrid systems. As digitalization continues to increase the software capabilities in different domains, the faster iteration speed made possible by this type of development process becomes evermore desired outside of the software environment. Hybrid systems in the field of mechatronics are one example of how software has increased in importance.

Initially the field of mechatronics was composed mainly of mechanical (often analog) systems, and some electrical components that perhaps had some small software component. Over time, the software components have not only evolved as an additional discipline (separate to electrical engineering), but have also gradually gained importance and capabilities in these hybrid systems. In order to address the difficulties of agile development in non-software contexts, Hostettler *et al.* propose that innovation teams developing hybrid systems iterate through desirability, feasibility and viability cycles, each with their own plan-do-check-act phases [HBLK17]. In order to do so, the authors propose that each cycle can be executed on individually, as long as they are synchronized at relevant milestones.

When Schwaber introduced the scrum development process in 1997, the main goal was to allow for faster iteration cycles to increase the feedback loop of the software development process [Sch97]. By having continuously delivered components be systematically released to customers, scrum aims to reduce the feedback iterations. A similar logic of component synchronization is used in an implementation of agile methodologies known as Scrum-of-Scrums. This approach was initially developed for large scale software projects, and it suggests that systems be divided into sub-components, each with it's own scrum team that iterate at it's own speed and synchronize with the big system at specific milestones [FS13]. While the concept has been successfully applied to prototype hardware projects [BA18], its usage in high scale commercial productions, as well as urban planning, is still limited.

Urban planning projects present similar limitations to hardware development when applying agile methods, in the sense that feedback iterations can take a long time (as construction projects take long to implement), and late change can have expensive and prohibitive consequences to the entire system. While the software domain solves this by decoupling components as much as possible, this is harder to apply to urban design, as the coupling of city systems is inherently determined by its human citizens. For example, if the government were to introduce new regulations limiting the fuel emissions by passenger cars, it would be forced to do so in a gradual approach, as drastic changes prohibiting citizens to use their expensively-bought vehicles would likely face great aversion from the population.

One valuable source of information for the necessary feedback loops between deployed systems and urban impact can be sensors installed throughout the city, which can allow for faster evaluations of the urban environment. In order to discuss the capabilities and state of the art architectures common to these, we now look into ubiquitous computing applied to the urban environment.

2.2 Ubiquitous Computing

The term Ubiquitous computing was coined by Mark Weiser and encompasses a paradigm where computing integrates information into everyday, physical objects [Wei93]. While describing a world where computational information is as seamless – or ubiquitous – as scrap paper and almost invisible to the end user, Weiser already anticipated some of the problems now addressed in the 21st century, such as the high bandwidth required from wireless networks when a large amount of devices are connected to it [Wei93].

Recently the problem of high necessary bandwidth is now being addressed by the introduction of 5G, which aims to make not only web browsing and virtual reality experiences a reality, but also enable the fast transmission of high quality video data [Qi+16]. These advances have enabled the introduction of the Internet of Things (IoT), which further adds real time capabilities and standard protocols to the high information exchange between connected ubiquitous devices.

While IoT has received much attention due to the diffusion of sensors and high bandwidth capabilities, the concept of connecting everyday objects to the internet is not new. The first non-computer device claimed to be connected to the internet was hooked up in 1982, when a group of researchers at the Carnegie Mellon University wanted to know whether the coke machine available on the building had (a) any cokes at all, and (b) if the cokes were cold. They connected the machine to the local network and could thus check the state before going up to get the coke[1] [Orn16]. While several

[1] Original webpage: `http://www.cs.cyber.edu/~coke/`

definitions of IoT exist, they generally encompass the integration of everyday objects into the internet.

Definition 6

The **Internet of Things** is "a world where physical objects are seamlessly integrated into the information network, and where the physical objects can become active participants in business processes. Services are available to interact with these 'smart objects' over the Internet, query their state and any information associated with them, taking into account security and privacy issues [HKS09]"

A further segment of Ubiquitous computing can be found in Cyber physical systems (CPS), which were first described in 2005 [Sch17]. More recently these have been described as "physical and engineered systems whose operations are monitored, coordinated, controlled and integrated by a computing and communication core" [RLSS10]. While the differentiation between CPS, IoT and Ubiquitous computing can be hard to make, CPS are composed of both physical (hardware) and digital (software or cyber) components. The development of CPS over time can be mapped to hybrid systems (as described in Section 2.1.3). CPS often requires close collaboration between both the software and physical components. In CPS, the software regulating the physical object has a more active role than data or internet connectivity management and can even regulate the object's complete behavior, as well as its relation with different systems. In this sense, both CPS and IoT are types of Ubiquitous Computing technologies, and they all contain some physical object related to them. This relation can be seen in Figure 2.1. Nonetheless, this dissertation uses the terms CPS and IoT interchangeably, as the systems developed by Smart Cities generally entail both aspects of internet connectivity and cyber physical components.

The vast amount of CPS systems and applications that can be mapped to their use case have motivated meta-models to be created in order to represent these systems, such as the one by Scheuermann [Sch17]. The meta-model described by Scheuermann and partially shown in Figure 2.2 illustrates how Smart Environments can be gener-

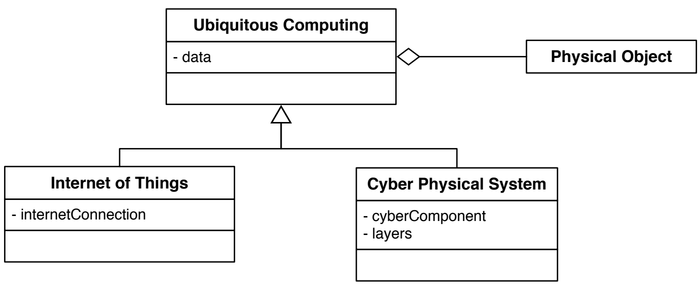


Figure 2.1: Relationship between Ubiquitous Computing, Cyber Physical Systems, and Internet of Things (UML Class Diagram).

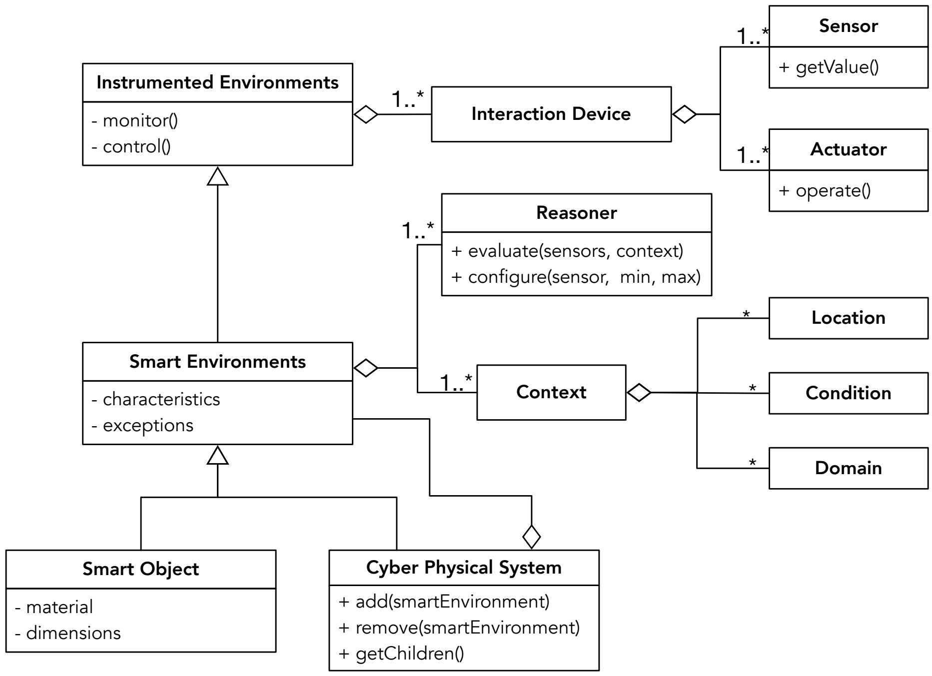


Figure 2.2: Cyber Physical Systems meta-model (UML Class Diagram). Adapted from Scheuermann [Sch17].

alized as entities that contain not only sensors and actuators, but also evaluate the data provided by these, according to a given context.

Scheuermann defines a smart environment as containing a context that includes domain knowledge, location information, and quality conditions of the environment. This context informations helps the reasoner makes informed evaluations about the environment. Furthermore, Scheuermann describes how smart objects can be combined through CPS to form entire Smart Environments, and how the context and information added by each of the smart objects can be passed to the different layers of the connected CPS [Sch17]. As can be seen in the Figure 2.2, these layers go on to form a composite pattern in which the CPS represents the composite component described by Gamma *et al.* [Gam+95].

An example of a Smart Object, or a component of a CPS, include gloves with integrated sensors that help execute a manufacturing function, as described by Scheuermann *et al.* [SSBV16]. Further examples can be found in the architecture and urban planning field, where buildings can be found that regulate temperature and energy requirements through sensors and actuators, as well as that try to balance individual and collective comfort. Since 1997 Hartkopf *et al.* have studied the advances toward electronically enhanced offices for the different functions of the workplace and which digital tools may aid in their execution [Har+97].

These buildings are a good example of CPS as they also present the layered aspect often necessary for these systems. A building can be seen as a collection of rooms, which are then a collection of objects, which may include or contain different sensors and actuators each. The requirements presented by the integration of these layers and components is the origin of the concept of Fog Computing, which is now presented in more detail.

2.2.1 Fog Computing

Fog Computing is an architectural paradigm that describes how to organize different sensors and actuators commonly found in IoT projects. It extends the concepts of cloud computing to allow lower latency between nodes, location distribution of the components, as well as a larger number of connected nodes [BMZA12].

Definition 7

Edge devices are sensors and actuators that connect to any ubiquitous computing system. The name relates to the fact that these devices are the outer most (thus, edge) components of a system, and usually interface with a single component.

To handle the scalability and reliability requirements mentioned, the fog paradigm creates an additional layer between cloud servers and edge devices, called the fog layer. The main purpose of this fog layer is to provide fast, physically close, and reliable computational resources to the edge devices. While the edge devices may or may not have some computational capabilities themselves, these are generally limited to processing information in their immediate neighborhood and captured by themselves, as the lack of connectivity to other devices limit their information acquisition. This information can, however, be forwarded to fog nodes, who can thus serve as an initial processing phase for the different edge devices present in a smart environment. A fog node can be anything from a simple Raspberry Pi or Arduino, to a more advanced computer, as long as it is physically close to the edge devices in order to ensure latency and connection quality. As described by Bonomi *et al.* it is relevant to note that while some edge devices may require mobile capabilities (such as sensors in a car), the fog nodes are usually static (such as a smart traffic light) [BMZA12].

Seitz describes Fog Computing as a multiple inheritance between centralized and decentralized architectures, and offers an architectural pattern to allow for leveraging problems between real-time access and synchronization of data between multiple edge devices [Sei19]. Fogxy, the architectural pattern presented by Seitz, further recognizes that while cloud computing paradigms does supports scalability and synchronization quality attributes, it is limited as far as real time capabilities are concerned and often requires expensive hardware [Sei19].

In order to solve this Seitz introduces the architectural style of the Fogxy Proxy, a combination of the proxy pattern with the concepts of Fog Computing that solves the problem of lack of availability when the cloud node is hard to reach [STB18]. In this architectural style, the fog node becomes the *Fogxy Proxy* component and allows the data from the edge devices to be cached before being forwarded to the cloud. This reduces the network load as requests from the edge devices only need to reach the Fogxy Proxy – and not the cloud server. Furthermore, the introduction of the proxy pattern allows for availability of the Fogxy node to the edge devices regardless of the state of the remote cloud. In order to communicate between the edge devices and the Fogxy node, the observer pattern [Gam+95] is used, where a Fogxy component can be an edge device or fog node, and publishes updates to any registered environments.

Further work on the formalization of fog architectures was done by Cicirelli *et al.* where the authors have proposed a meta-model as how to unify different edge devices in a smart environment context in [Cic+18]. Their work describes how relevant data for a smart environment can come from different types of data sources, and how several functionalities can be combined into composed functionalities of a smart environment. This leads to a composition similar to the one presented by Scheuermann, but which is instead focused on the functionalities of the system, rather than on the information sources.

Cicirelli *et al.* further formalize a stereotype for DataSources, which also include some of the attributes that Scheuermann introduced into Context [Cic+18; SSBV16]. According to Cicirelli *et al.* a DataSource is composed of Actors, a Location, different Smart Functionalities, a Type (which may in turn be composed of several types), an optional Timestamp, can be either volatile or persistent, and can be either event based, stream based, or have periodic updates. We have integrated De Cañeta's system terminology to the the taxonomy presented by Cicirelli *et al.* which can be seen in Figure 2.3. It is relevant to note, however, that Cicirelli *et al.* do not consider the difficulties of agile hardware development in their analysis of edge devices.

Observation 4

While the advances in IoT and CPS have allowed for the integration of several physical and digital components to highly integrated networks, the physical size, complexity, and testing capabilities of some hardware toughen the agile development of these. Fog computing and data composition allow for a step in the direction of integrating the data produced by these.

So that these aspects can be more easily visualized and studied, digitalization of these is sometimes required. An investigation of how virtualization architectures can assist on this is now presented.

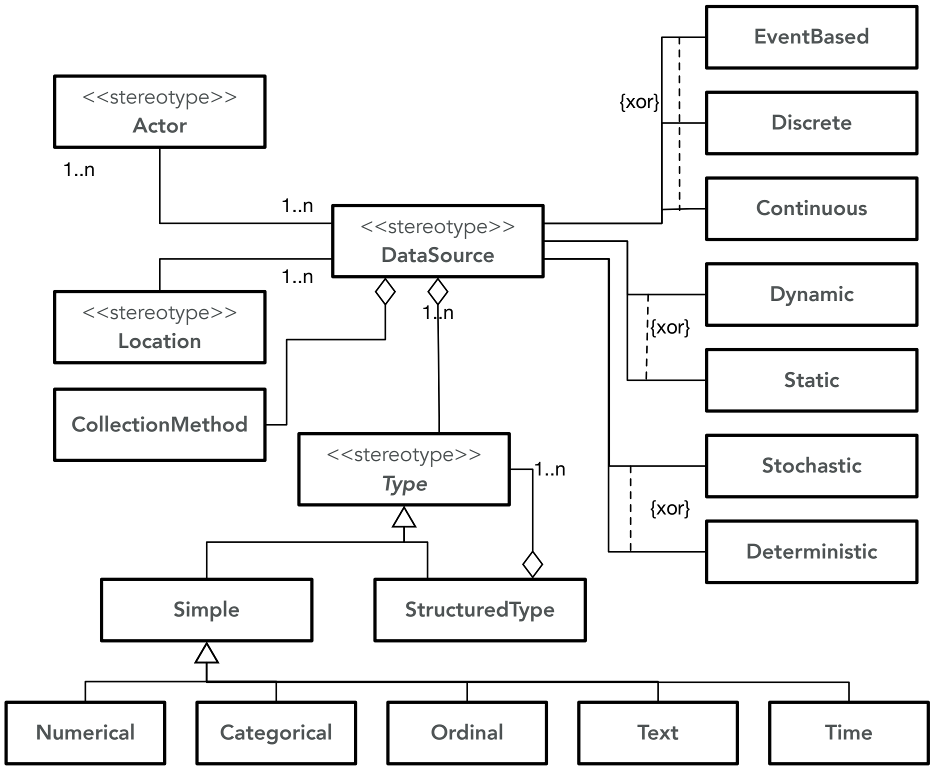


Figure 2.3: DataSource Taxonomy (UML Class Diagram). Adapted taxonomy from Cicirelli *et al.* to include terminology from De Cañeta *et al.* [Cic+18; DCGGM11].

2.2.2 Virtualization Architectures

In addition to analyzing the communication between different sensors and actuators, cities often rely on simulating parts of their infrastructure in order to predict how a given scenario would affect the existing community. As with other non-software systems, digital twins provide a powerful tool for these.

One attempt to define virtual equivalents to physical objects was done by Bauer *et al.* [Bau+13], where the concept of Augmented Entity is introduced, defining an abstraction that is composed of both a physical object (that can contain sensors or not) and a virtual entity counterpart. This virtual entity can then be used by a computer to carry simulations of the physical entity.

Definition 8

A **virtual entity** is either an active digital artifact, or a passive one. A digital artifact can include anything from a computer file, to a more complex active simulation. The main difference between the active and passive alternative is the possibility to interact with active artifacts through a provided service [Bau+13].

The concept of physical entity, however, is less clearly defined in the work by Bauer *et al.* [Bau+13], where it is specifically mentioned that these can not be clearly assigned to hardware or software components. According to the architecture presented by Bauer *et al.* devices such as Actuators, Tags, or Sensors are Physical Entities, but can also be attached to these [Bau+13]. While the relation between Devices and Physical Entities of Bauer *et al.* [Bau+13] resembles the composition between Smart Objects and CPS found in Scheuermann [Sch17], in Bauer *et al.* this is done less successfully as it effectively introduces a double composition, between Devices and itself, as well as between Physical Entities and itself, thus breaking the composite pattern as presented by Gamma *et al.* [Gam+95].

A further approach as how to formalize the relation between physical objects and their digital counterparts is presented by Ganslmeier, where the virtualization of physical components is described as a mean to minimize risk and costs in the development process of the automotive industry [Gan15]. While that work is partially limited to the domain of the automative industry, some the formalizations described there can be further applied to Smart Cities. As in [Bau+13], Ganslmeier also decomposes each entity that should be simulated into a virtual and a real (or physical) element, and uses common protocols for each of these. Differently than [Bau+13], however, Ganslmeier also depicts a virtual representation for human actors of the system, which further allows for the automatic testing of possible human interactions with the system. An overview of the framework presented in this work can be found in Figure 2.4. As the VU Framework developed in [Gan15] focuses specifically on the automotive development process, Ganslmeier broke down the framework into the main aspects of the driving task, such as driver, vehicle, and environment elements. While these are considered too domain specific to be generalized, it is interesting to note how the Configuration attribute of vehicle and environment represent the same concept as the Context class described by Scheuermann in [Sch17], basically the conditions that each component operates under.

Further aspects interesting to note in the VU framework is the consideration of four levels of reality or simulation abstraction [Gan15]. In the beginning of its development phase, the model in the loop scenario is described, where all components of the system are represented by their virtual aspects, thus defining one main model of the system. Directly after that the Software in the Loop phase is presented, in which the models are translated to a code compatible with the target hardware, thus allowing for a precise interface definition. The Hardware in the Loop phase follows, where physical hardware components are tested against simulated input values. In the last phase reality is then completely present, and all components can be monitored with their physical components. One can see how the virtual components are the only ones

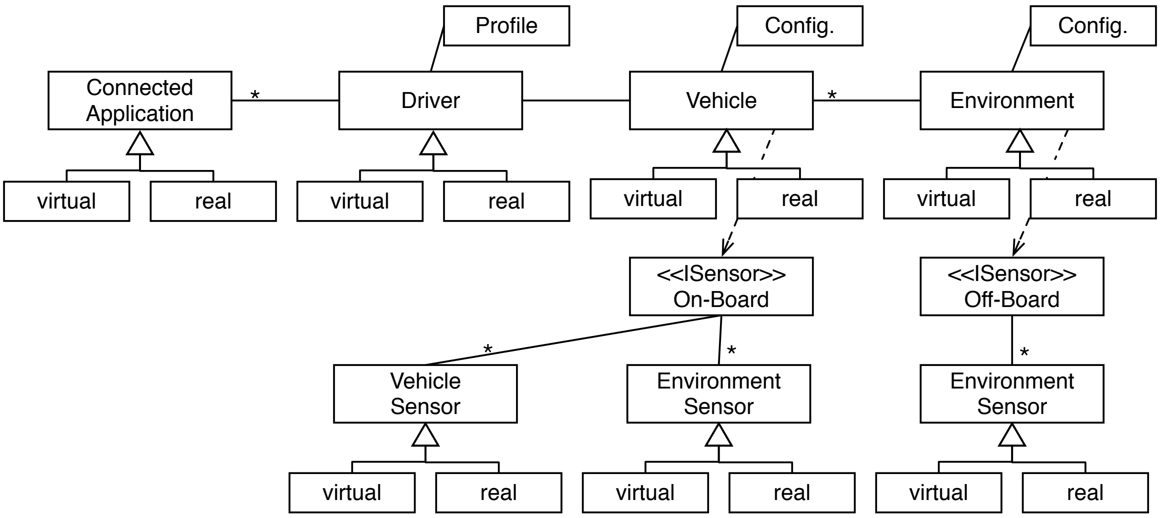


Figure 2.4: Overview of the VU-Framework described by Ganslmeier (UML Class Diagram). Translated from Ganslmeier [Gan15].

available in the beginning, and slowly evolve into physical ones.

Ganslmeier additionally considers what transformations are necessary from virtual components to real world ones. Besides the usage of data communication protocols for both the real and virtual components, he also described a coordinate system transformation from real world coordinates to the virtual world x, y, z coordinates, in order to ensure that objects were placed correspondent and consistent in each of the four virtualization phases.

While these works represent a solid basis for IOT applications, relevant architectures, device types, and data sources, they handle the topic from a broad perspective. Smart cities represent a very specific configuration of IOT and ubiquitous computing, that deal with specific required data and usage of these. In order to examine the consequences and requirements of these application, an analysis of what Smart Cities means, and how this relates to sustainability concepts is now presented.

Chapter 3
Investigation of Sustainable Cities

3.1 Smart Sustainable Cities 24
3.2 SSC System Architectures 26
3.2.1 ITU Smart Sustainable City 26
3.2.2 iSapiens 26
3.2.3 Multi-Level Smart City 27
3.2.4 Discussion 27
3.3 Sustainability Standards 28
3.3.1 ISO 37120: City Services Indicator 28
3.3.2 ISO 37122: Smart Cities Indicators 30
3.3.3 Smart Sustainable City Indicators 31
3.3.4 Sustainable Urban Mobility 32
3.3.5 Taxonomy of Indicators 33
3.4 Urban Planning Softwares 34
3.4.1 Urban & Traffic Simulations 34
3.4.2 Geographic Information Systems 35

The following chapter continues to analyze the urban design, and how sustainability aspects integrate into it in a digitally connected world. In Section 3.1 the term smart sustainable city is formally defined. That definition is then used to look into international standards relating sustainable and connected indicators to the urban domain in Section 3.3. Section 3.4 then considers current software options used by urban planners in order to understand how these can be used, supported, and extended by the components of the framework to be developed.

3.1 Smart Sustainable Cities

A wide range of smart city definitions can be found in literature, and most smart city projects have their own definition of the term. In an endeavor to categorize the different aspects of a smart city, 120 definitions of the term were investigated by the International Telecommunication Union (ITU), and then grouped together into 7 categories, as can be seen in Figure 3.1 [SL19].

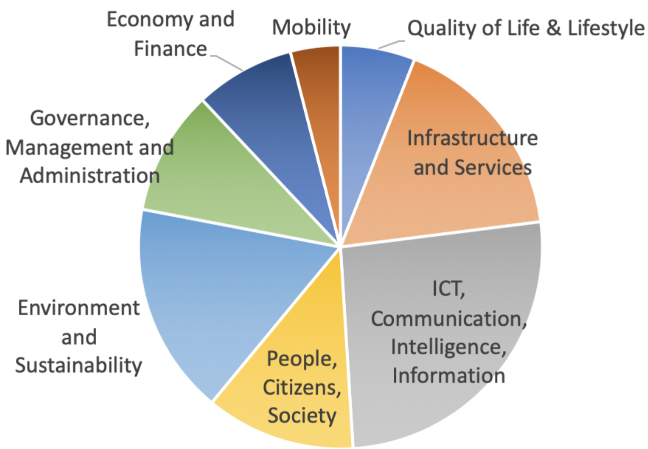


Figure 3.1: Pie chart based on the keywords found on the definitions of different smart city publications. Adapted from Sang *et al.* [SL19].

As Figure 3.1 shows, the most common terms in the smart city definition are aspects concerning Information and Communication Technology (ICT), Infrastructure and Services, as well as Environment and Sustainability. Kitchin considers how smart cities include "pervasive and ubiquitous computing built into the fabric of urban environments" to control city services [Kit14]. While that interpretation touches upon some important aspects such as using ubiquitous computing to monitor city flows, it does not take the notion of sustainability into consideration. In order to bring the concepts of connected and sustainable cities together, the United Nations Economic Commission for Europe (UNECE) defined Sustainable Smart Cities (SSC) as the following:

Definition 9

Smart Sustainable City:"an innovative city that uses information and communication technologies (ICT) and other means to improve quality of life, efficiency of urban operation and services, and competitiveness, while ensuring that it meets the needs of present and future generations with respect to economic, social, environmental as well as cultural aspects. [ITU-d]"

An important similarity that can be observed in the definitions above is the intent of using information technology to gather extensive data about a city's usage, and thus improve citizen's quality of life. This, however, leads to the need of multiple data sources being continuously monitored, which poses a challenge from a systems engineering point of view.

Furthermore, it is relevant to note the sustainability aspect of the definition above. The last segment of the definition includes several important concepts which are now analyzed. First it addresses societal needs, which is a key part of sustainable development. In a 1987, report the United Nations mentions that when discussing global needs, a priority should be given to the need of the poor, explaining how in order to meet essential needs, it must be assured that those poorer get their fair share of resources necessary for their growth [WCED87]. While societal needs is a broad concept, the notion of sustainable development helps us relate it not only to human need, but also to environmental usage.

These necessities, however, should be ensured also for future generations, which brings us to the second concept: the fact that current developments should assure that future generations have access to the same natural resources as present today, without limiting their potential development due to lack of assets. The United Nations report from Brundtland further explains that sustainable development includes the idea that technology and social organizations should not, neither in the present nor in future generations, present a limit on meeting needs [WCED87].

Finally, the definition above segments sustainability into three main pillars, of economic, social, and environmental aspects. While currently some definitions (such as [ITU-a]) include cultural development as a fourth pillar, the core concept remains the same. When discussing sustainable development, society development and impact should be taken into consideration, not only the effects on the environment. These pillars are important not only when discussing the resources of present and future generations, but also when considering the priority of poorer communities for sustainable development, as described earlier.

3.2 SSC System Architectures

After looking at the "Sustainable" aspect of SSC, we now consider the second S: "Smart". Several researchers and groups have already proposed standards for the technical implementation required for these systems, and the most relevant of these are now presented. A few research initiatives have already concentrated their focus on developing a system architecture for smart cities, which are now presented.

3.2.1 ITU Smart Sustainable City

The focus group from the ITU initiative has gone on to propose a system architecture for connecting the different sensors and data sources made possible by their approach. They divide the smart city system into four layers, from bottom to top: (1) Sensing layer, (2) Network layer, (3) Data and Support layer and (4) Application layer [ITU-f]. This closed layered architecture allows for each layer to access the one only directly above it, and provide services to the layer directly above it, similarly to the reference model of Open System Interconnection [BD10].

3.2.2 iSapiens

While the closed layered architecture approach presents important advantages as far as security and extensibility are concerned, it may run into a bottleneck when processing the high amount of data that smart cities require. As we have seen in 2.2.1, when integrating several data sources in real time, one systems approach that presents itself is the use of Fog Architectures. One such approach for smart cities was presented by Cicirelli *et al.*, where the authors describe *iSapiens*, a platform that is proposed and tested in the city of Cosenza (Italy) with multiple climate, air quality, noise and luminosity sensors installed and connected to it [CGSV17]. Importantly, Cicirelli *et al.* also use a layered architecture for their system, but apply a fundamental change to the second layer, where the data sources get aggregated into more abstract concepts, which are then presented to the third layer. This aggregation is important, as the sensors installed on the bottom-most layer are done so with an edge approach, meaning that the data delivered by these devices is completely unprocessed.

It is furthermore important to note that the architecture presented in [CGSV17] is not a closed layered one, and intermediate/top layers are allow to access services provided by all layers beneath it, not only the one directly below it. This allows the agents inside of each layer to expose the data aggregation and further processing to external third party providers, which can be a desired configuration, especially when dealing with open data initiatives.

3.2.3 Multi-Level Smart City

A further smart city architecture for IoT-based applications is the multi-level smart city, proposed by Gaur *et al.* [GSPM15]. As with the other research, a layered architecture is also proposed, but Gaur *et al.* consider more domains of a smart city. Furthermore, the authors consider the uncertainties arrived when merging sensor data from several sources in smart cities and use Dempster-Shafer rules [Sha76; ADB12] in order to minimize the effects of these uncertainties. While their system is not described as a fog architecture, their use of distributed sensors, which are then aggregated one layer above closely resembles fog aspects.

3.2.4 Discussion

While both of the research by Cicirelli *et al.* and Gaur *et al.* envision extensible sensor networks which are slowly aggregated and processed as the layers go higher [CGSV17; GSPM15], they lack the support of international standardization approaches, as is present in the results from the ITU organization [SL19]. Furthermore, both of the former publications focus mainly on smart cities, without taking the sustainability aspect deep into consideration, which limits their applicability into our problem.

It also important to note that system engineering architectures for smart cities are only part of the required infrastructure, as the connection between these IoT systems and the urban infrastructure (roads, housing, pipelines, etc) should also be addressed. While the architecture described by the ITU initiative covers both technical components and their integration into urban systems [SL19], it lacks the notion that the mentioned ICT systems track data relevant for sustainability indicators in the smart cities. This concept, handled by the data virtualization layer in [CGSV17], would allow to address the problem that components with similar sources, such as Intelligent Building Systems and Building Information Modeling in the diagram present no common data element between them. One further disadvantage of the architecture presented by the ITU initiative is the lack of definition of the interfaces among the components. While in some cases these can be implied (such as in the communication layer), the relation between ICT standards and the services provided by the smart sustainable cities is not clear.

3.3 Sustainability Standards

As Santana *et al.* describe, several maturity models exist for evaluating how sustainable or integrated a city is [SONS18]. As the problems addressed by cities worldwide often are similar to each other, the International Organization for Standardization has developed different standards to account for necessary indicators that a city should report on in order to be described as sustainable. The main indicator which covers the city services and quality of life is ISO 37120 and it follows the principles described in ISO 37101. The latter standard focuses on the requirements and processes for systems aiming to help sustainable development. Furthermore ISO 37120 also is expanded upon by ISO standards 37122 and 37123. Standard 37122 focuses on smart cities, and thus will be described in more detail below, standard 37123 on the other hand focuses on the resilience of cities after natural disasters, and thus is classified as out of scope for this research. Figure 3.2 demonstrates the relationship between each of these standards.

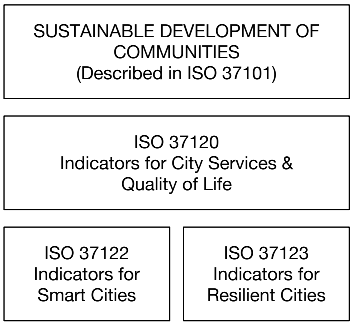


Figure 3.2: Relations between ISO indicators for smart cities. Adapted from Iso definitions [ISO 37120].

3.3.1 ISO 37120: City Services Indicator

In order to aid the sustainable development of communities, standard ISO 37120 focuses on indicators that measure the performance of city services and quality of life. It describes indicators in themes of a city that should be monitored, and classifies these indicators as core, supporting and profile indicators that must or could be measured in order to comply with the standard [ISO 37120].

In order to become ISO 37120 certified, a city should measure, in the minimum, the 45 core indicators listed in the document. These are classified into the following themes:

1. Economy
2. Education
3. Energy
4. Environment and climate change
5. Finance
6. Governance
7. Health
8. Housing
9. Population and social conditions
10. Recreation
11. Safety
12. Solid waste
13. Sports and culture
14. Telecommunication
15. Transportation
16. Urban/local agriculture and food security
17. Urban planning
18. Wastewater
19. Water

ISO 37120 defines an indicator as a quantitative, qualitative, or descriptive measure that demonstrate the performance in the delivery of city services and quality of life. The annex of the document further describes a mapping between the presented indicators and the UN's SDGs, thus visualizing how the measurement and data collection for each of the indicators allows to monitor progress toward the SDGs. A taxonomy of the most important concepts defined in the standard can be found in Figure 3.3.

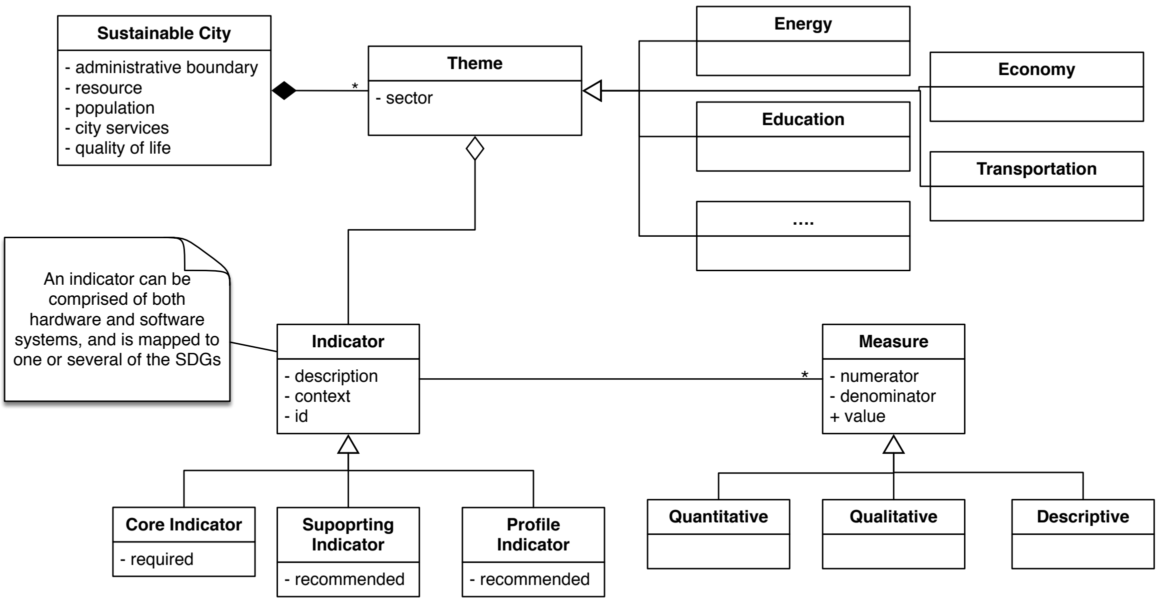


Figure 3.3: Indicators taxonomy for smart cities. Derived from ISO 37120 and ISO 37122 descriptions (UML Class Diagram).

3.3.2 ISO 37122: Smart Cities Indicators

ISO 37122 complements the original standard by identifying indicators and methodologies that increase the sustainability pace through the usage of data information and modern technologies [ISO 37122]. While ISO 37120 focuses on general city services, most of the indicators described in ISO 37122 focus on aspects that would affected if a city focuses on digitalization and connectivity of its infrastructure. Several of the indicators described in ISO 37122 would have a direct impact by diffusion of CPS in cities, such as “Percentage of the city population with access to real-time public alert systems for air and water quality advisories” or “Percentage of city streets and thoroughfares covered by real-time online traffic alerts and information” [ISO 37122]. As far as themes are concerned, ISO 37122 uses the same categories as listed above for ISO 37120.

Since measuring technological progress can be a hard task, the International Organization for Standardization defined criteria for describing the quality and principles of the indicators. These are defined below [ISO 37122].

1. Completeness of information
2. Technology neutrality
3. Simplicity of results
4. Validity of data
5. Verifiability of indicators
6. Availability of data

It is interesting to note how these criteria are related to non-functional requirements often used in software engineering. They, however, do not describe quality requirements for the systems measuring the data, but are one meta-level above them, describing quality requirements for the indicators themselves, that in turn describe quality attributes of the smart systems. The fact that these indicator quality criteria are listed for ISO 37122 but not for ISO 37120 could nonetheless be representative of how digital systems make it easier to monitor quality attributes.

While ISO 37120 and 37122 are the most relevant indicators for our study due to their emergence from the International Organization for Standardization, they are not the only existing set of indicators for measuring the success of smart cities. We now present some further relevant indicators.

3.3.3 Smart Sustainable City Indicators

In Section 3.2 we described the SSC architecture proposed by the International Telecommunication Union (ITU). In 2016, the same organization released a series of key performance indicators for smart sustainable cities [ITU-e; ITU-d; ITU-c; ITU-b]. In 2017 they partnered with UNECE and formed the United for Smart Sustainable Cities (U4SSC) initiative, where a document was released proposing indicators and collection methodologies for smart sustainable cities [ITU-a]. This document builds on top of the indicators presented in [ITU-e; ITU-d; ITU-c; ITU-b], and is considered to be more relevant as (a) it is more recent, and (b) several aspects of the indicators are described in more detail than the key performance indicator series. The U4SSC document is now described.

While the ITU publication date of 2017 is before the official publication of the ISO standards versions described above, it is after the publication of the previous version of ISO 37120, which was released in 2014. While there are some similarities between both indicators set (see Section 3.3.5, below), the indicator set belonging to ITU has an important element to consider in relation to the ones from ISO: it is distributed under a creative common license, thus making it available online without charge and making it easier to spread than the ISO ones. While this could be a good argument in favor of diffusion of these indicators, it also poses a limitation in regards to standardization, as the license used by the authors explicitly allows for adaptation of the indicators, thus making it harder to use for the comparison goals of our framework.

Furthermore, the indicators presented by the ITU are organized by sustainability impact classes rather than by urban infrastructure. This means that each indicator falls under the classes (1) Economy, (2) Environment, or (3) Society and Culture. These classes are then subdivided into the different urban infrastructure categories, which only partially overlap with the ones used by the ISO standards. Each indicator (referred by them as key performance indicator, KPI) is defined according to the following attributes:

1. Dimension
2. Sub-Dimension
3. Category
4. Rationale / Interpretation / Benchmarking
5. Methodology
6. Unit
7. Data Sources / Relevant Databases
8. SDG Reference(s)

While these attributes may appear to be more structured than the ones used by ISO – and indeed, the ITU document itself appears to have a more structured layout, as the indicators are presented in tables – there is a considerable overlap of these attributes to the ones described in text by Iso.

3.3.4 Sustainable Urban Mobility

We now consider the methodology and indicator calculation method for the Sustainable Urban Mobility document, from the World Business Council for Sustainable Development [SMP 2.0]. The major difference between this indicator set and the previously mentioned ones is the fact that it focuses only on urban mobility aspects, and is thus more specific than the other sets. While there is again an overlap between some of the indicators presented here, and some of the ones presented in the other sets, some metrics such as congestion and delays, or mobility space usage are defined here for the first time.

While the indicators presented in [SMP 2.0] are considered too specific to the mobility sector to serve as a general urban planning comparison set, they are still presented here due to the methodology described by them, which can be applicable to other domain areas. One key contribution by [SMP 2.0] is their analysis of available data sources, and how these can be further processed by merging existing data sources, into more refined metrics, which can again be further integrated into new metrics. They are the only indicator listed that describe this hierarchical relation between data sources, which is a key observation for our framework analysis later.

Furthermore, [SMP 2.0] is the indicator set that is the most mathematically precise. Initially each of the indicators in the mentioned set are presented with a well defined mathematical formula, where each of the relevant variables are clearly mapped to a data source. Then, the authors further map each of the indicators to a 0-10 scale, providing both reference values for the 0 and 10 of the range, and a reference value for each of the indicators for an existing city, which allows for a city implementing the indicator to have an idea wether the value of their indicator is good or bad.

Last but not least, [SMP 2.0] recognizes the value of population survey as a useful data source, which allows for a direct exchange of a cities' population when measuring the cities performance. When this data source is reasonable, the document goes as far as to propose survey questions that should be addressed by the citizens.

While the domain of [SMP 2.0] is too narrow to allow for a complete city comparison, the methodology described in the document is of great value when formalizing aspects of urban indicators.

3.3.5 Taxonomy of Indicators

Other than the indicators described above, further frameworks for measuring urban progress toward the SDGs exist, including the list of SDG targets published by the UN [UNDESA17]. While the UN targets provide a good mapping to each SDGs and thus contain several sustainability remarks, the digital focus is less present, and the targets are set as optimistic goals, such as achieving zero hunger worldwide. The methodology to calculate these targets is also described textually rather than mathematically.

All of the above presented indicators contain some similarities. Huovila *et al.* critically analyzed a subset of the presented indicators (and some additional ones) [HBA19]. The authors investigate how the indicators compare as far as considerations of sustainability, digitalization, city sectors and evaluation purposes are concerned [HBA19]. The information is organized into a taxonomy, and since both city sector and the difference between sustainability and digitalization aspects has previously been discussed, we now focus on the third dimension presented by Huovila *et al.*, evaluation purpose. The authors further refine this concept into five categories: input, process, output, outcome and impact [HBA19]. It is relevant to note that these categories assess not only the indicators itself, but also what the authors call interventions, or processes that aim to affect a given impact indicator. The classification aims to allow the measurement of progress toward a target throughout the entire lifecycle of an intervention.

According to Huovila *et al.*, 47% of the indicators present ISO 37120 are impact indicators, representing the biggest group for that standard [HBA19]. ISO 37122, however, mostly has outcome indicators, which represents a 48% of the standard. Generally, both ISO standards have a low percentage of process indicators, which is coherent with their aim of measuring the result of city transformations [HBA19].

A further sustainability indicator work was done by Mori *et al.*, who proposed a city sustainability index by considering constraints and maximization indicators [MY15]. The authors differentiate between values that should keep a constraint, and are thus ideal when within a certain range, or indicators that should simply be maximized [MY15]. This touches upon a discussion that becomes relevant in our later framework analysis, that a key limitation of indicators such as the ones presented in ISO 37120 and 37122 is the fact that the lack of a reference value for these indicators makes it harder for cities to know weather the values they are reporting for the indicator are good or bad.

Mori *et al.* further analyze how the pursue of indicators in one sustainability pillar can lead to the diminishing of indicators in other areas, and thus argues that the three sustainability pillars – environmental, economic and social – should be considered simultaneously when reporting indicators [MY15]. Since the ISO standards cover indicator in all of these categories, this is deemed an advantage of the indicator set.

While together these different standards offer concrete indicators for what a city should measure in order to call itself smart and sustainable, they serve only as a starting point when analyzing the measurements and information reported on.

3.4 Urban Planning Softwares

The usage of computational software to aid in the development of urban infrastructure systems is long common practice. From simulations to geographic systems, the existing software systems allow urban planning to plan, simulate and control the different service in their regions. This section describes some of the existing types of software that are most relevant for our proposed framework. We initially discuss the different types of urban simulations, and some projects that use these for sustainability tracking. Traffic simulations are then described in detail, as these will become especially relevant for the mobility components used in the reference applications of Scoup. Last but not least, geographic information systems are discussed as a basis for several urban analysis and planning.

3.4.1 Urban & Traffic Simulations

Mathematical models have long helped urban planners analyze and consider projects for their regions. From functions that map the spread of diseases or information, to energy consumption models, several models exist that represent different aspects of the urban development process. As with any model, one of the key aspects to consider is the level of abstraction proposed by each of these, and how that influences the calculated results. These individual models have evolved into simulations, and now slowly to digital twins [BR16]. One of the initial use cases for modeling of urban components included the monitoring and controlling of environmental factors. In 1995 researchers at the Carnegie Mellon University developed a system for Geographic Environmental Modeling that aided public policy researchers to consider various what if scenarios to analyze the impact of their public strategies [BRRM95].

Fast forward to the 21st century, and these software systems have slowly advanced to digital twins. As previously mentioned, although digital twins have become an interesting research area, the complexity of most cities make the implementation of a digital twin for a city a computationally difficult task. The main difference between a digital twin and classical simulations is the consideration of the entire life cycle of the city and the synchronization to physical objects [BR16].

While complete digital twins for cities are not yet ubiquitous, several projects exist for developing digital twins for individual systems. Relevant among these is the

Virtual Singapore initiative[1], where the Singapore government plans to create a 3D digital platform for public private and research usage. The platform should include detailed models of buildings in the city, including not only the components of the building, but even information on each material used. Furthermore, extensive data has been collected and modeled on the flora of the island [Gob+18].

One type of urban simulation that have received special attention due to their capabilities of visualizing and identifying important bottlenecks are traffic simulations. Traffic modeling has likewise evolved from mathematical modeling [She85] to cloud simulation for large-scale systems [Zeh18]. Traffic simulations can be categorized into three main types, depending on the level of detail that they represent: macroscopic, mesoscopic, or microscopic [Zeh18]. Macroscopic simulations typically model roads as flows, in order to allow for a fluid analysis of how fast or slow the traffic is in each road [KBA13]. Mesoscopic simulations may allow for different vehicle types, but these are at some point aggregated for computational reasons [KBA13]. Lastly, microscopic simulations model each individual vehicle or traffic participant as an agent. These can have not only travel behavior for each vehicle in the simulation, but also different driving behaviors [KBA13].

3.4.2 Geographic Information Systems

Urban planners and architects have long used geographic information to aid in their work. Historically, these originated as glass tables with a light in the back, which would allow urban planners to trace streets and areas in their map and design [MGPG11]. Maliene *et al.* describe how one of the first attempts to use a computer to optimize this process was in 1962, when geographers at the Washing University used machine readable punch cards to render coordinates of plant sites [MGPG11]. Since then, the usage of spatial information in the planning of urban environments has consistently increased in importance, as has the usage of computers to manage high precision maps.

Geographic information systems (GIS) comprise analytical models that encompass geo-referenced data into cartographic projections [MGPG11]. Both the projections and data involved can assume different formats and while common standards do exist for each, the usage of which one is preferable highly depends on the exact use case. While the consideration of a road network in a traffic simulation (as seen in Section 3.4.1) can require for streets and intersections to be modeled as lines in a graph, the evaluation of population density may require for a city to be sub-divided into smaller areas, and points of interests are often modeled as simple points in a graph.

[1] `https://www.nrf.gov.sg/programmes/virtual-singapore`

These three main data schemes (1) lines, (2) areas, and (3) points are all geo-referenced, and can be read through the usage of GIS software such as open source alternative QGIS[2], or professional commercial options such as ArcGIS[3]. Furthermore, these relations can contain further attributes that enhance the information available. In the population density example, it is evidently necessarily to not only store the area relative to the population density, but also the value of the population density itself, which is then simply a new column in the table. These additional columns and GIS software will usually allow for their query through varieties of database Structured Query Language (SQL).

Common data sources for GIS data include open databases such as OpenStreetMap (OSM)[4] or include satellite geodetic data [MGPG11]. The integration of GIS data and satellite imagery further allow urban planners to analyze changes in urban areas with a higher spatial and time resolutions [MGPG11].

Observation 5

While some of the GIS softwares allow for programming interfaces to automatically query and manipulate geographic information, the integration of these into other systems is usally not streamlined.

We now describe how the observations presented in this and the previous chapter can be integrated into a framework for constraint based urban planning.

[2] QGIS: https://www.qgis.org/en/site/
[3] ArcGIS: https://www.arcgis.com/index.html
[4] OpenStreetMaps: https://www.openstreetmap.org/

Chapter 4

SCOUP Framework

4.1 Requirements Elicitation 38
4.2 SCOUP Concepts 41
4.3 Formalizing Sustainability Indicators 43
4.4 Meta-model 50
4.5 City Input Process 53
4.6 SCOUP Overview 54

In this chapter Scoup is presented. We start by describing in Section 4.1 what challenges and features should be targeted by a framework that addresses the different data availability in urban planning and the necessity to integrate these. The requirements are presented as a use case diagram and the main terms necessary for the Scoup framework are defined. We then derive the main concepts which will be necessary for the framework and define them in Section 4.2. These are then used in Section 4.3 to formalize the sustainability indicators described in the previous chapter with the usage of Object Constraint Language (Ocl) and present a taxonomy for the indicator classification. We then use this formalization in the Scoup meta-model in Section 4.4 and connect the main stereotypes of the framework. We describe a process for how to input a city or region into the Scoup system and finally summarize the main components of the framework in Section 4.6.

While this Chapter presents the initial stage of Scoup's development, additional analysis and components get integrated into the framework at the end of each project presented in the following two chapters.

4.1 Requirements Elicitation

Each region is fundamentally different from another. Not only as far as population and culture go, but also regarding politics and legal requirements.

Definition 10

Region: "An administrative area, division, or district. [...] A broad geographic area distinguished by similar features"[1]. The word region is used to avoid confusion between cities, municipalities, and the different nomenclature that different countries use for these. A region is meant as a superclass for all different types and sizes of cities.

All of these concepts are integrated into the notion of context. Scheuermann defines that "a Smart Environment consists of contexts such as Location, Condition or a Domain"[Sch17]. While these are important considerations for cities, their context encapsulates further aspects. As a complex system which encapsulates several independents systems, a Smart Sustainable City has more than one condition and these can be measured and evaluated as indicators. Our definition of context is further explained in Section 4.2. A framework for sustainable urban planning should take into consideration the different aspects of the urban context and the relation among these, such as the urban categories presented in Section 3.3.1. We now present the functional requirements of SCOUP (SCOUP-FR) for the urban context:

SCOUP–FR1: Comparability among regions The problem investigation indicates that a major problem faced by cities and communities is the comparison among and inside themselves, including in a neighborhood scale. While Section 3.3 shows that various sustainability indicators exist that could allow for such comparison, several of the necessary data sources are not always directly available across the different neighborhoods. This is partially due to the disconnection that often exists among different departments that collect these data sources. Having data sources integrated into a single platform would allow for these indicators to be more easily calculated across different regions and thus easier to compare.

SCOUP–FR2: Representation of physical entities Cities are composed of physical streets, buildings, vehicles, and people. While some of these are digitally connected, this assumption does not hold true for all entities. As some of these offline entities also contain relevant qualitative data and execute important urban initiatives, it should be possible to incorporate the knowledge expressed by these. An example of

[1] "Region" *Merriam-Webster.com.* 2020. `https://www.merriam-webster.com` (13 May 2020).

this could be a trash collection initiative that has no sensor or connected data source, and where employees log, at the end of each round, how many tonnes of trash were collected in that region.

SCOUP–FR3: Addition of data sources As cities become more digitalized, they start to have more quantitative and continuous data available. However, it is still common to have a large part of a cities' information stored in offline or analogous paper sources. A framework for constraint-based urban planning should allow for the modular integration of several types of data, ranging from qualitative census information to geographic satellite imagery, to data flows incoming from any street sensor. It is essential that a framework should support these data sources and that they can be added at different points in time, referencing both different spatial and temporal aspects.

SCOUP–FR4: Consideration of multiple stakeholders Cities are complex environments, where government agencies, citizens, local businesses, and larger industry each have very different expectations and capabilities toward the public space. Data privacy considerations toward these stakeholders affect the measuring capabilities from the communities point of view. Nonetheless it must be ensured that different and possibly conflicting expectations between all stakeholders are addressed, while also respecting the data privacy of each of them. A framework for sustainable urban planning should support data entry and indicator retrieval for multiple stakeholders who may have conflicting interests.

SCOUP–FR5: Calculation of sustainability indicators While several solutions exist for tracking data types and sources, the same does not apply to urban initiatives that use and provide these data. Ultimately, a Smart City is only useful if the data collected is used to ensure that the city "meets the needs of present and future generations with respect to economic, social, environmental as well as cultural aspects" [ITU-d], and thus aids in the execution of sustainable initiatives. The fact that a standard definition for such indicators does not exist means that a taxonomy is required to allow not only for the comparison, but also for the evolution of these within the city.

SCOUP–FR6: Switching between possible initiatives It is possible that similar initiatives variations provide different outcomes. One simple example of this could be having varying speed limits throughout the day, which could affect air quality. While higher traffic flow is important during during rush hour, it could be feasible to decrease flow, or even close certain lanes, when the flow demand is smaller throughout the day, in order to regulate air quality. While some initiatives require complex infrastructure to be carried out, initiatives with a digital interface can often be dynamically adjusted. A framework that allows for initiative definition should also permit that these be dynamically switched whenever possible, and have the impact of these switches tracked.

SCOUP–FR7: Evaluation of initiatives through simulations As described in Section 2.1.3, the rapid evaluation of urban initiatives becomes a challenge, as these often require longer and less flexible development phases. For example, a city cannot test the results of a new road with the same ease that a developer can test the effects of a new function. To facilitate such initiative evaluations, the framework presented in this dissertation should support the digital representation of urban initiatives, making it possible to run simulations that predict the impact of the tracked indicators.

Use Case Model

Figure 4.1 describes the relations between the requirements described above. It is important to note how each of the use cases are executed through different actors, which is a direct consequence of how different stakeholders are involved in the urban context. The actor Urban Planner models the people who plan and execute the urban initiatives, and are responsible for the regions planning on a macro level. These are the stakeholders who commit to the SDGs, and thus implement projects or initiatives that take action in such direction. It is their responsibility to manage the activities and citizens of a city, and thus they are the entry point to the core use cases of SCOUP.

The actor Citizen models the residents in the region, who are affected by the Urban Initiatives. They can use any relevant part of the initiative, and may have an interest in monitoring the indicators of the regions where they live.

Furthermore, there also is the actor Data Providers, who makes information from heterogenous sources available to the city and Urban Planners. Data Providers may be individual sensors providing information relevant to the indicators, but can also be local companies, or services operating in the region who measure and report on some relevant information. It is relevant to note that while urban planners can (but must not) be citizens, both urban planers and citizens can also be data providers.

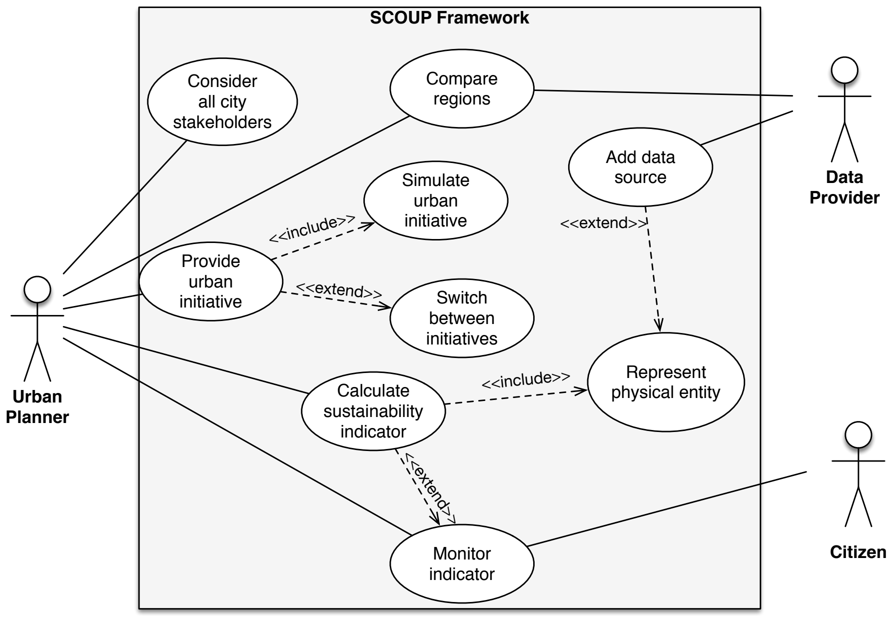


Figure 4.1: Scoup Requirements (Uml Use Case Diagram).

4.2 SCOUP Concepts

In the following section we define the stereotypes and concepts that compose the Sustainable Constraint-based Urban Planning (Scoup) framework. This framework contains abstractions and type definitions that are common for many sustainability projects. While most of the concrete implementations of Scoup presented are in the transportation sector, an attempt is made to keep Scoup's stereotypes generic. We start by defining the main concepts used in our model.

Definition 11

Sustainability: The definition of the International Telecommunication Union [ITU-a] is used, where the concept of sustainability consists of four aspects: social, economic, environmental, and cultural aspects. A component or system is considered sustainable in case it ensures that present and future generation will have access to the same resources in these four categories.

In other words, the consumption and production of all key resources should be kept at a stable level for the future. While systems can be sustainable in one of the four aspects, this does not guarantee that they are sustainable in all four of them.

Definition 12

Indicator: In keeping consistent with the ISO definitions, an indicator is defined as a measured index, which contains quantitative, qualitative, or descriptive aspects.

Indicators are the reference value that a system tries to collect and optimize. Sensors, actuators, and strategies available through other components aim to facilitate the control of these.

Definition 13

Urban Initiative: Any action, program, or service started by a City with the aim of improving an indicator related to one of the Sustainable Development Goal (SDG).

Urban Initiatives can be physical initiatives such as covering potholes (SDG 11[2]) or digital, such as making a given information available to the public (SDG 16[3]). Initiatives involve multiple stakeholders and a flow of events, which should ultimately lead to the difference in indicator performance. Different variations of an initiatives are possible, which may result in different impacts. While Urban Initiatives may be seen as the "actions" of a region, they are not implemented with the Strategy Pattern [Gam+95] as the concept of run time is hard to relate to cities.

Definition 14

Context: The circumstance or background data relevant to any system's component.

The context data often get saved as meta-data tags and includes any situational information such as location, time, temperature, or conditions of the controlled system. Context data can be static (such as location) or dynamic (such as time) and should get updated accordingly. Context data are not directly related to measured indicators of an object or environment and merely provide additional information about the relevant system.

Definition 15

Measurement: Any data or information expressed in quantitative or qualitative terms.

[2] `https://sustainabledevelopment.un.org/sdg11`
[3] `https://sustainabledevelopment.un.org/sdg16`

In the domain of smart sustainable cities, a data source generally has the format of a population survey or census, a geographic information system set, raw sensor data collected by any available sensors, or a combination of these in the form of a given analysis. Urban indicators are not seen as a data source, as they are seen as outputs of the system, and not inputs. Data sources come from either static sources or fog devices.

Definition 16

Simulation: An approximate imitation of the operation and results of a process, phenomenon, system, or system usage given initial conditions and represented over a parameter such as space or time.

Computer simulations are digital abstractions of physical objects, and use algorithms to calculate the results of the process, usually on the basis of physical laws. Simulations correctness can only be assured given the fulfillment of the initial conditions.

Having described the necessary requirements and concepts for our framework, we now analyze the existing aspects of sustainability indicators and how these can be translated into generic components, dependent on defined variables, which origin from different data sources.

4.3 Formalizing Sustainability Indicators

While several tools exist for traffic and urban infrastructure optimization based on the maximization of specific parameters, we introduce a formalization that allows for the modular integration of different variables, data sources, physical world limitations, and sustainability thresholds. A combination of Object Constraint Language (OCL), object orientation, and sustainability indicators is used.

Environmental standards, such as [SMP 2.0] or ISO 37120 [ISO 37120], make it possible to derive real world constraints for sustainability indexes. In the following section, we formalize these into OCL constraints that are then integrated into SCOUP. While a concrete example is provided for indicator 19.1 of ISO 37120, an emphasis is kept on keeping these as generic as possible, in order to allow for their extensions and further usage with other standards and applications.

As stated in Section 3.3.1, ISO 37120 describes 19 categories as a reference for their indicators that measure the quality of life and city services in any given city. While each of these indicators have specific input parameters and calculation, several of the required information is generic to all of them. As an example the following section uses Indicator 19.1:

Example 2: ISO 37120 - Indicator 19.1 Calculation

Kilometers of public transport system per 100 000 population [ISO 37120], a core indicator which measures the offer the public transportation systems in a city. "The extent of a city's transportation network can provide insight into traffic congestion, transportation system flexibility and urban form. Cities with larger amounts of public transport might tend to be more geographically compact and supportive of non-motorized modes of transportation." [ISO 37120]:

$$Indicator_{19.1}(Public\,Transport\,System\,Size) = \frac{LengthPT * 100000}{Population}$$

[ISO 37120]

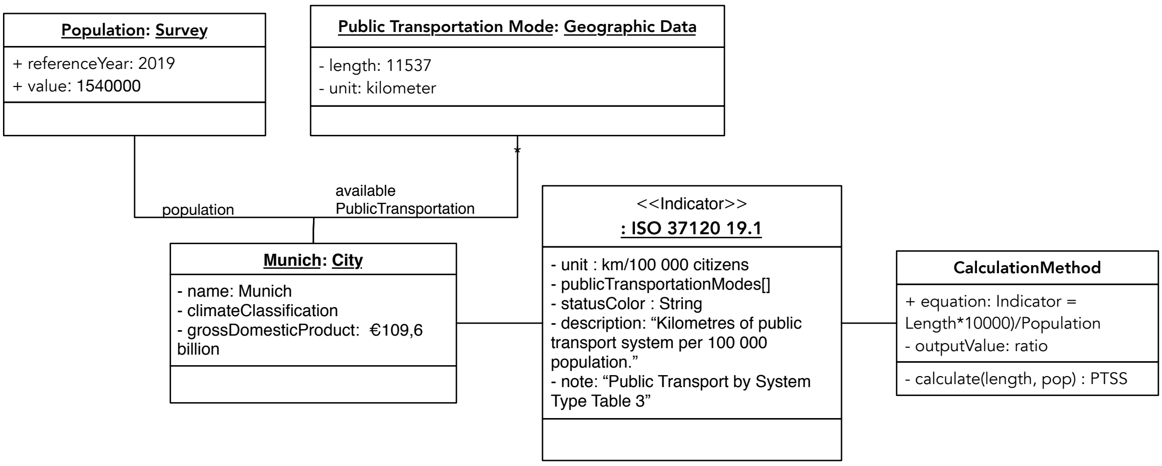


Figure 4.2: Elements of the public transport system length indicator for the city of Munich and its relations (UML Object Diagram). Munich values originate from GIS data analysis based on OpenStreetMaps [HW08] information, and online population databases.[4]

In this calculation, $PublicTransport\,System\,Size$ represents the calculated ratio for how many kilometers of public transportation ($LengthPT$) system are available in the region for every 100 000 citizens ($Population$). In order to incorporate this indicator into the described framework, we first present the components of the indicator in oriented object notation and then translate the variables into OCL constraints.

Figure 4.2 illustrates how the mathematical formula of the indicator can be illustrated as an object diagram, describing all of the Indicators parameters, measurements, their relations, and calculation. As shown in the figure, and compared

[4]OpenStreetMaps: `https://www.openstreetmap.org/` [HW08]
World Population Review: `https://worldpopulationreview.com/world-cities/`

against the original source [ISO 37120], the Indicator contains several characteristics, including unit and description. Other than the attributes defined in the Indicator stereotype, this Indicator instance also has an additional note describing which public modes of transpor types are to be considered for the Indicator. When translating this into object notation, these become attributes of the instance ISO 37120 19.1, of type Indicator. In order to calculate the value of the Indicator a formula is provided which receives two different parameters, each stemming from a different type of data source. The logic for iterating through the length of all types of public transport can be integrated into the CalculationMethod.

One difference between [ISO 37120] and [SMP 2.0] indicators (see Section 3.3) is the fact that [SMP 2.0] provides a reference value for the indicator, while the ISO indicators do not. While this lack of range could make the usage of ISO harder, it is compensated in ISO by the fact that most of the ISO indicators are either percentages or relative to the population.

Example 3: Munich Public Transport System Size

According to the formula above, the city of Munich has a public transport system of 750.0 $\frac{km}{100\,000\,people}$.

It is relevant to note that public transport system is an indicator that should be maximized and thus a bigger value is better than a smaller one. Once we have a calculated Indicator we are presented with the problem of how to compare this to other cities. By definition, a comparison always requires more than one element, and thus the first step is to get the same Indicator for different cities. Table 4.1 represents the value of the same indicator based on a GIS analysis for several cities.

City	Total Length [km]	Population	ISO 37120 19.1
Kirchheim	1,752	12,806	13,681,087.0
Munich	11,537	1,538,302	750.0
London	20,552	9,304,016	220.9
Singapore	10,980	5,935,053	185.0
Seoul	12,503	9,963,452	125.5
Beijing	12,251	20,462,610	59.9

Table 4.1: Reference values for Indicator ISO 37120 19.1 for different cities around the world. Total length of public transportation was the summation of different modes of transportation based on GIS data from OpenStreetMap. For population, values from 2020 were used for the major cities, and from 2018 for Kirchheim[5].

[5] OpenStreetMaps: `https://www.openstreetmap.org/`

It is also relevant to note the importance of the city boundaries when calculating these indicators. Many cities worldwide have quite undefined boundaries due to their extended metropolitan areas and unclear divisions exist between the main city area and other municipalities that have emerged around it. An example of this is the relation between Munich and Kirchheim, the latter being integrated into the public transportation service of the former and thus making it hard to define to which public transportation lines in Munich also belong to Kirchheim. For the calculation in the table above, an attempt was made to use the city areas as defined in OpenStreetMaps, but the accuracy of these highly varies throughout regions worldwide.

Table 4.1 also illustrates the importance of having normalized indicators. While the public transportation length is largely the same magnitude for most cities, the resulting indicator have a much wider range, due to the wide variance of the population in each of the cities. The municipality of Kirchheim (refer to Chapter 6) exemplifies how these indicators become out of scale when the region's population is smaller than 100 000. Since Kirchheim's population is less than 100 000, the calculated indicator value is highly out of range when compared against the other cities. While the example indicator ISO 37120 19.1 is normalized against population, [ISO 37120] does not provide a reference calibration to map the indicator value to a standard 0–100 scale. This raises the question of how to normalize these indicators to a standard scale or percentage that can be mapped to green/yellow/red references.

We approach this by using the existing indicators as reference values, and define the minimum point of the range as the current minimum existing indicator value and the maximum point in the range as the current existing maximum indicator value. In the ISO 37120 19.1 example, we disregard Kirchheim from now on because the small population makes it a big outlier in the data set and due to its shared usage of Munich's public transportation infrastructure. SCOUP addresses the population range problem by issuing a warning each time a small community attempts to compute a population normalized indicator. It is important to note that the usage of the current database as reference values presents relevant limitations, as in the worst case if a new city is input into the system that is outside of the current range, this would trigger a recalculation of the green/yellow/red status of all cities in the database. While the green/yellow/red scale makes sense in view of SCOUP-FR1 comparability requirements, it means that each time a new City is inserted into the database, it must be relatively compared to all existing cities, which may potentially be a computationally expensive operation.

We use the reference values $59.9\frac{km}{100\,000\,people}$ and $750.0\frac{km}{100\,000\,people}$ as the outer bound-

World Population Review: `https://worldpopulationreview.com/world-cities/`
Kirchheim Population: `https://www.landkreis-muenchen.de/landkreis/daten-und-fakten/`

aries of the range. In order to calculate the thresholds of the green-yellow-red stoplight signal, we use the median of the dataset ($185 \frac{km}{100\,000\,people}$), and add/subtract a 25% buffer to calculate the upper/lower boundaries. In the example for Public Transport System Size, the thresholds of the range boundaries result in the following thresholds:

Example 4: Threshold values for Iso 37120 19.1

Green: $\geq 231 \frac{km}{100\,000\,people}$
Yellow: $231 - 148 \frac{km}{100\,000\,people}$
Red: $< 148 \frac{km}{100\,000\,people}$

Note that while these thresholds are mapped to concrete values and units in this example, using a standard 25% buffer allows us to keep the same green-yellow-red stoplight signal mapping for any indicator, regardless if the value is represented as a percentage of not. These threshold values allow us to calculate OCL sustainability invariants for each indicator. Keeping consistent with the Public Transport System Size indicator example, we now convert these thresholds into OCL notation.

Listing 4.1: Km of public transport system signal mapping.

```
context PublicTransportSystemSize::StatusColor : String
derive: if (formula.output >= 231)
    then "Green"
else if ((formula.output < 231) and (formula.output >=  148))
    then "Yellow"
else
    then "Red"
```

Generalization

While each indicator standard considers different attributes, some commonalities are shared among all indicators. Figure 4.3 models these by relating indicators both to the methodology used to calculate them, the data sources related to these methodologies, as well as the categories each indicator falls into.

The category component is one that differs significantly between each indicator standard. While Iso 37120 and 37122 predefine several urban infrastructure categories to be measured and evaluated in each standard, the [SMP 2.0] indicators map each of their mobility indicators to one or two categories out of only four available. While some of the categories in each of these indicators overlap, several aspects covered in the Iso standards are not observed in [SMP 2.0], as this standard focuses mainly on mobility aspects, which is only one chapter of the Iso standards. In each case it is possible to notice the sector description of each of the described categories.

As far as indicator methodology goes, each indicator has its own set of rules and parameters to be measured and taken into consideration. One could argue that the output of the methodology is the core value of an indicator, however, it is important to also note its limitations. While the Iso standards define what parameters of a city should be measured in order for a given community to call itself compliant to the sustainability and digitalization standards, it does not provide reference values for the result of these measurements and most of its indicators are described as simple percentages or parts per (hundred thousand) capita. The [SMP 2.0] indicator collection, on the other hand, provides a scale for each of the presented indicators and usually also provides a city example as an indicator reference. Furthermore, the [SMP 2.0] group always provides a concrete formula for the calculation of the indicator as part of its methodology, which makes it easier to directly compare different cities. Since the Iso indicators often are simple percentages, calculating them usually means performing a specific division of the mentioned parameters, which is also described in the indicator standard. While the Iso indicators set does not provide the indicator calculation as a mathematical equation, a textual description of how to calculate these is always provided, which then allows us to formalize the equation.

One thing that all methodologies have in common is their reliance on available information. Availability of data sources is one of the key differences between different communities and cities, as governments tend to favor the measurement of parameters that are better aligned with their political and social interests. Even for parameters that do often get measured by different governments, such as air quality index, this is usually done by completely different systems and sensors, which is one of the key motivations behind using the mentioned indicator standards.

Several types of data sources are available and – depending on the indicator considered – a single data source or combination of these is required. Other than sensor measurement which are typical for fog systems, urban infrastructure projects also tend to rely on two further types of data sources: geographic informations systems (Gis) and surveys distributed to the affected population, such as a census. The combination of these types of data allow for analysis about the locations in which the citizens are thinking or doing what, which can also be further extended by sensor measurements. The necessity to integrate non-digital data sources such as survey, Gis, or analysis results, is one of the reasons why Smart City planning becomes a challenge. Simply integrating these data sources in the first place can already be a problem and indicators allow for a standardized approach on how to do so.

Figure 4.3 illustrates the relations between these components, when aggregating all of these into one indicator definition. It is notable that although both taxonomies were developed independently from each other, several aspects of the described taxonomy resemble the aggregation described in [HBA19]. The major difference among these indicator taxonomies is the inclusion of the concept of Measurement, which Huovila *et al.* simply describe as "Input". The composition among different measurements depicted in Figure 4.3 is the key to the integration of different possible data sources, used in the Indicator calculation.

Furthermore, in order to make the OCL conversion possible, we add a value attribute to the Indicator stereotype, which is then compared against the median indicator value in the database. This makes it possible to convert any indicator output to the same green-yellow-red traffic signal, and the comparison then is dependent on a simple multiplication. The result of the OCL conversion can be seen below.

Listing 4.2: OCL Representation of Indicator Status Color.

```
context Indicator::StatusColor : String
derive: if (indicator.value >= indicator.median*1.25)
    then "Green"
else if ((indicator.value >= indicator.median/1.25) and
         (indicator.value <  indicator.median*1.25))
    then "Yellow"
else
    then "Red"
```

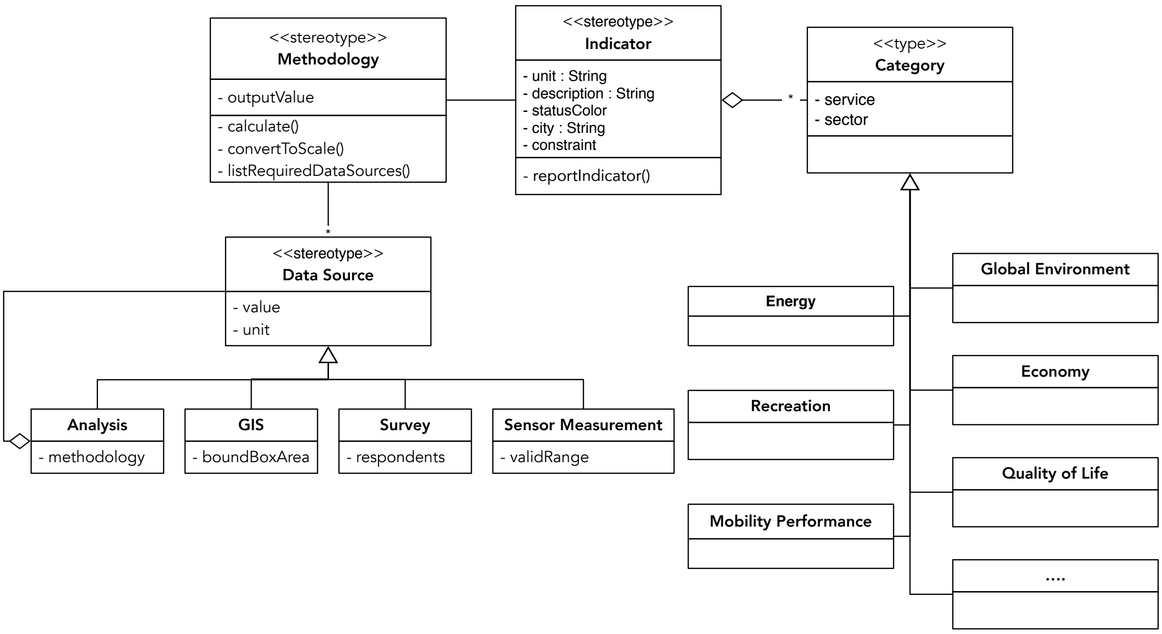


Figure 4.3: Indicator taxonomy (UML Class Diagram).

4.4 Meta-model

Following the description of the basic concepts necessary for our framework, as well as a general representation of Indicators, we advance in the analysis process and generalize our abstractions into a Smart Sustainable City (Ssc) meta-model, which is presented below. We start by describing the meta-model as an Analysis Object Model, then discuss which relations are necessary among these and how design patterns [Gam+95] help achieve some of Scoup's requirements. It is relevant to note that while the meta-model is derived now from the start, it will get iterated and improved upon at the end of the following two chapters. For simplicity reasons, we present the meta-model only once and present the elements that get inserted into it later as a different color.

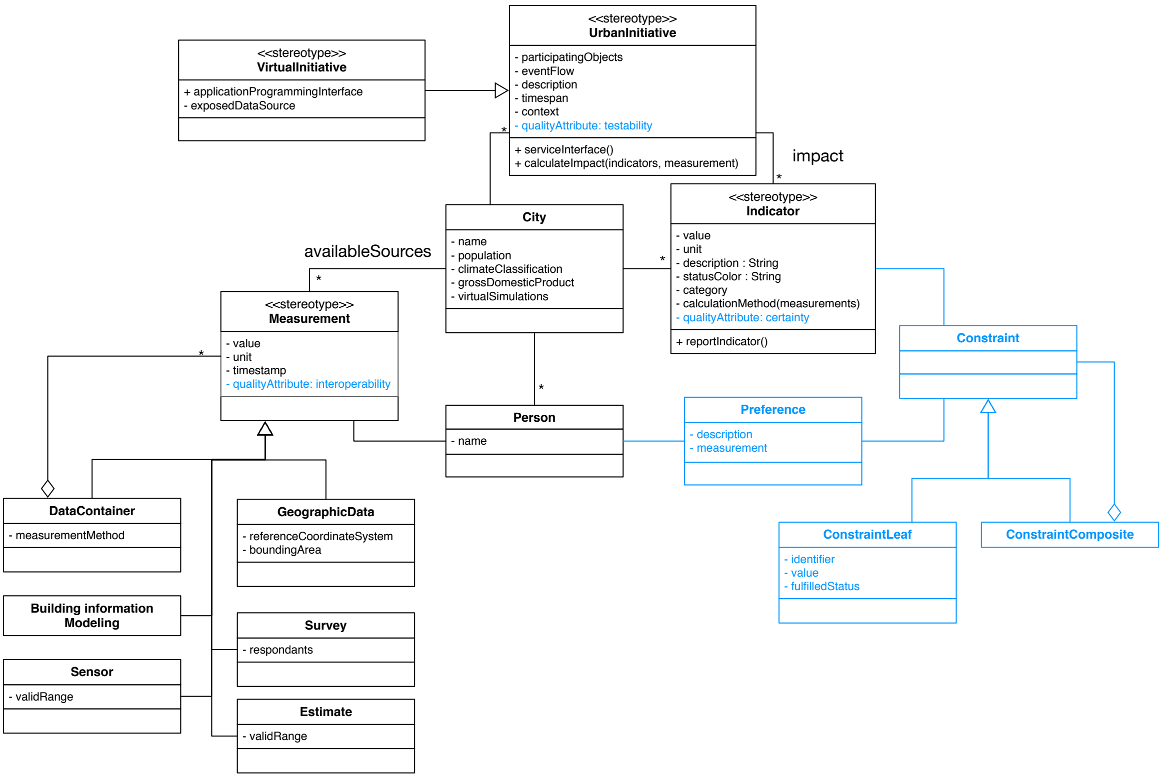


Figure 4.4: Scoup meta-model. Elements in blue are added to the meta-model in iterations after Chapter 5 and 6 (Uml Class Diagram).

The main goal of the Scoup framework is to facilitate the implementation of the Ssc. It is relevant to note that while the word “City” is used to be compliant with the referenced standards, in theory this could be any municipal or administrative region with a given name and population, including smaller communities. For normalization and comparison purposes, we additionally include gross domestic product and the Köppen-Geiger climate-classification [PFM07] as attributes to a City. This information is useful for the Context considerations of an Urban Initiative and thus are integrated into the meta-model. Figure 4.4 shows how a Ssc is modeled as a simple City and the "Smart" aspect is given by the relation between the City and all available Measurement sources.

As seen in Defintion 15, Measurements can be of several sub-types. While fog computing allows for the easy integration of real-time Sensors, it is important to note that several Cities rely on more static and historical information, which can thus be summarized as Surveys or Estimates. As seen in the Indicator example of Section 4.3, Gis data becomes crucial for the automation of the input processing required in some of the Indicators. The fact that Gis information can be queried through python libraries such as GeoPandas[6] makes the automatic parsing of these Measurements for any required composition considerably easier. Additionally, we consider Building Information Modeling (Bim) Measurements, which may be useful for calculating some of the Indicators in the Energy sector.

As seen in Section 4.1, one of the key requirements for Scoup is to allow for the comparison of different cities or communities through the usage of common indicators. As previously presented, these indicators can be modeled by a common taxonomy (Figure 4.3) and one mode of organizing the different data sources involved in Smart City assessment is to structure them in a Composite Pattern [Gam+95]. We keep this relation for the meta-model in Figure 4.4, as the DataContainer composition element allows the for the building of more complex Measurement types, that encapsulate two or more fo the other Measurement types, thus allowing for more detailed analysis. When relating the City to Measurements, we save the available sources as a list of Measurements of the City. As a logical constraint of the Scoup framework, a City will report on an Indicator if and only if it has all of the data sources required by its measurement method. This bi-conditional logical connection has the advantage of allowing for an efficient query among the different sources.

In keeping consistent with Section 4.3, we model how Indicators contain the Constraints that were represented using Ocl notation and how Indicators are able to send a notification to the Scoup framework when that Constraint is broken. While we recognize the need of having Constraints integrated into the Scoup meta-model, we model these simply as Ocl constraints and revisit this concept later in Section 5.3.

To improve the performance of the measured Indicators, a City generally provides different types of Urban Initiatives. These Urban Initiatives require different types of resources from different stakeholders, which come into use at different steps of their workflow. The procedure that the Urban Initiative needs to go through is represented

[6]GeoPandas library allows for Python queries on Gis data: `https://geopandas.org/`

by an event flow, which is carried out by participating objects and stakeholders. Note that these participating objects can be virtual components, edge devices, or even human stakeholders. Urban Initiatives are often dependent on City attributes, and thus the City aspects relevant for the Urban Initiative may get stored as an additional context. Likewise, when planning an Urban Initiative, a City must consider the available options and how this context affects the policy decision. This modeling is inspired by the Strategy Pattern [Gam+95], even if the pattern is not completely applicable as it is usually hard to switch in real-time between Urban Initiatives. The fact that the terminology of one of the key design patterns described by Gamma *et al.* [Gam+95] is so applicable to the urban environment (even if the execution conditions are not), is seen as further evidence of the relation between Cities and complex Systems. Besides the standard Urban Initiative, we also consider Virtual Initiatives, which are provided by an Application Programming Interface (API).

An example of an Urban Initiative would be the covering of potholes in a City or region, where it is first necessary to collect the location (part of the context) of the potholes (a participating object), and then a vehicle (new participating object) should be deployed to that location and pours asphalt (new participating object) to cover the pothole. The vehicle may then continue to the next location, of the next pothole in its description.

A Virtual Initiative, on the other hand, is made available through an API which can be accessed by external stakeholders. It is relevant to note that while these will generally have fewer participating objects and only expose a data source, they are consistent with the initiative definition since they allow progress for indicators presented in ISO 37122 such as “Number of datasets offered on the municipal open data portal per 100 000 population” or “Number of online databases available through public libraries per 100 000 population” [ISO 37122], these contribute to SDG 16[7].

[7] `https://sustainabledevelopment.un.org/sdg16`, which includes accountable and inclusive institutions at all levels

4.5 City Input Process

To compare cities among each other it is first necessary to bring the available measurements into a common format and thus Scoup allows for a form-based input of information. This process can be seen in Figure 4.5. When an officer first registers a City, Scoup tries to automatically request the City meta-data defined in Figure 4.4 from the internet, through publicly available open data portals[8]. It then initializes the Urban Initiative, Indicator, and Measurement relations, and ask the city officer for the upload of any available measurements.

Since the list of available indicators is saved into Scoup, the system knows what are the required input data for each of the indicator. When a city officer inputs measurements into the system, Scoup automatically cross references the input data against the variables required for each indicator and informs the user which indicator would be feasible to calculate. The identified Indicators with the highest number of available input are then suggested to the user, and if any Indicator has almost all required data input into the system but has a few fields missing, Scoup then informs the user about this and allows them to input the missing Measurements.

As far as Urban Initiatives are concerned, this input process happens after the Measurements, possibly independently from it. After Scoup has processed what are the existing Measurements for the City, it allows the officer to input Urban Initiatives into the database. As a result, the fields in Figure 4.4 are presented, so that the officer can describe the flow of events of the Urban Initiative. Scoup again tries to identify which Indicators could be affected by this initiative and presents these to the user for confirmation. If all necessary Measurements for this Indicator are already available in the system, Scoup creates the correlation between the Urban Initiative and the Indicator and tries to derive the impact of the Urban Initiative on the Indicator.

[8] For example: `https://public.opendatasoft.com/explore`

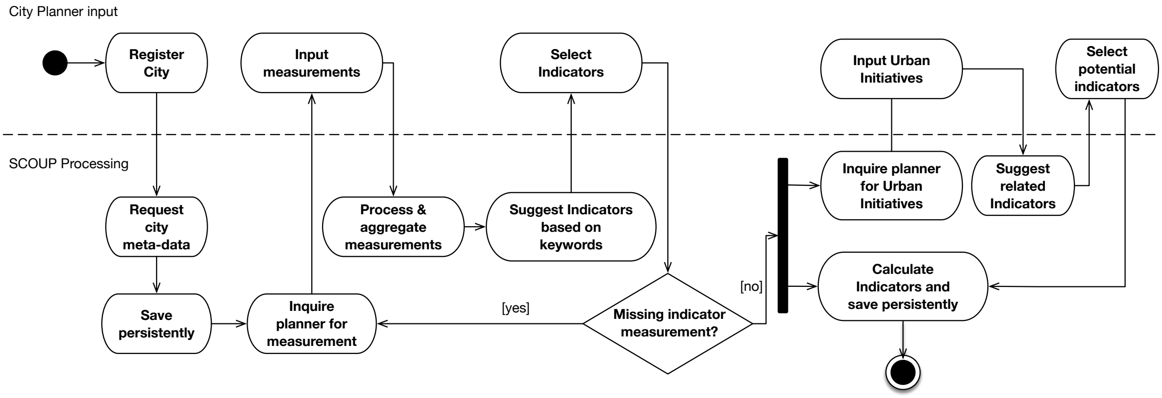


Figure 4.5: Dynamic Model of how to input a new City, Measurements and Indicators into Scoup (Uml Activity Diagram).

4.6 SCOUP Overview

After presenting a first version of Scoup, the following chapters apply the framework to two different mobility projects, one from the perspective of the citizen or traveler and a second one from the perspective of the community. At the end of each chapter considerations are made on how to extend Scoup based on the lessons learned from each chapter.

We provide an overview of how the presented related work and problem investigation build into the core components of Scoup and summarize them. From the Cps background it can be derived how decentralized architectures such as fog systems can improve scalability and real time measuring of a vast amount of information. Furthermore, it shows the relevance of of data integration, and how virtualization concepts can help in that direction. The analysis of the current state of Smart Sustainable Cities leads not only to a concrete definition of what should be aimed for, but also to the selection of Iso standards 37120 and 37122 for comparisons among regions. As far as constraints that affect each region are concerned, these are different in each aspect of urban life. While an attempt to formalize constraints has been presented above with the usage of Ocl, this is revisited in Section 5.3.

These analyses are brought together by Scoup that integrates them into three components: (1) the Ocl formalization presented above, (2) the the meta-model applied and validated in the following chapters, as well as (3) urban development software components that were developed for both of the reference applications. This is presented in Figure 4.6. The components of Scoup are applied to two reference applications in the following chapters and at the end of each a discussion is presented as to what extent Scoup was able to satisfy the applications' needs, and how it can still be extended.

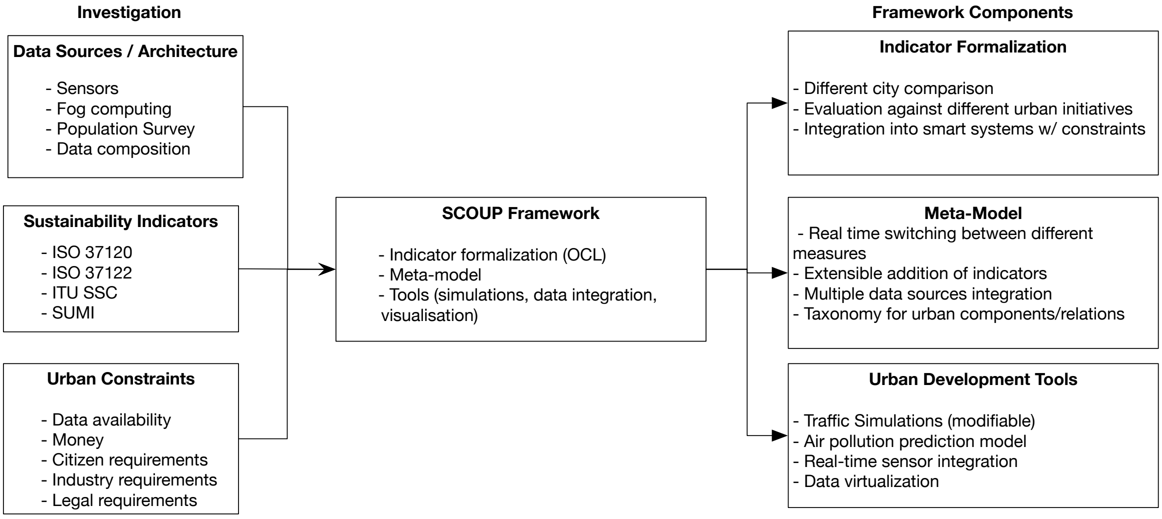


Figure 4.6: Depiction of how the problem investigation relates to the components of the SCOUP framework.

Chapter 5

Urban Routing

5.1 Problem Investigation 58
5.1.1 Functional Requirements 59
5.1.2 Related Work 60
5.2 Reference Application: Multi-Modal BR Router 61
5.2.1 Route Cost Function 64
5.2.2 Personal Constraints 66
5.2.3 Mode of Transportation Detection 68
5.3 SCOUP Extension 70

The framework and tools presented in the previous chapter allow for an analysis of the urban planning process from a region perspective. However, while computing the shortest path from A to B has been long researched with the use of classical algorithms such as Dijkstra [Dij59] and A^* [HNR68], the personalization of routes continues to represent several computational challenges. This chapter investigates how personal constraints can be algorithmically computed, the effect that these have on Smart Sustainable Cities (SSC) indicators, and what changes are necessary to SCOUP to integrate these. A multi-modal router is presented as a reference implementation, that applies the SCOUP concepts and presents an approach to solving route personalization.

5.1 Problem Investigation

Since Dijkstra proposed his shortest path algorithm in 1959, routing remains a central problem of computer science. While Dijkstra's algorithm guarantees to always deliver the shortest path between two nodes A and B in a given graph, it's runtime of $O((|V|+|E|)log(|V|))$ for V vertices and E edges in the graph can become high for a city map with hundreds of thousands of streets and intersections as vertices and edges [Moeh+05].

Among the improvements made to Dijkstra's algorithm is the A^* algorithm, which uses a fast computable heuristic to estimate the distance from each node to the destination node, thus first researching the edges that are more likely to return the fastest route [HNR68]. Generally, the heuristic necessary for the A^* is implemented in the form of the euclidean distance between the current node and the destination. This does ensures that in the average case, the runtime is faster when compared to Dijkstra's algorithm. Nonetheless, neither of the two algorithms succeed in taking into consideration any personal preference from the travelers nor do they directly integrate different modes of transportation into one route.

Multi-modal route planning is the process of computing a route from A to B so that different modes of transportation are used within the same route. In theory, any public transportation route that uses a combination of walking, bus, and subway could already be seen as a multi-modal route, but in practice routes that integrate modes such as bikes or ride sharing are more useful to consider. While multi-modal routes are harder to compute as they consider a higher number of vertices in the graph and their combinations, they also offer an optimization potential on both route duration and sustainability impact. Classical multi-modal routes include cycling until a specific subway station and thus requiring one public transport transfer less in a given trip.

Example 5

A concrete example for a multi-modal route could include the scenario of a sports fan driving to watch a match. One option would be driving the car to a "park & ride" subway station, and continuing onto the game arena with public transport. On a game day, this could highly decrease the parking search time, and thus represent an important optimization.

One of the relevant limitation of routing algorithms is that they classically approach the computation of routes from an individual traveler's perspective, which may or may not be the best route from the cities perspective. While for the traveler the most time-efficient route may include a transfer in a busy interchange station, for example, where she could change to a different line, reducing the over-crowdedness of these interchanges is often an important objective for city planners.

Observation 6

Current routing algorithms optimize specific routes, without taking urban conditions and impact into consideration. The integration of such considerations first require additional information.

A system that allows for a city to optimize their complete mobility profile, will usually require several other variables to be taken into consideration, and thus we derive the following functional requirements for the *Bayerischer Rundfunk* (BR) system (BR-FR):

5.1.1 Functional Requirements

BR-FR1 **Calculate urban indicator:** The system should allow for the calculation of indicators that measure the complete mobility profile. While this should take into consideration the routes each traveler is taking to their destination, it should aggregate meta-data about the routes into city-wide statistics, and thus not focus on the individual traveler.

BR-FR2 **Personalize routes:** The system should allow for traveler-specific input to be taken into consideration when calculating an urban route from A to B. A route should be personal in the sense that each traveler has their own preferences as far as mode of transportation and their usage is concerned. Ideally, this should include individual constraints, such as a maximum biking distance on a given route.

BR-FR3 **Integrate different mobility providers:** Different mobility providers such as public transportation, bikes, bike sharing and their integration should be taken into consideration when calculating routes. These multi-modal routes should be related to the user's individual preferences as defined in BR-FR2 and the modular addition of different providers in the future should be supported.

BR-FR4 **Integrate different data sources:** As routes should be calculated from a city perspective, it should be possible to integrate urban data sources, such as weather information which may affect cycling conditions or road traffic and construction states which may affect driving time.

BR-FR5 **Collect feedback about usage:** Travelers should be able to provide simple feedback to the city, if the suggested route was good or bad. This should be seen as a further data source about the quality of the routes and can be used to adapt the system in the future.

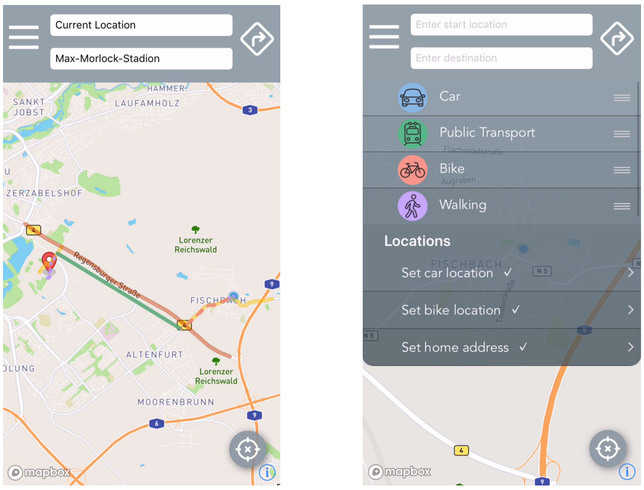
5.1.2 Related Work

While some of the requirements above pose new challenges in the field of urban routing, individual aspects can be partially found in some existing algorithms or applications, which are now presetned.

Beyond Dijkstra and the A^* algorithms already presented above, on the list of static routing algorithms, recent years have seen the introduction of Contraction Hierarchies [GSSD08]. This algorithm adds a pre-processing layer that contracts important edges (such as highways) into single nodes, thus allowing for a significant performance and memory speed up. These algorithms are implemented into several modern applications such as Google Maps [Sve10] or OpenRouteService [NZ08] and through these provide routing from any given point A, to any given point B inside of an urban road network. While computationally these seem to be solved problems, in reality a traveler will take other factors into consideration when planning their personal route from A to B. With this in mind, multi-modal and personalization algorithms started to be recently suggested.

Major works in relation to multi-modal routing include the open source project Open Trip Planner (OTP) [NT16], in which public transit routes can be integrated with other modes such as cycling. The application uses a modification of the A^* that allows for exchanging between the road network and the public transportation network on each station.

The work in OTP represents some of the state of the art methods as far as multi-modal routing is concerned, but presents only limited capability as far as personal preferences of the routes goes. In order to advance this research, Bucher *et al.* propose a heuristic based procedure, where individual preferences are stored as constraints that are applied to the route [BJR17]. These constraints are defined as user and mode of transportation specific, and the authors postulate how variables such as weather conditions can affect, for example, the eagerness of a given traveler to cycle for part of the trip. While the work by Bucher *et al.* is considered state of the art as far as personalization of multi-modal routes is concerned, it still suffers limitations related to the impact of the calculated routes.



(a) (b)

Figure 5.1: Multi-Modal BR router. On the left: A calculated multi-modal route consisting of both bike and public transportation segments. On the right: configuration screen of the modes of transportation available as well as what are the main locations relevant to the user.

5.2 Reference Application: Multi-Modal BR Router

To tackle the problem of multi-modal routing, a system including a router and an iOS application was developed during the iPraktikum in partnership with the public broadcasting service *Bayerischer Rundfunk* (BR). A screenshot of the iOS application can be seen in Figure 5.1. In the developed application, routes were calculated by extending upon the presented OpenTripPlanner (OTP) project, with a focus on personalization options which aim to make a given route more useful. From a traveler's perspective, a multi-modal trip is considered "useful" if the resulting route is either faster, cheaper, or more sustainable than the traveler's usual commuting alternative. Furthermore, it is relevant to note that travelers have different route preferences in regards to their work commute, or normal recreational activities. In the latter case they tend to be much more attentive to environmental factors such as air quality or noise, and thus consider other variables than only time or distance for their route [BCHK19]. From thee perspective of a city implementing the SCOUP framework, a multi-modal trip is considered useful if it improves the performance of one of the ISO indicators.

In order for any of these things to be feasible, however, the relevant data should be available, which is often a limiting factor when calculating multi-modal trips. Conventional cities distinguish between four main modes of transportation: (1) walking, (2) cycling, (3) public transportation, and (4) individual cars. While new modes of transportation – such as motorcycles, electric scooters, ride sharing options, etc – are becoming more present, integrating these new modes of transportation into routing applications is often limited by public data about their infrastructure and usage. Even when looking at the four main modes of transportation, data availability already highly depends on the city being discussed.

While some cities publish their public transportation schedule openly online under the General Transit Feed Specification (GTFS) format [AB13], in other cases an individual request to public authorities may be necessary, which may or may not work. The presented OTP project requires GTFS data to calculate trips on the public transportation network, which usually is a vital aspect of most multi-modal traffic reduction measures. While it is often common to assume that routes for bicycles can be calculated using the normal car road-network, this is not always true in the real world. In practice, some roads – such as highways – are not feasible for bikes, while pedestrian passages are off-limits for cars, but often acceptable for cyclists. This road-level classification becomes highly location specific, and an integrated approach to this problem is presented by the OpenStreetMap (OSM) group, which rely on tags for different road types. Even then different stakeholders may often have different opinions about which tags are correct for a given road [MC12].

As a test environment for the BR application, the Bavarian city of Nürnberg was chosen, due to their GTFS public transportation schedule being publicly offered online[1]. Other than public transportation, personal bicycles were integrated as a mode of transportation. Since this application tackled the problem from a traveler's perspective (BR-FR2), attention was paid to offer the maximum customizability of routes. This means that enabling, disabling, and configuring of individual modes of transportation is made possible on a per-trip-basis, and a graphical user interface is provided for doing so. Figure 5.1 shows the route and settings interface of the developed application, and Figure 5.2 displays the subsystem decomposition of the system developed for BR.

Since the application was developed in cooperation with BR, radio-based traffic information was available, which provided an extra layer of information to drivers. One of the challenges in the development of this application addressed by the SCOUP framework was the integration of different data sources. While OpenStreetMaps (OSM) data provided the basis for the map and route calculation, these were first integrated to the Nürnberg GTFS, in order to allow for routing on the public transportation network. Traffic data, presented by BR through a File Transfer Protocol (FTP) server, allowed some roads to be flagged as too congested, but the difficulty of integration meant that these congested roads were considered for routes nonetheless. The main reason for

[1] `https://www.vgn.de/web-entwickler/open-data/`

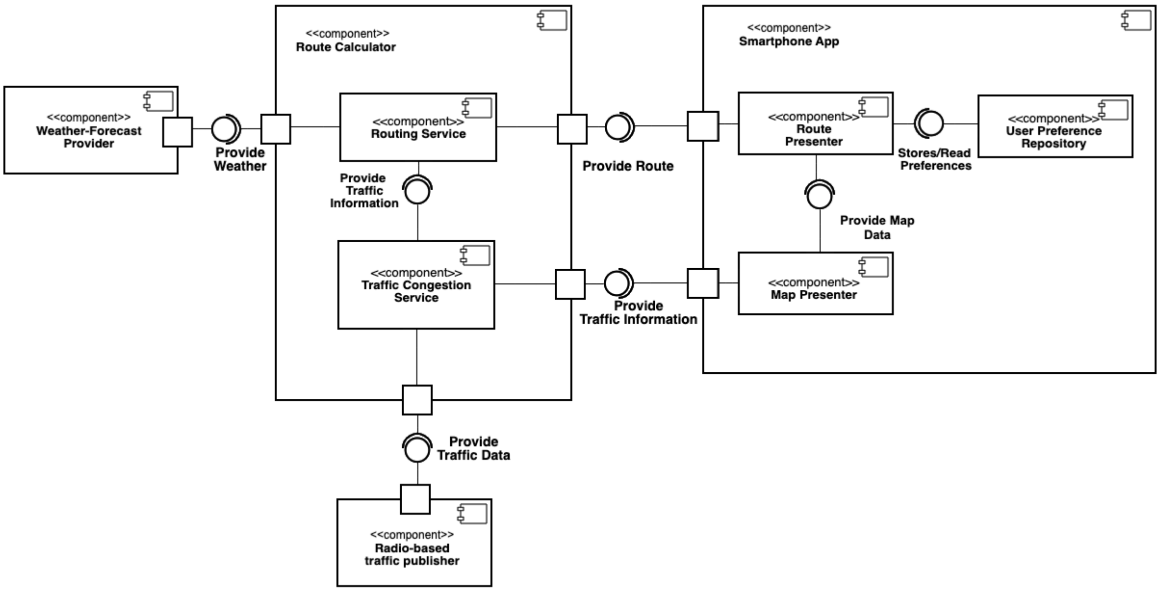


Figure 5.2: Subsystem Decomposition for the Multi-Modal Br Router (Uml Component Diagram).

this was the different road indexing used by Osm and by the traffic data reference system. The fact that these were in different formats, meant that a transformation to a common model was first necessary, which was then saved onto a database, which was then accessible to both the routing service and the iOS application for traffic requests. While the provided data was displayed in the Br app, the routing service only flagged the affected streets, as this request was only made after the calculation of the route.

Furthermore, weather data was also present, which allowed for a user-specific personalization of how this should affect routing capabilities. An example scenario would be to avoid cycling under rain. While ideally these preferences should be user-centered, the time restrictions of the project meant that this had to be implemented as a general rule, server side.

From a routing perspective, the capabilities of the Otp project were extended to include some basic personalization options, such as saving the location of the user's car or bike and then starting the route from there. These were packed into a routing service component that ran in the router server. A user-configurable maximum biking distance was implemented. These features were integrated both on the iOS application, as well as in the Otp server. The traveler configures these preferences in their iOS application and these would be encoded as route request parameters to the Otp-based router component.

As stated in Br-Fr3, one of the goals of the developed system is the integration of different modes of transportation. While Otp already provided support for cycling and public transportation integration, the same is not true for options such as ride sharing and electric scooters. From an urban perspective, however, the integration of these modes is highly desirable as in several trips they represent feasible alternatives to personally-owned vehicles. To enable such an integration, it is necessary to analyze what are the routing challenges in this direction. Ride sharing, for example, has elements both of personal vehicles –such as the usage of the road network–as well as taxis or even public buses, where a pre defined route is available and passengers request an extra stop. By combining the formats and methods of these individual modes, it is possible to easily set up new ride sharing as a new mode of transportation. This represents a clear example of how inheritance and composition can be used to extend existing system limitations.

An initial version of this system was implemented during the iPraktikum capstone project [BKA15] of 2018. During it, a group of 8 students developed the system which is described above, and the application depicted in Figure 5.1. After the capstone project, several aspects of the system were extended through final theses. The analysis and results of these researches are now presented.

5.2.1 Route Cost Function

Beyond the extension of routing options to include different modes of transportation, one of the aspects raised during testing of the application is how each person takes into consideration different variables when choosing a route. To handle this, we developed a cost model function together with Joseph Saroufim and the *Industrieanlagen-Betriebsgesellschaft mbH* (IABG, see Section 6.2) that considers more complex variables than time or distance for route planning [Sar19].

In order to analyze the cost of a trip, each multi-modal route was associated with different modes of transport, which could either be car-based, or mobility-based. Mobility-based modes were further subdivided between public transportation and mobility services offered by the private sector, such as commuter buses or ride hailing services. The cost function was then implemented by a commuter bus project, which we detail in Section 6.3.

To calculate the complete cost of a multi-modal route, we divided the route into several segments, each with a single mode of transportation. For each segment, we calculated a total cost to it, and the sum of the cost of each segment comprised the route cost. As far as variables per segment are concerned, we defined the segment cost as the sum of the environmental cost (measured in grams of CO_2 equivalent), monetary costs (measured in euros), time cost (measured in minutes), and distance cost (measured in kilometers). While each of these costs is added, it is relevant to note the dependency that both time and distance have in the other variables. The time, monetary and environmental costs are all dependent on distance and mode of transportation. While the time cost was calculated algorithmically the monetary and environmental costs were based on a literature review, which is now presented.

For the monetary costs of each segment, further consideration was necessary. While the public transportation provider in Munich, the *Münchner Verkehrs- und Tarifverbund* (MVV), has a complex price structure, a rough average price of 0.12 $\frac{€}{km}$ can be estimated based on the total revenue and travelled passenger kilometers provided by the public transportation company [Mvv]. For cars, however, the monetary model needs to take several costs into consideration, and thus the total cost of ownership analysis by the *Allgemeiner Deutscher Automobil-Club* (ADAC) was used. This analysis calculates the total cost of ownership for different levels of cars, while considering their maintenance, depreciation value, insurance and operational costs. A complete overview of the different costs per km based for each car type can be found under [ADAC19].

For the environmental costs, a life cycle analysis from the Swiss organization mobitool was used, available under [FMST16]. Mobitool offers a model that considers production, fuel consumption, usage, operation, maintenance and disposal costs in grams of CO_2 equivalent, for a variety of modes of transportation [FMST16]. This information is then divided by average consumption rates to get a grams of CO_2 equivalent value per passenger kilometer [FMST16].

SCOUP allowed now only for all of these variables to be taken into consideration, but also to address BR-FR4 and aggregate them for the observed trips. Our framework furthermore facilitates the calculation of some of the ISO 37120 indicators, such as “Percentage of commuters using a travel mode to work other than a personal vehicle”, or “Average commute time”.

The calculation of these indicators help the transportation analysis from a city perspective, yet our work with Saroufim in [Sar19] reveals further variables that could also be integrated into a modular cost function as the one developed. An example of such variables include the time value of the trip and the psychological impact it has. For example, a train commute where you can read tends to have a greater time value than time spent searching for a parking space. Varotto *et al.* presented a work measuring and quantifying this effect [VGSB14], but the lack of this information for the Munich scenario meant that the integration of this into the cost model remained future work. Furthermore, the flexibility of a trip could also be taken into consideration by such a cost function, as defined by Abkowitz *et al.* [Abk81]. Abkowitz *et al.* analyze which variables affect time planning decisions for commutes, and formalized how the expected loss for the destination arrival influences the selection of mode of transportation [Abk81].

Together, these variables allowed for an analysis of how transportation mode selection allows for a transportation profile for a city or community to be calculated, and thus provide initial measurements for indicator 19.8 of Iso 37120: "Transportation profile indicators".

5.2.2 Personal Constraints

To allow for such a magnitude of variables to be integrated into the actual route calculation, a model was then developed of how to define flexible constraints during the route calculation. As described in the Problem Investigation, several routing algorithms currently exist that address both personalization and multi-modality aspects. In the following we present the results of a work in cooperation with David Budischek, where a taxonomy was developed to integrate personalization constraints into the OpenTripPlanner (Otp) project [Bud19].

To personalize routes based on individual preferences and constraints, it is first necessary to define a route constraint. The taxonomy illustrating this can be seen in Figure 5.3. Any constraint must invariantly be applied to something in order to be fulfilled. A route constraint limits the route calculated over a graph in a given way. Together with Budishek we developed a taxonomy, where a route constraint is composed of further constraints, forming a composite pattern [Bud19; Gam+95]. The leaf node of this pattern is then composed of one condition value, which may limit the time or distance travelled through a maximum value, and an identifier. This identifier determines which edges to be limited by the constraint, and the combination of different identifiers allows for constraints such as "Don't cycle when it's raining" to be implemented, as rain can be defined as a combination of a certain area during a certain time.

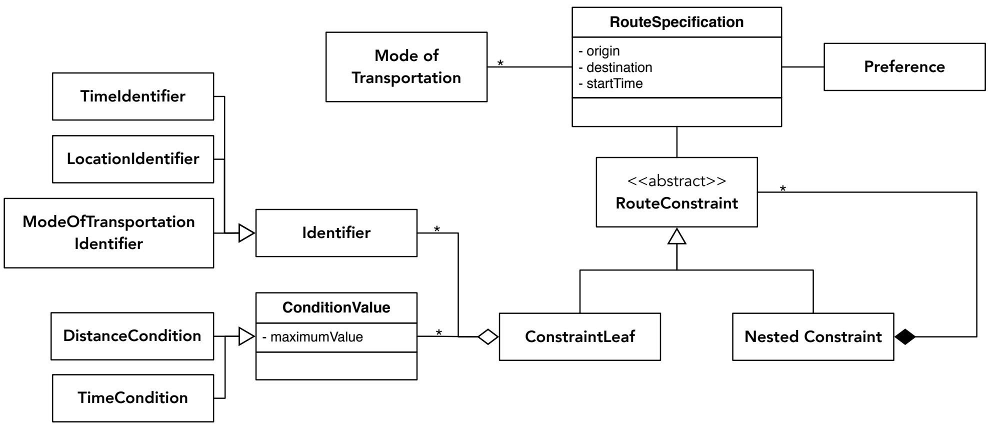


Figure 5.3: RouteSpecification relations and constraint taxonomy (Uml Class Diagram). Adapted from Budischek [Bud19].

The conceptual difference between identifiers and conditions is important. While identifiers relate to the edges of a graph to be ignored, the conditions relate to the usage of these edges. For example the constraint “Cycle for a maximum of 5km” contains a mode of transportation identifier and a distance condition. “Cycle for a maximum of 15min” analogously contains a time condition.

These constraints are evaluated during the routing algorithm, which is then extended so that each state being considered for the calculated path checks if it fulfills the constraint, and is only further considered in the priority queue if that is the case. If the constraint is broken at a given state, the state is then removed from the queue, and thus not further considered for the route calculation. This constraint evaluation was implemented as an extension to OTP[2].

As can be seen in Figure 5.3, the condition value element contains a maximum value, but not a minimum value. This is due to the graph theory limitation that checking if a state (and thus current path) fulfills a constraint in the past (i.e., has surpassed the maximum) is considerably easier than enforcing a state to fulfill a constrain in the future of the execution, which would be necessary for a minimum value consideration. Biking for maximum 5km is easy to check against, as you only need to validate if all the bike segments to this point add up to 5km. If the constraint were to be cycling for a minimum of 5km, in the other hand, it would be necessary to enforce a limitation on the next states that get added to the priority queue of the A^* algorithm, which would thus invalidate the entire concept of the priority queue. This is considered a technical limitation, and thus minimum constraints are not supported.

Through the concatenation of several conditions and identifiers in the composite pattern, this taxonomy allows for a greater personalization of the route, evaluated during the route calculation. While this covers a wide variety of constraint cases, it is interesting to note how the time component seems redundant, as it is present once as an identifier, and once as a condition. The identifier usage is meant as a method to allow for rush hour or weather conditions considerations, while the condition value is meant as the time equivalent of distance values, such as a 15min limit. The real time consideration of constraints is an relevant contribution of [Bud19], however the obvious confusion caused by having two time elements is addressed when integrating these constraints to SCOUP, in Section 5.3.

[2]Source code available under: `https://github.com/Budischek/OpenTripPlanner`

5.2.3 Mode of Transportation Detection

After calculating a personalized route for the individual traveler, the city may be interested in knowing if the travelers actually used the suggested route. To address Br-Fr1 and Br-Fr5 from Section 5.1.1 and collect aggregated data about the usage of the application, a mode of transportation detection algorithm is now presented. By developing an algorithm to detect if the calculated route was actually taken, it is possible to gather data for the following Iso 37120 indicators: (1) "Percentage of commuters that use other travel modes than personal vehicle", (2) "Annual number of public transport trips per capita", and (3) "Average commute time".

To allow for this calculation, we first had to guarantee that the calculated mode of transportation was actually used to travel the route. Since the described router was implemented as a mobile iOS application, it is reasonable to use the sensors available on the smartphone for this prediction. In a collaboration with Jens Klinker we developed a series of models to predict which mode of transportation were used, based on mobile Global Positioning System (Gps) data [Kli19]. The results of this work were published under [AKB19] and are now described.

Initially a Gps tracker application was used to collect 128 individual trips from 17 different users, who classified their trips into 5 classes for modes of transportation: (1) Walk, (2) Bike, (3) Car, (4) Ubahn/Sbahn, (5) Tram/Bus. As the study was carried out in the region of Munich, the Ubahn and Sbahn public transportation options available in the city were considered. An Ubahn is considered a shorter distance (usually underground) subway, while the Sbahn in the city has long overground components, and a more regional and long distance aspect to it. This data collection resulted in 56000 individual Gps points, which were then used as input data for a classification model [AKB19]. As a focus is kept on multi-modal trips, it is the aim of the prediction algorithm to correctly identify modes of transportation also for trips that contain several modes. To do so, we rely on the work of Bobol *et al.* in which the authors identify a transition matrix between each mode of transportation [BCTH12].

Bobol *et al.* present how in over 87% of the cases, each mode of transportation is followed by a walking segment in the trip [BCTH12]. Intuitively, this makes sense as a traveler usually walks between two modes of transportation when switching between them. This observation is then used to split the collected trips between walking and non-walking segments. To classify the walk segments and thus split the trip, the approach from Bobol *et al.* [BCTH12] is adapted so that a data window of 60 seconds traverses the entire trip, and classifies each window as a walking or non walking segment. This resulted in a decision tree with only two classes, walking and non walking, which was trained using average speed, as this feature was identified as having the most accurate results. It is interesting to note that the decision value calculated for this node is consistent with references values for walking speed found in literature: $8.8\frac{km}{h}$ [FBT06; AKB19]. This simple test based on the average speed achieved a 94.6% accuracy when labelling walking segments [AKB19].

Upon segmenting the trip, the following features are calculated for each single mode segment: Average duration between GPS data points, maximum duration between GPS data points, average speed for the entire segment, maximum speed, maximum acceleration, and maximum deceleration. The maximum features, however, were defined as to not be the absolute maximum speed or duration of the trip, but rather the 99% quantile, so as to not be biased by possible data outliers. As with as the previous features, it was analyzed that calculating the 85% speed quantile also proved a valuable feature, as it facilitated the distinction between cars and long distance trains.

After having calculated all these features for each single mode segment, a decision tree was trained with the labeled data set, and the resulting model can be seen in Figure 5.4. The feature's value in each node of the tree is used to make a decision. If the feature in the trip is smaller than the value in the node, the left branch is further considered, and further decisions are made, until a mode of transportation leaf node is reached. If the values are bigger than or equal to the value of the tree's node, the right branch is further evaluated.

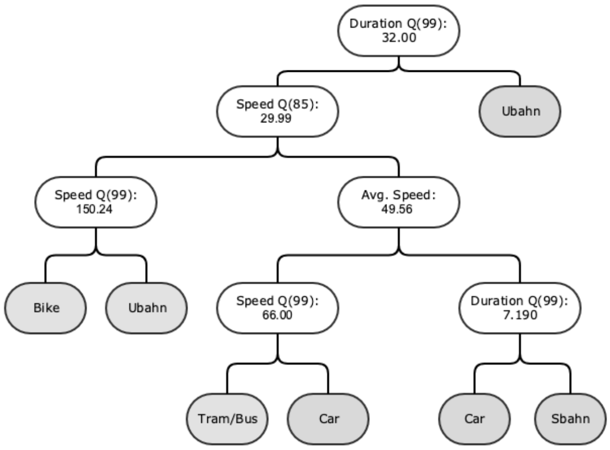


Figure 5.4: Mode of transportation decision tree. Source: [AKB19] © [2019] IEEE.

Running a validation set against the calculated decision tree produced an accuracy of over 80% for bike, Sbahn, and tram/bus predictions. The biggest confusion was when predicting cars, which often got mixed with Sbahns, due to their similar speed and acceleration profile. Figure 5.5 shows the final achieved confusion matrix between the predicted and actual modes of transportation for the model presented [AKB19].

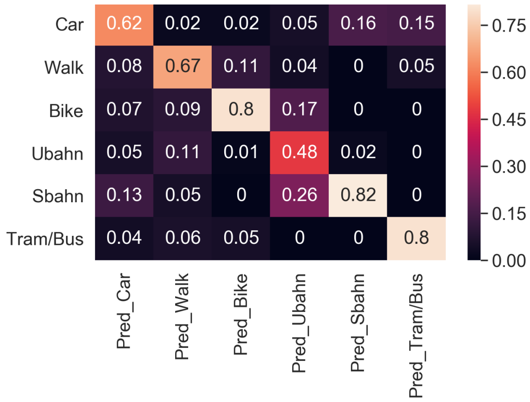


Figure 5.5: Confusion matrix representing the accuracy of the decision tree model achieved. Source: [AKB19] © [2019] IEEE.

5.3 SCOUP Extension

Together with the multi-modal routing application, the extensible route cost function, the personalization of constraints and the mode of transportation recognition algorithm represent additional artifacts that were developed with the usage of the Scoup framework. We now analyze how well the Scoup framework suited the development of these artifacts, and what changes are necessary to the framework to more accurately depict the development needs.

As presented in the previous sections, one of the aspects missing in Scoup for the development of this artifact was the consideration of constraints. In order for an application developed with Scoup to be useful from a personalization – and even sustainability – perspective, it is important for it not only to present the different information and analysis available, but also to allow for system constraints to be inserted. Based on the lessons learned from the Br application, we now consider how to define such constraints.

As seen in Section 5.2.2, the concatenation of constraints allows for the consideration of more complex scenarios, such as avoiding an area and mode of transportation at the same time. This led to the realization that the insertion of constraints should be accompanied by the composite pattern, to allow for more complex constraint structures. When inserting the composite pattern, however, it is important to analyze what the leaf node should contain.

From Section 5.2.2 we can observer how separating a constraint's identifier and value can be beneficiary. However, the implementation showed to have some confusion in the distinction between the two, as could be seen in the time aspect of the constraint. To improve upon that, we describe each concept. A constraint identifier relates specifically to the aspect that gets limited by the constraint. This should be

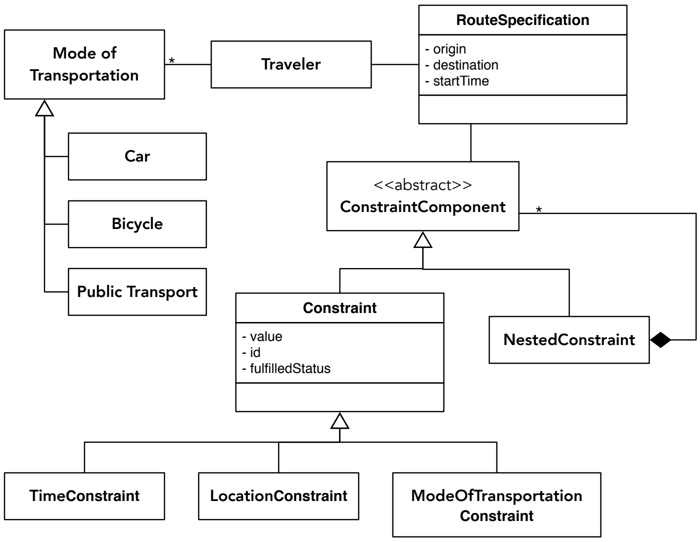


Figure 5.6: Updated multi-modal Br router analysis object model after refactoring of constraints abstractions and considerations of the Scoup meta-model (Uml Class Diagram).

as specific as possible and in the context of cities it is thus feasible for it to relate to the measured indicators. A constraint's value should relate to a concrete attribute that can be measured. This can be a number representing the constraint's threshold or a simple state that is passed or not passed. A value can thus be either boolean or decimal and should be as concrete as possible. The combination of value and identifier results in the constraint's status, which is a boolean attribute that defines whether the constraint has been fulfilled or not. When applying these changes to the Br multi-modal router, the Analysis Object Model is thus updated to Figure 5.6

The cost function in Section 5.2.1 expressed the possibility of addressing sustainability and route planning from a single traveler's point of view, as these may have preferences that must be considered by the urban infrastructure. In order to support these applications, a new preference class is inserted into the Scoup meta-model, which is depicted in Figure 5.7. The relation between preference and person is inserted. Furthermore, it is interesting to note how the relation between city and indicator is analogous to the relation between person and preference, as a preference can also be controlled by a constraint, and a person's preferences can ideally be measured, either by their own input or by data available through the city. For simplicity reasons, the meta-model depicted in Figure 5.7 does not display the previously described subclasses of measurement and urban initiative abstraction, since their attributes and usages has not been modified.

As discussed in Section 5.2 and required by Br-Fr3, one critical aspect of the developed multi-modal application is the possibility to integrate several data sources.

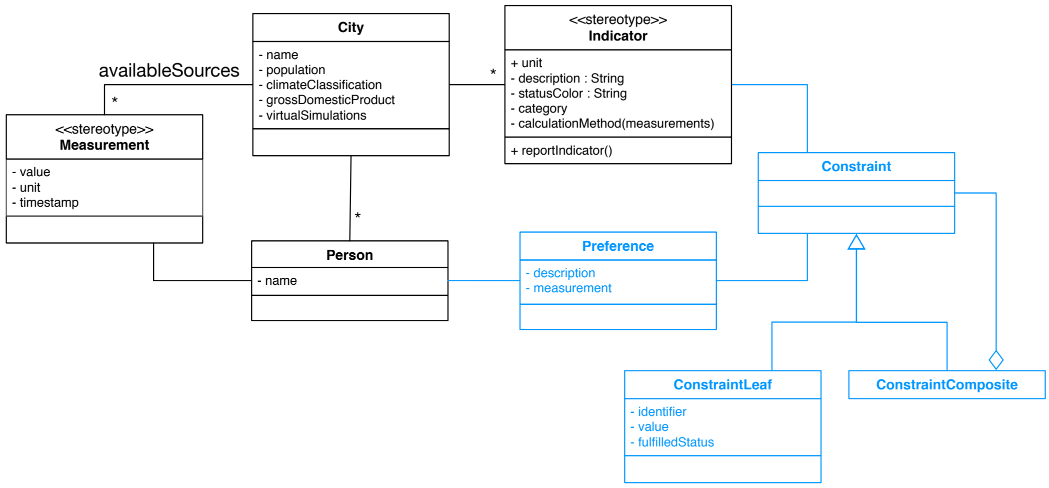


Figure 5.7: Updated section of the Scoup meta-model after the insertion of constraints and preference abstractions (Uml Class Diagram). Extensions made to Scoup now are marked in blue.

An application that supports multi-modal routing should ideally support all modes of transportation available in the city and be scalable in support for new modes of transportation. The Br multi-modal application made use of the different variants of the Gtfs format to support this. While current application supports only public transportation, cars, and private bicycles due to the limited data availability, the format and project it is built upon can integrate bike and car sharing options. This shows us how aggregating different data sources into a single Data Container (as allowed by the Scoup meta-model in Figure 4.4) makes it easy to add further data sources. Nonetheless, it is relevant to note the role that the open source community plays in such aggregation. One of the main reasons why the integration of different modes of transportation was possible in this application was the extensive support of the Gtfs format which allowed the encapsulation of several different modes of transportation into a single standardized format and data file, thus making it's integration into the project considerably easier.

One threat to validity that can be derived from the implementation of the Br multi-modal application is the lack of usage of the sustainability aspects of Scoup. While this is partially due to the individual's focus of the developed application – rather than the city's perspective – it is important to consider how to motivate citizen's behavior as an urban initiative, which should address sustainability aspect and be measured by urban indicators. This highlights the observation that simply measuring data is not enough to classify a community as a Ssc, and that active initiatives should be pursued. In the interest of addressing the sustainability aspects from a city's perspective, we now apply Scoup to a project implemented on the scale of a small community.

Chapter 6

Smart Mobility

6.1 Problem Investigation 74
6.2 Kirchheim Stakeholders 78
6.3 Commutify Initiative 79
6.4 Reference Application: SmartHeim2 82
6.4.1 Data Availability 83
6.4.2 Simulation 84
6.4.3 Traffic Flow Sensors 88
6.4.4 Air Quality Sensor 90
6.4.5 Indicator Analysis 92
6.5 SCOUP Extension 94

In the previous chapter we described an approach to consider personalized preferences for transportation options. We now analyze a similar problem from a city perspective and examine what sustainability considerations should be taken into consideration. One expectation of Smart Sustainable Cities (SSC) is to enable efficient transportation optimizations [BDD16]. Benevolo *et al.* gather six main smart mobility objectives: (1) reduce pollution, (2) reduce traffic congestion, (3) increase people's safety, (4) reduce noise pollution, (5) improve transfer speed, and (6) reduce transfer costs.

While these objectives have been long followed by city governments, comparing initiatives that pursue them from around the world continues to be an open problem. Reduction of traffic pollution and congestion has been the objective of systems developed in big cities across the world such as Singapore[1], London[2], or São Paulo [Swi+14]. With the goal of developing a system that could be extended to different

[1]National Geographic: `https://www.nationalgeographic.com/environment/urban-expeditions/green-buildings/green-urban-landscape-cities-Singapore/`

[2]Here Mobility: `https://mobility.here.com/learn/smart-city-initiatives/london-smart-city-tackling-challenges-20-initiatives`

regions and to solve the problems described in the previous chapters, the German Chamber of Industry and Commerce (in the German original: Industrie- und Handelskammern, IHK) started a pilot project to increase traffic connectivity and better measure it's values in the outskirts of Munich[3].

We describe the system developed for IHK, and how this relates to the problems associated with smart urban management. In Section 6.1 we describe the mobility problems to be addressed in a local municipality and their causes, we then derive the requirements for improving the situation and present a requirements diagram. Section 6.2 presents all stakeholders involved in the initiative promoted by IHK. In Section 6.3 an urban initiative is presented where these requirements are partially accessed with the usage of SCOUP. Section 6.4 shows how the software components available for SCOUP help access the sustainability aspects of urban planning. Section 6.5 finally revises the SCOUP framework for additional components.

6.1 Problem Investigation

Benevolo *et al.* analyze different types of smart mobility initiatives that a city can consider when optimizing it's mobility offerings [BDD16]. The authors further note the differentiation between digital offerings and sustainable ones, and explain that while initiatives can tackle both, that is not necessarily always the case . For each of the proposed initiatives, an analysis is made as to what extent they address each of the six benefits in smart mobility described above, and what is the level of digital systems that they require (classified simply as low, medium, high) [BDD16].

The implementation of these initiatives, however, is often more simple in theory than in practice. To evaluate the complexity of implementing them in practice, the IHK pilot project chose the municipality of Kirchheim as a starting point to implement their smart mobility initiative. Kirchheim is a a municipality east of Munich, whose population increased by a factor of 6 between 1970 and 2017 [BLS18]. This population growth has been accompanied by an increase in employees working in the municipality. In the last 10 years, this growth has been as big as 15.2% [Cul19]. Even though Kirchheim only has a population of around 12000 citizens, the traffic problems in the town becomes more understandable when looking at a map.

The municipality is composed of two smaller communities: Kirchheim and Heimstetten, which were previously only connected by an arterial road. As becomes clear in Figure 6.1, while there is a highway exit from the main ring to Kirchheim, the same does not apply to Heimstetten. On the other hand, Heimstetten has a connection to

[3] `https://www.sueddeutsche.de/muenchen/landkreismuenchen/smart-city-die-vermessung-kirchheims-1.4508747`

the public suburban trains, but passengers arriving there have a problem arriving in Kirchheim, as it only contains bus stations.

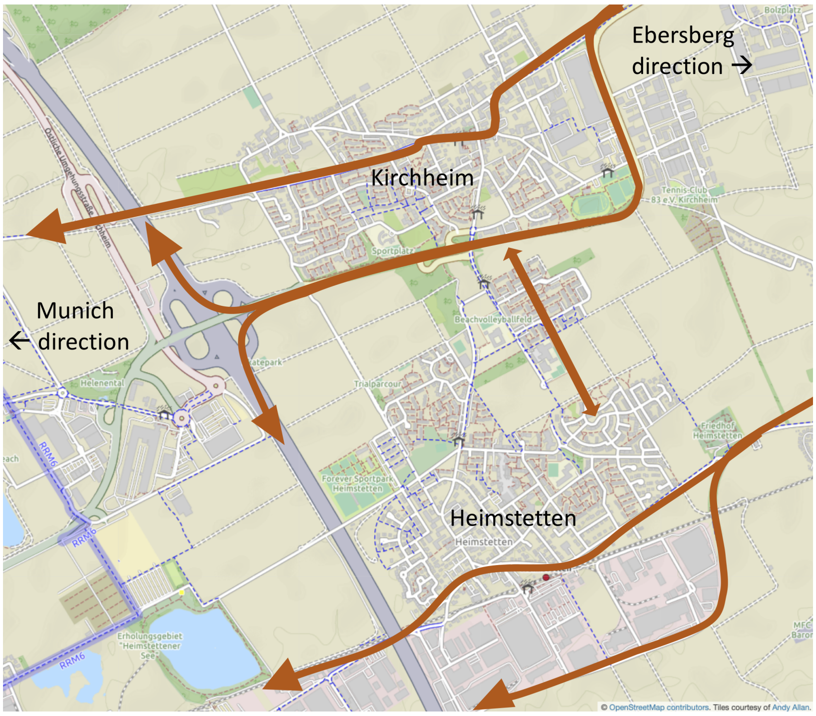


Figure 6.1: Map displaying most important traffic flows between the city of Munich and the communities of Kirchheim, Heimstetten, and Erding. Source: Translated from internal Kirchheim analysis.

Including Kirchheim and Heimstetten, the municipality has three different industrial centers, which employ a few thousand citizens. The employees arriving at these work stations during the morning and evening commute struggle, as not only is the single highway exit a bottleneck, arriving at the industry centers with public transport is also hard, due to the few bus lines available.

To tackle the problems caused by the lack of connecting roads and required reachability of the industrial centers, Kirchheim's municipality decided to develop a Smart Mobility project, which tackles four main objectives:

1. Visibility: How does the traffic flow?
2. Transparency: Why does the traffic flow that way?
3. Predictability: How will the traffic flow?
4. Automatization: How can automated reactions be implemented?

While those goals are a good starting point, it was necessary to refine them into concrete functional requirements before designing the system. It is also relevant to note the data integration necessary to achieve them, especially as far as transparency and predictability are concerned. Even before the Kirchheim project, this effort was already observed in a collaboration with Schlichtmann [Sch18] where (a) GIS information for the Munich metropolitan region, (b) a survey of the region's citizens, (c) population density, and (d) company locations were combined to derive a model of the Munich commute. Although in the collaboration, a model was trained through accurate data of the *Wohnen Arbeiten Mobilität*[4] study [Thi+16] the end prediction of the model had big discrepancies between the calculated and actual inter-regional commute volume, due to the amount of assumptions that took place during the data integration [Sch18].

Based on the learnings from the collaboration with Schlichtmann [Sch18], a decision was made to install sensors in the community, and thus have a calibration basis for the integration. The objectives above and the Smart Mobility objectives by Benevolo *et al.*, [BDD16] were used to guide the development of the Smart Mobility initiative in Kirchheim. The system developed for Kirchheim expected the following functional requirements (K-FR):

K-FR1 **Improve mobility:** The main goal of the system is to improve mobility in the municipality of Kirchheim, according to the categories described by [BDD16]. The main points addressed should be the reduction of pollution and traffic congestions, and safety standards should be maintained. This should be achieved by addressing the traffic relations, and recommending non congested roads.

K-FR2 **Visualize Available Data:** The system supports real time visualization of the traffic points in Kirchheim, allowing for long term data analysis and traffic scenario evaluation.

K-FR3 **Understand Traffic Demand Relation:** The relation between traffic demand and road capacity is displayed, as well as the consequences of these as far as sustainability aspects of the municipality such as air pollution are concerned.

K-FR4 **Simulate Traffic Conditions:** Traffic conditions are simulated at a driver-level, representing the different flow patterns throughout times of days, and days of week. This simulation should accurately represent the demand as far as the different regions of the municipality are concerned, and data collected from the traffic sensors are used as a calibration basis for better simulations results.

K-FR5 **Modify Road Network:** Modifications on the road network are assisted by visualizing the impact that such modifications could have on traffic congestions.

[4] translation: "living, working, mobility"

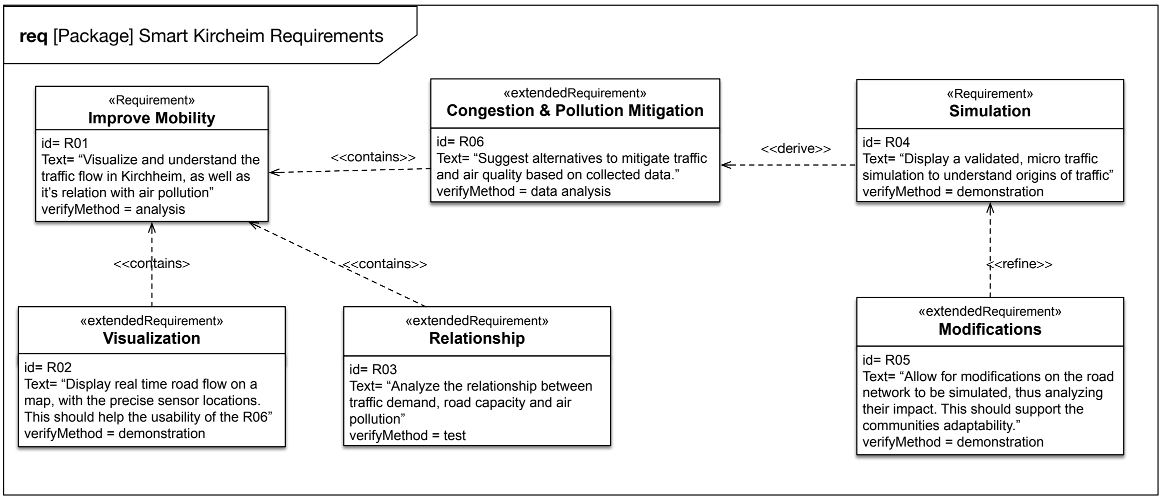


Figure 6.2: Requirements for the Smart Mobility project in Kirchheim – Systems Modeling Language (SysML) Requirements Diagram.

An interface between possible road network modifications and the presented traffic simulations should be presented to better evaluate the effects of these changes.

K-Fr6 **Mitigate Congestion & Pollution:** Alternatives are suggested to mitigate traffic congestion and air pollution, based on both historic data and real time values. These alternatives should be calculated based on literature research of similar systems as well as an analysis of the applicability of these data models to Kirchheim’s situation. Traffic reduction alternatives are evaluated by comparing both traffic simulations and gathered data in order to continuously improve the accuracy of the system.

The relationship between these different requirements and how the traffic improvement initiatives and simulations are the top-level elements can be seen in Figure 6.2. From a technical point of view, however, quality attributes should also be taken into consideration. It should be ensured that the developed system supports the continuous addition of road sensors, as these become available. Different types of flow sensors and air pollution sensors will eventually be connected to the infrastructure, which should thus also allow for the integration of public open data Application Programming Interfaces (Api). Although each of these sensors is to be installed in a different location, access to their data should be enabled by a single interface, to reduce the integration overhead of the connecting applications.

A further quality attribute is that each sensor has the necessary technical infrastructure and bandwidth to broadcast it’s gathered information in realtime. While connecting a sensor to a power socket seems easy in a controlled room, this becomes a challenge when installing sensors in the middle of roundabouts and street intersections, which occasionally have neither energy nor data connections available to the required locations. This availability quality attribute [BCK03] should also be addressed through the usage of Scoup.

6.2 Kirchheim Stakeholders

As is common with urban initiatives, one relevant challenge in implementing smart mobility initiatives is the wide range of stakeholders involved in the project. Benevolo *et al.* describe how the stakeholders of such a project can act like agents for change, receive the benefits of the initiative, or both [BDD16]. In order to obtain a better understanding of each of the partners involved in the Ihk Smart Mobility initiative, a short description of each of them is provided.

- **Kirchheim Government[5]:** The municipalities' town hall serves as the main point of contact who not only presented some of the requirements described above, but also acts as a common point of contact between all the other stakeholders.
- **IABG[6]:** The company Industrieanlagen-Betriebsgesellschaft mbH (Iabg) offered the Commutify urban initiative to the project (see Section 6.3 below). The company used employer and residential data to calculate optimal commuter bus routes. Iabg was further responsible for conducting a citizen survey about the mobility preferences in the municipality [Cul19].
- **HawaDawa[7]:** The air quality startup provided the air quality sensors that were installed in the municipality, and were thus also a partner for the part of the study that uses this data set.
- **Cesonia[8]:** The data management startup was responsible for running the data virtualization service, and were thus a partner for the part of the project that controlled this [Sim19]. They further hosted the database that persisted all collected data.
- **Advanced Urban Analytics:** The project from the University of Applied Sciences Landshut (*Hochschule für Angewandte Wissenschaften Landshut*) developed an additional traffic flow sensor that aimed to solve some of the shortcomings of the ones previously installed in Kirchheim [Wal19]. While their sensor

[5]Municipality website: `https://www.kirchheim-heimstetten.de/`
[6]Company website: `https://www.iabg.de/en/`
[7]Company website: `https://hawadawa.com/`
[8]Company website: `https://www.cesonia.io/de/startseite`

was later integrated into the provided infrastructure, their sensor development was mostly independent from the rest of the project.

- **Omega Lambda Tec[9]:** The data science startup was responsible for most of the practical coordinations in the initiative, such as acquiring sponsors and permits for each step of the project. They were thus also an important source of (technical) requirements for the developed system.
- **SpaceNet[10]:** The company was responsible for providing the servers running the entire infrastructure. The software developed in partnership with Cesonia was running in their server.
- **Chair for Applied Software Engineering:** The research group from the Technical University of Munich was responsible for developing several of the components for the system (see Section 6.4), which aggregated into the results presented in this chapter.

6.3 Commutify Initiative

After realizing an origin and destination analysis of the citizens of Kirchheim, one of the first urban initiatives examined was the introduction of Commutify[11] commuter buses. Commutify is a project inside of the *Industrieanlagen-Betriebsgesellschaft mbH* (IABG) where an ant colony optimization was used to calculate commuter bus routes connecting the residences of employees of a company or region to their work locations [Sar19]. The idea behind is that by allowing employees to be picked up close to their residence and clustering them together inside one commuter bus, for a better price than car ownership or public transportation, it would be possible to reduce the number of cars in the road, thus addressing K-FR1 from Section 6.1. Furthermore, providing special commuter buses guarantees that the employees have a seat for the commute ride, and must thus not stand during the ride – as is common in public transportation providers – thus providing a safer mobility option than what is usually available.

In order to evaluate the effectiveness of such an urban initiative, we first model it according to SCOUP, as can be seen in Figure 6.3. In the context of the Commutify urban initiative, the most relevant indicator is the same one as used in Section 5.2: Percentage of commuters using a travel mode to work other than a personal vehicle. As the number of users of the service can be tracked by the company providing

[9]Company website: `https://omegalambdatec.com/`
[10]Company website: `https://www.space.net/`
[11]Commutify project: `https://www.commutify.de/`

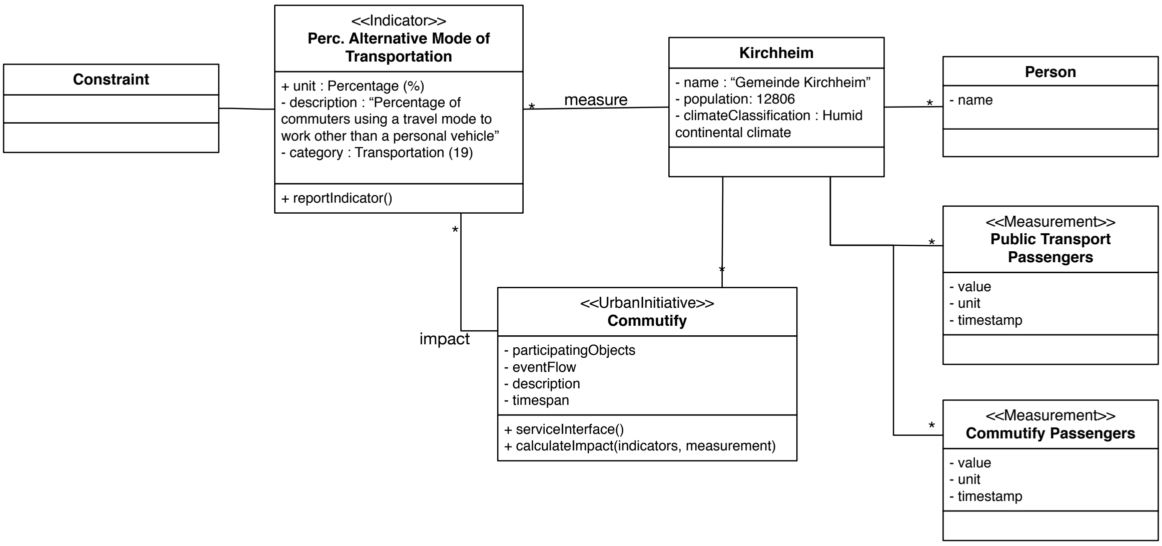


Figure 6.3: Application of the Scoup meta-model to the Commutify urban initiative (Uml Class Diagram).

it, this number should be integrated into other metrics available from the public transportation provider.This displays a threat to validity for the indicator, as it is hard to make a generalization about all citizens in the municipality through only this measurement.

Observation 7

The fact that a measurement is available for one specific urban initiative, does not implicate that the measurement is complete for the entire analyzed region, nor that it is representative thereof. Not all available measurements have the same value for the region.

Scoup allows us to map the Commutify urban initiative to the presented taxonomy. As with any urban initiative, the first step of the realization process is to determine its feasibility. The entire consideration and execution process for Commutify in Kirchheim is presented in Table 6.1, where an adapted version of the use case template [BD10] is used to describe the urban initiative. It is noteworthy how the attributes defined in Scoup for urban initiatives directly relate to the execution of a use case according to software engineering standards. The attributes and relations of an urban initiative are transformed into a table format, to analyze the connection between the initiative and the environment it is applied to. The main modification between Table 6.1 and the use case template defined in [BD10] is the addition of the urban initiative name, as well as the indicators that this initiative expects to impact.

While the flow of events described in Table 6.1 was implemented in the municipality of Kirchheim, unfortunately the decision at step (6) resulted that the majority of

Urban Initiative Name	Introduce Commutify
Indicator	Percentage of commuters using a travel mode to work other than a personal vehicle [ISO 37120] Annual number of public transport trips (per capita) [ISO 37120] Average commute time [ISO 37120]
Participating actors	Commutify, Company, Kirchheim local authorities, Citizens, Citizen's Car, Bus Driver, Bus
Entry conditions	Citizens commute to work in Kirchheim mostly by car Commutify has available buses
Flow of events	1) Citizens commute to work in Kirchheim mostly by car 2) Traffic increases 3) Citizens get annoyed at traffic and report this to the local authorities 4) Kirchheim contacts Commutify to help with traffic flow 5) Commutify calculates a bus route connecting the interested citizens and companies 6) Commutify asks citizens if they would be willing to use the bus route instead of their car 7a) If affirmative, Commmutify contacts bus & bus driver provider for price and availability. Commutify forwards costs to Kirchheim, Company, and Citizens. 7b) If negative, Buses are not driven. 8) If bus operation approved by all, Bus starts driving the route calculated in (5) during rush hour period. 9) Indicator increases.
Exit conditions	Bus regularly drives the calculated route, carrying the affected citizens to the Company. Indicator has changed (increased)

Table 6.1: Commutify urban initiative description, table adapted from the scenario template in Brügge *et al.* [BD10].

people who would be offered the Commutify bus route did not wish to relinquish their cars, as they still required the flexibility made available by it. This demonstrates a problem both with the Commutify concept, as well as with other urban initiatives: what is 'best' at the community level is often not the same thing as is 'best' at an individual level. Since at step (7b) the decision was made not to drive the buses through the community, the Kirchheim project then evolved into analyzing the situation from the community's perspective, and the resulting project and results are now presented.

6.4 Reference Application: SmartHeim2

As described in requirement K-Fr2 above an important specification was the real time visualization of the traffic flow on a map and the integration of these into simulation capabilities. The following section presents a system we developed for Kirchheim in order to address these requirements. The name of the system, SmartHeim2, is a reference to the fact that there are two *Heim*s, Kirchheim and Heimstetten, which should be made 'smart' and better connected between them, by the integration of digital systems.

The community of Kirchheim first completed a study of their mobility situation in cooperation with Iabg (see description above). This included a questionnaire to the local population, to investigate what were their biggest annoyances in regard to transport. A complete report of the questionnaire and the findings of this study can be found in the work by Cullen [Cul19]. One of the major findings of the reported study was the high annoyance that most of the commuters in Kirchheim have in relation to the reliability of the public transport availability in the region. Furthermore, it was shown how a high percentage of commuters thus prefer to use their cars on the daily commute. This leads to a higher traffic volume during the commute hours.

The SmartHeim2 system was decomposed into smaller components, which are now described. It is relevant to note that while we refer to the entire system developed for Kirchheim as SmartHeim2, it was developed modularly through several different and only partially connected components, some of which have their own names below. Initially we accessed the data availability of the community, and set up a simulation to gain further insights. To calibrate the simulation against reality, we installed traffic and air quality sensors throughout the community. The total data was then leveraged against each other to calculate different indicators for the Kirchheim community. In the following sections we explain how the traffic flow and air quality data were used in combination with historical data sets to calibrate the provided simulations and enable indicator calculations. Figure 6.4 illustrates the major components of the system, which are presented in detail in the following sections.

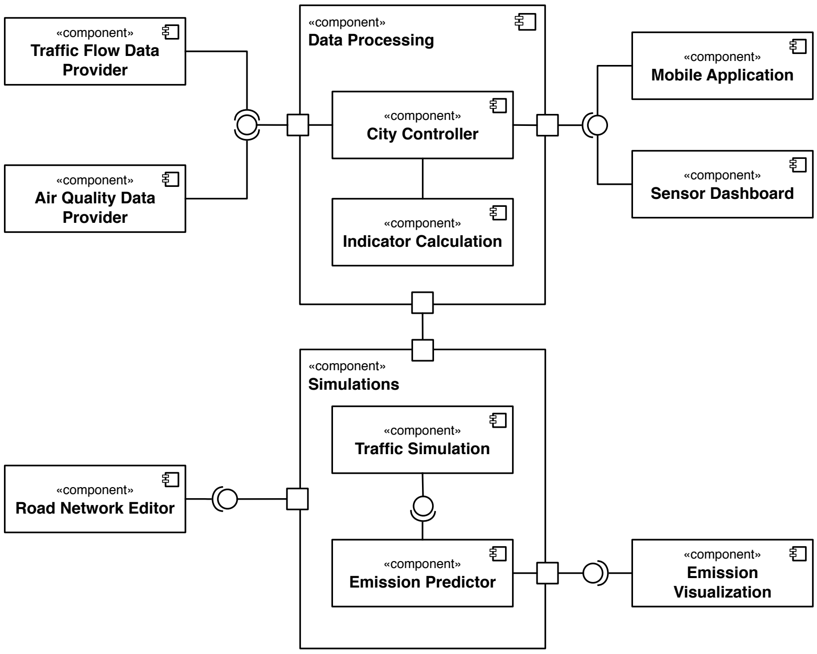


Figure 6.4: Subsystem Decomposition of the SmartHeim2 System (UML Component Diagram).

6.4.1 Data Availability

As most communities and cities throughout the world, Kirchheim has long been collecting different information about the region. Although a lot of data exists, most of it is only present in Portable Document Format (PDF) or Microsoft Excel format. Most of the available data is unstructured, being only available for some time periods, and constantly having the methodology behind these change. At the start of the project a list of available data sources was compiled, which can be found in Appendix B.

To make the available information useful, it was necessary to integrate them into a common application. Historically, this has been done through the usage of data mashups, which Daniel *et al.* describe as "a composite application developed starting from reusable data, application logic, and/or user interfaces" [DM14]. Mashups consists of reusable data components made available for the application, as well as of the mashup logic that controls how these components are combined and transformed [DM14]. The logic layer may require some considerable manual labor, as integrating each new component may potentially require an additional logic interface to extract the necessary information.

Observation 8

This manual labor will inevitably have different quality levels for each city or person doing it, and thus the value of the aggregated data may vary significantly.

In order to address the lack of quality when integrating data sources, we also considered a data virtualization approach for SmartHeim2. As presented in Section 2.2.2, virtualization of the components provides an approach on how to abstract different data sources. As the approach relies on automatically parsing the columns and attributes of data sources, it is only useful for machine readable formats such as comma separated values or application programming interfaces. Together with Cesonia (see Section 6.2), we virtualized the Kirchheim measurements available in machine-readable format and thus brought to a common sensor measurement format [Sim19] as defined in Scoup. When more than one measurement was available for the same indicator (such as with the traffic flow sensors, see Section 6.4.3), the measurement composite defined in Scoup was used to integrate these.

Since an important goal of the SmartHeim2 system (K-Fr2) was to also support the previously available Pdf human readable data, both data aggregation approaches were partially used in the end. While this worked well in the SmartHeim2 system, the individual considerations as to which approach to use for which measurement makes it hard to generalize for all cities.

6.4.2 Simulation

The rapid increase in Kirchheim's population led to the roads connecting it to Munich to quickly become congestionned. In order to analyze the effects of possible urban initiatives, the community desired to have a traffic simulation, which accurately represents the road conditions, as was presented in K-Fr3 of Section 6.1.

As we have seen in Section 3.4.1, different types of traffic simulations exist, so the first step for the SmartHeim2 project was to decide which level of abstraction and simulation software was to be used. Since it was not required for a huge area to be simulated, but rather only the communities of Kirchheim and Heimstetten, a decision was made to use a microscopic simulation. When neither a big physical area nor a high number of traffic participants/vehicles are required, these simulations can usually be run on personal computers with a fast enough runtime.

Furthermore, the SmartHeim2 project also had limited project funding and thus open source projects were preferred. When taking both of these constraints into consideration, a decision was made to use the Simulation of Urban MObility (Sumo), which was developed by the *Deutsche Zentrum für Luft- und Raumfahrt* (translation: German Aerospace Center). Sumo offers several useful features, such as multi-lane streets with lane changing, junction-based right-of-way rules, and lane-to-lane connections [KHRW02]. The fact that the software is open source further allowed for the extension of functionalities described in Section 6.1. A screenshot of the Sumo simulation in Kirchheim can be seen in Figure 6.5

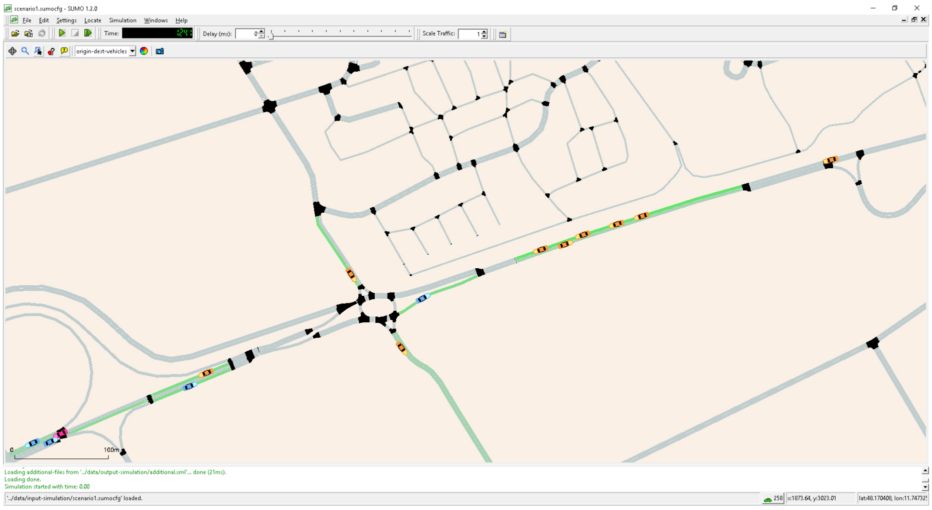


Figure 6.5: Sumo simulation of Kirchheim

As stated in K-Fr4 in Section 6.1, the urban planners in Kirchheim aimed to evaluate the effect of road network modifications on the urban traffic. K-Fr5 furthermore describes the need to make modifications to the road network. For example, before constructing a new road or lane connecting Kirchheim and Heimstetten, it was desired to predict the effect that this would have on the traffic incoming from the Munich highway exit. In order to allow for urban planners to more easily edit and visualize the road network, we developed a Sumo plugin in collaboration with Kirchheim and Jakob Smretschnig [Smr19], where a communication was implemented between Sumo and Java Open Street Map editor (Josm)[12], in which one could modify the road network in the tool, and the updated network would be used as a basis for the simulation. The developed plugin was named TraLAMA, and is also made available open source [13].

Two main features of Scoup allowed for an easy integration between TraLAMA and the rest of the SmartHeim2 system. First, the measurement composite allow for an integration of the Gis data necessary for the simulation with the Sensor data available for the traffic flows. The composite pattern allowed for each sensor to be mapped to their respective Gis region, which was then aggregated to include both data sources. These were then processed and made available for the simulation, which Scoup interprets as a VirtualInitiative, as the communication between TraLAMA and Sumo followed an Api specification.

While Sumo does provide some basic network editing capabilities, we provided an interface to an external software since Josm more easily integrates to other geographic tools, thus allowing for the visualization of building types and geolocation traces,

[12] Josm: `https://josm.openstreetmap.de/`
[13] TraLAMA source code: `https://github.com/jetoff41/TraLAMA`

which are tools that can aid urban planners. SmartHeim2 further provides an interface which allows for the comparison of different scenarios within SUMO, thus allowing for traffic analysis throughout the day and in different road conditions. A screenshot of this comparison can be seen in Figure 6.6.

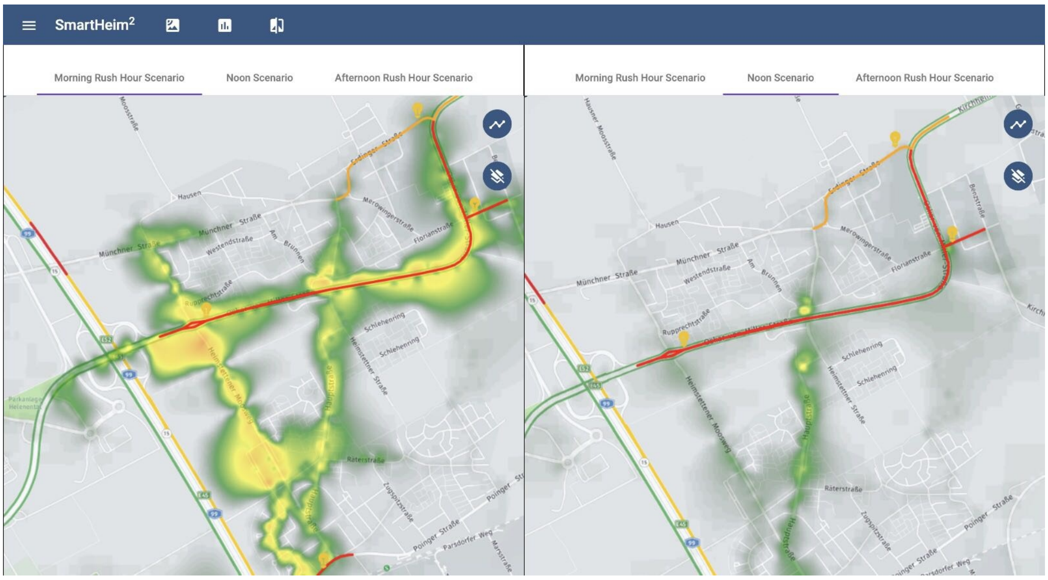


Figure 6.6: Kirchheim traffic flow comparison based on calibrated simulation data.

Both the SUMO traffic simulation, the developed TraLAMA plugin and the SmartHeim2 system are meant as urban development software components to aid planners in their job. While their goal is to facilitate the implementation of urban initiatives, they are not an urban initiative themselves, as they do not directly change any indicator and only help in the planning of such initiatives. Since they are not urban initiatives, we describe the usage of the tool with a scenario template [BD10] rather than with the urban initiative one we have used otherwise. Table 6.2 outlines how the ideal flow of events should look like, once the system is deployed in the community of Kirchheim.

As with most simulations, validation of the traffic flow present in the SUMO model was relevant so that the results predicted by the software were useful for the urban planners. In order to have a basis to calibrate the traffic simulation against, traffic flow sensors were installed in key locations throughout Kirchheim, and the data collected by these then processed through the virtualization approach presented in Section 6.4.1. These are now described.

Use case name	Measure traffic in Kirchheim Eye Roundabout and analyze possible change
Participating actors	John: Traffic planner
Flow of events	1) John wants to know the current traffic situation in the roundabout that provides access to Kirchheim. 2) John opens the SmartHeim2 application, and selects the Kirchheim Eye roundabout 3) The application connects to the installed traffic flow sensor, retrieves the speed values for the last hours and displays the average speed of the cars passing by the roundabout. 4) John wants to know how this evolves over time and opens the historic view 5) John selects the desired date 6) The system displays the average speed at the roundabout for the selected date. 7) John realizes how the traffic build up in the roundabout is consistent over time and considers changing the traffic flow in the location. 8) John edits the road network, to directly connect the exit of the highway to one of the main industry centers in Kirchheim. 9) John starts a traffic simulation to analyze the effects of this possible change.
Entry conditions	Traffic flow sensor installed in the roundabout Traffic flow sensor has power and internet connection
Exit conditions	John knows real time traffic situation in the roundabout
Special requirements	For the information gained by John to be useful, this scenario should be executed during rush hour, such as 8am

Table 6.2: Flow of events for the roundabout traffic analysis in Kirchheim.

6.4.3 Traffic Flow Sensors

As mentioned earlier, one of the goals of the SmartHeim2 project was to increase the visibility of the traffic situation in the community. With the aim of better visualizing the current traffic state and calibrating the simulation, we installed traffic flow sensors in the main access points of the community.

When deciding which type of sensor was ideal for the required measurements, several options came into play. One traffic flow sensor provider was already present in the community, the Bremicker-Veris system [14]. The Veris system was developed with the aim of helping to control traffic flow and is the state-of-the-art solution for speed monitoring from Bremicker. It can not only display different traffic speeds, but also has an integrated flow sensor and camera that helps to measure if the cars passing by are keeping true to that speed. As there was already an installed Veris sensor in the community, the company contact and required elicitation process had already been processed with Bremicker, thus making the Veris sensors a candidate for the further required sensor locations.

While the Veris sensors are good at recording midrange- and high- speeds of passing by cars, the same does not apply to low speeds, which are mostly present during traffic. The Veris sensor works by measuring the road-gaps between one car and the next one, and then comparing their lengths to estimate vehicle quantity and speed. While this approach works well when cars are going fast, and therefore have a high breaking gap in front of them, this gap becomes smaller during traffic situations, thus making it hard for the sensor to accurately count the number of cars and their speed. The data recorded in the community of Kirchheim shows how during peak hours, the Veris sensors would occasionally record a smaller number of cars, as it would see one car as a long vehicle due to not being able to detect the gap in between them. As the number of vehicles during rush hour was exactly one of the key points relevant in this project, alternative sensor systems were considered.

As detecting traffic bottlenecks was a major goal of the project (K-Fr3 & K-Fr6), the above limitation became a bigger challenge, and further traffic flow alternatives were investigated. In order to better measure vehicle flow, also during congestion situations, the Advanced Urban Analytics (see Section 6.2) project developed a radar-based sensor, which more accurately measures the speed of slow vehicles. The results of this group can be found in the work by Wallner [Wal19].

As the sensor developed by the Advanced Urban Analytics project was just a prototype, only one unit of it was available. This led to the situation where both sensor systems were in use, the Veris and the newly developed one. Since both of these were

[14]Further Information: `https://www.bremicker-vt.de/de/produkte/geschwindigkeitswarnanlagen/verkehrsdatenerfassungs-und-informationssystem-veris-01`

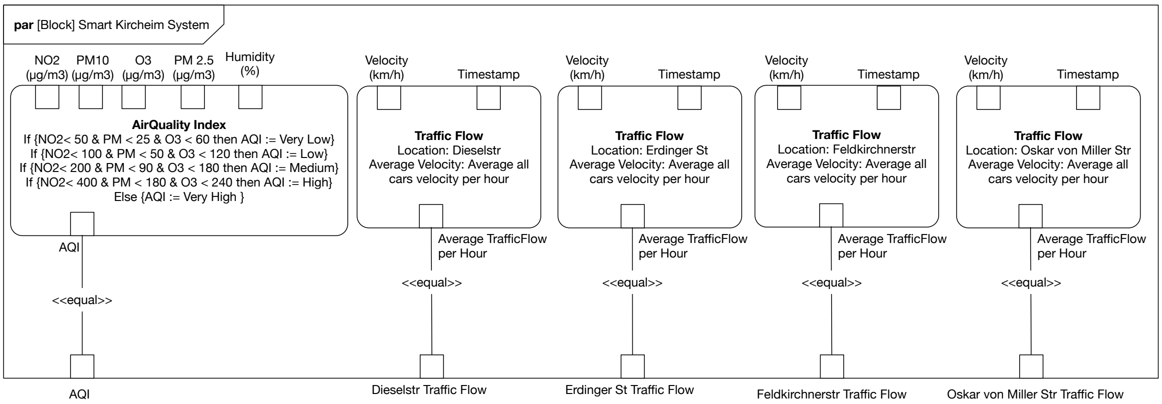


Figure 6.7: Sensors installed in Kirchheim (SysML Parametric Diagram).

measuring similar types of data, it signified the need for these to be aggregated into a similar format. Since both of the sensors were measuring flow, it was agreed that both would be integrated to a $Speed\,per\,Vehicle$ unit.

While the unit is not one of the indicators measured by the standards used in Scoup, when used in combination with imagery (which was for example available for the Veris sensors), they do allow for the calculation of two relevant indicators: Iso 37120 19.8.1: *Number of personal automobiles per capita* and Iso 37122 19.3 *Percentage of vehicles registered in the city that are low-emission vehicles.* If information about the emission types of the passing vehicles were to be made available later, this could be integrated to the data collected by the sensors to report on that indicator as well. This means that SmartHeim2 should support the management of data not directly related to the considered indicator standards.

In an attempt to track the impacts of traffic in air quality, pollution sensors were then installed in key locations in Kirchheim. Figure 6.7 shows a SysML parametric diagram of all the sensors and information gathered by the different sensors installed in Kirchheim. The air quality sensors are now presented.

6.4.4 Air Quality Sensor

To correlate the traffic flow data with the air pollution in the community, particulate matter sensors from HawaDawa[15] were installed to measure pollution at key locations throughout the city, a few of which also had the traffic flow sensors described in Section 6.4.3. These pollution sensors measure Air Quality Index based on three main indicators: O_3, NO_2, and PM_{10}, which respectively stand for ozone, nitrogen dioxide, and particulate matter of size $10\mu m$ or less in diameter.

Differently than traffic flow, air quality components represent key indicators present in ISO 37120, where a total of 6 indicators tracked by SCOUP correlate to the different chemical components affecting the environment and climate change (ISO 37120 8.1-8.7) [ISO 37120]. Good air quality information can be crucial to understand the health impacts of traffic, as studies in the Munich region have demonstrated a correlation between transportation emitted NO_2 and respiratory problems in school children [Nic+03].

Other than measurement collection and indicator reporting, the installation of air quality sensors supported three main goals: (1) the analysis of correlation between air quality and traffic flow, (2) the modeling of what changes to air quality could be expected when introducing changes to the road network or traffic flow, and (3) the calibration of the presented simulations with real world data. The correlation between air quality and traffic flow had already been partially investigated by HawaDawa in a previous study in a different region of Munich, which also investigated the effect of weather between these parameters [Hel19]. This previous study showed that wind speed was one of the variables that most correlated with air quality and thus it was desired to improve the modeling of the pollution components and their dispersion.

To efficiently model the changes in air quality, we wanted to integrate the air quality model into the available traffic simulation, and calibrate the simulation with the existing sensor data. The result of these goals was the KoAiry component of the SmartHeim2 system, which we did in collaboration with David Schemm and HawaDawa. The complete KoAiry description can be found under the work by Schemm [Sch20] and is also made available through an open source license[16]. For the calibration of the simulation the area of Kirchheim was segmented into three different regions, representative of the main traffic origin and destination hubs in the community, where traffic flow sensors were also installed. The traffic originated, passing, and arriving in these regions was then extracted from the traffic flow sensor data and a travel demand was generated based on these to serve as the input for the SUMO simulation.

[15] HawaDawa: `https://www.hawadawa.com/our-solution/`
[16] KoAiry source code: `https://github.com/scheda74/koairy`

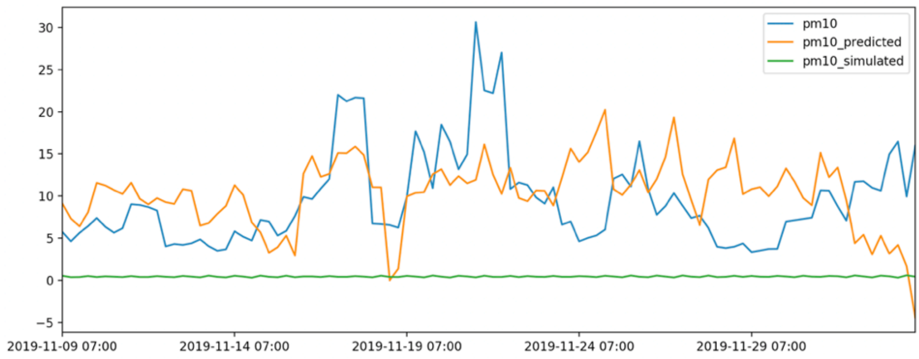


Figure 6.8: PM10 values for a sensor location in Kirchheim according to the measured sensor values, the original simulated ones, and the predicted ones with the multi-layer perception regressor. As presented by Schemm [Sch20].

SUMO has a mode that allows the user to simulate air pollution based on the vehicle emission classes of the simulation agent's, using the *Handbook Emission Factors for Road Transport* (HBEFA) [Lop+18]. While we input the traffic simulation with the vehicle class distribution present in Kirchheim, the results obtained show that the simulated emissions were often not accurate [Sch20]. When relating this to work done by Held with HawaDawa [Hel19] it makes intuitive sense as the traffic simulation does not have any weather models in it. To have more accurate predictions of what air quality should be expected after introducing road network changes, the available sensor data was used to train a linear regression and multi-layer perception regressor, which more accurately predicted the different components of the emissions. Figure 6.8 demonstrates how the prediction model achieved compares against the original simulation data, as well as the ones measured by the sensors. These prediction models were integrated into a KoAiry web-dashboard [Sch20]. The dashboard further displays the measurements of the air quality sensors and allows the SUMO simulation to be started. The dashboard is seen in Figure 6.9.

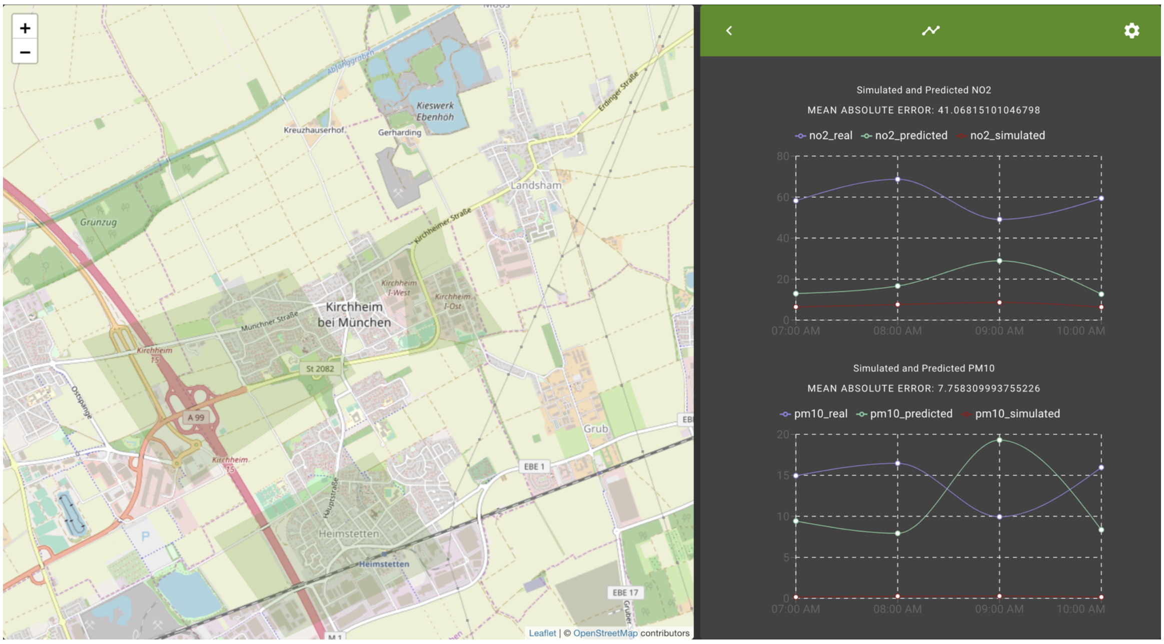


Figure 6.9: KoAiry Dashboard displaying the segmented areas of the Kirchheim region, and the sensor data. As presented by Schemm [Sch20].

6.4.5 Indicator Analysis

Once all sensors were installed in Kirchheim, the indicators mentioned earlier could be calculated, and the last thing missing in the Scoup integration were the considerations of which urban initiatives were possible by these. The instantiation of the Scoup meta-model can be seen in Figure 6.10.

As described in Section 6.4.1, beyond the traffic simulation and sensors described in this chapter, Kirchheim also made available extensive data and measurements which were historically kept by the community. This allowed us to calculate several of the relevant Iso indicators, a complete list of which can be found in Appendix B. As far as the urban initiatives implemented in Kirchheim are concerned, several were considered. The Commutify initiative has already been described, as has the virtual initiatives where the aggregated data from the sensors and measurements were made available through an Api a dashboard and used in the simulation. We further decided to install a smart traffic light in the roundabout exit from the highway and while the purchase has been approved, the installation process is not yet completed at the time of writing, thus making it impossible to evaluate the results.

The idea behind the smart traffic light is that it contains an actuator that should be connected to the Veris traffic flow sensors, thus ensuring a smoother flow in the street connecting the highway to the city and the industry areas. For the Kircheim 2030 initiative, several of these smart traffic lights are being considered. The evaluation of

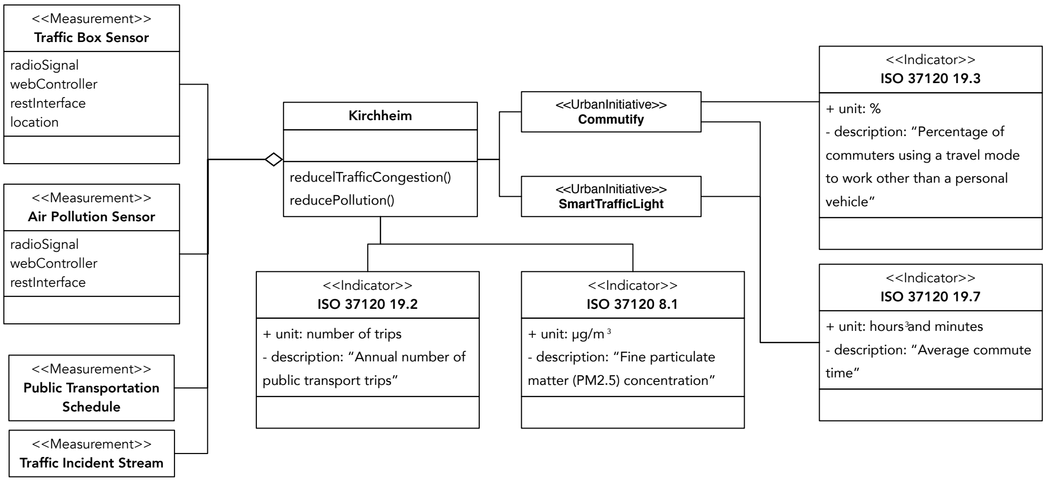


Figure 6.10: SmartHeim2 System Analysis Object Model according to M1 Instantiation of the Scoup meta-model (Uml Class Diagram).

these is regarded future work for the SmartHeim2 system.

The simulations and models presented in this section are used in Kirchheim, but they are still far from an over-arching digital twin that allows for the calculation of all indicators described in Iso 37120 and Iso 37122. While a project that integrates Bim modules is currently planned in Kirchheim, it is important to note that even with the best digital twin models, aspects predicted must alway be verified by physically sourced information. As we have seen in Figure 6.8, descripencies between even complex prediction models and actual sensor values are not negligible.

6.5 SCOUP Extension

Observation 8 remarked on the manual work necessary for the data integration phase. This means that not all indicator calculations will present the same accuracy and reliability for every region. When considering the region comparability goal of Scoup, this presents a high threat to validity, as even if the same indicators are calculated, different data sources, methodologies, and processes may lead to a different reliability level. To address these differences, we introduce quality attributes to the Scoup stereotypes, an overview of which can be found in Table 6.3.

Stereotype	Quality Attribute
Measurement	Interoperability
Indicator	Certainty
Urban Initiative	Testability

Table 6.3: Newly introduced quality attribute for each of Scoup's stereoytpes.

For the measurement stereotype, interoperability is the main quality attribute. According to Bass *et al.* this means the "degree to which two or more systems can usefully exchange meaningful information via interfaces in a particular context"[BCK03]. In the Ssc context, this relates to the degree in which different measurements can exchange meaningful data about their respective values through the Scoup interface provided for different Sscs. Bass *et al.* define interfaces (in the context of interoperability) as the set of assumptions that can safely be made about an entity, which is exactly what Scoup aims to provide for the Sscs. For the sake of clarity, these assumptions for the measurement stereotype are partly derived from the data quality assessment work by Pipino *et al.*, [PLW02] and enunciated below.

1. Metadata relevant to a given measurement is always available
2. The timestamp for the collected measurement includes year and month
3. The value attribute of a measurement is relevant for the calculation of one of the Iso indicators.
4. The calculation methodology for a given measurement is consistent over time. If a change is required, it shall represent a new, independent, measurement.
5. The unit of the measurement is consistent over time. If a change in this is required, it shall represent a new, independent, measurement.

For the indicator stereotype, the certainty of the provided measurements and calculated value is an important quality attribute. In our Kirchheim analysis we often encountered the case in which only partial information for an indicator was available, and not complete required data. This implicates that while the calculated indicators can serve as a base for comparison, caution is still required when making a definitive statement about these. In order to address this uncertainty and keep track of the quality of an indicator's calculation, a certainty attribute is added to the indicator stereotype, which should be reported on as a 5-step Likert scale, answering the question "The acquired measurements are scientifically certain" ranging from Strongly disagree (1) to Strongly Agree (5).

The concept of urban initiatives is centrally related to having a city take action to impact an indicators. Thus, for an urban initiative to have quality it should have a high change in the specific indicator based on the available context and measurements. The quality attribute that relates to this is testability, as defined by Bass *et al.* [BCK03], who describe testability as how well a system (in our case, urban initiative) "gives up" his faults easily. If we consider that in our Ssc context a fault is the lack of change in an indicator, an urban initiative's testability then relates to how well the difference in an indicator's value is assignable to an urban initiative. For example, if several urban Initiatives tackle the same indicator in a similar time range, it may be hard to assign the measured indicator difference to any individual urban initiative.

These attributes should be collected and tracked with the same rigorous as the indicators themselves. Furthermore, it is important to persist not only the indicators and their respective measurements, but also any available measurement in the community, as measurements previously available (even when they are not currently used for an indicator) can become indicator relevant in the future, when combined to newly available measurements. An overview of the Meta-model after these extensions can be found in Figure 4.4.

Chapter 7
Validation

7.1 Comparison of Urban Initiatives across Regions 98
7.2 Representation of Physical Entities 100
7.3 Data Source Integration 102
7.4 Multiple Stakeholder Support 104
7.5 Discussion 105

To investigate how Scoup can be used in Ssc applications, we evaluate to what extent the core requirements of Scoup are applicable to further projects. For each of the core requirements defined in Section 4.1, we conduct a small experiment where a hypothesis is first made about the applicability of the Scoup requirement, the study is then described and Scoup is applied to it, before finally evaluating which concrete results could be derived from it.

We initially address the comparison requirement in Section 7.1. In Section 7.2 we then use a search and rescue project to evaluate the applicability of Scoup's meta-model to further projects and initiatives. Section 7.3 discusses the outcomes of the addition of data sources proposed by Scoup. Section 7.4 debates how sucesful the consideration of multiple stakeholders was in the perspective of urban initiatives. Last but not least, Section 7.5 discusses how each of the projects presented in Chapters 5 and 6 address each of the requirements proposed by Scoup, and to what extent each of these requirements were actually achieved.

7.1 Comparison of Urban Initiatives across Regions

As presented in Section 4.1, one of the main requirements for Scoup, is to facilitate the comparison of different regions and urban initiatives. Referring back to our meta-model and definition, an urban initiative should be implemented by a city, contain a set of participating actors and objects, associated events, a description and a timespan. Importantly, an urban initiative impacts one specific Iso indicator. We now consider the following the observation:

Observation 9

Scoup allows for urban initiatives that were implemented in different regions by different groups to be compared against each other.

To validate this observation, we aim to compare the urban initiatives described in Chapter 6 in Kirchheim, to an urban initiative implemented in Pittsburgh, which was planned and executed independently from our research. In May 2012, the city of Pittsburgh and a research group from the Carnegie Mellon University agreed to install a pilot of the Scalable Urban Traffic Control (Surtrac) traffic light system in 9 intersections in the region[SBXR13]. The idea behind the system is to regulate the crossings as a scheduling problem, where each lane in a traffic flow is seen as queue. Cameras are used in each intersection to count the vehicles queuing at the respective traffic lights. This queue is then used to decide whether to switch or extend the traffic light phase. This schedule is then communicated to neighboring intersections, which can then optimize their own switching based on the incoming traffic flow [SBXR13].

In order to validate our observation, we map the Surtrac urban initiative to the same table structure which was previously applied in Kirchheim. Table 7.1 describes the Surtrac system as an urban initiative within the Scoup framework.

The final report for Surtrac [SBXR13] cites the impact in traffic flow the system generated. From that, it is possible to derive the average commute time. The impact on air quality limits in the region are also monitored. Surtrac thus allows for the calculation of the same Iso 37120 indicator as described for Kirchheim.

Our study shows that the structure and indicators provided by Scoup do indeed allow for urban initiatives that were independently developed to be compared against each other. When doing so, however, it is important to note the difference in each of the communities. Being consistent with the Scoup city attributes, defined in Chapter 4, these regions have very different population sizes, economic resources, and climate conditions. These attributes are bound to affect how many sensors and smart traffic lights they are able to install, as well as the average flow expected from roads. One approach to address these differences could include to normalize the Iso indicator

Urban Initiative Name	Introduce Smart Traffic Light [SBXR13]
Indicator	Average commute time [ISO 37120] Greenhouse gas emissions measured in tonnes per capita [ISO 37120]
Participating actors	Pittsburgh, Traffic Light, Vehicles, SURTRAC System, Congested Intersections, Traficon cameras
Entry conditions	9 congested intersections are present, about 116m from each other SURTRAC is available
Flow of events	1) Traffic in East Liberty neighborhood increases, and Pittsburgh realizes this. 2) East Liberty undergoes a redevelopment, and converts one way ring road into a two-way traffic, thus creating new intersections. 3) New intersections are equipped with Traficon cameras monitoring inflow directions which communicate to the SURTRAC scheduler. 4) SURTRAC independently computes for each intersection a schedule for servicing all currently approaching vehicles. 5) This schedule is used by the smart traffic light to determine when to switch phases. 6) Schedule is recomputed every few seconds. 7) SURTRAC propagates schedule to neighboring intersections, which then use the outflow from one intersection as inflow for the next one during the computation. 8) Due to scheduled traffic light phasing, vehicles passing through the intersection need to stop less often.
Exit conditions	SURTRAC is installed on the intersection. Average commute time decreases

Table 7.1: SURTRAC Urban initiative description implemented in Pittsburgh.

impact based on invested dollars, but this would require additional data on the urban initiatives, which some communities may be reluctant to share.

This lack of normalization and difference on regional attributes represents a major limitation to Scoup's comparability claim. While analyzing the same indicator helps compare and track sustainability and digitalization efforts across communities, the difference in a regions context may often mean that even the same indicator may suffer from vastly different conditions, thus impacting their reliability. Although the quality attribute introduced in Section 6.5 tries to address this, it must be assumed that the true reasons that affect an indicator's reliability remain mostly unknown.

7.2 Representation of Physical Entities

The second core requirement of Scoup was to enable the representation of physical entities relevant for the urban context (requirement Scoup-Fr2). As described in Chapter 4, this was done through the introduction of a meta-model that related measurements, urban initiatives, cities, their population and constraints. The projects presented in this dissertation applied the meta-model in the mobility and transportation context. In the following we investigate the following observation:

Observation 10

Scoup's meta-model can be tailored to smart sustainable cities applications that can impact the Iso indicators.

To validate this claim, we apply Scoup's meta-model to a project that was developed independently from the mobility ones described in this dissertation. Furthermore, the described Ssc system does not relate to the mobility category, and can thus identify to what extent Scoup's meta-model can be applied to a different context.

In the *Ferienakademie* 2018, a group of 19 students from three universities participated in an Interdisciplinary System Course [SAWB19] to develop a Multi Operational Drone Collaboration Platform (Modcap)[ASB19]. The aim of the system was to enable the usage and coordination of multiple autonomous drones to help in search and rescue situations after natural disasters. By leveraging several unmanned aerial vehicles which individually had missions ranging from object identification to supply delivery, the fog architecture proposed by Scoup allowed for efficient communication between mission base and available drones, as well as a wider search area [ASB19]. While this project was only implemented at a prototype level, the presented proof of concept already allows for a consideration of the Scoup meta-model.

The result of the application of Scoup's meta-model to Modcap's system can be seen in Figure 7.1. To handle with the aspect that Modcap should be deployed to remote areas rather than cities, the region affected by the urban initiative is modeled as a Catastrophe Region, which more accurately depicts Modcap's deployment target. Furthermore, the system describes that the data collection is carried out by

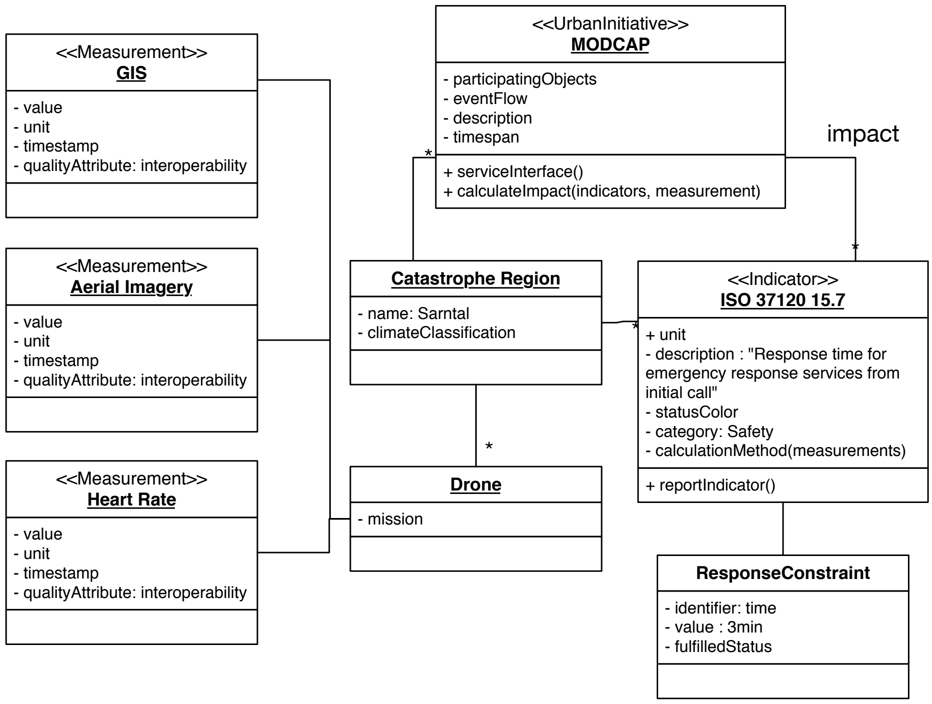


Figure 7.1: MODCAP system according to SCOUP's meta-model (UML Object Diagram).

drones which follow different missions, and thus the measurements are connected to the deployed drones. ISO 37120 has an indicator which measures the response time for emergency services, which should ideally be impacted by the MODCAP system. SCOUP's constraint further allows for the introduction of a response time for this constraint, that could the activation of further urban initiatives.

While Figure 7.1 shows that SCOUP's meta-model can be applied to MODCAP, some considerations are necessary. It is relevant to note that the MODCAP system does not apply for the classical SCOUP scenario, as MODCAP is aimed for usage in remote areas, where SSCs (or even internet connectivity) are usually not available [ASB19]. This means that while the meta-model could be applied and three of the ISO indicators could be impacted by the urban initiative (ISO 37120 15.3, ISO 37120 15.7, and ISO 37122 15.1), the indicators calculated only partially measure the urban initiatives success.

One approach to address this limitation could be the inclusion of the ISO 37123 indicators into the SCOUP framework, the third indicator standard in ISO's series that addresses the resilience of a city. While this indicator set was initially deemed out of scope for the SCOUP framework, if the encompassing of such systems becomes required the framework could be extended.

One further aspect of the MODCAP system that may require further tailorability of SCOUP is the calculation and display of geographic elevation differences over time. While GIS elevation can be modeled through measurements, and their difference through a composition of measurements, the mapping of this information to indicators is not direct.

7.3 Data Source Integration

The projects described in this dissertation had to handle vastly different data sources, and the integration of these was generally successful. From the different modes of transportation support in Chapter 5 to the diverse sensors and formats available in the Kirchheim community, the ability to handle different data formats is one of the core advantages of SCOUP. We formulate this as:

Observation 11

SCOUP supports the continuous addition of data sources.

To validate this, we consider the 2019 edition of the *Ferienakademie*, where an urban initiative was developed that used unmanned aerial vehicles to fly over remote water sources, gather water samples, and automatically analyze the water quality of the collected samples. As with the MODCAP system, this project was developed in the *Ferienakademie* course, where a group of students built a prototype for the Smart Water Monitoring (SWARM) system.

As with several of the systems analyzed throughout this dissertation, the measuring of water quality is a complex procedure, with no single defined or established methodology for it as well as several possible data sources [Kun+20]. To solve this, the SWARM system supported the collection of different water quality measurements, such as temperature and pH, in different regions of the lake. As "Clean Water and Sanitation" is present as goal number 6 of the SDGs, water quality indeed is one of the indicators tracked under ISO 37120, specifically two indicators of the standard apply: "percentage of the city population with potable water supply service" and the "percentage of population with access to improved sanitation" [ISO 37120]. It is nonetheless hard to map the measurements collected by the SWARM system to the mentioned indicators. To track these indicators, the SWARM system proposed in [Kun+20] maps good water availability to the indicators listed above, pH and temperature as a relevant measurement for these indicators, and the collection system as an urban initiative. Figure 7.2 presents the SCOUP meta-model applied to the SWARM system.

The usage of SCOUP in SWARM's system allowed for the data integration between the different sensors and drones to be carried out in the fog layer proposed by SCOUP. For this, each drone served as an intermediate fog layer, and aggregated the sensor's

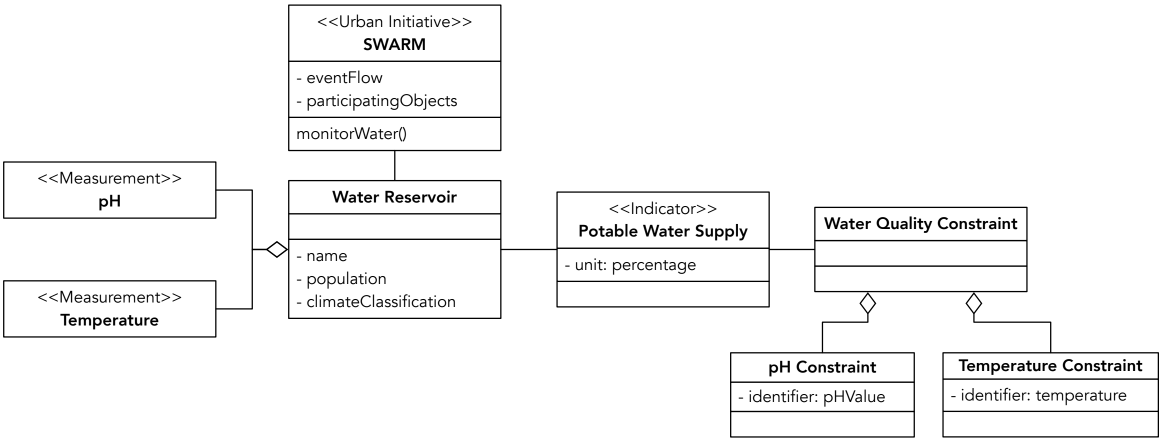


Figure 7.2: Scoup meta-model applied to Swarm system (Uml Class Diagram).

measurements for that area of the lake into intermediate results, before sending these to a persistent storage. In a context where it was hard to compare the collected measurements against further information, the certainty and interoperability quality attributes for indicators and measurements became more relevant, as single pH measurements could be inaccurate, and not represent the quality of the entire water reservoir [Kun+20].

The study with Swarm demonstrated that it is feasible to integrate different data sources with the architecture and relations proposed by Scoup. It is important to note, however, that throughout the other projects described in this dissertation, the integration of new data sources usually came with at least partial additional work. In Chapter 5, data source integration was possible as an industry standard had already been established and thus adding new modes of transportation meant using that standard, which made it easier. For the Kirchheim project, however, adding new data sources required the developed virtualization to parse all columns of the data into an extensible database. While this effort is worth it for automatic data sources which receive frequent updates, it may be a big overhead for static sources or for community sources which are only available as Pdfs and rarely updated.

The advantages of fog computing and data virtualization allow for fast integration of real time data sources. While these advantages are definitely relevant, the information required to calculate most Iso indicators tend to vary little over a year, which may mean that a different update and input mechanism these may be reasonable. The additional work necessary from each project for the successful integration is seen as a limitation of Scoup.

7.4 Multiple Stakeholder Support

A further important requirement for Scoup was the support of different, and sometimes conflicting, stakeholders (Scoup-Fr4). We now evaluate to what extent this was achieved.

Observation 12

Scoup allows for compromises when supporting multiple stakeholders.

As was seen in Chapter 6, the Kirchheim Smart Mobility project profited from the multiple stakeholder support from Scoup. The community of Kirchheim had to handle different expectations from their citizens, industry partners, and local businesses. When analyzed together with the data source integration just described, the multiple stakeholder support requirement leads to a data privacy consideration, which especially in a public office context may become critical.

The experience in Kirchheim has shown that when all information is public, industrial partners become more open to share their own information in return for reasonable urban initiatives. Both the Munich public transportation company and Iabg made information available inside of the initiative, that would otherwise be hard to come across. While perhaps not all of the urban initiatives planned in Kirchheim were carried out at their full extent, the fact that they were even considered and analyzed presented an advantage to these industries big enough to be willing to share their information.

Different stakeholders will inherently provide and require both different datas and initiatives from any smart city system. While the idea behind Scoup is to aggregate the available information into the same consistent indicators, the data required for this calculation may not be available to all stakeholders involved. Furthermore, political motivations may mean that some of the indicators may not be desired to be publicly reported upon. While Sdg 16 “Peace, Justice, and Strong Institutions” supports the public availability of public information, it may be unrealistic to expect every smart city worldwide to make an honest effort toward that goal.

In order for the different stakeholders to agree to share their information in a public platform, they may often expect something in return, and access to the existing data may or may not be enough. In order for Sdg 16 to become attainable, it is important that the communities show the value of sharing information through continuous introduction of urban initiatives that improve quality of life and sustainability. Scoup facilitates this sharing and visualization.

7.5 Discussion

In general the requirements elicited for Scoup were addressed at different levels on each of the described projects and the general framework. In the following we compare which project addressed which Scoup requirement described in Section 4.1, as can be seen in Table 7.2.

Requirements	SURTRAC	Urban Routing	Kirchheim
Comparability among regions			
Representation of physical entities			
Addition of data sources			
Consideration of multiple stakeholders			
Calculation of sustainability indicators			
Switching between possible initiatives			
Evaluation of initiatives through digital representations			

Table 7.2: Analysis of which Scoup requirements were necessary in each of the main projects developed with the framework.

When we compare the realization of Scoup's requirements to the systems developed with Scoup (Urban Routing and Kirchheim) versus the one without (Surtrac), it is no surprise that Surtrac addresses the presented requirements to a lesser degree. While all of the Scoup requirements were addressed in either the Urban Routing or the Kirchheim projects, this was done at a different level in each. The representation of physical entities, for example, was initially aimed to be a relevant requirement for more accurately representing the complex urban context, and while this was partially addressed through the usage of simulations in Chapter 6, several abstractions were made for these simulations, which meant that the original requirement considerably changed in scope. The Urban Routing project, however, did not address this at all, as the urban initiatives presented there are entirely digital.

The comparability among regions was envisioned as a core Scoup requirement, and while the presented Iso indicators do provide useful benchmarks in that direction, our analysis shows that is is hard to ensure the quality of each measurement involved in these indicators. While some projects, such as the one described in Chapter 6, may provide relevant measurements for an indicator, it is hard to ensure the accuracy of each measurement, while keeping the resultant indicator comparable. It is relevant to note that the calculation of sustainability indicators and comparability among regions are the requirements that differ the most between Surtrac and the applications that implemented Scoup.

A further requirement which was only partially implemented is the switching between urban initiatives. The main challenge in this respect is the implementation of more than one urban initiatives simultaneously. While this was initially planned in Kirchheim, a more detailed consideration of the Commutify initiative proved to be unsuitable, and its implementation was thus rejected.

Chapter 8

Conclusion

8.1 Summary 108
8.2 Future Work 109
8.3 Epilogue 110

The goal of Scoup was to develop a computational framework to strategically use growing sensor and urban data sources to modularly and comparatively benchmark and evaluate progress toward the United Nation's (Un) Sustainable Development Goals (Sdg), specifically toward goal 11: Sustainable Cities and Communities. Cities are continuing to collect extensive data about their own regions and initiatives and thus a framework that allows these to be compared has value.

The following chapter first summarizes the results of this dissertation in Section 8.1. We then provide an outlook on possible future work to these results in Section 8.2 and finalize with a discussion about how the concepts presented in this dissertation can be applied to urban initiatives worldwide in Section 8.3.

8.1 Summary

This dissertation presented SCOUP, a framework that aims to help smart and sustainable cities track their progress toward the SDGs. SCOUP itself is composed of three main components: (1) an indicator formalization that allows ISO standards 37120 and 37122 to be mapped into OCL language, (2) a meta-model relating cities' measurements to urban initiatives and indicators to track them, and (3) software components that aid in the planning and tracking of these indicators.

The indicator formalization is important as it allows SCOUP to have a verifiable method of tracking if an indicator is being reported on and kept compliant to. This further allowed us to form a constraint definition, that uses hierarchical composition to encapsulate different layers of constraints. As far as the sustainability aspect of SCOUP goes, an implementation of the constraints to ISO 37120 and 37122 was deemed to address progress toward the SDGs, although potentially different indicator sets could be used with the same formalization. These two indicator sets were chosen due to their direct relation to many of the SDGs, as well as their coverage of not only several categories within urban planning but also of both digitalization and sustainability efforts in Smart Sustainable Cities (SSC).

The meta-model presented in SCOUP uses two compositions to aggregate both measurements and constraints in a hierarchical aspect. Constraints use SCOUP's formalization results to calculate indicators with measurements and thus track whether the indicators are within acceptable ranges. Furthermore it is explained how the relation between cities, urban initiatives and indicators can be analogous to the one between people and preferences. Urban initiatives are described to be one of the key instruments cities can use to advance toward the SDGs, and thus it is important to track their impact on relevant indicators.

Last but not least, the implementation of SCOUP into urban initiatives was aided by the usage of data integrations that allowed for simulations to be calculated of some of the major indicators. The development of SCOUP was done through an iterative process, where each of the components were implemented and extended through reference applications. During this development, two main systems that included several additional components were presented: (1) the BR router for multi modal urban routing, and the (2) SmartHeim2 project, a dashboard for smart mobility that correlates traffic and air quality indicators.

The multi modal router developed in partnership with BR presents a first step in the goal of integrating all different mobility providers into potentially the same route from A to B. While at a graph theory level this proves an ongoing challenge, the consideration of personal variables for route evaluation was shown as an approach to prioritize which routes are preferable for which person. Furthermore, these preferences were generalized into constraints that could be parsed into the route and an algorithm was presented to automatically detect which mode of transportation a passenger used.

Last but not least the SmartHeim2 project in Kirchheim showed how different available data sources can be aggregated to calculate sustainability indicators. Several urban development software components were presented, including one that facilitates simulation of modifiable road networks and the usage of these for the evaluation of road construction initiatives. Furthermore a dashboard was presented where not only the correlation of air quality indicators and traffic flow could be observed, but also prediction models as to what would happen to one based on changes of the other. These dashboards and simulations were callibrated based on sensors installed throughout the streets of Kirchheim, and the aggregated measurements were then used to calculate some of the ISO indicators for the Kirchheim municipality.

8.2 Future Work

In the reference applications presented in Chapters 5 and 6, SCOUP was shown to be valuable in aiding the integration of different data sources and keeping a consistent system architecture. There are nonetheless areas where SCOUP can still be further extended to support new applications.

A visualization of all the information gathered could be an important step toward the data and correlation analysis of the different indicators. Ideally such a platform should connect to different open data initiatives of cities worldwide, to more easily allow for an automatic pipeline of indicator calculation and tracking. By integrating an indicator visualization to open data initiatives, it would be possible to easily show the value of the gathered measurements, and perhaps incentive other regions to make more of their data publicly accessible.

Furthermore, such a visualization for SCOUP's gathered information could allow for an additional relevant data source: feedback from the region's citizens. Cities and their urban planning is inherently shaped by their citizens. A mechanism to allow for an active citizen participation in the prioritization and execution of urban initiatives could express an important feedback loop for urban planners, and thus represent a further step in making cities more agile systems. Polst *et al.* present some initial research of how citizen feedback can serve as requirement source for smart cities, and what is the best method to inquiry citizens [PE20]. Integrating this into SCOUP could allow the framework to serve as an interface between citizens and urban planners.

On the vision toward a digital twin of the SSC, an important consideration is the architectural planning of buildings and their internal structures. The integration of

Building Information Modeling softwares into the simulation and data aggregation capabilities of Scoup could allow for an integrated approach between the different buildings in a city. This could allow for the consideration of energy and water profiles for neighborhoods inside the city, and thus aid urban planners in the management of the cities' resources.

Extending the simulation aspects of Scoup could also provide relevant future work for the framework. City aspects such as energy consumption, waste management, health and education capabilities, and safety concerns represent some of the categories in Iso 37120 and 37122 that could also be better planned with the aid of more extensive simulations. While simulations for aspects such as energy consumption do exist, the integration of these with the measurements and indicators available in Scoup could lead to important insights for urban planners.

8.3 Epilogue

During the finalization of this dissertation, the worldwide Covid-19 pandemic started. The ongoing situation provides an interesting case study for the difficulties of city comparison. Throughout the pandemic, almost every country worldwide started to track and report on the same two indicators: (1) current number of reported cases in the country and (2) current number of deaths. This became one of the rare examples of where data truly seems comparable across regions, the progress toward the fight of the disease could be measured, and the results of the different health initiatives compared.

Indeed, within weeks of the World Health Organization having declared the disease a worldwide pandemic (and in some cases, even before that) several online dashboards started to appear, where real-time data between different countries, states, or regions could be directly compared against one another. While this led to a direct comparison between the commonly measured indicators – that coupled together with news sources allowed for an analysis of the effectiveness of urban initiatives – this did highlight the importance of a common methodology. In several cases the vast difference in testing capabilities, case reporting frequency, and exact case definition in each country led to numbers that were hard to rely on and analyze during the pandemic. An example of this can be seen in the variability reported in the daily new cases for several cities. Due to reporting and testing infrastructure in the different days of the week, it was common to see a big drop in the new cases over weekends in several regions worldwide.

Furthermore, the speed with which the pandemic spread across the world – as did the news and reporting of it – goes along to make a good example of how interconnected not only cities, but also society and the world as a whole, have become. The effects on the economics of a health crisis show how one section of urban life continues to be intrinsically linked to all other areas, both locally and globally. If used to its' full extension, technology and data reporting can be a key enabler for that essential integration.

While the goals set forth by the SDG's may seem to be too optimistic based on today's technology and status quo, achieving them will require true cathedral thinking. The concept of planning long-term goals in the present to problems that will affect and must be continued to solve by future generations[1] has been exemplified in the construction of several cathedrals worldwide. When construction began on the Santa Maria del Fiore Florence Cathedral in 1296, neither the architect nor the construction workers of the time had concrete plans of how to build the dome. It was not until over 100 years later, in 1418 that Filippo Brunelleschi proposed the technique that would allow the structure to sustain itself[2]. While the concrete techniques to achieve the SDGs may not all be known, the foresight necessary, for now, is to lay the foundations that will allow the construction, monitoring, and comparison of urban initiatives aiming to move toward them.

[1] Cathedral thinking: https://cathedralthinking.com/
[2] Brunelleschi's Dome: https://www.nationalgeographic.com/magazine/2014/02/Il-Duomo/

Appendices

Appendix A

Urban Initiatives for the Sustainable Development Goals

While the 17 SDGs proposed by the United Nations are well known, it is not always clear what can be concretely done to achieve them. The following list aims to provide examples of urban initiatives that can be undertaken by urban planners, society leaders, and citizens to help achieve the goals. This list was compiled through initiatives partially proposed by the "SDGs in Action" application[1], the community of Kirchheim, stakeholder conversations during the composition of this dissertation, as well as the author.

Goal 1) No Poverty

- Solar powered irrigation
- Provide micro loans
- Incentive local businesses to hire locally
- Provide skill workshops
- Provide contraception methods

Goal 2) Zero Hunger

- Solar powered irrigation
- Improve food distribution
- Reduce food waste

[1] SDGs in Action application: `https://sdgsinaction.com/`

Goal 3) Good Health & Well- Being

- Integrate hospitals & health systems
- Promote vaccination
- Improve sanitation knowledge
- Provide nursing homes

Goal 4) Quality Education

- Incentivize school participation, for example through food distribution at school
- Offer quality online courses
- Provide internet access in schools
- Technical skills workshop
- Add online database access to public libraries

Goal 5) Gender Equality

- Measure & incriminate pay gap
- Provide skill workshops for women
- Provide longer father leaves
- Paid parent leave for both fathers and mothers
- Incentivize women to take leadership roles

Goal 6) Clean Water and Sanitation

- Provide mechanisms to filter water in pollution sources
- Incentivize and educate population on recycling campaigns
- Separate storm and sewer water systems
- Digitalize water monitoring stations, and provide these as data sources for population

Goal 7) Affordable and Clean Energy

- Allow for energy storage
- Use renewable energy to power all public spaces
- Activate public lights only when needed with detection sensors
- Incentive private (home) energy production, through solar energy, for example
- Incentivize use of digital thermostats, to use less energy warming/cooling buildings

Goal 8) Decent Work and Economic Growth

- Provide loans for small business
- Track which companies use sustainable materials and make that information public. Incentivize population to prefer these
- Provide job training for unemployed youth

Goal 9) Industry, Innovation and Infrastructure

- Limit air pollution per industry and source
- Contract local industries
- Incentivize start up creation
- Modernize energy and water infrastructure

Goal 10) Reduced Inequality

- Construct mixed usage public areas
- Provide affordable public housing
- Incentivize and enable civic participation in public infrastructure planning

Goal 11) Sustainable Cities & Communities

- Provide true mobility & active lifestyle infrastructure
- Plant trees and increase sidewalks to make cities more walkable
- Limit use of single occupancy vehicles
- Promote usage of high occupancy vehicles

Goal 12) Responsible Consumption and Production

- Promote recycling of industrial goods
- Improve waste management to enable a circular economy
- Creation of a digital information and education center

Goal 13) Climate Action

- Educate the population on how to consume less red meat
- Incentivize sustainable farming
- Air quality monitoring of the entire municipal area
- Identification of causes & measures for the reduction and avoidance of pollutants

Goal 14) Life Below Water

- Treat gray water for reuse
- Reduce spillage of black water

Goal 15) Life on Land

- Don't print and save on paper
- Integrate sustainability compliance into automatic building permit process

Goal 16) Peace, Justice and Strong Institutions

- Promote open data initiatives among urban planners and citizens.
- Improve accountability of institutions reporting on sustainability indicators.

Appendix B

Kirchheim Indicators

Category	ISO Number	Calculated Indicator	Communiyt Data Source
Economy	ISO 37120 5.1	City's unemployment rate	Arbeit - Arbeitslose.xlsx
	ISO 37120 5.4	Youth unemployment rate	Arbeit - Arbeitslose.xlsx
	ISO 37120 5.7	Annual number of visitor stays (overnight) per 100.000 population	Unternehmen - Tourismusbetriebe.xlsx
Education	ISO 37120 6.4	Primary education student-teacher ratio	Bildung - Lehrer zu Schüler Verhältnis.xlsx
Energy	ISO 37120 7.1	Total-end-use energy consumption per capita	Versorgung - Stromverbrauch.xlsx
	ISO 37120 7.2	Percentage of total end-use energy derived from renewable resources	Versorgung - Stromerzeugung.xlsx
Environment and climate change	ISO 37120 8.1	Fine particulate matter (PM2.5) concentration	Real-time sensors
	ISO 37120 8.2	Particulate matter (PM10) concentration	Real-time sensors
	ISO 37120 8.7	03 (ozone) concentration	Real-time sensors
	ISO 37120 8.5	NO2 (nitrogen dioxide) concentration	Real-time sensors
	ISO 37120 8.6	SO2 (sulfur dioxide) concentration	Real-time sensors
	ISO 37122 8.2	Number of real-time remote air quality monitioring stations per square kilometre	Real-time sensors
Governance	ISO 37120 10.4	Voter participation in last municipal election	Wahl - Gemeinde Wahlbeteiligung.xlsx
Housing	ISO 37120 12.5.1	Total number of households	Bau - Bestand Gebäude.xlsx & Bevölkerung - Entwicklung Kirchheim.xlsx
	ISO 37120 12.5.2	Persons per unit	Bau - Bestand Gebäude.xlsx & Bevölkerung - Entwicklung Kirchheim.xlsx
	ISO 37120 12.5.4	Living space (square metres) per person	Bau - Bestand Gebäude.xlsx & Bevölkerung - Entwicklung Kirchheim.xlsx
Population and cocial conditions	ISO 37120 13.4.1	Annual population change	Bevölkerung - Entwicklung Kirchheim.xlsx
	ISO 37120 13.4.3	Population demographics	Demographie - Entwicklung detailliert.xlsx
Transportation	ISO 37120 19.3	Percentage of commuters using travel mode to work other than personal vehicle	Commutify & MVG
	ISO 37120 19.5	Transportation deaths per 100.000 population	Verkehr - Unfälle Kirchheim.xlsx
	ISO 37120 19.7	Average commute time	real-time sensors & MVG
	ISO 37120 19.8.1	Number of personal automobiles per capita	Verkehr - Kraftfahrzeugbestand Kirchheim.xlsx
	ISO 37120 19.8.2	Number of two-wheeled motorized vehicles per capita	Verkehr - Kraftfahrzeugbestand Kirchheim.xlsx
	ISO 37122 19.3	Percentage of vehicles registered in the city that are low-emission vehicles	Verkehr - Kraftfahrzeugbestand Kirchheim.xlsx

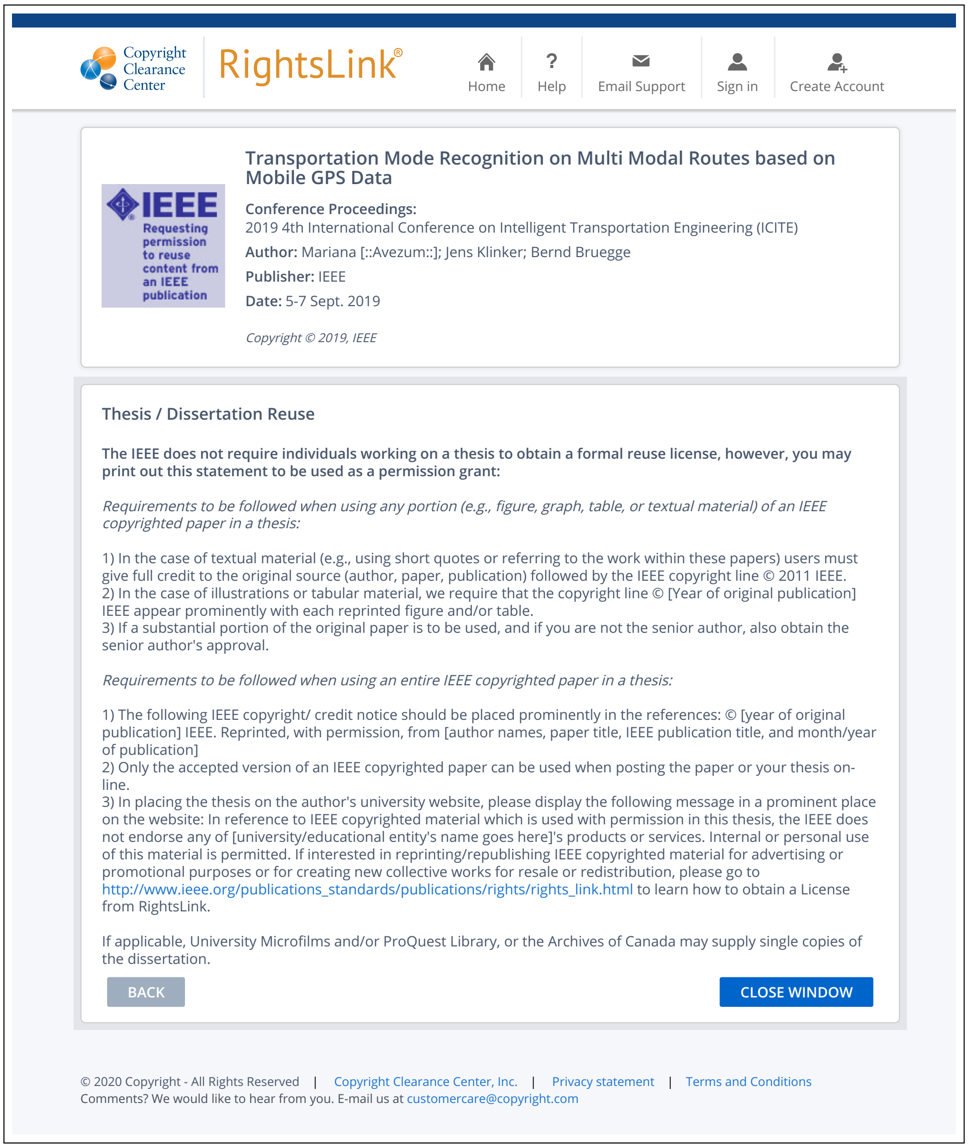
Urban planning	ISO 37120 21.1	Green area (hectares) per 100.000 population	Fläche - ALKIS Nutzen.xlsx & Bevölkerung - Entwicklung Kirchheim.xlsx
	ISO 37120 21.3	Jobs-housing ratio	Bau - Bestand Gebäude.xlsx & Arbeit - Soz.Pfl. Beschäftigte am Arbeitsort - nach Wirtschaftsbereich.xlsx
	ISO 37120 21.4	Basic Service Proximity	Open Street Maps
	ISO 37120 21.5.1	Population density	Fläche - ALKIS Nutzen.xlsx & Bevölkerung - Entwicklung Kirchheim.xlsx

Table B.1: List of indicators that were calculated for the municipality of Kirchheim, their respective ISO numbers and internal data sources in the community.

Appendix C

IEEE License

Copyright material from IEEE was used for the referenced work from Avezum *et al.* [AKB19]. Reuse permission was obtained as can be seen in Figure C.1.



Copyright Clearance Center | RightsLink® | Home | Help | Email Support | Sign in | Create Account

IEEE Requesting permission to reuse content from an IEEE publication

Transportation Mode Recognition on Multi Modal Routes based on Mobile GPS Data

Conference Proceedings:
2019 4th International Conference on Intelligent Transportation Engineering (ICITE)

Author: Mariana [::Avezum::]; Jens Klinker; Bernd Bruegge

Publisher: IEEE

Date: 5-7 Sept. 2019

Copyright © 2019, IEEE

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK | CLOSE WINDOW

Figure C.1: Permission grant for [AKB19].

List of Abbreviations

ADAC Allgemeiner Deutscher Automobil-Club

API Application Programming Interface

BIM Building Information Management

BR Bayerischer Rundfunk

CMU Carnegie Mellon University

CPS Cyber Physical System

EU European Union

FR Functional Requirement

FTP File Transfer Protocol

GIS Geographic Information System

GPS Global Positioning System

GTFS General Transit Feed Specification

HBEFA Handbook Emission Factors for Road Transport

IABG Industrieanlagen-Betriebsgesellschaft mbH

ICT Information and Communication Technologies

IHK German Chamber of Industry and Commerce (in German: *Industrie- und Handelskammern*)

IoT Internet of Things

ISO International Organization for Standardization

ITS Intelligent Transportation Systems

ITU International Telecommunication Union

JOSM Java OpenStreetMap Editor (Software)

MODCAP Multi Operational Drone Collaboration Platform

MOF Meta Object Foundation

MVV Münchner Verkehrs- und Tarifverbund

OCL Object Constraint Language

OSM OpenStreetMap

OTP Open Trip Planner

PDF Portable Document Format

SCOUP Sustainable Constraint-based Urban Planning framework

SDG Sustainable Development Goal

SQL Structured Query Language

SSC Sustainable Smart Cities

SUMO Simulation of Urban Mobility (Software)

SURTRAC Scalable Urban Traffic Control

SWARM Smart Water Monitoring

SysML Systems Modeling Language

TUM Technical University of Munich

UML Unified Modeling Language

UN United Nations

UNECE United Nations Economic Commission for Europe

VU VU Framework, presented by Ganslmeier [Gan15]

List of Figures

1.1 Research Process (UML Activity Diagram) 6

2.1 Ubiquitous Computing Taxonomy (UML Class Diagram) 16
2.2 Cyber Physical Systems Meta-Model (UML Class Diagram) 17
2.3 DataSource Taxonomy (UML Class Diagram) 20
2.4 VU Framework (UML Class Diagram) 22

3.1 Smart City Keyword Pie Chart 24
3.2 ISO Smart Cities Indicators Overview 28
3.3 ISO Indicator Taxonomy (UML Class Diagram) 29

4.1 SCOUP Use Case Diagram 41
4.2 Public Transport System Size Indicator (UML Object Diagram) 44
4.3 SCOUP's Indicator Formalization (UML Class Diagram) 49
4.4 SCOUP Meta-Model (UML Class Diagram) 50
4.5 SCOUP City Input Process (UML Activity Diagram) 54
4.6 SCOUP Overview 55

5.1 Multi-Modal BR Router Screenshots 61
5.2 Multi-Modal BR Router Subsystem Decomposition (UML Component Diagram) 63
5.3 RouteSpecification & Constraint Taxonomy (UML Class Diagram) . . . 66
5.4 Mode of Transportation Decision Tree 69
5.5 Mode of Transportation Confusion Matrix 70
5.6 Multi-Modal BR Router Analysis Object Model (UML Class Diagram) 71
5.7 Extension of SCOUP Meta-Model with Preferences & Constraints (UML Class Diagram) 72

6.1 Kirchheim Traffic Flow 75
6.2 Kirchheim Smart Mobility Project Requirements (SysML Requirements Diagram) 77
6.3 Commutify Urban Initiative According to SCOUP (UML Class Diagram) 80
6.4 SmartHeim2 System Subsystem Decomposition (UML Component Diagram) 83
6.5 SUMO Simulation of Kirchheim 85
6.6 Kirchheim Traffic Flow Comparison 86
6.7 Sensors Installed in Kirchheim (SysML Parametric Diagram) 89
6.8 PM10 Values in Kirchheim Based on Sensors & Model 91
6.9 KoAiry Dashboard 92
6.10 SmartHeim2 System Analysis Object Model (UML Class Diagram) . . . 93

7.1 MODCAP Instantiation of SCOUP (UML Object Diagram) 101
7.2 SCOUP Meta-Model Applied to SWARM system (UML Class Diagram) . 103

C.1 IEEE Permission Grant for [AKB19] 121

List of Tables

2.1 MOF Abstraction Layers 13

4.1 ISO 37120 19.1 City Comparison 45

6.1 Commutify Urban Initiative 81
6.2 Kirchheim Roundabout Analysis Use Case 87
6.3 Quality Attributes for SCOUP's stereotypes 94

7.1 SURTRAC Urban Initiative 99
7.2 SCOUP Requirements Analysis 105

B.1 Kirchheim Indicators 120

Bibliography

[AB13] A. Antrim and S. J. Barbeau. "The Many Uses of GTFS Data–Opening the Door to Transit and Multimodal Applications." In: *Location-Aware Information Systems Laboratory at the University of South Florida* 4 (2013).

[ABD15] V. Albino, U. Berardi, and R. Dangelico. "Smart Cities: Definitions, Dimensions, Performance, and Initiatives." In: *Journal of Urban Technology* 22 (February 2015). DOI: 10.1080/10630732.2014.942092.

[Abk81] M. D. Abkowitz. "An Analysis of the Commuter Departure Time Decision." In: *Transportation* 10 (1981), pp. 283–297. DOI: 10.1007/BF00148464.

[ADAC19] *ADAC Autokosten Herbst/Winter 2019/2020.* Report. 17.03.1000 - IN 30274. ADAC: Allgemeiner Deutscher Automobil-Club, 2019.

[ADB12] T. Ali, P. Dutta, and H. Boruah. "A New Combination Rule for Conflict Problem of Dempster-Shafer Evidence Theory." In: *International Journal of Energy, Information and Communications* 3.1 (2012), pp. 35–40.

[AE17] K. M. Alam and A. El Saddik. "C2PS: A Digital Twin Architecture Reference Model for the Cloud-Based Cyber-Physical Systems." In: *IEEE Access* 5 (2017), pp. 2050–2062. DOI: 10.1109/ACCESS.2017.2657006.

[AKB19] M. Avezum, J. Klinker, and B. Bruegge. "Transportation Mode Recognition on Multi Modal Routes Based on Mobile GPS Data." In: *2019 4th International Conference on Intelligent Transportation Engineering (ICITE).* IEEE. 2019, pp. 141–146. DOI: 10.1109/ICITE.2019.8880187.

[ASB19] M. Avezum, A. Seitz, and B. Bruegge. "MODCAP: A Platform for Cooperative Search and Rescue Missions." In: *Proceedings of the Workshops of the Software Engineering Conference 2019.* Vol. 2308. 2019, pp. 63–66.

[Ave17] M. Avezum. "An Agile Process for Hardware & Software Co-Design. WARR Hyperloop: A Case Study." Master Thesis. Technische Universität München, 2017.

[BA18] L. Boskovski and M. Avezum. “Combining Hardware and Software Development: A Case Study on Interdisciplinary Teaching Projects.” In: *Combined Proceedings of the Workshops of the German Software Engineering Conference 2018 (SE 2018)*. 2018, pp. 12–15.

[Bau+13] M. Bauer, M. Boussard, N. Bui, F. Carrez, C. Jardak, J. De Loof, C. Magerkurth, Meissner, Stefan, A. Nettsträter, A. Olivereau, M. Thoma, J. W. Walewski, J. Stefa, and A. Salinas. *Internet of Things–Architecture IoT-A Deliverable D1. 5–Final architectural reference model for the IoT v3. 0*. IoT-A. Seventh Framework Programme, July 2013.

[BCHK19] K. S. Bunds, J. M. Casper, J. A. Hipp, and J. Koenigstorfer. “Recreational Walking Decisions in Urban Away-From-Home Environments: The Relevance of Air Quality, Noise, Traffic, and the Natural Environment.” In: *Transportation research part F: traffic psychology and behaviour* 65 (2019), pp. 363–375. DOI: /doi.org/10.1016/j.trf.2019.08.006.

[BCK03] L. Bass, P. Clements, and R. Kazman. *Software Architecture in Practice*. Addison-Wesley Professional, 2003. ISBN: 9780132942799.

[BCTH12] A. Bolbol, T. Cheng, I. Tsapakis, and J. Haworth. “Inferring Hybrid Transportation Modes from Sparse GPS Data Using a Moving Window SVM Classification.” In: *Computers, Environment and Urban Systems* 38.6 (2012), pp. 526–537. DOI: 10.1016/j.compenvurbsys.2012.06.001.

[BD10] B. Bruegge and A. H. Dutoit. *Object-Oriented Software Engineering Using UML, Patterns, and Java*. Pearson Education, Inc, 2010. ISBN: 978-0-13-606125-0.

[BDD16] C. Benevolo, R. P. Dameri, and B. D’Auria. “Smart Mobility in Smart City.” In: *Empowering Organizations*. Ed. by T. Torre, A. M. Braccini, and R. Spinelli. Springer International Publishing, 2016. DOI: 10.1007/978-3-319-23784-8_2.

[BJR17] D. Bucher, D. Jonietz, and M. Raubal. “A Heuristic for Multi-Modal Route Planning.” In: *Progress in Location-Based Services 2016*. Springer, Cham, 2017, pp. 211–229. DOI: 10.1007/978-3-319-47289-8_11.

[BKA15] B. Bruegge, S. Krusche, and L. Alperowitz. “Software Engineering Project Courses with Industrial Clients.” In: *ACM Transactions on Computing Education (TOCE)* 15.4 (2015), pp. 1–31. DOI: 10.1145/2732155.

[BLS18] *Gemeinde Kirchheim b. München 09 184 131: Eine Auswahl wichtiger statistischer Daten*. Statistik Kommunal. Kirchheim 09 184 131. Bayerische Landesamt für Statistik, 2018.

[BMZA12] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli. "Fog Computing and Its Role in the Internet of Things." In: *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*. New York, NY, USA: Association for Computing Machinery, 2012, pp. 13–16. DOI: 10.1145/2342509.2342513.

[BR16] S. Boschert and R. Rosen. "Digital Twin—The Simulation Aspect." In: *Mechatronic Futures: Challenges and Solutions for Mechatronic Systems and their Designers*. Ed. by P. Hehenberger and D. Bradley. Springer International Publishing, 2016, pp. 59–74. ISBN: 978-3-319-32156-1. DOI: 10.1007/978-3-319-32156-1_5.

[BRRM95] B. Bruegge, E. Riedel, A. G. Russell, and G. J. McRae. "Developing Gems: An Environmental Modeling System." In: *IEEE Computational Science and Engineering* 2.3 (1995), pp. 55–68. DOI: 10.1109/99.414880.

[Bud19] D. Budischek. "Constraint-based Multimodal Transportation Routing." Bachelor Thesis. Technische Universität München, 2019.

[CGSV17] F. Cicirelli, A. Guerrieri, G. Spezzano, and A. Vinci. "An Edge-based Platform for Dynamic Smart City Applications." In: *Future Gener. Comput. Syst.* 76.C (2017), pp. 106–118. DOI: 10.1016/j.future.2017.05.034.

[Cic+18] F. Cicirelli, G. Fortino, A. Guerrieri, A. Mercuri, G. Spezzano, and A. Vinci. "A Metamodel Framework for Edge-Based Smart Environments." In: *2018 IEEE International Conference on Cloud Engineering (IC2E)*. IEEE. 2018, pp. 286–291. DOI: 10.1109/IC2E.2018.00067.

[Cul19] S. Cullen. "Causes and Effects of Suburban Traffic Dynamics." Master Thesis. Uppsala Universitet, 2019.

[DCGGM11] J. F. De Canete, C. Galindo, and I. Garcia-Moral. *System Engineering and Automation: An Interactive Educational Approach*. Springer-Verlag Berlin Heidelberg, 2011. DOI: 10.1007/978-3-642-20230-8.

[Dij59] E. W. Dijkstra. "A Note on Two Problems in Connexion With Graphs." In: *Numerische Mathematik* 1.1 (1959), pp. 269–271. DOI: 10.1007/BF01386390.

[DM14] F. Daniel and M. Matera. "Mashups: Concepts, Models and Architectures." In: *Mashups*. Springer, 2014, pp. 137–181. DOI: 10.1007/978-3-642-55049-2.

[Eng+20] E. Engström, M. A. Storey, P. Runeson, M. Höst, and M. T. Baldassarre. "How Software Engineering Research Aligns With Design Science: A Review." In: *Empirical Software Engineering* (2020). DOI: https://doi.org/10.1007/s10664-020-09818-7.

[EU19] *CO_2 Emissions from Cars: Facts and Figures*. Electronic Article. Accessed: 2019-04-08. European Parliament, 2019. URL: https://www.europarl.europa.eu/news/en/headlines/society/20190313STO31218/co2-emissions-from-cars-facts-and-figures-infographics.

[FBT06] K. Fitzpatrick, M. A. Brewer, and S. Turner. "Another Look at Pedestrian Walking Speed." In: *Transportation Research Record* 1982.1 (2006), pp. 21–29. DOI: 10.1177/0361198106198200104.

[FMST16] R. Frischknecht, A. Messmer, P. Stolz, and M. Tuchschmid. "mobitool–Grundlagenbericht." In: *treeze Ltd.* (November 2016). DOI: 10.13140/RG.2.2.24306.09921.

[FS13] L. Faria Stefanini. "Scrum of Scrums: Running Agile on Large Projects." In: *Scrum Alliance* (June 2013).

[Ful09] J. Fulmer. "What in the World is Infrastructure." In: *Infrastructure Investor* 1.4 (2009), pp. 30–32.

[Gam+95] E. Gamma, R. Helm, R. Johnson, J. Vlissides, and D. Patterns. *Elements of Reusable Object-Oriented Software*. Addison-Wesley Publishing Company, 1995. ISBN: 0201633612.

[Gan15] T. Ganslmeier. "Framework für die Realisierung eines hybriden Gesamtsystems zur hardwareunabhängigen Entwicklung elektronischer Fahrzeugapplikationen." Dissertation. München: Technische Universität München, 2015.

[Gob+18] L Gobeawan, E. S. Lin, A. Tandon, A. T. K. Yee, V. H. S. Khoo, S. N. Teo, S. Yi, C. W. Lim, S. T. Wong, D. J. Wise, et al. "Modeling Trees for Virtual Singapore: From Data Acquisition to CityGML Models." In: *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences* 42 (2018). DOI: 10.5194/isprs-archives-XLII-4-W10-55-2018.

[GSPM15] A. Gaur, B. Scotney, G. Parr, and S. McClean. "Smart City Architecture and its Applications Based on IoT." In: *Procedia computer science* 52 (2015), pp. 1089–1094. DOI: 10.1016/j.procs.2015.05.122.

[GSSD08] R. Geisberger, P. Sanders, D. Schultes, and D. Delling. "Contraction Hierarchies: Faster and Simpler Hierarchical Routing in Road Networks." In: *International Workshop on Experimental and Efficient Algorithms*. Springer, Berlin, Heidelberg. 2008, pp. 319–333. DOI: 10.1007/978-3-540-68552-4_24.

[Haf+04] S. Hafstein, R. Chrobok, A. Pottmeier, J. Wahle, and M. Schreckenberg. “A High-Resolution Cellular Automata Traffic Simulation Model with Application in a Freeway Traffic Information System.” In: *Computer-Aided Civil and Infrastructure Engineering* 19.5 (2004), pp. 338–350. DOI: 10.1111/j.1467-8667.2004.00361.x.

[Har+97] V. Hartkopf, V. Loftness, A. Mahdavi, S. Lee, and J. Shankavaram. “An Integrated Approach to Design and Engineering of Intelligent Buildings — The Intelligent Workplace at Carnegie Mellon University.” In: *Automation in Construction* 6.5-6 (1997), pp. 401–415. DOI: 10.1016/S0926-5805(97)00019-8.

[HBA19] A. Huovila, P. Bosch, and M. Airaksinen. “Comparative Analysis of Standardized Indicators for Smart Sustainable Cities: What Indicators and Standards to Use and When?” In: *Cities* 89 (2019), pp. 141–153. DOI: 10.1016/j.cities.2019.01.029.

[HBLK17] R. Hostettler, A. I. Böhmer, U. Lindemann, and A. Knoll. “TAF Agile Framework Reducing Uncertainty Within Minimum Time and Resources.” In: *2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC)*. IEEE. 2017, pp. 767–775. DOI: 0.1109/ICE.2017.8279962.

[Hel19] M. Held. “Potential of Dynamic Speed Advisory Equipment to Reduce Air Pollution.” Study Project at the Chair of Traffic Engineering. Technische Universität München, 2019.

[HKS09] S. Haller, S. Karnouskos, and C. Schroth. “The Internet of Things in an Enterprise Context.” In: *Future Internet – FIS 2008*. Ed. by J. Domingue, D. Fensel, and P. Traverso. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 14–28. DOI: 10.1007/978-3-642-00985-3_2.

[HNR68] P. Hart, N. Nilsson, and B. Raphael. “A Formal Basis for the Heuristic Determination of Minimum Cost Paths.” In: *IEEE Transactions on Systems Science and Cybernetics* 4.2 (1968), pp. 100–107. DOI: 10.1109/tssc.1968.300136.

[HW08] M. Haklay and P. Weber. “OpensStreetMap: User-generated Street Maps.” In: *IEEE Pervasive Computing* 7.4 (2008), pp. 12–18. DOI: 10.1109/MPRV.2008.80.

[ISO 37120] *Sustainable Cities and Communities — Indicators for City Services and Quality of Life*. Standard ISO 37120. International Organization for Standardization, July 2018.

[ISO 37122] *Sustainable Cities and Communities — Indicators for Smart Cities*. Standard ISO 37122. International Organization for Standardization, May 2019.

[ITU-a] *Collection Methodology for Key Performance Indicators for Smart Sustainable Cities.* Recommendation. ITU 978-92-61-25231-1. International Telecommunication Union, 2017.

[ITU-b] *Key Performance Indicators for Smart Evaluation and Assessment Sustainable Cities to Assess the Achievement of Sustainable Development Goals.* Recommendation. ITU-T Y.4903/L.1603. International Telecommunication Union, 2016.

[ITU-c] *Key Performance Indicators Related to the Sustainability Impacts of Information and Communication Technology in Smart Sustainable Cities.* Recommendation. ITU-T Y.4902/L.1602. International Telecommunication Union, 2016.

[ITU-d] *Key Performance Indicators Related to the Use of Information and Communication Technology in Smart Sustainable Cities.* Recommendation. ITU-T Y.4901/L.1601. International Telecommunication Union, 2016.

[ITU-e] *Overview of Key Performance Indicators in Smart Sustainable Cities.* Recommendation. ITU-T Y.4900/L.1600. International Telecommunication Union, 2016.

[ITU-f] *Overview of Smart Sustainable Cities Infrastructure.* Report. ITU-T FG-SSC 05/2015. International Telecommunication Union, 2015.

[KBA13] H. N. Koutsopoulos and M. Ben-Akiva. "Advanced Public Transport Systems, Simulation-Based Evaluation." In: *Transportation Technologies for Sustainability.* Ed. by M. Ehsani, F.-Y. Wang, and G. L. Brosch. New York, NY: Springer New York, 2013, pp. 100–121. DOI: `10.1007/978-1-4614-5844-9_297`.

[KHRW02] D. Krajzewicz, G. Hertkorn, C Rössel, and P. Wagner. "An Example of Microscopic Car Models Validation Using the Open Source Traffic Simulation SUMO." In: *Proceedings of Simulation in Industry, 14th European Simulation Symposium.* 2002, pp. 318–322. ISBN: 3-936150-21-4.

[Kit14] R. Kitchin. "The Real-Time City? Big Data and Smart Urbanism." In: *GeoJournal* 79.1 (2014), pp. 1–14. DOI: `10.1007/s10708-013-9516-8`.

[Kli19] J. Klinker. "Transportation Mode Recognition on Multi Modal Routes based on Mobile GPS Data." Guided Research. Technische Universität München, 2019.

[Kun+20] J. Kunze, V. Mayer, L. Thiergart, S. Javed, P. Scheppe, T. Tran, M. Haug, M. Avezum, B. Bruegge, and E. C. Ezin. "Towards SWARM: A Smart Water Monitoring System." In: *3rd International Conference on Industrial Cyber-Physical Systems (ICPS).* To be published. IEEE. 2020.

[Lop+18] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y. P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner. "Microscopic Traffic Simulation Using SUMO." In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. 2018, pp. 2575–2582. DOI: `10.1109/ITSC.2018.8569938`.

[MC12] P. Mooney and P. Corcoran. "The Annotation Process in OpenStreetMap." In: *Transactions in GIS* 16.4 (2012), pp. 561–579. DOI: `10.1111/j.1467-9671.2012.01306.x`.

[MGPG11] V. Maliene, V. Grigonis, V. Palevičius, and S. Griffiths. "Geographic Information System: Old Principles with New Capabilities." In: *Urban Design International* 16.1 (2011), pp. 1–6. DOI: `10.1057/udi.2010.25`.

[Mil12] M. A. Miller. "Bus Rapid Transit, Institutional Issues Related to Implementation." In: *Encyclopedia of Sustainability Science and Technology*. Ed. by R. A. Meyers. New York, NY: Springer New York, 2012, pp. 1818–1833. ISBN: 978-1-4419-0851-3. DOI: `10.1007/978-1-4419-0851-3_495`.

[Moeh+05] R. H. Möhring, H. Schilling, B. Schütz, D. Wagner, and T. Willhalm. "Partitioning Graphs to Speed up Dijkstra's Algorithm." In: *International Workshop on Experimental and Efficient Algorithms*. Springer, Berlin, Heidelberg. 2005, pp. 189–202. DOI: `10.1007/11427186_1`.

[MOF 1.4] *Meta Object Facility Core Specification Version 1.4*. Core Specification. Version 1.4. Object Management Group, 2002.

[MOF 2.5.1] *Meta Object Facility Core Specification Version 2.5.1*. Core Specification. Version 2.5.1. Object Management Group, 2016.

[Mvv] *Der MVV in Zahlen*. Electronic Article. Accessed: 2020-02-25. Münchner Verkehrs- und Tarifverbund, 2019. URL: `https://www.mvv-muenchen.de/ueber-den-mvv/der-verbund/mvv-in-zahlen/index.html`.

[MY15] K. Mori and T. Yamashita. "Methodological Framework of Sustainability Assessment in City Sustainability Index (CSI): A Concept of Constraint and Maximisation Indicators." In: *Habitat International* 45 (2015), pp. 10–14. DOI: `10.1016/j.habitatint.2014.06.013`.

[Nic+03] T Nicolai, D Carr, S. Weiland, H Duhme, O Von Ehrenstein, C Wagner, and E Von Mutius. "Urban Traffic and Pollutant Exposure Related to Respiratory Outcomes and Atopy in a Large Sample of Children." In: *European respiratory journal* 21.6 (2003), pp. 956–963. DOI: `10.1183/09031936.03.00041103a`.

[NT16] C. Narboneta and K. Teknomo. "OpenTripPlanner, OpenStreetMap, General Transit Feed Specification: Tools for Disaster Relief and Recovery." In: *CoRR* abs/1609.01472 (2016).

[NZ08] P. Neis and A. Zipf. “Zur Kopplung von Opensource, OpenLS und OpenStreetMaps in OpenRouteService.org.” In: *AGIT 2008. Symposium für angewandte Geoinformatik. Salzburg. Austria.* 2008.

[Orn16] S. Ornes. “Core Concept: The Internet of Things and the explosion of interconnectivity.” In: *Proceedings of the National Academy of Sciences* 113.40 (2016), pp. 11059–11060. DOI: 10.1073/pnas.1613921113.

[PBDL05] K. Pohl, G. Böckle, and F. J. van Der Linden. *Software Product Line Engineering: Foundations, Principles and Techniques.* Springer-Verlag Berlin Heidelberg, 2005. DOI: 10.1007/3-540-28901-1.

[PE20] S. Polst and F. Elberzhager. “Eliciting Requirements from Citizens–What Can We Learn from Other Disciplines?” In: *REFSQ 2020 Joint Proceedings of the Co-Located Events.* 2020.

[PFM07] M. C. Peel, B. L. Finlayson, and T. A. McMahon. “Updated World Map of the Köppen-Geiger Climate Classification.” In: *Hydrology and Earth System Sciences Discussions, European Geosciences Union* 11.5 (2007), pp. 1633–1644.

[PLW02] L. L. Pipino, Y. W. Lee, and R. Y. Wang. “Data Quality Assessment.” In: *Communications of the ACM* 45.4 (April 2002), 211–218. DOI: 10.1145/505248.506010.

[Qi+16] Y. Qi, M. Hunukumbure, M. Nekovee, J. Lorca, and V. Sgardoni. “Quantifying Data Rate and Bandwidth Requirements for Immersive 5G Experience.” In: *2016 IEEE International Conference on Communications Workshops (ICC).* IEEE. 2016, pp. 455–461. DOI: 10.1109/ICCW.2016.7503829.

[RLSS10] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic. “Cyber-Physical Systems: The Next Computing Revolution.” In: *Design Automation Conference.* IEEE, 2010, pp. 731–736. DOI: 10.1145/1837274.1837461.

[Sar19] J. Saroufim. “A Modular and Extensible Mobility Cost Model for Urban Commuter Route Planning.” Bachelor Thesis. Technische Universität München, 2019.

[SAWB19] A. Seitz, M. Avezum, S. Wagner, and B. Bruegge. “Interdisciplinary System Courses - Teaching Agile Systems Engineering.” In: *Proceedings of the Workshops of the Software Engineering Conference 2019.* 2019, pp. 11–18. DOI: 10.18419/opus-10361.

[SBXR13] S. Smith, G. Barlow, X. F. Xie, and Z. Rubinstein. “SURTRAC: Scalable Urban Traffic Control.” In: *Proceedings of Transportation Research Board 92nd Annual Meeting Compendium of Papers.* Transportation Research Board, 2013.

[Sch17] C. Scheuermann. "A Metamodel for Cyber-Physical Systems." Dissertation. München: Technische Universität München, 2017.

[Sch18] R. Schlichtmann. "Data Analysis Using Machine Learning Approaches for a Traffic Simulation in the Munich Metropolitan Area." Master Thesis. Technische Universität München, 2018.

[Sch20] D. Schemm. "Simulation, Prediction and Visualization of Air Quality based on Meteorological Conditions and Urban Mobility Data." Bachelor Thesis. Technische Universität München, 2020.

[Sch97] K. Schwaber. "Scrum Development Process." In: *Business Object Design and Implementation*. Ed. by J. Sutherland, C. Casanave, J. Miller, P. Patel, and G. Hollowell. Springer, 1997, pp. 117–134. DOI: `10.1007/978-1-4471-0947-1_11`.

[Sei19] A. Seitz. "An Architectural Style for Fog Computing: Formalization and Application." Dissertation. München: Technische Universität München, 2019.

[Sha76] G. Shafer. *A Mathematical Theory of Evidence*. Vol. 42. Princeton University Press, 1976. ISBN: 069110042X.

[She85] Y. Sheffi. *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*. Prentice Hall, Inc, England Cliffs, 1985. ISBN: 0139397299.

[Sim19] H. Simon. "Data Integration and Virtualization for Smart Mobility." Bachelor Thesis. Technische Universität München, 2019.

[SL19] Z. Sang and K. Li. "ITU-T Standardisation Activities on Smart Sustainable Cities." In: *IET Smart Cities* 1 (2019), pp. 3–9. DOI: `10.1049/iet-smc.2019.0023`.

[SMP 2.0] *Methodology and Indicator Calculation Method for Sustainable Urban Mobility*. Indicators Work Stream. Sustainable Mobility Project 2.0, World Business Council for Sustainable Development, December 2015.

[Smr19] J. Smretschnig. "An Analysis of Traffic Simulations Based on Modifiable OpenStreetMap Data." Bachelor Thesis. Technische Universität München, 2019.

[SONS18] E. d. S. de Santana, É. de Oliveira Nunes, and L. B. Santos. "The Use of ISO 37122 as Standard for Assessing the Maturity Level of a Smart City." In: *International Journal of Advanced Engineering Research and Science (IJAERS)* 5 (2018). DOI: `10.22161/ijaers.5.12.42`.

[SSBV16] C. Scheuermann, M. Strobel, B. Bruegge, and S. Verclas. "Increasing the Support to Humans in Factory Environments Using a Smart Glove: An Evaluation." In: *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*. IEEE. 2016, pp. 847–854. DOI: 10.1109/UIC-ATC-ScalCom-CBDCom-IoP-SmartWorld.2016.0134.

[STB18] A. Seitz, F. Thiele, and B. Bruegge. "Fogxy: An Architectural Pattern for Fog Computing." In: *Proceedings of the 23rd European Conference on Pattern Languages of Programs*. ACM. 2018, p. 33. DOI: 0.1145/3282308.3282342.

[Sve10] G. Svennerberg. *Beginning Google Maps API 3*. Apress, 2010. DOI: 10.1007/978-1-4302-2803-5.

[Swi+14] D. Swiatek, M. S. Pokorny, J. A. Tatto, J. Gonçalo, V. A.Z. C. Melo, and E. M. Dias. "Traffic Management Solutions in Large Cities-the Integrated Centre of Urban Mobility (CIMU) in São Paulo." In: *Proceedings of the 18th International Conference on Systems*. 2014.

[Thi+16] A. Thierstein, G. Wulfhorst, M. Bentlage, S. Klug, L. Gilliard, C. Ji, J. Kinigadner, H. Steiner, L. Sterzer, F. Wenner, et al. "WAM Wohnen Arbeiten Mobilität. Veränderungsdynamiken und Entwicklungsoptionen für die Metropolregion München." In: *Lehrstuhl für Raumentwicklung und Fachgebiet für Siedlungsstruktur und Verkehrsplanung der Technischen Universität München* (2016).

[UNDESA17] *Global Indicator Framework for the Sustainable Development Goals and Targets of the 2030 Agenda for Sustainable Development*. Report. (A/RES/71/313). United Nations, Department of Economic and Social Affairs, 2017.

[UNDESA18] *World Urbanization Prospects: The 2018 Revision*. Report. (ST/ESA/SER.A/420). United Nations, Department of Economic and Social Affairs, 2019.

[VGSB14] S. Varotto, A. Glerum, A. Stathopoulos, and M. Bierlaire. "Modelling Travel Time Perception in Transport Mode Choices." In: *14th Swiss Transport Research Conference*. 2014.

[Wal19] D. Wallner. "Entwicklung eines Radarsensorsystems zur echtzeitnahen Verkehrsflussmessung." Master Thesis. Hochschule für Angewandte Wissenschaften Landshut, 2019.

[WCED87] *Our Common Future*. World Commission on Environment and Development. Oxford: Oxford University Press, 1987. ISBN: 019282080X.

[Wei93] M. Weiser. “Hot Topics-Ubiquitous Computing.” In: *Computer* 26.10 (1993), pp. 71–72. DOI: 10.1109/2.237456.

[Wie14] R. J. Wieringa. *Design Science Methodology for Information Systems and Software Engineering*. Springer-Verlag Berlin Heidelberg, 2014. DOI: 10.1007/978-3-662-43839-8.

[Zeh18] D. Zehe. “Cloud Simulation for Large-Scale Agent-Based Traffic Simulations.” Dissertation. München: Technische Universität München, 2018.