U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Semantic Integration of Urban Mobility Data through Ontologies for Supporting Data Visualization

Thiago Sobral Marques da Silva

Supervised by:
Professor Maria Teresa Galvão Dias, Ph.D.
Professor José Luís Cabral Moura Borges, Ph.D.

Thesis submitted to the Faculty of Engineering, University of Porto
for the Doctoral Program in Transportation Systems
MIT Portugal Program

November 2018

We kindly acknowledge Fundação para Ciência e Tecnologia (FCT) for providing the scholarship PD/BD/105910/2014, under the scope of the Doctoral Program in Transportation Systems - MIT Portugal Program.

Abstract

The widespread adoption of Intelligent Transportation Systems by cities allows the generation of vast multisource data, which can be used to better understand urban mobility dynamics. Data visualization has been acknowledged as a valuable approach to assist domain users, e.g. analysts and decision makers, on answering their questions (analytical tasks). This thesis proposes an ontology-based approach to the problem of visualizing multisource heterogeneous urban mobility data. We focus on spatiotemporal data about events of a transportation network, e.g. ticket validations, accidents, travel intentions, among others. The problem is assumed to be twofold. Firstly, from the data perspective, datasets should be interoperable prior to visualization; a formal foundation is required to integrate urban mobility data and interrelate it to visual structures. Researchers have successfully applied ontologies to other problems in transportation, although there are no applications to visualization. Secondly, from the users' perspective, related studies do not consider the role of human factors to support tasks such as formalizing analytical tasks or finding appropriate visualization techniques to those tasks. Moreover, scientific literature shows evidence of little involvement of domain users throughout the process of design, development and evaluation of visualization techniques. This thesis provides four theoretical and practical contributions to the state of the art.

The first theoretical contribution is a formal conceptual model that interrelates three facets: structural characteristics of spatiotemporal urban mobility data, visualization techniques and empirical domain user knowledge. The model is materialized as the Visualization-oriented Urban Mobility Ontology (VUMO), the first and main practical contribution of this thesis. The ontology leverages metadata to provide a semantic foundation for user-oriented visualization systems. VUMO allows modeling various types of events and interrelates their structure with visualization concepts. To account for human factors, the ontology allows the annotation of empirical knowledge about visualization techniques and domain users, such as their profile description and feedback about visualization techniques. For instance, those may consist of quantitative or qualitative ratings. Analytical tasks can be formalized as data queries. We show how rule-based inference uses metadata to infer implicit knowledge from events data and analytical tasks, and to evaluate the compatibility of visualization techniques with analytical tasks. As the second theoretical contribution, we describe how VUMO supports user-oriented approaches to the development of visualization techniques, by relating the development phases and their artifacts to VUMO components.

To evaluate the validity of our approach, the second practical contribution is the *SUMVis* visualization system, which served as a testbed for case studies related to the analysis of public transportation ridership, origin-destination flows, and bus engine emissions. We collected public transportation data and involved domain users from the cities of Porto, Portugal, and Boston, United States. We demonstrate the integration of data, and discuss how VUMO can tackle inherent technical challenges, like extending the ontology to support other events, and building new rules for knowledge extraction. We exemplify the semantic annotation of visualization techniques and analytical tasks in terms of domain users' needs and requirements, and how rules evaluate compatibility and extract features from analytical tasks. Finally, we apply a user-oriented design methodology to demonstrate the annotation of artifacts, which are used to the development of a rule-based visualization recommendation method. The case studies show that the proposed approach could be applied to relevant real-world problems in urban mobility analysis. From a theoretical standpoint, we found the approach to be generalizable, as VUMO can also be used to model other types of spatiotemporal events and inference rules. This suggests that the ontology can effectively support other practical settings that require visualization of heterogeneous multisource data.

Keywords: urban mobility, spatiotemporal data, data visualization, data integration, ontologies

Resumo

As cidades têm adoptado Sistemas Inteligentes de Transporte que permitem gerar uma grande quantidade de dados. Embora estes sistemas permitam uma melhor compreensão das dinâmicas de mobilidade urbana, ainda apresentam desafios quanto à integração dos dados, frequentemente heterogéneos. A visualização de dados é uma abordagem relevante para auxiliar *stakeholders*, como agentes de decisão e peritos em mobilidade urbana, a responder às suas questões (tarefas analíticas). Esta tese propõe uma abordagem baseada em ontologias para resolver o problema de visualização de dados heterogéneos de mobilidade urbana provenientes de múltiplas fontes. Focamo-nos em dados espaciotemporais relacionados a eventos que podem ocorrer numa rede de transportes, tais como validações de bilhetes e intenções de viagem. O problema possui duas facetas. A primeira, relativa à perspectiva dos dados, introduz a necessidade de que os *datasets* sejam interoperáveis, i.e. é necessário definir uma conceptualização formal capaz de integrar os dados, e interrelacioná-los a estruturas visuais. O uso de ontologias mostra-se eficaz em algumas áreas de investigação em transportes, mas ainda não foram aplicadas à visualização de dados. A segunda faceta relaciona-se com o escasso envolvimento dos utilizadores no processo de concepção, desenvolvimento e avaliação das técnicas de visualização. A presente tese oferece um conjunto de contribuições para colmatar o referido problema.

A primeira contribuição teórica consiste num modelo conceptual formal que interrelaciona três componentes: características estruturais dos dados espaciotemporais de mobilidade urbana; técnicas de visualização; e conhecimento empírico obtido através dos *domain users*. Este modelo constitui a base para a materialização da ontologia VUMO (Visualization-oriented Urban Mobility Ontology), que consiste na primeira e principal contribuição prática desta tese. A ontologia aproveita o potencial dos metadados para fornecer a semântica necessária aos sistemas inteligentes de visualização (*knowledge assisted*). A ontologia VUMO permite a modelagem de diversos tipos de eventos, e a interrelação das suas estruturas com conceitos de visualização. Os fatores humanos também são considerados relevantes no processo de visualização. A ontologia permite a anotação semântica do conhecimento empírico relativo aos utilizadores, p.ex. os seus perfis e *feedback*, e às técnicas de visualização. As tarefas analíticas podem ser formalizadas através de *queries*. A presente tese demonstra a importância da inferência semântica na descoberta de conhecimento implícito nos eventos e tarefas analíticas. As regras lógicas subjacentes à inferência são utilizadas para avaliar a compatibilidade das técnicas de visualização com as tarefas analíticas. Como segunda contribuição teórica, a presente tese descreve como a ontologia VUMO pode ser aplicada a abordagens orientadas ao utilizador no desenvolvimento de técnicas de visualização, através da anotação semântica dos *artifacts* resultantes das fases de desenvolvimento.

Para avaliar a fiabilidade da abordagem proposta nesta tese, a segunda contribuição prática consiste na ferramenta de visualização *SUMVis*, que serviu como uma plataforma *testbed* para a condução dos estudos de caso relacionados à análise de procura em transportes públicos, matrizes origem-destino, e emissões de poluentes em autocarros. Os dados de transportes públicos e *domain users* considerados para a avaliação pertencem às cidades do Porto e Boston. Demonstramos o processo de integração de dados, e discutimos como a ontologia pode dar suporte a alguns desafios, como estender a ontologia a fim de suportar novas classes de eventos, e definir novas regras de inferência para extracção de conhecimento. Exemplificamos a anotação semântica de técnicas de visualização e tarefas analíticas em termos das necessidades e requisitos dos utilizadores, e como as regras são capazes de avaliar a compatibilidade e inferir propriedades das tarefas analíticas. Finalmente, aplicamos uma metodologia de desenho centrado no utilizador para demonstrar a anotação de *artifacts* que foram utilizados no desenvolvimento de um método de recomendação de visualizações baseado em regras de inferência. Os estudos de caso mostram que a abordagem proposta nesta tese pode ser aplicada a problemas reais e relevantes no âmbito da mobilidade urbana. Do ponto de vista teórico, a abordagem é considerada generalizável, uma vez que a ontologia também pode ser usada para modelar outros tipos de regras de inferência e eventos espaciotemporais.

Palavras-chave: mobilidade urbana, dados espaciotemporais, visualização e integração de dados, ontologias

Acknowledgments

The road was long but fruitful. I express my gratitude to Professors Teresa Galvão and Zé Luís for their kind friendship, confidence and support. Their guidance since my Master studies was fundamental to maturing my research ideas, which are now consolidated in this thesis. I also thank Professor Jorge Pinho de Sousa for the pleasant conversations and friendship, and supporting me to pursuit the doctoral degree.

I would like to thank Carla Monteiro, Isabel Ribeiro, Soledade Medeiros and Mónica Silva for the kind friendship, support and assistance. Moreover, the support of many professors from the Department of Industrial Engineering and Management shall not be forgotten. In particular, I would like to thank Professors Ana Camanho, António Brito, José Barros Basto, Gabriela Beirão, João Falcão e Cunha, Jorge Freire de Sousa, Jorge Teixeira, Henriqueta Nóvoa, Lia Patrício, Pedro Amorim and Vera Miguéis. I also thank the head of our department, Professor José António Sarsfield Cabral. His energy and dedication to our department are, undoubtedly, an inspiration.

This research benefits from a research period at the Massachusetts Institute of Technology (MIT). I am grateful to Professor Chris Zegras for the warm welcome as part of the CoAXs team, during my stay at the Department of Urban Studies and Planning. I would also like to thank Professors Jinhua Zhao and Gabriel Sánchez-Martinez (MIT), OPT S.A. and TIP S.A. for providing the datasets which were used in this research. A special thank you to Flávia Cardarelli and Christian Prothmann for receiving me so well at MIT, and for the nice conversations at coffee time.

No work environment is complete without good and supportive friends. I am thankful to my friends from room L201, namely Ana Loureiro, Ana Kustrak, Joana Hora, João Dias, Marta Ferreira, Renata Oliveira and Vera Costa. Likewise, I would like to thank my neighboring department friends Bernardo Ferreira, Eduardo , Jamil Farkatt, Zé Machado and Nina Costa.

The support of friends who are far away cannot be forgotten. A special thank you to Bruno Andrade, Allan Prudlik and Felipe Calazans for closely following the development of this thesis, and my whole academic life. I am grateful to Carlos José Neto, Clarissa Taques, Daniel Victor, Daniela Taques, Débora Reichert, Fernando Neu, Tatiane Spagnol, Flávia Taques and Lucas Espíndola for their friendship.

I would like to thank the kind friendship, support and motivation of Bárbara Costa, Sílvia Loureiro, Sérgio Duarte, Nabila As'ad, Ana Soares, Natália Machado, Eduardo Vasconcelos and Joana Sousa, since the time of our Master's studies. I am grateful for having met you.

During my doctoral studies, I was able to build noble friendships: Ana Rita Ferreira, André Carvalho, António Campos, Beatriz Gonçalves, Carlos Teixeira, Catarina Barbosa, Catarina Pinto, Cátia Bandeiras, Cátia Salgado, Cláudia Amorim, Fernando Caixeta, Fabian Heynmann, Flávia Sousa, Gonçalo Pereira, Guillermo Pereira, Jaqueline Viegas, José Rodrigues, Lígia Coelho, Lígia Conceição, Luís Coimbra, Mayara Moraes, Marisa Pedro, Melissa Mariana, João Pires (two of them), Melissa Mariana, Octávio Correia, Pedro Roque, Rui Coelho, Rui Pedrosa, Tiago Dionísio, Vanessa Cunha, Vanessa Tavares. I also owe a special mention to Gonçalo Pereira and Guillermo Pereira.

I am grateful for my family's love and support during these years away from home. In particular, I am grateful to my parents Maria de Lourdes and José Carlos who have always supported my academic career and the decision to study abroad. This thesis is dedicated to both of you.

Publications

Book chapters, first author

- T. Sobral, T. Galvão, J. Borges. Towards a conceptual framework for classifying visualisations of data from urban mobility services. In Lecture Notes in Business Information Processing, Vol. 201, pp. 228–242, 2015

Abstracts and presentations

- T. Sobral, T. Galvão, J. Borges. An Ontology-based Approach to Visualizing Urban Mobility Data. 7th Industrial Engineering and Management Symposium (IEMS'16), p. 65, FEUP, 2016
- T. Sobral, T. Galvão, J. Borges. An Ontology of Urban Mobility Events for Supporting Visualization Tools. 8th Industrial Engineering and Management Symposium (IEMS'17), p. 72, FEUP, 2017

Articles in international conference proceedings, first author (most recent first)

- T. Sobral, T. Galvão, J. Borges. VUMO: Towards an Ontology of Urban Mobility Events for Supporting Semi-Automatic Visualization Tools. Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC 2016), 5 pp., Rio de Janeiro, Brazil, November 2016
- T. Sobral, T. Galvão, J. Borges. Semantic Integration of Urban Mobility Data for Supporting Visualization. Proceedings of the 3rd Conference on Sustainable Urban Mobility (CSUM 2016), 8 pp., Volos, Greece, May 2016
- T. Sobral, T. Fontes, V. Costa, T. Galvão, J. Borges. OBAVUM: An Ontology-based Approach to Visualizing Urban Mobility Data. Proceedings of the IEEE International Conference on Big Data Analysis (ICBDA 2016), 5 pp., Hangzhou, China, March 2016

Article in international conference proceedings, co-author

- J. Hora, T. Galvão, A. Camanho, T. Sobral. Estimation of Origin-Destination matrices under Automatic Fare Collection: the case study of Porto transportation system. Proceedings of the EURO Working Group on Transportation Meeting. Transportation Research Procedia, Vol. 27, pp. 664–671, Budapest, Hungary, September 2017

Contents

List of Figures

List of Tables

List of Acronyms

AFC	Automated Fare Collection
ANPR	Automatic Number Plate Recognition
APC	Automated Passenger Counting
AVL	Automatic Vehicle Location
CSV	Comma-separated Values
DC	`DataConcept` (VUMO Superclass)
DSS	Decision Support System
DUC	`DomainUserConcept` (VUMO Superclass)
EDA	Exploratory Data Analysis
FEUP	Faculty of Engineering of the University of Porto
GIS	Geographic Information System
GPS	Global Positioning System
GTFS	General Transit Feed Specification
ITS	Intelligent Transportation Systems
MBTA	Massachusetts Bay Transportation Authority
MIT	Massachusetts Institute of Technology
OD	Origin-destination
OPT	Optimização e Planeamento de Transportes, S.A.
OWL	Web Ontology Language
RDF	Resource Description Framework
RDF-S	RDF Schema
RIF	Rule Interchange Format
SIRI	Service Interface for Real Time Information
SKOS	Simple Knowledge Organization System
SPARQL	SPARQL Protocol and RDF Query Language
SPIN	SPARQL Inference Notation
SQL	Structured Query Language
STCP	Sociedade de Transportes Coletivos do Porto, S.A.
SUMVis	Semanic Urban Mobility Visualization
SWRL	Semantic Web Rule Language
TIP	Transportes Intermodais do Porto
UCD	User-centered Design
UMC	`UrbanMobilityConcept` (VUMO Superclass)
VC	`VisualizationConcept` (VUMO Superclass)
VUMO	Visualization-oriented Urban Mobility Ontology
W3C	World Wide Web Consortium
WKT	Well-known Type
XML	Extensible Markup language
XSD	XML Schema Definition

Part I

Thesis Overview

Chapter 1

Introduction

This thesis studies the problem of visualizing heterogeneous multi-source urban mobility data, to support the development of semantically-rich, knowledge-assisted visualization tools. Those tools can foster the application of Information Visualization to practical situations, as they provide support for the integration of data from multiple sources, and consider the importance of human factors throughout the visualization process.

Visualizing datasets is still a technical task, despite the existence of tools that strive to simplify it. We assume the problem to be twofold. Firstly, heterogeneous data should be interoperable prior to visualization, i.e. the semantics of attributes and their values should be unambiguous across datasets. Secondly, domain users may lack the required visualization knowledge for executing tasks like choosing the right visual encodings. Scientific literature shows evidence of little involvement of domain users on the process of design, development and evaluation of visualization techniques. In this thesis, domain users are transportation stakeholders who are involved with urban mobility issues, e.g. decision makers, researchers, analysts, community advocates, among others. To the best of our knowledge, the second part of the problem remains unexplored in the Transportation literature. In parallel, some studies propose solutions that have been tested with generic domain data, e.g. movies and artists databases. The first part of the problem has been partially addressed in previous studies, although modeling spatiotemporal urban mobility data still requires theoretical and practical advances. Moreover, no studies seem to address the modeling of spatiotemporal urban mobility data for visualization.

To solve the problem, we propose a foundational ontology-based approach for integration of heterogeneous spatiotemporal mobility data, often regarded as *data* hereinafter, and annotation of visualization knowledge, to support the development of semantically rich, knowledge-assisted visualization systems. The principal idea consists of building a knowledge representation model - an ontology - which semantically describes different types of mobility data and interrelates its structure to visualization concepts. To account for human factors, the ontology allows the annotation of empirical knowledge about visualization techniques and domain users, such as their profile description and feedback about visualization techniques. For instance, those may consist of ratings or qualitative tags. In brief, the framework also supports user-oriented methodologies for developing visualizations by providing the necessary semantics to annotate the artifacts that yield from those methodologies.

As a practical implication, visualization tools can exploit the knowledge described in terms of the ontology elements to assist domain users in finding appropriate visualizations for their data and the questions they want to ask about data (analytical tasks), e.g., through semi-automatic recommendation. Moreover, researchers and practitioners can share and reuse empirical knowledge across systems to better understand which techniques meet the needs of domain users, provided that the knowledge has been described with the proposed ontology.

The remainder of this chapter is structured as follows: the underlying research context is presented in Section 1.1. In Section 1.2, we define the problem statement based on the research gaps discussed in Section 1.1. In Section 1.3, we state our research objectives and questions. 1.4 explains the theoretical and practical contributions of this thesis. Section 1.5 describes the methodology we carried throughout our study. Finally, Section 1.6 provides the outline for the following chapters.

1.1 Research Context

Mobility dynamics are nontrivial to human perception. The proportion of people living in urban areas may rise to figures up to 66% of the world population by 2050 [1]. Intelligent Transportation Systems (ITS) generate an enormous volume of data which can be analyzed to provide better understanding of mobility phenomena, and to devise actions to improve transportation systems. Data, however, is not always analyzed in depth, which may suggest a dichotomy between information abundance and knowledge starvation.

Visualization is a powerful means for extracting knowledge from datasets. Early applications of visualization to Transportation were based on Geographic Information Systems (GIS), as discussed in Chapter 3. Recently, visualization tools and frameworks allow to create novel, GIS-independent tools. Figure 1.1 exemplifies the potential of some of those frameworks. It shows commuting patterns based on the Hubway bike sharing system of Boston, U.S.A., and travel time cost in comparison with an equivalent travel by public transportation system (PTS). A handful of innovative visualization techniques do not only appear in Transportation literature, but in public data visualization challenges hosted by institutions such as universities or companies. For instance, the technique shown in Figure 1.1 was developed for the Hubway Data Challenge 2012 [2]. Hubway is a shared bicycle service available to citizens in Boston, Massachusetts. The technique allows the user to explore time savings of people who commute by bicycle instead of public transportation, i.e. bus and subway. After the user selects its origin (a Hubway bicycle station), the visualization technique provides information about the total number of trips made by citizens for every origin-destination pair, as well as the comparison between the time it would take if the same trip was made using public transportation. Destinations are represented as radial arcs.

Domain users usually possess data in various file formats and schemas. Such data may not be provided with metadata, i.e. data about data. Metadata can provide semantics to datasets by aligning their attributes and values to a pre-defined knowledge representation model. Our experience with domain users during previous studies showed a gap regarding the actual use of visualization techniques in practical contexts [3, 4]. We argue that data heterogeneity is a factor that contributes to this gap. Scientific literature also shows evidence of lack of close contact between visualization experts and domain users, which can negatively impact the utility of the proposed visualization techniques. Some standards for urban mobility data have been proposed to facilitate interoperability across systems. For instance, in the context of public transportation, the GTFS standard allows agencies to publish information about their services, e.g. lines, schedules and fares [5]. The SIRI standard is a protocol for exchanging real time information about public transportation systems and vehicles [6]. Nonetheless, those standards can be better defined as schemas, as they do not provide semantics to data.

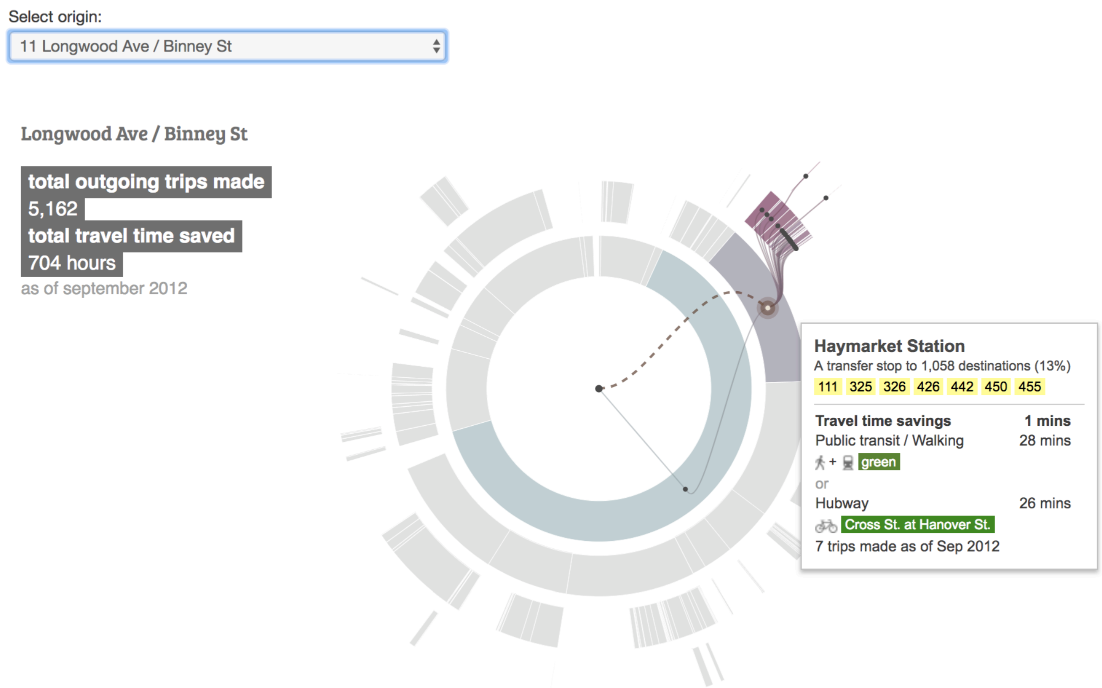


Figure 1.1: Commuting patterns within the Hubway bike sharing system. The visualization proposed by [2] allows one to explore the number of outgoing trips, and the reduced commuting time in comparison to public transportation alternatives

The visualization pipeline is usually organized in three phases: (1) raw data preprocessing, (2) visual transformation and (3) visual mapping [7]. In (1), a tool ingests raw data and transforms it into a format supported by a visualization technique. In (2), instance data is transformed to meet the structural requirement of a visualization technique. Finally, phase (3) is responsible for mapping instance data onto the visual variables that form a visualization, e.g. shape, size, texture. We discuss the theory of Information Visualization and the role of visual variables on Chapter 2.

Scientific literature in Transportation contains various examples of visualization techniques to support urban mobility analysis. Frequently, those techniques are meant to support particular datasets related to a certain theme. Further work is still required on the issue of visualizing data from multiple heterogeneous sources.

The motivation for our research also stems from the perceived difficulty that domain users face regarding visualization knowledge. This work is a natural sequel to the author's Master dissertation [3], which carried an exploratory study for the development of user-centered visualization techniques for mobility data. The results showed that the involvement of domain suers during the design and development of visualization techniques could increase their effectiveness. Moreover, we argue that the choice of appropriate visualization techniques can be influenced by data characteristics, the profile of domain users, the context of analysis, e.g. strategic, operational or tactical, and the questions they have about data [4]. In that study, we defined that further work should consider a formalization of such knowledge, due to its subjective nature, so that visualization tools could exploit it to improve their effectiveness and reduce the technical burden associated to the process of visualizing data.

1.2 Problem Statement

Section 1.1 introduced the following issues:

I1. Data heterogeneity due to the plurality of ITS specifications;

I2. Visualization techniques may have limited data interoperability, as they depend on specific data structures which are often unprovided with semantics;

I3. Infrequent involvement of domain experts on the design, development and evaluation of visualization techniques. Moreover, choosing appropriate visualization techniques is non-trivial, and domain experts may lack the necessary visualization knowledge.

We consider that I1 is unlikely to be mitigated, despite all the efforts in that direction. The introduction of any standard is constrained by temporal, technical and budgetary factors. Finally, it may be impractical to propose unified standards for publishing spatiotemporal data, as transportation systems span through diverse contexts and should satisfy distinct stakeholders and their complex requirements. Currently, there are various standards for transportation data, e.g., SIRI [6] and GTFS [5], each with their own scope and characteristics.

I2 highlights a research gap on regards to visualization studies in Transportation. As visualization techniques usually depend on specific schemas and formats, it becomes technically demanding to visualize data from multiple sources without a common representation model. Visualization tools such as Tableau [1] allow users to explore heterogeneous data simultaneously, although the user is limited to the available visualization techniques.

I3 hinders the spread of use of visualization techniques in practice. In fact, visualization knowledge is required so that experts can choose appropriate visual encodings. In the long run, visualization techniques should strive for reducing the technical burden, to allow experts to focus on data exploration.

The aforementioned issues allow us to raise the following questions:

- How to visualize spatiotemporal data from heterogeneous data sources?
- How to formalize the representation of empirical knowledge retrieved from domain users during user-oriented processes for the development of visualization techniques?
- How such formalization could be made reusable by other researchers and practitioners, so that it could be applied and modified to meet the requirements of other contexts?

1.3 Research Objectives and Questions

This thesis aims the following objectives by answering their respective research questions:

O1. To develop a visualization-oriented domain ontology to integrate spatiotemporal mobility data and support user-oriented methodologies for developing visualization techniques.

  Q1. How to conceptualize the fundamental structure of spatiotemporal urban mobility data, visualization techniques and empirical knowledge?

[1] https://www.tableau.com/

Q2. Which concepts should the ontology cover?
The definition of the ontology components should be built upon concepts found in the main types of spatiotemporal mobility data, and Information Visualization theory. The outcome of this research question should be an ontology that can be reused, altered and extended to meet distinct contexts and requirements.

O2. To propose an ontology-based foundational framework for supporting semantically rich, knowledge-assisted visualization systems.

Q3. How to use the ontology comprehensively on the development of visualization tools for spatiotemporal urban mobility data?
The theoretical basis for answering this question consists of identifying and understanding the different stages that data should pass, from integration to visualization, and considering the evaluation of visualization techniques with domain users. The outcome of this research question should be a novel approach to spatiotemporal urban mobility data visualization, which comprises the development of visualizations that support semantic data, and meet the requirements and preferences of domain users.

O3. To evaluate the ontology in practical contexts for validating O1 and O2.

Q4. Does our approach succeed in tackling issues I2 and I3?
The assessment and validation of our findings should be performed by evaluating functional visualization prototypes with domain users and data from PTS.

1.4 Research Contributions

This thesis provides the following contributions:

1.4.1 Theory

- An updated literature review (see Chapter 3) on the applications of Information Visualization and Ontologies to urban mobility analysis;
- A conceptual model that formalizes the structure of spatiotemporal urban mobility data, visualization techniques and empirical knowledge, built upon acknowledged frameworks for modeling general spatiotemporal data and Information Visualization theory;
- An ontology-based foundational framework to support the development of semantically rich, knowledge-assisted visualization systems.

1.4.2 Practice

The main practical contribution is VUMO, an acronym for Visualization-oriented Urban Mobility Ontology. The ontology was built in Web Ontology Language (OWL), a Semantic Web standard which has been applied to various domains outside the scope of the World Wide Web. VUMO also reuses existing ontologies, e.g. Geo [8], to express spatial information, and GTFS[2], which translates the GTFS schema into its semantic counterpart.

[2]The GTFS ontology is not officially provided by Google, but it is a community-driven effort based on the actual GTFS standard [5]

We also adopted the SPARQL Inference Notation (SPIN) notation/rule language to express the logical reasoning rules that infer implicit visualization knowledge from data, and support recommendation algorithms for visualization techniques. To the best of our knowledge, this thesis is the first study that seem to apply rule-based inference, particularly SPIN, to the development of visualization tools. The adopted technological stack is freely available, with no commercial restrictions.

1.5 Methodology

This research followed a mixed approach, due to its multidisciplinary nature that involves intersecting areas such as Computer Science and Software Engineering, Mathematics and Human-Computer Interaction. To answer Q1 and Q2, the conceptual model was built in accordance to existing methodologies for ontology development, and Information Visualization theory. To answer Q3, we recall and extend our previous work in [3, 4], according to Human-Computer Interaction principles. The same applies to Q4, as we approached domain experts through qualitative interviews and usability tests to evaluate the effectiveness of our framework.

1.6 Outline

The remainder of this thesis is organized as follows:

- Chapter 2 describes the theoretical background on Information Visualization, Semantic Web and ontologies required for this work. A link is established between Information Visualization and Semantic Web technologies by introducing the concept of Knowledge-assisted Visualization and related studies.

- Chapter 3 presents the applications of Information Visualization and Ontologies to urban mobility research. We identified a research opportunity due to the lack of knowledge-assisted approaches to Visualization in the context of transportation research, in particular to urban mobility studies. We also identified that the involvement of domain users is still small.

- Chapter 4 introduces two fundamental contributions of this thesis, answering to research questions Q1, Q2 and Q3. Firstly, a conceptual model to formalize spatiotemporal mobility data, visualization techniques, and empirical knowledge. We define a relation between data and visual features, and between empirical knowledge, data, and visualization techniques. The conceptual model answers to research question Q1. Secondly, the chapter introduces the VUMO ontology specification, which implements the aforementioned conceptual model. This chapter presents several examples to assist the reader on understanding the use of ontology components. It is also described how VUMO can support the user-centered design process of visualization techniques. This chapter also answers to Q2 and Q3.

- Chapter 5 describes a multi-case study using the PTS of Porto and Boston. The case study describes the *SUMVis* functional prototype, which is based on the VUMO ontology. This chapter answers to Q4.

- Chapter 6 concludes this thesis by revisiting the research questions and discussing important remarks. We also provide further research directions.

Figure 1.2 illustrates the thesis outline for Parts II and III, the research questions and where they are answered, and a broader depiction of the interrelations between chapters.

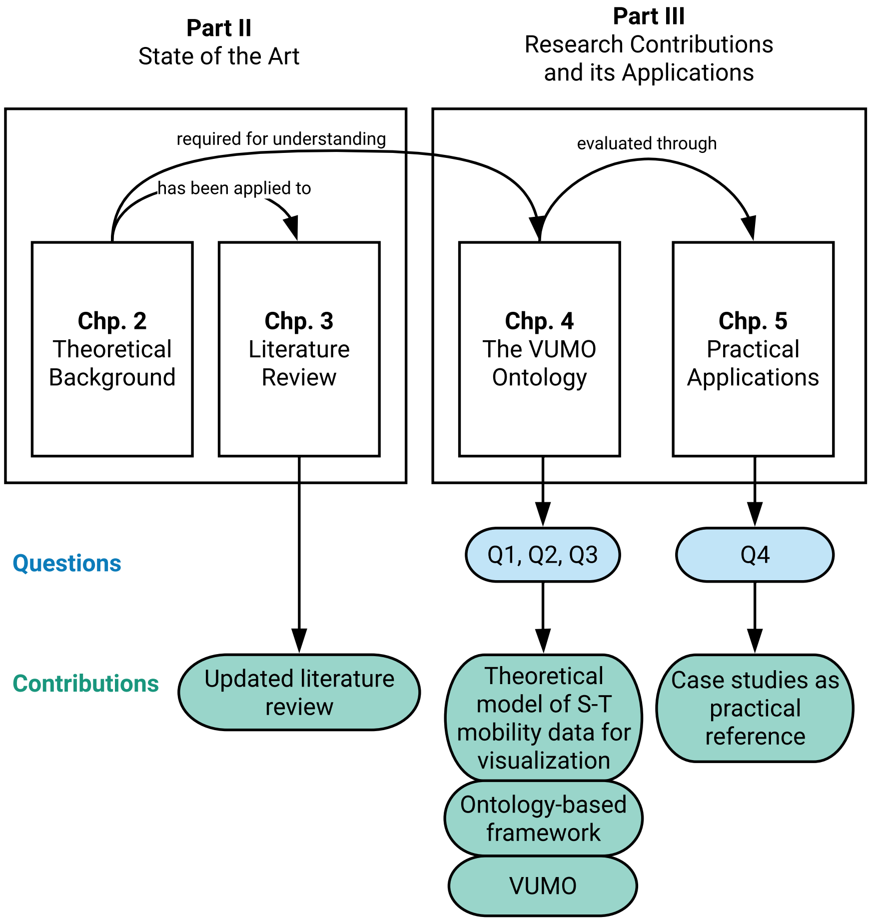


Figure 1.2: Illustration of the thesis outline for Parts II and III, emphasizing the relationship between chapters, and their respective contributions and answers to research questions

Part II

Background Theory and Literature Review

Chapter 2

Background Theory

This chapter introduces the theoretical background required for this research. In Section 2.1, we provide the main definitions of Information Visualization, and the theoretical models that were proposed to support the development of visualizations. The role of interactive visualizations and the involvement of end users on the process of visualization are also discussed. In Section 2.2, we introduce the concept of Semantic Web, its technologies, and ontologies. We present several perspectives for ontology classification, and explain the benefits of ontologies for information integration. Finally, we establish a link between Information Visualization and Semantic Web technologies by introducing the concept of Knowledge-assisted Visualization and existing works on this topic, which intends to overcome the inherent difficulties of the visualization process by providing means of automation on tasks such as visual mapping and recommendation of visualization techniques.

2.1 Information Visualization

Information Visualization consists of a multi-disciplinary area that intersects fields such as Computer Science, Human-Computer Interaction, Design and Cognitive Psychology. The maturation of the Information age motivates the need of visually representing data from large datasets to support knowledge extraction. In particular, computer-aided visualization has a crucial role in human cognitive systems, especially for the fact that visual displays provide the highest bandwidth channel from computers to humans [9].

Much earlier than the era of computer-aided visualizations, the work of William Playfair is considered the pioneer in the field of statistical graphics, for having invented the traditional line, bar, area and pie charts [10]. Figure 2.1 provides a well-known example from his book, The Commercial and Political Atlas, published in 1786, in which England's exports and imports from and to Denmark and Norway are compared.

Other examples of breakthrough works include the rose diagrams from Florence Nightingale [11], and the map of Napoleon's Russian campaign by Charles Minard [12]. Nightingale intended to visualize the number of death tolls in hospitals due to the lack of sanitation during the Crimean War, as shown in Figure 2.2, to fight for better conditions in hospitals. The work of Minard, shown in Figure 2.3, presents a map that describes the loss of the French army over the advance on Moscow, and its withdrawal.

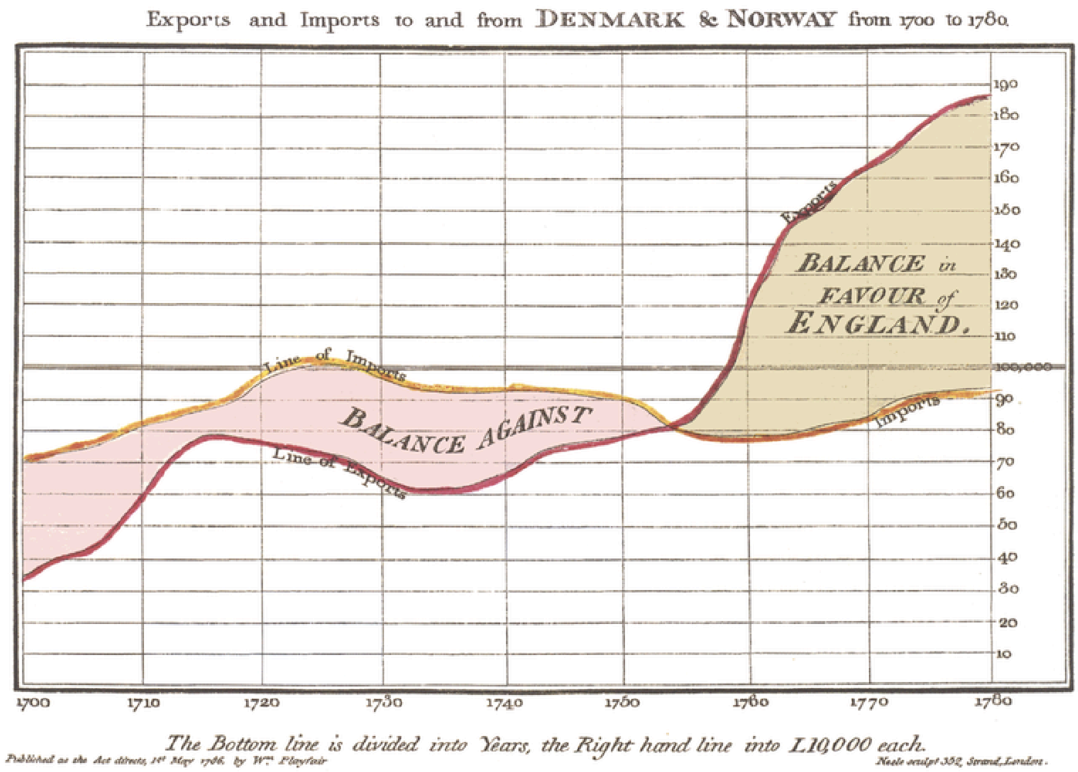


Figure 2.1: A visualization by Playfair which depicts England's exports and imports from Denmark and Norway [10]

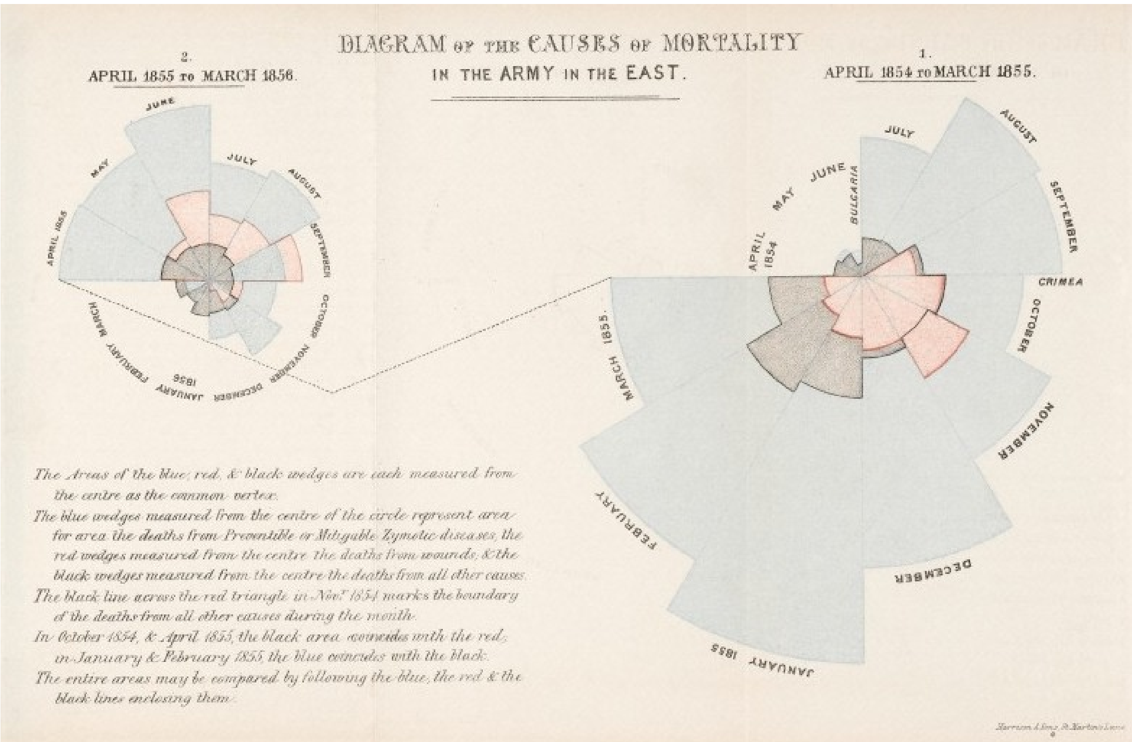


Figure 2.2: A visualization by Nightingale which depicts the number of death tools in hospitals during the Crimean War [11]

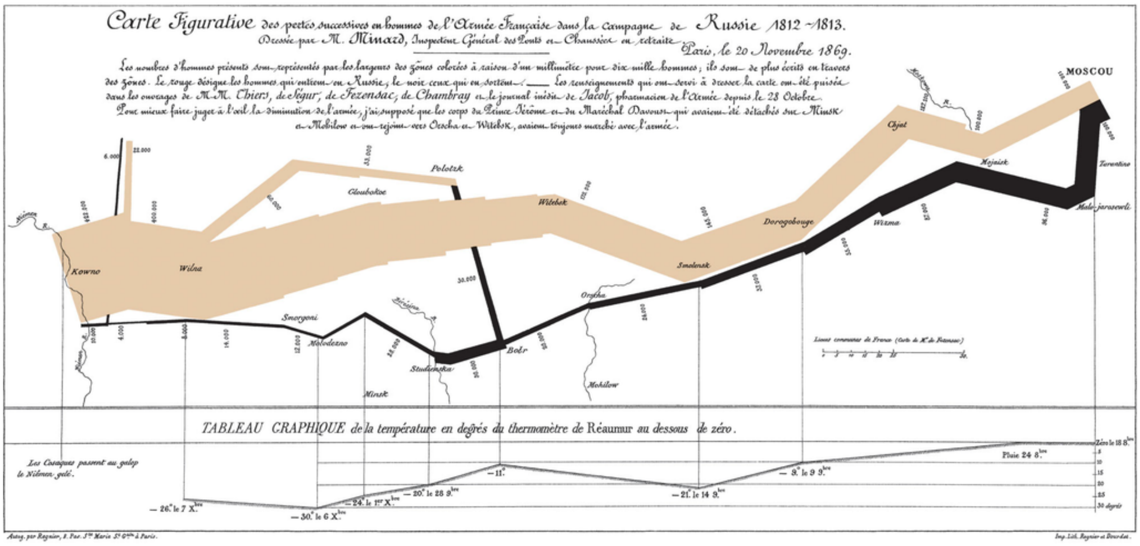


Figure 2.3: A visualization by Minard which depicts the loss of the French army over the advance on Moscow [12], and its withdrawal

2.1.1 Definition

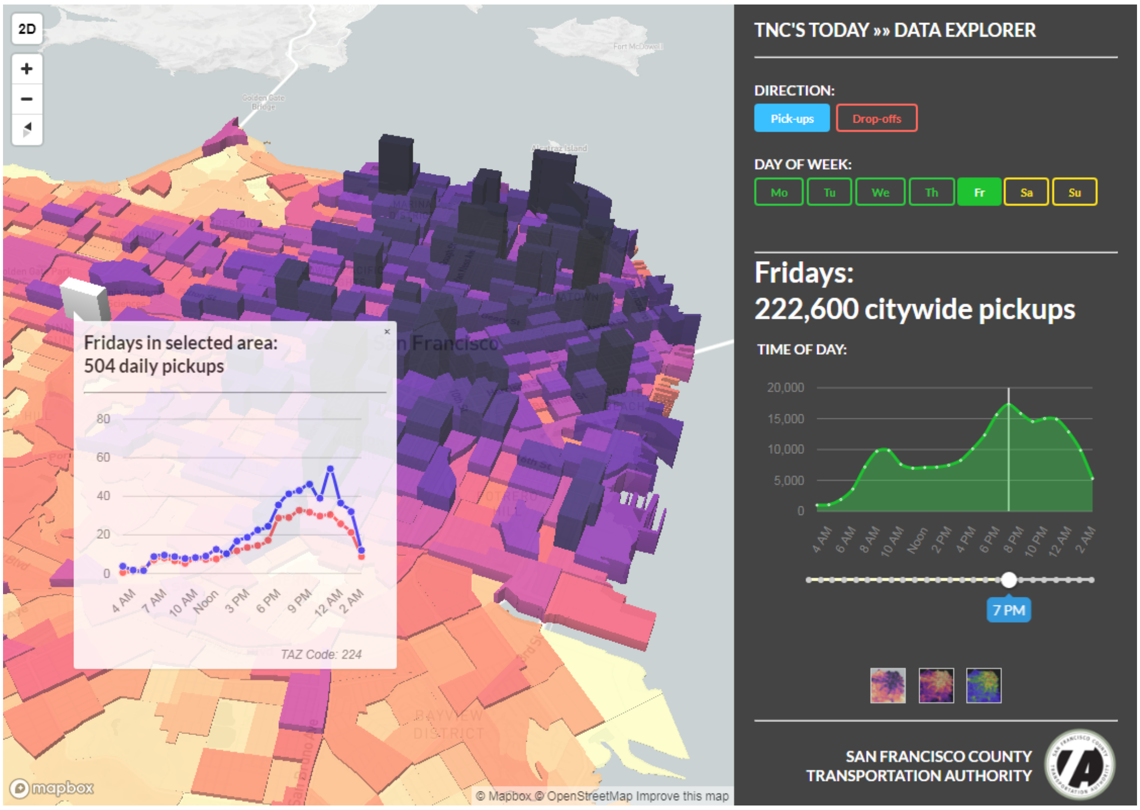
Card et al. define Information Visualization as *the use of computer-supported interactive visual representations of abstract data to amplify cognition* [13]. An alternative definition by Spence states that Information Visualization is the formation of a mental model or mental image of data [14]. The widespread adoption of visualization confirms that visualization techniques can help informing, improving analysis and decision-making. Ware et al. emphasize that Information Visualization does not rely solely on computer graphics; the connection to data is also crucial [9].

Information Visualization is one of the major areas of Visualization besides Scientific Visualization and Geo-visualization. Although that nomenclature is generally acknowledged, we argue that it lacks precision. One can identify substantial overlap between those areas. Information Visualization generally concerns the representation of abstract data like prices or social habits. Scientific Visualization concerns scientific data that may include a spatial component, such as numerical simulations (see Figure 2.4). Geo-visualization is considered to be similar to Scientific Visualization, differing by having maps as the main element onto which data is displayed. We naturally admit the overlap (or a better term: synergy) between Information Visualization and Geo-visualization, given that abstract information about urban mobility is embedded into a spatiotemporal context but it is not necessarily restricted to geographic representations.

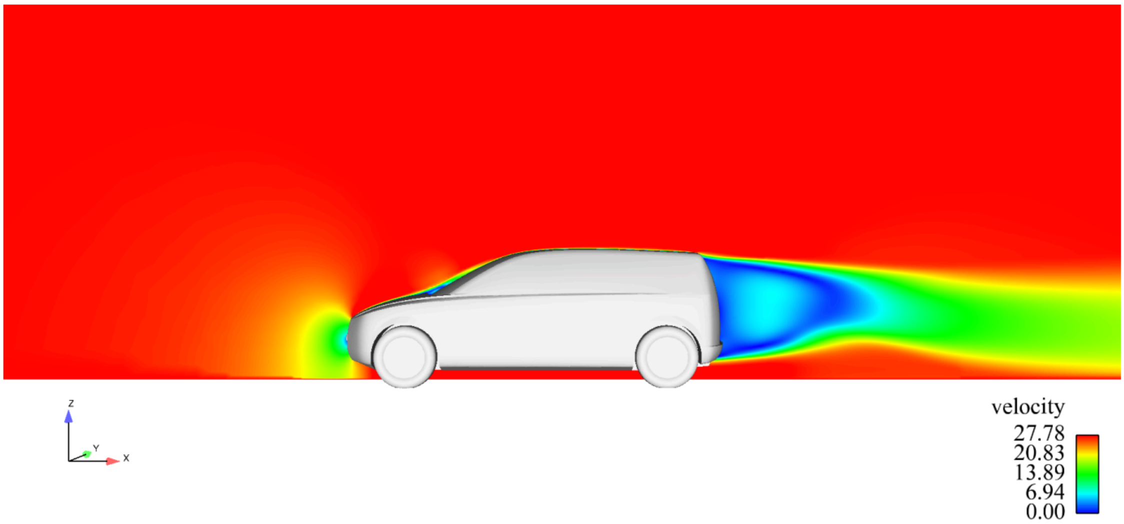
One can often identify misconceptions on regards to overlapping the definitions of Information Visualization and Visualization tools. The latter can be defined as computer-based systems which are designed to display encoded data with a view in order to support the visualization process [14].

In order to effectively map abstract data onto visual tokens, Card et al. proposed a human-centric reference model (see Fig. 2.5) [13], based on the seminal work of Jacques Bertin [17] on the Semiology of Graphics.

The human-centric reference model is composed by stages through which data should pass. *Raw data* comprises data in any format. *Data Transformations* transform raw data into *Data Tables*, which contain relational descriptions of data and are enriched with metadata, i.e. information about data. Data tables describe data with respect to certain variable types. Card et al. proposed



(a) Example of Information Visualization tool for exploration of Uber and Lyft usage in San Francisco



(b) Example of Scientific Visualization of velocity flows on a vehicle through computational fluid dynamics

Figure 2.4: An example of information visualization (a) and scientific visualization (b) (extracted from [15] and [16], respectively)

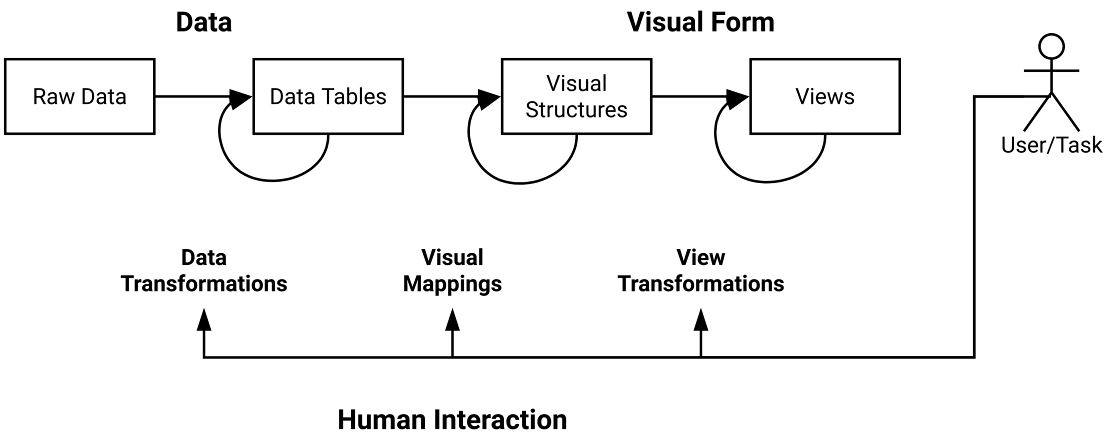


Figure 2.5: Human-centric visualization reference model (adapted from [13])

a broad classification based on three basic types of variables: nominal, ordinal, and quantitative [13]. Nominal variables are unordered sets, in the sense that a value can only be equal or different to other values. Ordinal values are subject to an order relation. Quantitative variables are defined within pre-defined numeric ranges and can be used for numerical calculations.

Visual Mappings transform data tables into visual variables - a concept coined by Bertin and extended by other authors (including Card et al.) - to account for the computational perspective of visualization, not yet available at Bertin's time, and further theoretical advances in the field of Visualization. Figure 2.6 illustrates the original visual variables proposed by Bertin [17].

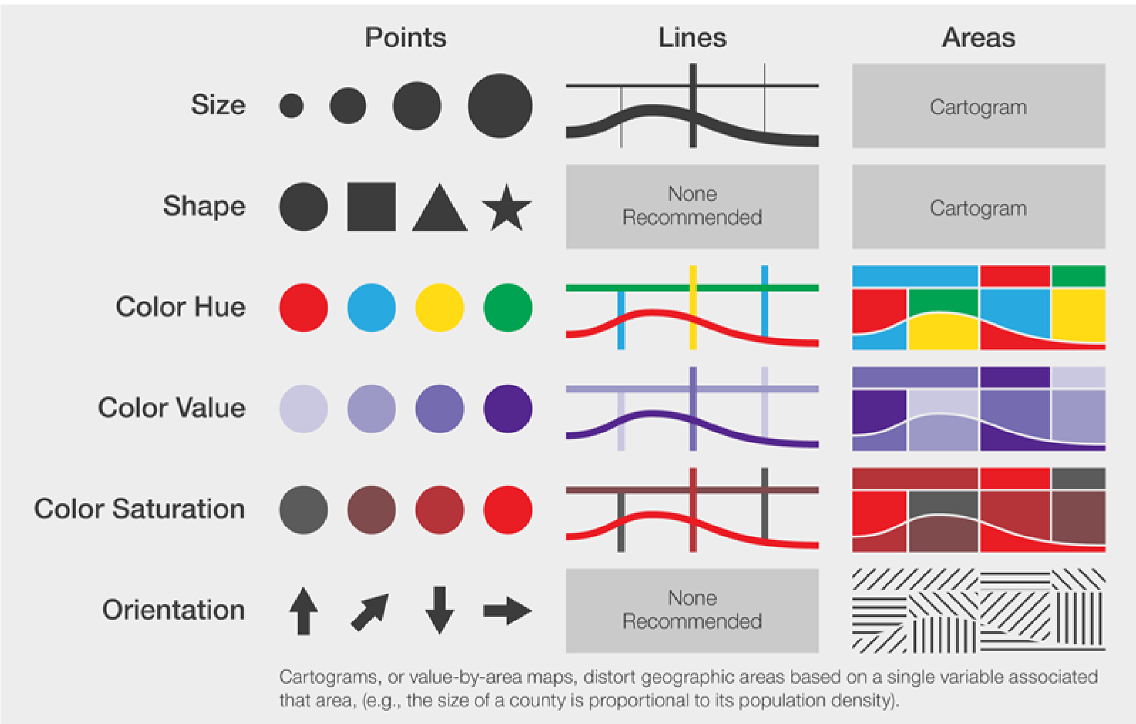


Figure 2.6: Visual variables proposed by Bertin (adapted from [17])

View transformations are responsible for the interaction layer between the human and the visualization environment. They transform static graphical representations in order to create new, distinct views of the visual structures. This stage completes the interaction loop between the human and visualization.

Visualization techniques are an essential component of Exploratory Data Analysis (EDA). We share the view of EDA proposed by Andrienko and Andrienko [18]: the exploratory process begins with the analyst motivated to investigate a certain subject, which may consist of a series of concrete questions. New questions may arise throughout the exploratory process.

Since last decade, the field of Visual Analytics (VA) was branched from Information Visualization. The distinction between them lies on the scope. The former benefits from the advances of the latter to support the process of analytical reasoning, and addresses analytical methods for knowledge extraction. The same authors outline the main areas of VA:

- *Analytical reasoning techniques*: to provide insights about data and to support decision making;
- *Visual representations and interaction tasks*: to enable visual means to interactively explore and understand data;
- *Data representations and transformations*: to convert data of multiple types to support analytical reasoning and visualization.

This work focuses on the second and third areas, although we argue that our contributions can motivate and support new developments of the first area.

2.1.2 Interaction in Visualization Tools

Technology has made responsive interaction possible within the context of visualization. Visualization tools now benefit from the capability of facilitating users to change their view about data. This view change can be defined as movement within the information space [14]. Rather than static visual representations of information, visualization tools allow users to interact with it through navigation and have immediate feedback.

To serve as a foundation of the Task by Data Type Taxonomy of Information Visualization, Shneiderman suggested a useful, widely acknowledged starting point for designing visualizations named "*visual information seeking mantra*": "Overview first, zoom and filter, then details-on-demand" [19]:

- *Overview*: gain overview about the object in analysis;
- *Zoom*: zoom in on objects that are of interest;
- *Filter*: filter unwanted objects;
- *Details-on-demand*: select an object (or a group of) and get further details.

Figure 2.7 shows an example of a highly responsive visualization tool that provides an initial overview of the density of immigration settlement of different foreign-groups across the United States. It allows further exploration by filtering for specific nationalities (Mexico, for instance), skimming through a timeline or zooming for details, in accordance to what has been proposed by Shneiderman.

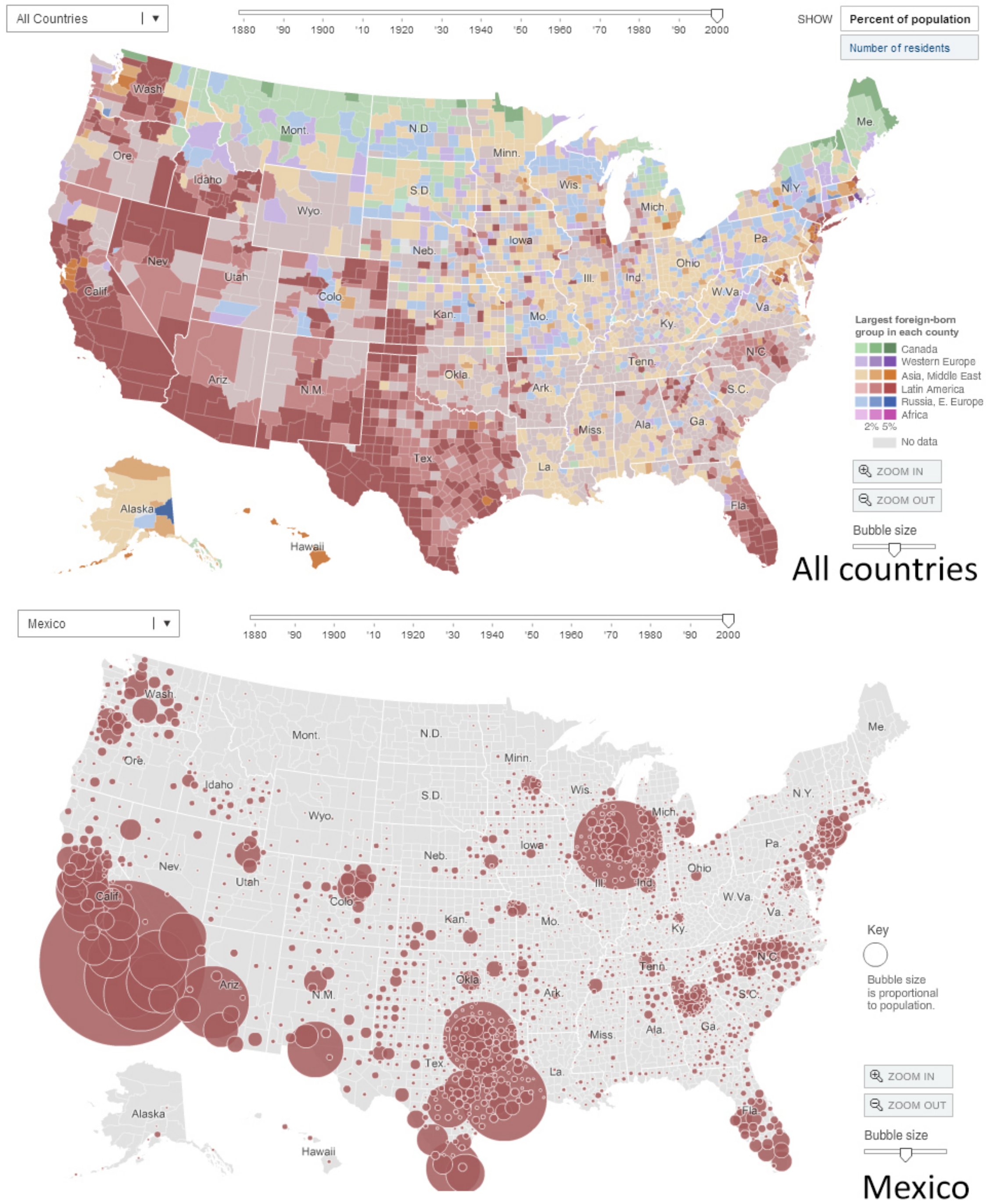


Figure 2.7: Immigration settlement across the USA. Overview of immigration by different nationalities is shown on the top. Filtering by Mexican citizens immediately provides the visualization on the bottom (Extracted from [20])

Such interactions on the visual environment are sequential and expected to be iterative. An enhancement to this mantra was proposed by Cockburn et al. to categorize visualization techniques [21]. The authors defined ”overview plus context” as Spatial Separation between focused information entities and contextual information. ”Zoom” was reduced to temporal separation, and ”Focus plus context” minimizes the seam within views.

Hearst extended the classification for visual interaction tasks based on users' interactions context [22]:

Brushing and Linking provide visual cues that highlight different visual representations of data on multiple canvas of the same visualization interface. It is useful to provide various perspectives for the same data instance. Examples of cues include change in color, size or position of the objects. Figure 2.8 provides an example of a visualization technique in which the user is able to select a region (red rectangle) within one of the cells of a scatterplot matrix. All the points within this region are highlighted on the remaining scatterplots cells.

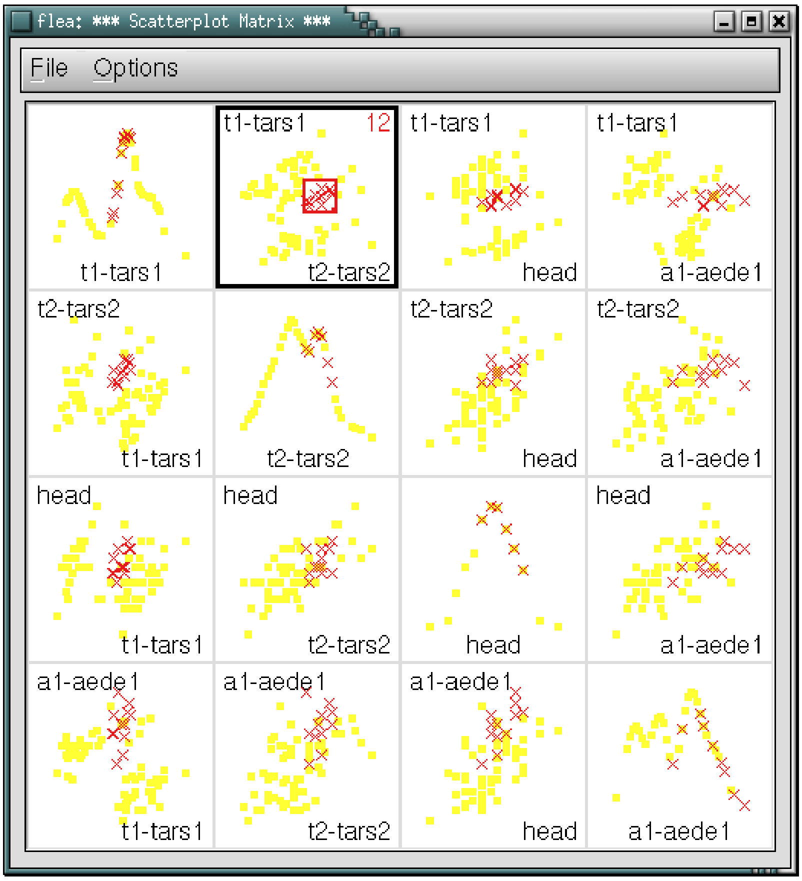


Figure 2.8: Brushing and Linking interaction task in a scatterplot matrix visualization. As the user selects a region within a scatterplot, all points within this region are highlighted in all scatterplots [23]

Panning and Zooming are equivalent to a change of viewpoint by a camera and a further zoom [13], allowing users to move the screen and zoom in an area of interest.

Focus plus Context tackle the inherent loss of information about the surroundings of the zoomed area. As the zoom level increases in a particular area of interest, overall structure is lost. An example of focus plus context is the Fish Eye view shown in Figure 2.9.

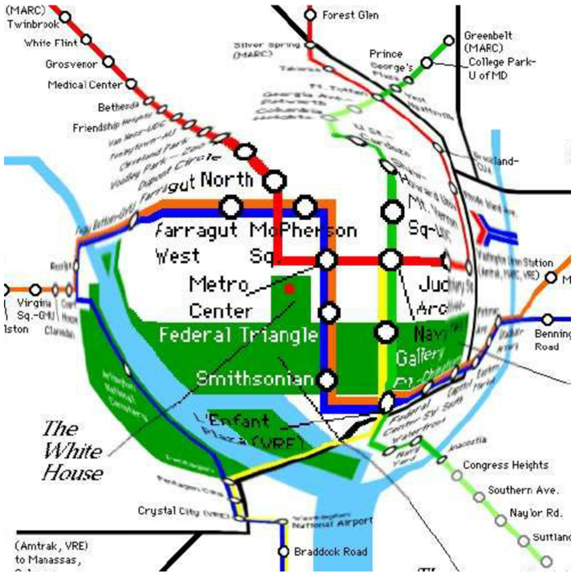


Figure 2.9: Fish Eye view with focus on the White House. Information about the context of the public transportation network is still preserved [24]

Semantic Zooming, in contrast to zooming, unveils more granular information about the target object or area, to reveal its context and meaning [25].

Animation is a consequence of one or more visual interactions, and do not provide manipulation functionalities [26]. It is suggested that animation aids improvement of interaction and understanding.

Overview plus Detail is used to display different levels of detail in two or more linked visualizations.

Dynamic Queries allow users to formulate and query data through visual interactions, reducing data overload. Examples of such mechanisms include temporal or attribute filters.

Direct Manipulation allows for a more natural user experience, as it consists of providing graphical metaphors on the user interface, in contrast to entering commands to interact with the visualization.

Interaction tasks and visualization tasks are terms that may be used interchangeably, which can lead to conceptual inconsistencies. Here we introduce some taxonomies found in literature for classifying visualization tasks.

The work of Shneiderman uses the already mentioned *Visual Information Seeking Mantra* to define the *Task by Data Type Taxonomy*, which includes three additional tasks [19]:

- *Relate*: view relationships between objects;
- *History*: keep track of actions to support undo, replay and progressive refinement;
- *Extract*: allow extraction of subgroups of objects and of query parameters.

Keller and Keller proposed a user-centered approach for classifying tasks [27]. The approach is divided into nine task categories, and only considers analytical aspects for interacting with visualizations, in the sense that broader tasks, such as *relate* or *extract*, are not considered. Table 2.1 presents such classification.

Yi et al. proposed a comprehensive classification of tasks based on user goals and interactions, to achieve seven interaction categories, which are presented in Table 2.2 [28].

Table 2.1: Taxonomy for Task Classification as proposed by Keller and Keller [27]

Task	Description
Identify	Recognition of objects based on the presented characteristics
Locate	Identification of an object's possition
Distinguish	Determination of differences between objects
Categorize	Classification of objects into various categories
Cluster	Grouping similar objects, given a set of criteria
Rank	Ordering objects according to a given parameter, e.g. relevance
Compare	Examination of objects
Associate	Drawing relationships between objects
Correlate	Finding correlations or cause-effect relationships between objects.

Table 2.2: Visual Task categorization as proposed by Yi et al. [28], adapted from Nazemi [29]

Category	Description
Select	Mark something as interesting to enable the following of the object
Explore	Show something else e.g., different subsets of data
Reconfigure	Provide a different view or arrangement of the underlying data
Encode	Provide a different fundamental view by selecting another visualization technique
Abstract/elaborate	Provide a different level of detail on the data e.g., by details-on-demand techniques
Filter	Provide a view with certain (predefined) criteria
Connect	Provide visual connection between the same objects on different views

2.1.3 Data Types and Classifications

This section introduces the aspects of data that are important for visualization. A number of classifications have been proposed. Data classifications can be abstracted to three perspectives: *level of measurement, data transformations,* and *data dimensions*.

Card et al. proposed a classification based on data values and their ordering capabilities, as already mentioned in Section 2.1.1. Data values can be nominal, ordinal and quantitative [13].

Chi proposed the *Data State Reference Model* (DSRM) (see Figure 2.10) a taxonomy for classifying visualization techniques [30]. Such classification defines data transformations and types as a baseline, in the sense it can be used for classifying data. The DSRM consists of three types for data transformation, four data stages and four types of operations. The initial phase is the *value* (raw data), from which *analytical abstractions* (data transformations) are applied. Transformed data contains structural information about data, i.e. metadata. In the remaining phases, *visualization transformations* and *visual mapping,* appropriate visualization techniques are chosen.

Shneiderman divided data into seven categories according to their dimension, in the scope of his *Visual Information Seeking Mantra* [19]. Such categorization was not meant to be exhaustive. In fact, Shneiderman argued that other data types may exist, and that his categorization "reflects an abstraction of reality". Table 2.3 presents the classification.

Keim et al. provided modifications to Shneiderman's classification [31]. The authors proposed that each timestamp can be assigned to multiple variables, thus being multi-dimensional. The category *Multi-dimensional* data comprises Shneiderman's Three- and multi-dimensional data; *Tree and Networks* were classified as *Hierarchies and Graphs*. Additional categories include *Text and Hypertext,* and *Algorithms and Software*. Table 2.4 presents the classification.

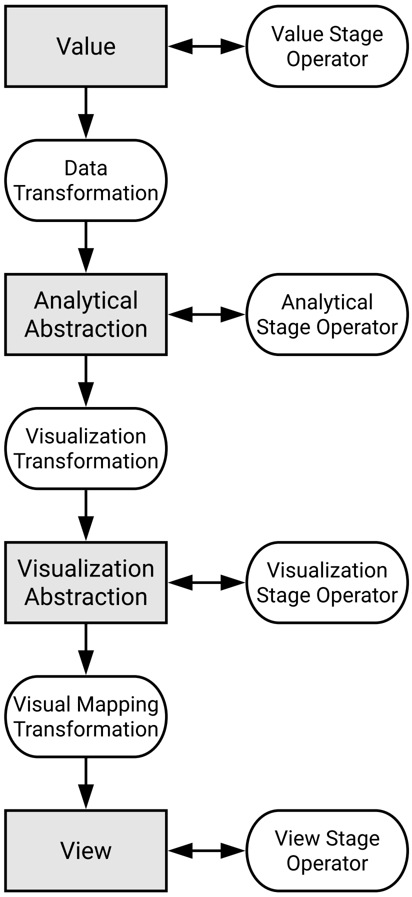


Figure 2.10: The Data State Reference Model proposed by Chi (adapted from [30])

2.1.4 Visualization of spatiotemporal data

Depending on the domain of application, visualizing spatiotemporal data can also make use of non map-based techniques, thus adopting other graphic entities such as lines, bars, circles or other geometric forms. However, when the actual spatial representation is of visual concern by the users who will benefit from those techniques, approaches take place so as to suggest ways of presenting space and time simultaneously. Such representations would always be subject to simplification depending on the presentation media, e.g. paper.

Two main visualization approaches have been applied for visualizing spatiotemporal data. The first approach appeared within the context of Time Geography coined by Hägerstraand, and consists of representing paths of moving instances through a time dimension which is perpendicular to a geographic plane, defined as a Space-Time-Cube [32]. Given that time is an intrinsic component of the visualization, it is possible to continuously track the movement of an entity through space over time. However, the amount of entities to be plotted might imply intense visual cluttering, which would require workarounds for reducing it. Hägerstraand's approach precedes the era of Geographic Information Systems, even though it is possible to identify applications of the Space-Time-Cube in other visualization-related developments (see Figure 2.11) [33].

The second approach consists of using animated maps to allow users to visualize changes in spatial data over time. At every frame of the animation, the position of an entity is redrawn. In order to avoid visual clutter, animated maps should preferably plot spatiotemporal data on a static

Table 2.3: Data type classification as proposed by Shneiderman [19]

Type	Description
One-dimensional	Linear data type
Two-dimensional	Planar or map data
Three-dimensional	Real world objects
Temporal	One-dimensional data with start and finish time
Multi-dimensional	Data in relational and statistical databases with n attributes.
Tree	Data with a link to (one) parent
Networks	Data items linked to an arbitrary number of other items

Table 2.4: Data type classification as proposed by Keim et al. [31]

Type	Description
One-dimensional	Data with one dimension, e.g. temporal data.
Two-dimensional	Data with two dimensions, e.g. spatiotemporal data.
Multi-dimensional	Data with more than three dimensions (multivariate data), e.g. relational databases.
Text and Hypertext	Data with unknown dimensions and number.
Hierarchies and Graphs	Data with relationships to other information entities.
Tree	Data with a link to (one) parent
Algorithm and Software	Program code that describes complex algorithms.

bi-dimensional map. However, this requires additional measures in order to avoid designing visualizations that force users to create mental associations between different time intervals. Instead, they should allow instant view of changes of an entity through time. Methods for displaying data within animated maps were proposed by Adrienko et al. [34].

2.1.5 Involving end-users in the development of visualizations

The involvement of users during the whole development process of visualizations is well acknowledged by researchers [35–40]. Visualizations have mostly been technology-driven, and it is now possible to identify a shift towards more user-centered approaches. These approaches differ on the way users are engaged throughout the process. These methods appear in diverse contexts such as epidemiology, hydrography and crime spotting visualizations, although they can be analyzed from a general UCD perspective for visualization.

Slocum et al. [39] proposed a UCD process consisting on a six-phase cascade for the creation of a visualization tool for water-balance issues: 1) prototyping; 2) domain expert evaluation; 3) software refinement; 4) usability expert evaluation; 5) software refinement and 6) decision maker evaluation. A noticeable disadvantage of the process is that end-users only participate in the end of the design process, after domain and usability experts have addressed the design and functionality related issues.

The method proposed by Robinson et al. extensively involve users throughout the process along six phases: 1) work domain analysis; 2) conceptual development; 3) prototyping; 4) interaction and usability studies; 5) implementation and 6) debugging [37]. Phases 2 to 5 occur multiple times in a loop. The work domain analysis phase consists of the first contact between stakeholders and developers, in which they communicate the initial ideas and requirements. The

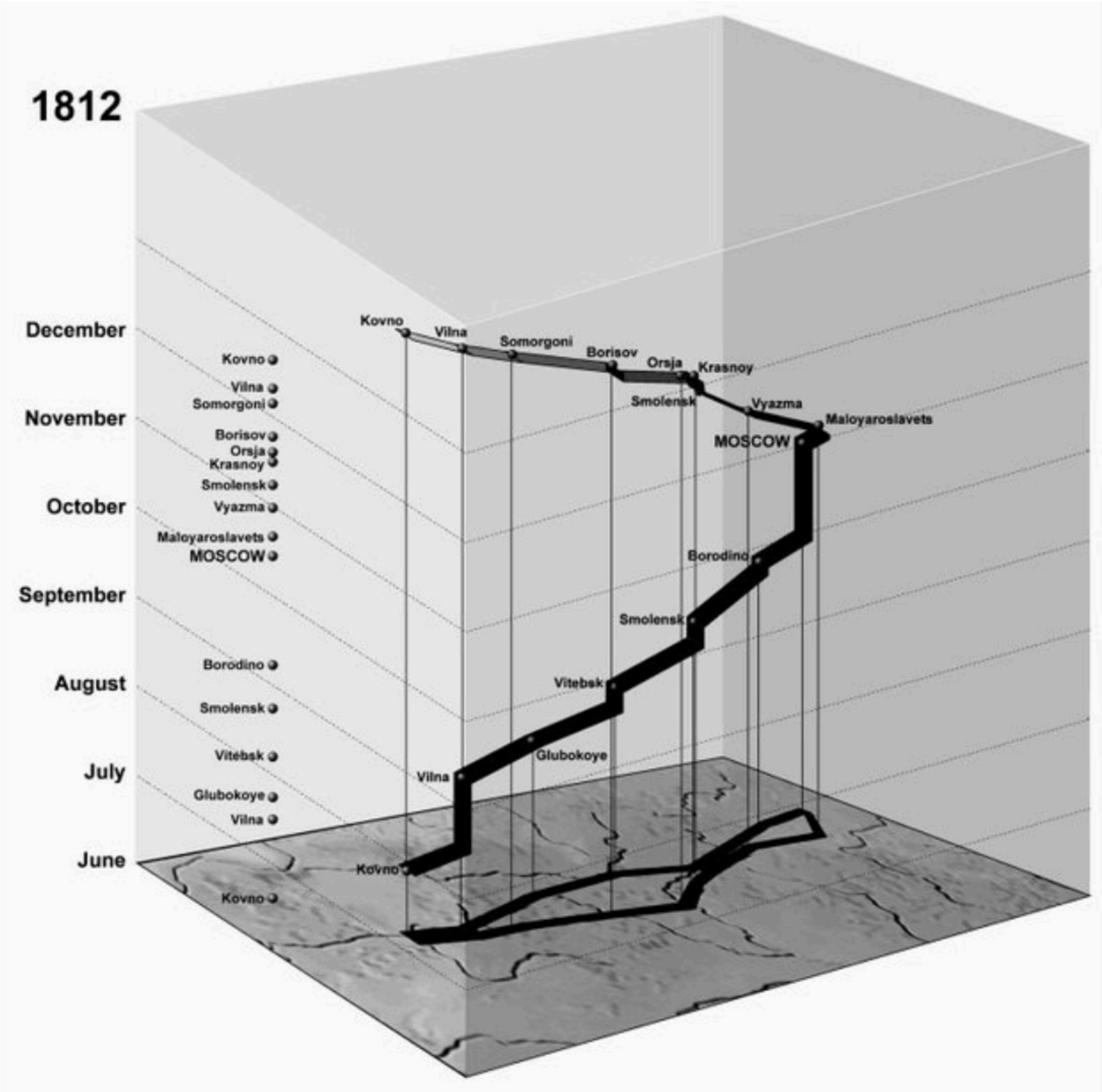


Figure 2.11: A space-time cube that represents Napoleon's Russian campaign, based on the original visualization proposed by Charles Minard [33]

conceptual development outlines the results from the work domain study. This approach suggests that if the work domain is not completely understood, the prototyping phase shall not proceed. From a critical perspective, it is still possible to proceed with prototyping, looping back to the work domain analysis phase if needed.

In response to this issue, Roth et al. [38] propose a modification to Robinson et al.'s UCD process with the following phases: 1) prototyping; 2) interaction and usability studies; 3) work domain analysis; 4) conceptual development; 5) implementation and 6) debugging. This time, prototyping has an initial role in the process, where designers develop visualizations according to the way they think about them. After performing interaction and usability studies, the prototype is used as a final component of the work domain analysis to catch new ideas from users. Finally, the results from the interaction and usability studies and work domain analysis formalize the conceptual development phase.

It might happen that the potential users of visualization tools are not familiar with visualization concepts, or are not aware of their true potential. This might compromise the effectiveness of their participation. In order to overcome these issues, Koh et al. [35] propose relevant additional phases based on the approaches of Robinson et al. and Roth et al (see Figure 2.12). These additional phases improve awareness of users about the power of visualization in a general context, but also within their own context by visualizing their own data.

The Visualization Awareness phase consists of introducing general concepts of Information Visualization to users in case they are not familiar with the field. In this phase, the first discussions for retrieving ideas of the development of their own visualizations are held. The Domain Visualization phase incorporates the development of the first prototypes tailored to the users' own data.

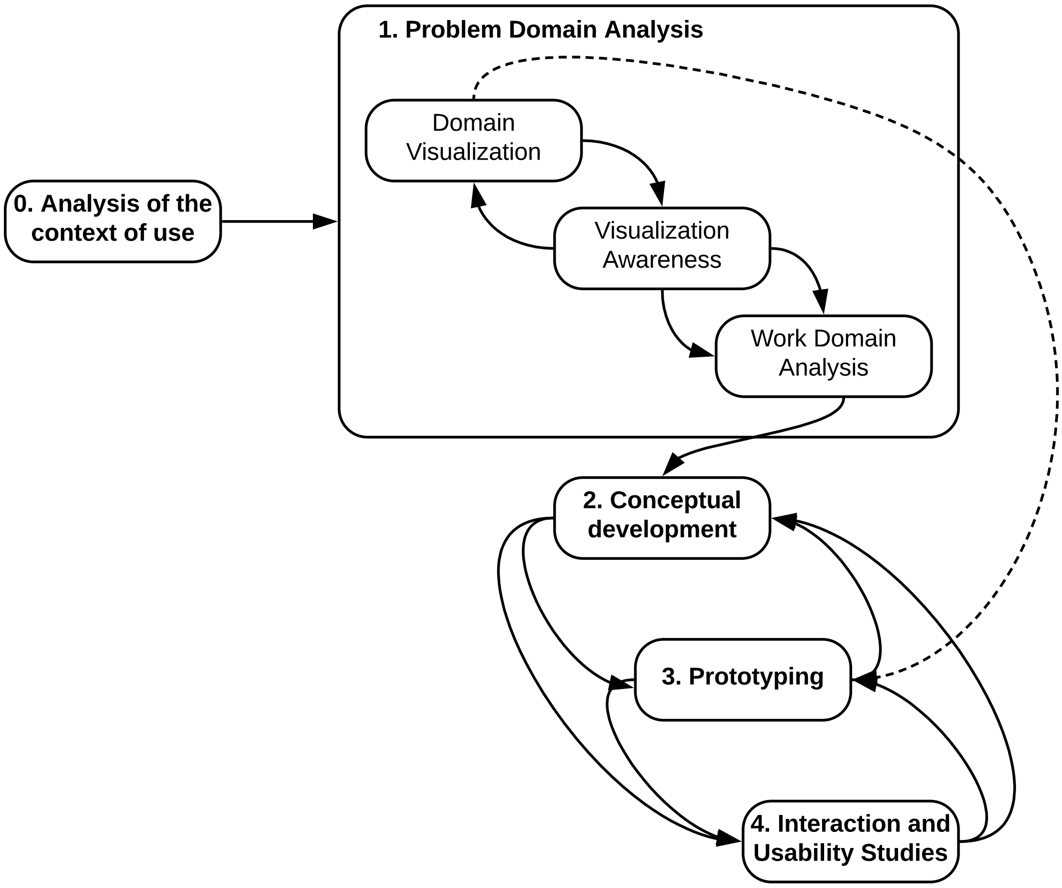


Figure 2.12: The UCD process proposed by Koh et al. (adapted from [35])

Visualization techniques and tools abound, but it is still possible to identify lack of engagement of visualization researchers with users. In fact, visualization tools are useless if users cannot effectively interact with them, thus naturally compromising the processes of knowledge extraction and decision-making. A study by Ellis and Dix [41] – one of the few about evaluation for general Information Visualization – supports this statement: after analyzing 65 papers describing new visualizations, it found that only 12 of them engaged users within evaluation processes. We identified the same issue in the Transportation domain, as discussed in Chapter 3. A further complicating factor is the inherent nature of visualization itself that turns evaluation into a complicated process. Again, the authors indicate some factors that contribute to making evaluation in Information Visualization a hard process:

- Variety of datasets: despite some earlier efforts in creating a standard for datasets, datasets are heterogeneous and hardens the evaluation process, as it might limit the availability and quality of data to be visualized;
- Indeterminacy of tasks: the tasks to be performed during the evaluation process are usually more structured, which differs from the ones to be performed in "real life" that are more exploratory; these ones are harder to replicate in an experiment;
- Participants in context: depending on the complexity of the application context, participants need to have a clear understanding of the problem that the visualizations are trying to solve. Some authors suggest that it is possible to obtain better information by involving domain or usability experts, even though it is typically harder to have access to those people [36, 40].

2.2 Semantic Web Technologies and Ontologies

The World Wide Web has become one of the main sources of information for people. Initially, the provision and access to information were restricted to IT specialists. Nowadays, billions of users have prompt access to the Web, and can share any information about any topic. Information overload is a direct implication, which requires users to search, filter, evaluate and select information that meet their interests.

The web contains numerous smart applications, such as complex search engines, journey planners with real-time information, and e-commerce services that can suggest products according to customers' shopping habits. Those applications depend on consistent data to produce acceptable output. The Semantic Web intends to provide an appropriate infrastructure for information integration on the Web, so that applications can perform to their potential [42].

The technologies presented in this section were originally developed to be used in the Web environment. However, their capabilities have been proven powerful to be applied to other domains of knowledge, such as Biology [43], Knowledge Management [44], and Agent-based Systems [45]. Chapter 3 describes the application of ontologies to Transportation.

2.2.1 Definition and Context

The Semantic Web is a collaborative effort led by the World Wide Web Consortium (W3C), and teams of researchers and industry practitioners. Its main goal is to provide a framework for sharing, finding, reusing and integrating data generated by applications and humans. This framework allows data to be processed and understood by computers, for the sake of switching the users' focus from data-driven to knowledge-driven activities.

The architecture of the Semantic Web is formed by hierarchical layers. Each layer is governed by one or more formal languages, and uses information and resources from layers that are immediately below. Such hierarchy is called Semantic Web Stack, which is illustrated in Figure 2.13.

The first layer, *Identifiers*, consists of Uniform Resource Identifiers (URIs), which uniquely identify resources, e.g. documents or websites, using a string in standardized form. Identifiers form the basis of data representation for Semantic Web applications. A notable type of URIs is the Uniform Resource Locator (URL), which contains the access protocol and network location

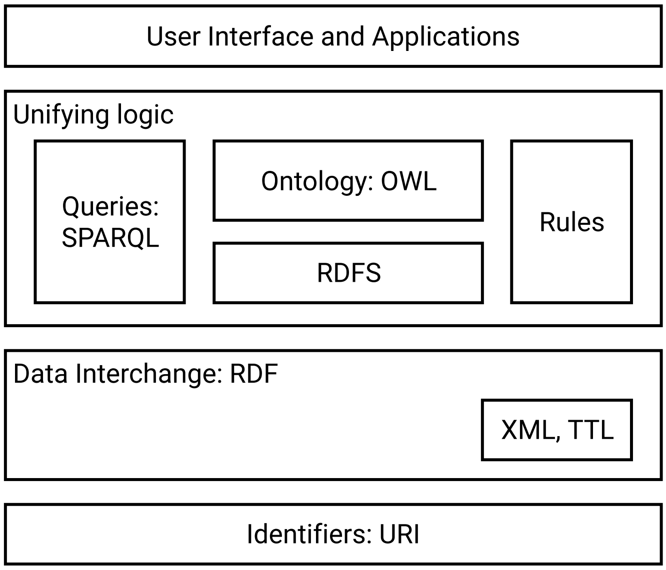


Figure 2.13: Semantic Web Stack (Adapted from [42])

of a resource. For instance, `http://www.fe.up.pt` is the URL and URI for the website of the Faculty of Engineering of the University of Porto (FEUP). URIs provide a trackable, understandable identification for all resources.

The *Data Interchange* layer provides a common syntax layer for representing identifiers and interchange. The standard data model for such representation is the Resource Description Framework (RDF). In RDF, data is expressed as subject-predicate-object triples. Subject and Object are nodes connected by an edge (Predicate). Subjects, predicates and objects are identifiers, hence have a URI. An illustrative example of a triple is given in Figure 2.14. The combination of multiple triples yields a graph structure, which characterizes the term *linked data* that is used to refer to data within the context of the Semantic Web.

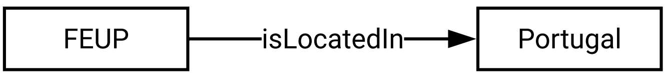


Figure 2.14: A triple according to the RDF data model

RDF data can be serialized using Extensible Markup Language (XML). Such serialization is not meant to be human-readable. Other serialization formats exist, such as the Terse RDF Triple Language (Turtle or TTL), which was designed with human readability in mind. This thesis uses the Turtle serialization to express semantic data, unless stated otherwise. RDF is detailed in Section 2.2.2.

RDF is used to build vocabularies that describe relationships and properties about resources. However, RDF *per se* has limitations with respect to building more formal taxonomies and to specifying part-whole relations, e.g. "Red Wine is a type of wine". RDFS, detailed in Section 2.2.3, provides such capabilities, and is used to build lightweight ontologies, i.e. simple taxonomies. The various classifications for ontologies are given in Section 2.2.7.

OWL, an acronym for Web Ontology Language, is a language derived from the Mathematics field of Description Logics, and allows for the creation of heavyweight ontologies, i.e. formally axiomatized. It provides additional features in comparison to RDFS. Moreover, OWL uses the standardized vocabulary of RDF and RDFS, and it is divided into profiles according to the expressiveness levels. OWL is detailed in Section 2.2.4.

The Rules layer provides logical reasoning to ontologies, using the semantics of RDFS and OWL. There is not an acknowledged standard for rule languages, although two have been used more frequently: Semantic Web Rule Language (SWRL) and SPARQL Inference Notation (SPIN). SPIN is a recent development that uses SPARQL for building complex rules that cannot be expressed with RIF and SWRL, and is considered a *de facto* standard for building semantic applications with rule-based inference. Rules are explained in Section 2.2.6.

SPARQL provides the capability of querying data for RDF data and RDFS/OWL ontologies. The structure of SPARQL queries is similar to the one used by SQL-based languages. Since RDF data is stored as a graph, queries aim to find a subgraph which matches a list of conditions. SPARQL is explained in Section 2.2.5.

2.2.2 RDF - Resource Description Framework

RDF is a framework for representing the relationships between resources, their properties and interrelations as a graph. As mentioned in Subsection 2.2.1, data is represented as Subject-Predicate-Object triples.

For example, consider the following graph in which triples describe a bus in terms of some properties. In a triple, subjects and predicates are always *resources*, i.e. an identifier that can be described in terms of one or more properties, and may belong to one or more classes. The object, however, can also be a literal. In semantic web applications, literals consists of strings, numerical values or complex values such as timestamps. Literals are well-defined datatypes that follow the XML Datatypes specification [46].

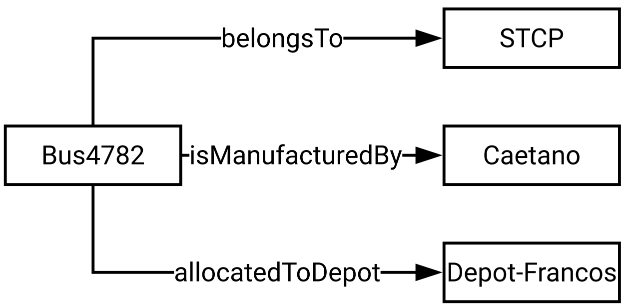


Figure 2.15: Example of a RDF graph that describes a bus

In RDF, resources are preceded by their namespace prefix. Namespaces - a technical term for *scope* - ensure that no ambiguity exists when referring to an identifier. For instance, the namespace for the RDF vocabulary is given by the URL below:

`http://www.w3.org/1999/02/22-rdf-syntax-ns#`

In Turtle notation, prefixes can be represented with a shorter namespace term, defined as qname. In the example of Figure 2.15, the qname `geo` refers to the following URL:

`http://www.w3.org/2003/01/geo/wgs84_pos#`.

The following example provides the serialization for the RDF graph of Figure 2.15. The lines beginning with `@prefix` are used to specify all qnames. The example provides three triples for the same subject `:Bus4782`. In Turtle notation, subject repetition is avoid to facilitate reading. Triples are separated with `;` until the last one, which uses `.`

```
@prefix : <http://web.fe.up.pt/~thiago/example.rdf#> .
@prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#> .

:Bus4782 :belongsTo :STCP ;
     :isManufacturedBy :Caetano ;
     :allocatedToDepot :Depot-Francos .
```

RDF allows the creation of blank nodes (bnodes), i.e. a node that does not have a URI. Despite not having an actual identification, semantic web applications can still identify them by providing internal labels, so that information is referenced correctly. Blank nodes are useful for representing information that does not require an identifier. Recalling the example of Figure 2.15, the following example provides the location of `Depot-Francos` as a blank node. The `geo` namespace indicates that we are using resources from that namespace.

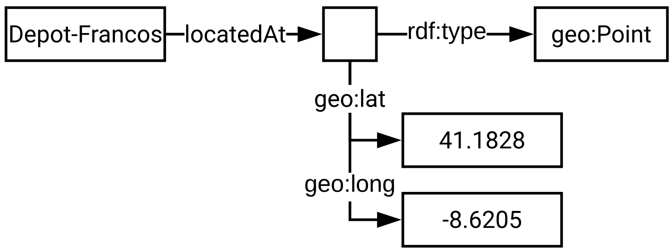


Figure 2.16: Description of a location with a blank node

The serialization of a blank node makes use of brackets, as in the following example:

```
:Depot-Francos :locatedAt [ rdf:type geo:Point ,
                           geo:lat "41.1828" ,
                           geo:long "-8.6205" . ]
```

2.2.3 RDFS: RDF Schema

RDFS extends the RDF vocabulary with components that describe taxonomies of classes and properties. Main concepts include the specification of subclasses and subproperties, range and domain of properties. Table 2.5 describes the main classes found in RDFS and RDF. All resources belong to a certain type (class). Every class is also a resource. For instance, `rdfs:Class` is a subclass of `rdfs:Resource`, which is a superclass for all possible resources. Using the same rationale, classes are instances of `rdfs:Class`. In the following example, BusStop is an instance of `rdfs:Class`. Table 2.6 describes the main properties of the RDFS vocabulary.

The concepts of domain and range are analogous to their mathematical counterparts. As an illustration, consider the property `operatedBy` that indicates the public transportation operator of a bus stop. The following assertions mean that the Subject of any triple that contains such property should be an instance of BusStop. Likewise, the Object should be an instance of Operator. Consider the triple "`BS039 operatedBy MBTA`". Assuming that our dataset did not have explicit i nformation about the types (classes) of BS039 or MBTA, an inference engine would still

Table 2.5: List of all classes in RDFS/RDF vocabulary

Class	Collection of
rdfs:Resource	All resources
rdfs:Class	All classes
rdfs:Literal	Literal values
rdfs:DataType	Data types
rdfs:XMLLiteral	XML Literals
rdf:Property	All properties
rdf:Statement	All statements
rdf:List	All lists
rdfs:Container	All containers
rdf:Bag	All unordered containers
rdf:Seq	All ordered containers
rdf:Alt	All alternative containers
rdfs:ContainerMembershipProperty	All properties that express membership

Table 2.6: Main properties in RDF/RDFS vocabulary

Property	Description
rdf:type	Instance of
rdfs:subClassOf	Subclass of
rdf:subPropertyOf	Subproperty of
rdfs:range	Restricts domains
rdfs:domain	Restricts subjects
rdf:first	First element of a list
rdf:rest	The remaining of a list
rdf:_1, rdf:_2,..., rdf:_n	Container membership properties
rdf:subject	Subject of a statement
rdf:predicate	Predicate of a statement
rdf:object	Object of a statement

be capable of deducting such facts, due to the domain and range assertions that were established before.

2.2.4 OWL: Web Ontology Language

OWL extends RDF and RDFS with additional components found in Description Logics, resulting in increased expressiveness and reasoning capabilities. However, increased expressiveness implies a trade-off with respect to reasoning efficiency. To overcome this issue, different OWL profiles were implemented. A profile is a subset of OWL that guarantees reasoning in practical computing time by sacrificing some of the logical expressiveness. We provide a brief description about the specifications and subsets of OWL.

The first specification of OWL implemented three profiles: Lite, DL and Full. OWL Lite was designed to model taxonomies with simple constraints, including 0-1 cardinality restrictions, i.e. restrictions about the number of possible instances of a class. OWL Full has no expressiveness constraints, but it cannot guarantee decidability. OWL DL provides many of the capabilities of Description Logics. Despite having the entire vocabulary of OWL Full, it introduces a restriction in which a URI cannot be simultaneously treated as an individual and class or property.

OWL 2, the second specification of OWL, implemented three profiles based on OWL DL:

EL, QL and RL.

OWL EL is designed to provide reasoning in polynomial time, with respect to the size of the ontology. It suits ontologies that contains a large number of properties and classes, and do not make use of computing-intensive rules.

OWL QL focuses on efficient query answering, and handling large volumes of instance data. This profile is based on work involving database integration, and provides features found in UML and ER models. Query answering ideally performs in logarithmic time, with respect to the size of instance data.

Finally, OWL RL is designed to provide maximum expressiveness possible while allowing the use of rules and rule-processing systems, i.e. inference engines. Rules must be conjunctive, i.e. clauses (atoms) can only be linked by the logical AND operator. In comparison to the other two profiles of OWL 2, OWL RL provides more constructs than OWL QL for defining classes.

2.2.5 SPARQL Query Language

SPARQL provides a protocol for accessing RDF data local and remotely, and a set of SQL-like instructions for querying RDF graphs. In contrast to languages based on SQL, there are no table join operations. All conditions and operations consist of finding a subgraph that matches all the conditions stated in a query's body. SPARQL provides four query result types:

- `SELECT`: returns values of variables that are bound to a query pattern;
- `CONSTRUCT`: returns an RDF graph by replacing variables in a query pattern;
- `DESCRIBE`: returns an RDF graph describing the resources that were found;
- `ASK`: returns a true boolean value if there is a subgraph that matches the given conditions.

As an illustration, the following query selects the identification and name of all students who belong to the University of Porto:

```
PREFIX : <http://example.com/student.owl#> .
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
PREFIX owl: <http://www.w3.org/2002/07/owl#> .

SELECT ?id ?name
WHERE {
  ?x :studiesAt :UPorto .
  ?x :hasStudentID ?id .
  ?x :hasName ?name . }
```

The `PREFIX` section works in the same way as in RDF files. Variables followed by a `?` are placeholders, which are used for graph matching purposes. For instance, `?id` and `?name` are variables that were placed as objects of triples `?x :hasStudentID ?id` and `?x :hasName ?name`, so that they will receive values stored in RDF data, in case the query succeeds in finding a subgraph with such conditions. Variable names do not matter in SPARQL. In practice, however, they should be chosen in such a way to facilitate readability. Variables can return values from subjects, predicates and objects.

A CONSTRUCT query returns a graph by replacing variables in a query pattern. For instance, the following query associates students to the class `WorkerStudent` in case it is known that they work at an institution:

```
PREFIX : <http://example.com/student.owl#> .
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
PREFIX owl: <http://www.w3.org/2002/07/owl#> .

CONSTRUCT { ?x a :WorkerStudent }
WHERE {
  ?x :studiesAt :UPorto .
  ?x :isActive "true" .
  ?x :worksAt ?company . }
```

A DESCRIBE query returns a single result RDF graph, which describes a given resource. As it is not still standardized, different SPARQL engines may return different results.

An ASK query returns a boolean value (`true` or `false`) according to whether the graph matching succeeded or not. For instance, the example of SELECT query could be modified to become an ASK query. If at least one student of University of Porto was found, the result would be `true`. Otherwise, `false` is returned.

As in SQL-based languages, SPARQL supports aggregate functions and constructs for building more complex conditions. The current stable version, SPARQL 1.1, became a W3C standard in 2013. A complete reference can be found in [47].

2.2.6 Rules and Reasoning

Rules reveal implicit knowledge. The importance of rules is evident in some ontology applications, such as rule-based systems for simulation purposes. In this section, we present a brief list of available technologies for reasoning.

Ontology rules can be built using a rule language. SWRL [48], Jena Rules[1] and SPIN [49] are the main ones adopted in Semantic Web applications. SWRL and Jena Rules share several similarities in terms of rule construction. SPIN uses SPARQL to represent rules and constraints. It is the industry *de facto* standard for building rules.

There are several Semantic Web frameworks for reasoning over ontologies. Some of them were originally written as description logic reasoners, such as Pellet[2], FaCT++ [50], and TopSPIN [49]. In this thesis, we focus on Apache Jena[3], a Semantic Web framework for Java. Jena provides reasoning for RDFS and OWL.

2.2.7 The role of Ontologies

Ontologies are considered to be the core of the Semantic Web. In brief, ontologies consist of a common set of terms for describing and modeling knowledge of a domain (e.g. urban mobility). Semantic models can be used by applications to make inferences about data or to analyze it considering its surrounding context. The term ontology originated in the domain of metaphysics, concretely in the Aristotelian studies on the nature of things and their existence.

[1]https://jena.apache.org/index.html
[2]https://github.com/stardog-union/pellet
[3]https://jena.apache.org/index.html

In the domain of Information Systems, especially in Artificial Intelligence, a number of authors propose definitions for the concept of ontology [51–53]. Despite specific differences, those definitions agree on its purpose of formalizing knowledge through a structured representation. For the purposes of our research, we adopt a concise definition coined by Thomas Gruber [51], which is adopted in many ontology-related studies:

> *An ontology is a formal, explicit specification of a shared conceptualisation. A 'conceptualisation' refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. 'Explicit' means that the type of concepts used, and the constraints on their use are explicitly defined. 'Formal' refers to the fact that the ontology should be machine readable, which excludes natural language. 'Shared' reflects the notion that an ontology captures consensual knowledge, that is, it is not private to some individual, but accepted by a group.*

Existing classifications of ontologies regard their level of formalization, expressiveness and specificity [51–54]:

Classification according to Formalization

Formalization regards the language that is used to represent the ontology.

- *Highly informal:* the ontology is expressed in loose natural language and is prone to ambiguity.
- *Semi-informal:* the ontology is still expressed in natural language with a certain structure, so as to reduce ambiguity.
- *Rigorously formal:* the ontology is defined with formal semantics and may contain theorems and proofs.

Classification according to Expressiveness

Expressiveness regards if an application is able to make or not reasoning tasks with the ontology.

- *Heavyweight Ontology:* the ontology is extensively axiomatized. All constraints are explicitly represented in order to eliminate terminological and conceptual ambiguities. Such ontologies are usually built to support computing-intensive activities such as reasoning and database integration.
- *Lightweight Ontology:* the ontology depicts a simple taxonomic structure of primitive or composite terms (see Figure 2.17), along with their definitions. Constraints or reasoning rules are not specified.

Classification according to Specificity

Specificity comprises the domain(s) that the ontology will define.

- *Generic Ontology:* Commonly regarded as "top-level" ontology, it contains concepts that are considered generic across many fields, such as state, event and action.

- *Core Ontology:* The ontology specifies concepts that are generic across many domains.

- *Domain Ontology:* The ontology specifies concepts that are specific to a certain domain (e.g. urban mobility). Due to their specificity, they seldom figure in other domain ontologies. Otherwise, a core ontology would be considered more appropriate for including such concepts, as they would belong to many domains.

Ontologies share the basic elements of other modeling languages in software and database engineering, such as Universal Modeling Language (UML):

- *Entities*: Subjects that belong to a certain domain;

- *Properties*: Attributes that each entity owns;

- *Relationships*: Relationships among entities;

- *Actions*: Events that occur with entities.

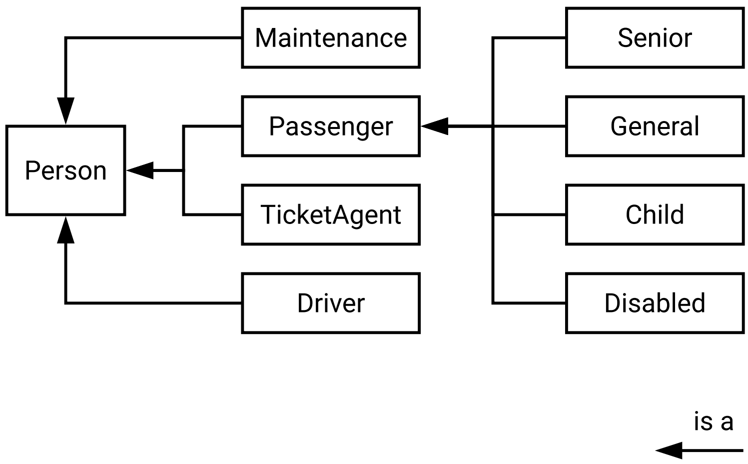


Figure 2.17: An excerpt of an ontology showing subclasses of the class *Person* (adapted from [55])

2.2.8 Benefits of Ontologies for Information Integration

Information Systems modeling is often performed using UML and Entity-Relation (E-R) models. Such models better suit information sharing in closed, constrained environments, i.e. environments in which entities are defined and agreed by a limited number of sources (e.g. elements of an organization) [56]. In environments with multiple data sources, there might be different definitions of an entity and its elements. For instance, an entity might have different names. UML or E-R models cannot account for this type of heterogeneity.

As a simple example, consider two databases (A and B) containing information about types of buses. Suppose that the "articulated bus" type is represented in A as "Articulated", whereas B represents it as "Gelenkbus", its German counterpart. Despite the language difference, both are equal at the semantic level, as they refer to the same type of bus. Extracting knowledge from those databases or integrating them would not be possible without manually identifying and fixing language incongruence (e.g. editing and translating entries manually). Domain ontologies are able to overcome this issue.

Ontologies overcome those issues as they support the following notions: (i) *Anyone can say Anything about Any topic (AAA assumption)*; (ii) *Open-world assumption*, i.e. there might exist more concepts or facts beyond those that are already known (expressed in the ontology); (iii) *Non-unique naming*, i.e. different sources (e.g. people) can provide distinct names for all entities that belong to an ontology [57].

Aligned with the scope of our research, ontologies can provide interoperability of heterogeneous data sources. The schema of each data source needs to be mapped to one or more ontologies [58–60]. Wache et al. [44] analyzed several studies on the use of ontologies for information integration and identified three general approaches: *single*, *multiple* and *hybrid*.

Single approaches use one ontology for specifying semantics for all sources, and is generally used when all sources belong to the same domain. Such approach requires that all sources are compatible with the ontology. Multiple approaches have one ontology for each source and eliminate the requirement of compatibility of all sources. On the other hand, it requires an additional ontology for allowing interoperability between the different ontologies, thus imposing additional formalism. In practice, it is consensual that multiple ontology approaches are very difficult to be specified. Hybrid approaches lie between single and multiple approaches. Again, there are ontologies for semantically describing each source, but they are built according to a shared, consensual vocabulary [61].

2.2.9 Semantic data in the context of Information Visualization

A study by [62] proposed the Linked Data Visualization Model (LDVM), an abstract data process for dynamically connecting semantic data with visualizations. LDVM is an adaptation of the Data State Reference Model (DSRM) proposed by [30] and can be considered a workflow starting with semantic data and ending with visual representations of such data. However, the model assumes that a certain ontology is provided beforehand.

LDVM provides a general formalization of the process for building visualizations based on linked data and consists of 4 stages (see Figure 2.18) through which data must pass:

- *RDF Data:* input of raw semantic data in Resource Description Framework (RDF) format;
- *Analytical extraction:* data extractions retrieved from raw data, such as aggregated values;
- *Visualization abstraction:* the information to be displayed by the visualization technique;
- *View:* the visualization itself, as presented to the user.

2.2.10 Knowledge-assisted Visualization

Empirical user knowledge can be used by visualization tools to assist users on finding appropriate visual representations for their data, by means of (semi)-automatic recommendations. There are still few studies on this area, mostly consisting of applications to general domain data. Studies can be classified into three approaches: rule-based, behavioral, and personalized [63].

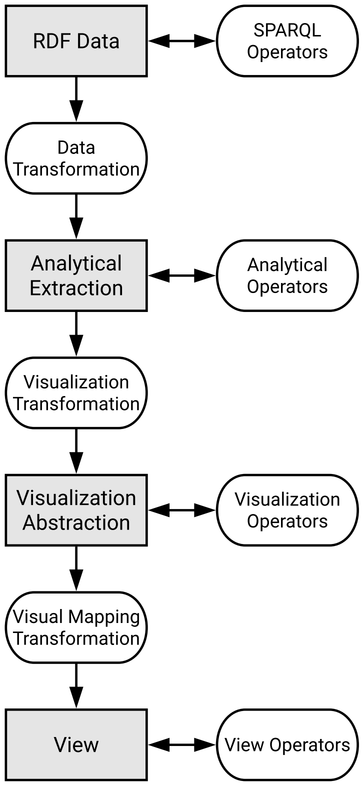


Figure 2.18: Overview of the LDVM model [62]

An early study by Stolte and Hanrahan [64] is one of the first that proposed automated generation of visualizations. Polaris, the initial backbone engine of *Tableau*, suggested visualizations for tables in relational databases. The system, however, required users to manually map data onto visual variables of each visualization technique.

The ShowMe system proposed by Mackinlay et al. [65] aimed to automatically generate visualization techniques for *Tableau*, by helping users on searching for the most appropriate visual representations for their tasks. Recommendations were based on data properties, and were ranked according to globally defined scores for every visualization type supported by the tool.

Voigt et al. [66] proposed another rule-based approach, in which a knowledge base of various ontologies was used to recommend visualization techniques. The system pre-selects visualization techniques based on data properties and tasks. Such techniques are then ranked according to several criteria like visualization facts and user context. Their system is supported by the Visualization Ontology (VISO), which describes data schema and the components of visualization techniques. The approach assumes that data are semantically integrated.

Nazemi et al. [29] proposed a behavioral approach to visualization recommendation. As the users interacted with the visualization system with the various techniques, their use profiles were stored for analyzing their behavior through interactions.

Mutlu et al. [63] carried out a comprehensive approach by involving lay users in a crowdsourced study with general public. Several dimensions for rating visualizations were defined according to usability factors. In both works, ontologies were used as a semantic foundation. They have been used to address other problems, e.g. knowledge management and travel planning.

Knowledge-assisted visualization is usually based on two main classes for content recommendation: Collaborative filtering (CF) and Content-based methods (CB) [67]. CF assumes that users with similar behaviors or preferences in the past are likely to have the same preferences in the future. CF-based methods generate recommendations based on a collection of user preferences, e.g. explicit ratings to items, or implicit ratings which can be inferred from behavior over time.

CF algorithms organize ratings as a matrix $A_{m \times n}$, where each entry $a_{i,j}$ corresponds to a rating given by user i on regards to item j. The top n recommendations for a user u can be obtained by calculating the nearest neighbors to u, i.e. most similars users or items. CF methods are divided into two broad groups: memory-based (user-based) and model-based (item-based).

Memory-based methods predict the nearest neighbors from similar users who have rated the same items. The prediction averages the ratings of the active user and the similar ones, thus generating a numerical value within the same rating scale. Model-based methods calculate similarity between items that were rated by the active user.

Various measures can be used to calculate similarity in CF methods. For instance, the Pearson correlation measures the strength of linear correlation between two variables, thus varying from -1 (extremely negative) to +1 (extremely positive). Memory-based methods add and subtract the neighbors' bias from the user average. Such measure is used for predicting a certain item. Model-based methods calculate the average of similar items rated by the user.

Content-based methods recommend items based on the correlation between the content of items, and the user's profile. The item content can be described by a set of features. CB methods typically use Vector Space Model (VSM) and Term Frequency - Inverse Document Frequency (TF-IDF) weighting to calculate the correlation between users and items. Each item is represented as a vector of term weights. A weight comprises the association degree between an item and a term. Analogously, user profiles can be represented by profile vectors. The following cosine measure can be used to determine the similarity between an item and a user profile:

$$\cos(\theta) = \frac{\mathbf{U} \cdot \mathbf{V}}{\|\mathbf{U}\| \|\mathbf{V}\|} = \frac{\sum\limits_{i=1}^{n} U_i V_i}{\sqrt{\sum\limits_{i=1}^{n} U_i^2} \sqrt{\sum\limits_{i=1}^{n} V_i^2}}$$

CF methods require low to no cost for acquiring knowledge from users, and such knowledge is independent of the items. However, such methods are prone to *cold start*, a situation in which no feedback from any user has been collected [67]. Data sparsity, in the case of a small amount of collected feedback, may imply failure to provide accurate recommendations. CB methods do not require direct user involvement, and they can provide recommendations even if no ratings have been collected. A shortcoming is that recommendations may end up being too general. For instance, Mutlu et al. [63] considered a hybrid approach by mixing both methods, and compared its accuracy to approaches based on CF and CB methods. The authors found that the hybrid approach provided more accurate results.

2.3 Summary

This chapter presented the major theoretical foundations and significant studies in the areas of Information Visualization and Semantic Web. The latter focused on ontology modeling. The use of visualization techniques in the domain of urban mobility is acknowledged as a relevant mean for extracting knowledge about a city through visual cues. Such techniques generally have the shortcoming of being designed for a specific data schema and pose a natural barrier for domain experts without technical knowledge that could take advantage of them. The development of visualization techniques still presents a heavy focus on technology rather than on their end-users. Involving end-users on the development cycle can enhance the efficacy of knowledge extraction.

This chapter introduced the concept of Semantic Web and its technologies. The Semantic Web aims at providing a framework that turns computers capable of processing and understanding data, so that users can switch focus from data-driven to knowledge-driven activities. Although such developments were oriented towards the World Wide Web, the potential of those technologies have been acknowledged and applied to various scientific areas.

Semantics is brought to data through formal languages. Semantic data is expressed in RDF triples. A triple consists of resources that can be connected through many other triples, yielding a graph. RDFS and OWL extend RDF with constructs that allow the creation of ontologies. OWL is the current W3C standard for building ontologies, and consists of profiles that guarantee reasoning in practical computing time by sacrificing some level of expressiveness. Semantic data can be queried using SPARQL, a SQL-like language where queries return results based on graph matching operations. Data may contain implicit knowledge that can be made explicit using logic reasoning rules declared in an ontology.

The existence of few studies on knowledge-assisted visualization, combined with the lack of related applications to Transportation, indicates an important research opportunity. An overview of recommendation methods was presented, with focus on collaborative filtering and content-based techniques. CF-based methods generate recommendations based on a collection of user preferences, e.g. explicit ratings to items. Content-based methods recommend items based on the correlation between the content of items, and the user's profile. To account for their limitations, hybrid approaches have been proposed.

Chapter 3

Literature Review

This chapter describes the current state of the art on the applications of Information Visualization (Section 3.1), Semantic Web technologies and ontologies to Transportation (Section 3.2), with focus on urban mobility analysis.

3.1 Information Visualization

This section aims to answer the following questions:

1. Which phenomena related to transportation have been analyzed using visualizations and which types of data have been exploited?
2. How traditional techniques have been used, and which novel techniques have been proposed?
3. To which extent end-users have been involved in the design and development of visualizations?

Past studies highlighted the myriad of opportunities for visually exploring transportation data, including those related to urban mobility [68, 69]. Pack [69] stated that the following research areas were considered promising:

- Real time visualization;
- Visual mining of archived data;
- Virtual design for construction.

The same study emphasized the role of the end-user during the design and development of visualization techniques. Evaluation is fundamental to ensure that techniques can actually support data exploration and knowledge extraction.

Early studies that addressed visualization in some way were supported by Geographic Information System (GIS) components. The availability of novel, free visualization frameworks, e.g. Processing[1], D3.js[2], paved the way for the development of innovative visualization tools and

[1] http://processing.org
[2] http://d3js.org

techniques that do not necessarily depend on a particular GIS. Furthermore, Intelligent Transportation Systems also had a crucial role in the advance of visualization, due to availability and size of collected data.

In 2004, Hughes et al. proposed a research agenda for the application of data visualization to Transportation Systems, based on the state of practice at that time [70]. In one of topics proposed by the authors, "Development of Visualization Standards for Source Data and Interoperability", they discussed the importance of meta-data on ensuring interoperability across systems, and visualization system architectures to support heterogeneous data sources. To the best of our knowledge, no related applications to urban mobility were found. In parallel, some authors acknowledged that visualization of heterogeneous data is an upcoming research trend [71]. Applications to urban mobility analysis started to gain momentum since the last decade. In 2011, a survey by Zhang et al. found that the number of visualization studies in transportation was small in comparison to other topics [71]. In 2015, a study by Chen et al. [7] provided a survey on visualization of traffic data of many means of transportation, including vessels and airplanes. In 2018, a survey by Andrienko et al. shed light on the lack of involvement of domain users throughout visualization development. The authors posited that insufficient communication can hinder visualization developers' knowledge about the problems and needs, and reduce the potential usability of visualization tools [72]. Our literature review endorses that statement; few studies described some level of involvement of domain users on the design, development and evaluation of visualization techniques.

The figures shown in this chapter were selected based on the relevance and originality for the advance on visualization of urban mobility data. We classified the surveyed studies according to several topics, which will be explained throughout this chapter. The list below indicates the topics with higher proportion of surveyed studies:

- Urban traffic flows and monitoring;
- People dynamics in urban environments;
- Road traffic incidents;
- Air pollution.

Based on the surveyed studies, we were able to provide two tables, shown in the end of this section, that summarize our findings. In Table 3.1 we list the surveyed topics and their related studies. Topics were sorted according to the number of related studies. Table 3.2 provides a classification of data source types which extends the one found in [7], and the studies that used such types.

3.1.1 Urban traffic flows and monitoring

Urban traffic flows analysis and monitoring is the most studied topic in mobility visualization. We begin by mentioning two early works on this subject. Firstly, in 2000, Claramunt et al. stressed the limitations of GIS software for managing data of very dynamic geographical phenomena [73]. Their work proposed a GIS-based prototype for visualizing urban traffic data. Various interactive visualization techniques were used to represent data with different perspectives and levels of aggregation (see Figures 3.1 and 3.2), thus providing a good reference for further works on how different visual perspectives could be useful for domain experts who act on different contexts.

For instance, on Figure 3.1 provides an overview on how traffic flows could be analyzed (a) spatially, through bidimensional maps, (b) thematically, with area charts, (c) temporally, through line plots that represent time series, and (d) with aggregation, by using bar charts to group intervals of values.

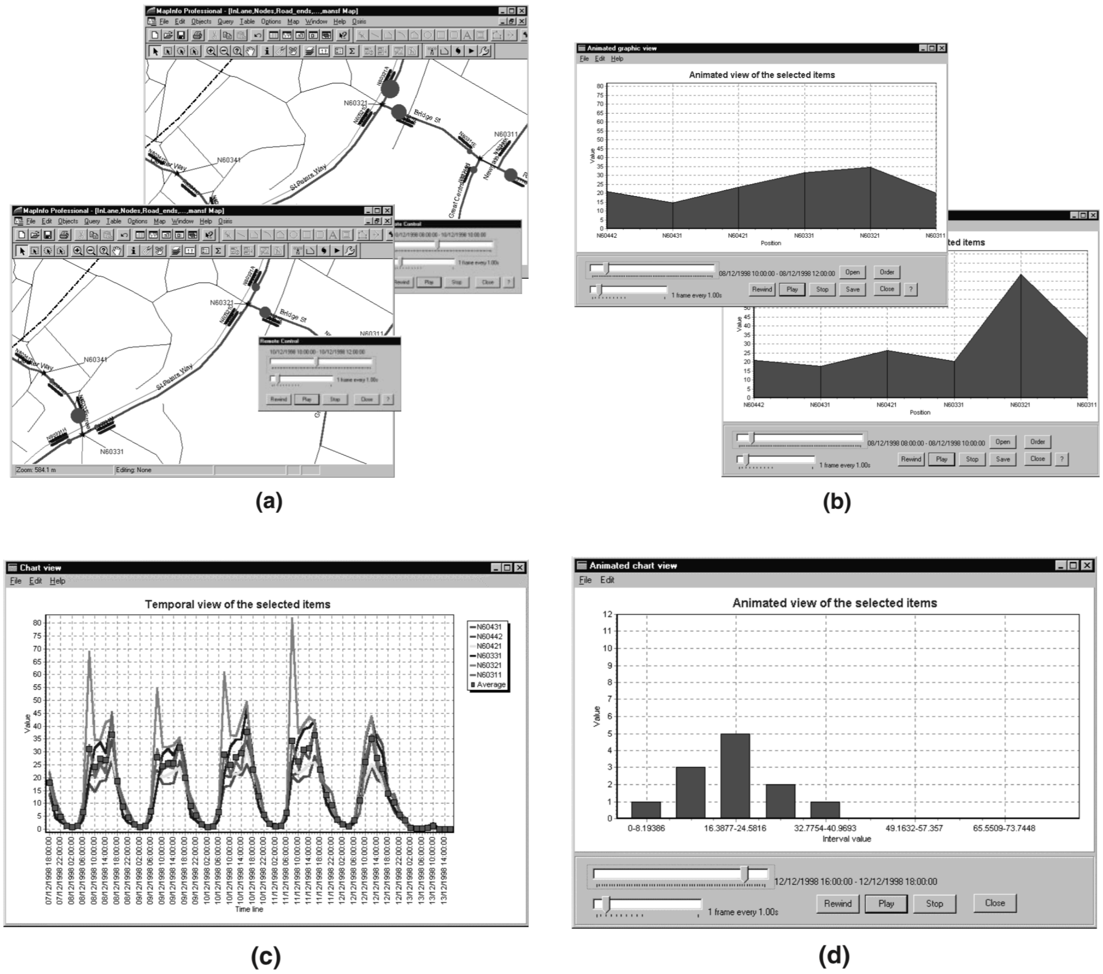


Figure 3.1: Different views on urban traffic data [73]

The second historic work, published in 2002, by Shekhar et al, seems to be the first that developed a non-GIS visualization tool for traffic flows data [74]. The *CubeView* system could be publicly accessed from a web browser, and displayed traffic video, maps with highway traffic intensity and outlier stations for user-specified date and time. Wang proposed the use of three-dimensional (3D) visualizations for a simulation-based traffic impact analysis system [75]. To facilitate interaction with the 3D environment, the visualization interface allows the user to interact with a 2D representation of road network. Through brushing and linking interactions, the selected road segment can be seen on the 3D canvas. 3D visualization of roads and moving vehicles were used by Sewall et al. to represent reconstructed traffic flows from discrete spatiotemporal data [76].

Guo et al. developed *TripVista* (see Figure 3.3), an innovative visualization tool for analysis of microscopic traffic behaviors, e.g. at road intersection, and abnormal patterns [77]. The tool tackles the intrinsic difficulty due to multidimensional nature of such data by introducing several visualization techniques. Geographic visualization is combined with abstract representations such as parallel coordinate plots, which have been used to represent multidimensional data, and scat-

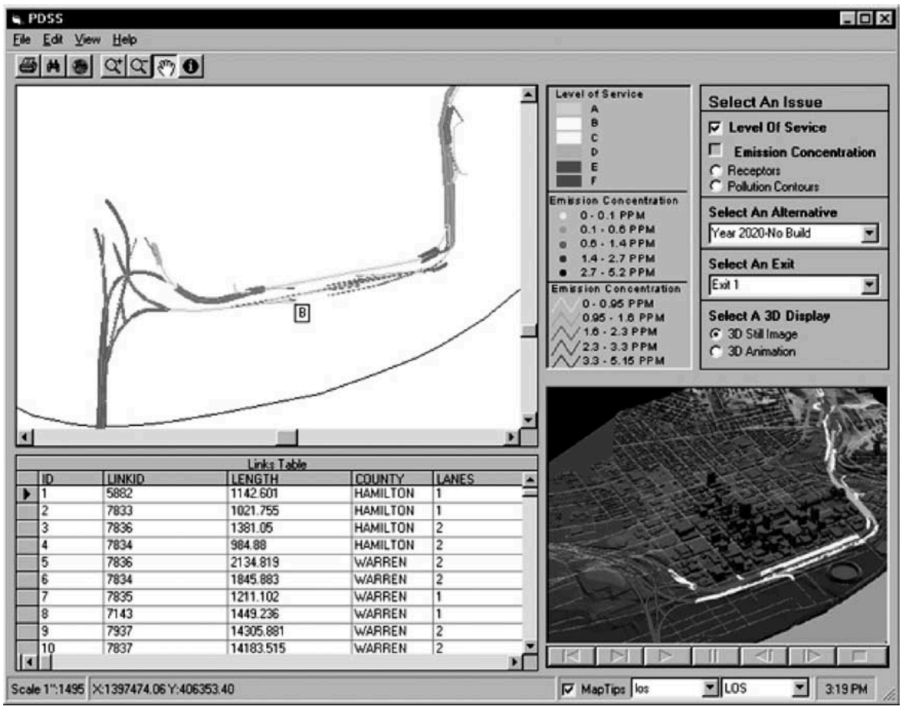


Figure 3.2: Traffic analysis of a road network section. Through brush and linking, the 2D visualization of the selected road segment can be seen on the 3D canvas [75]

terplots. The user can interact simultaneously with all visualization techniques through brushing and linking. The system also makes use of the ThemeRiver visualization technique, which is used to depict thematic variations over time [78]. Users have been involved on the system evaluation. Although the evaluation protocol has not been specified on the paper, the authors stated that the feedback was positive among domain experts.

Bak et al. [79] provided an interesting application of the rose diagrams of Florence Nightingale (see Figure 2.2) to the analysis of spatiotemporal stops due to traffic congestion (see Figure 3.4). In accordance to the original visual metaphor, circle segments represent time (hours of the day). The transparency of each circular segment is used to depict the number of occurring traffic jams. Finally, the size of each circular segment represents the duration of traffic jams. The authors tested the visualization technique with public transportation system data from Helsinki, Finland. In Figure 3.4, A, B, C, D, and E represent city landmarks.

Some authors focused on the temporal perspective of traffic flow analysis [80–83] using abstract visualization techniques. Song and Miller proposed a heat map matrix to analyze congestion patterns across two temporal granularities: days of the weeks or months, and time of the day [80], as shown in Figure 3.5. Such matrices can be effective on identification of abnormal patterns and have been applied to other visualization tools [84]. Liu et al. and Pu et al. applied circular heat maps for the same purpose [81, 82], which were overlaid on a map.

The work of Chen et al. [83] highlights the importance of the semantic zoom interaction for analyzing phenomena at different levels (see Figure 3.6). Speed bottlenecks retrieved from vehicle sensors can be analyzed in various temporal granularities in a heat map matrix. As the user selects the desired time period, the visualization technique adapts to show the respective vehicle speeds and flow intensity. The study provided extensive evaluation with domain experts and explicitly followed several visualization principles found in theory.

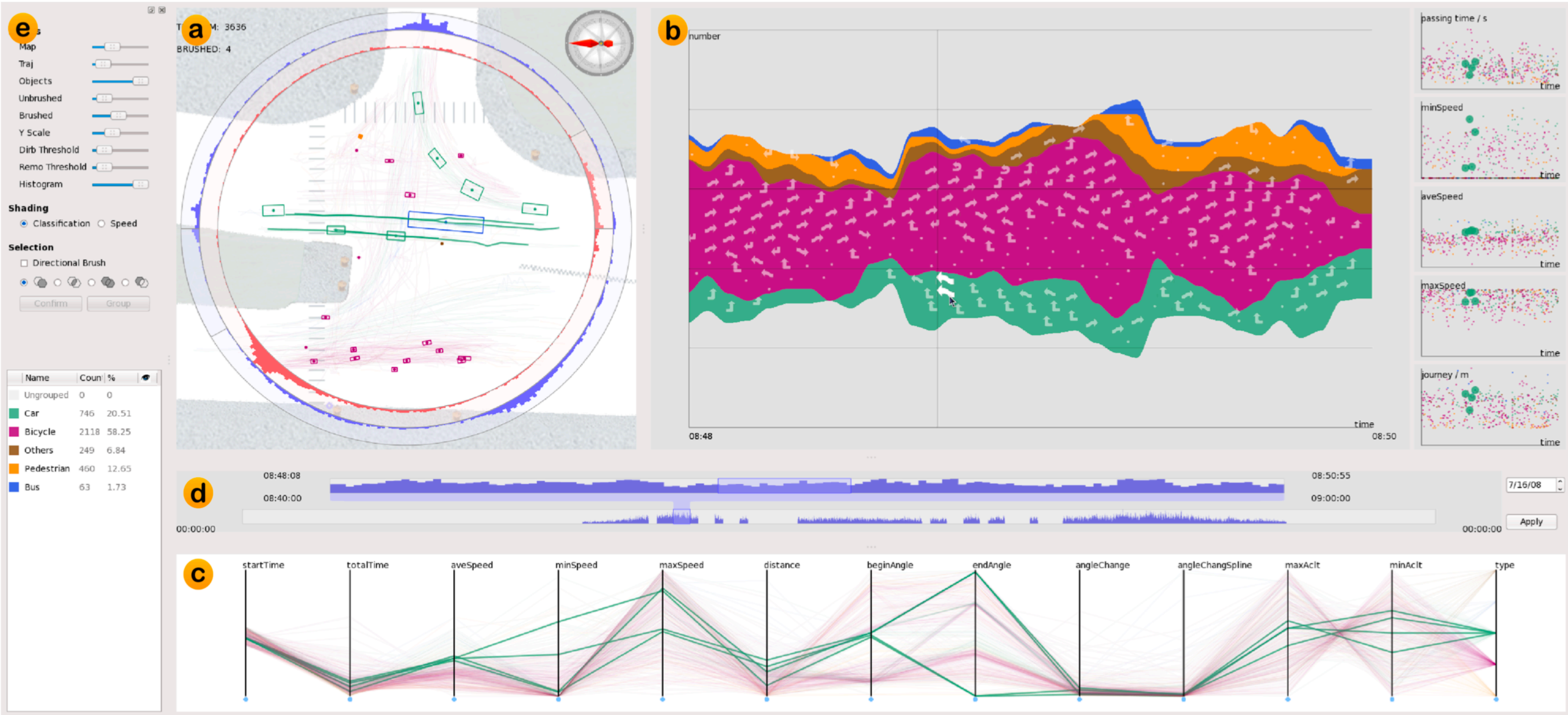


Figure 3.3: The TripVista system combines various types of visualization techniques for microscopic traffic data [77]

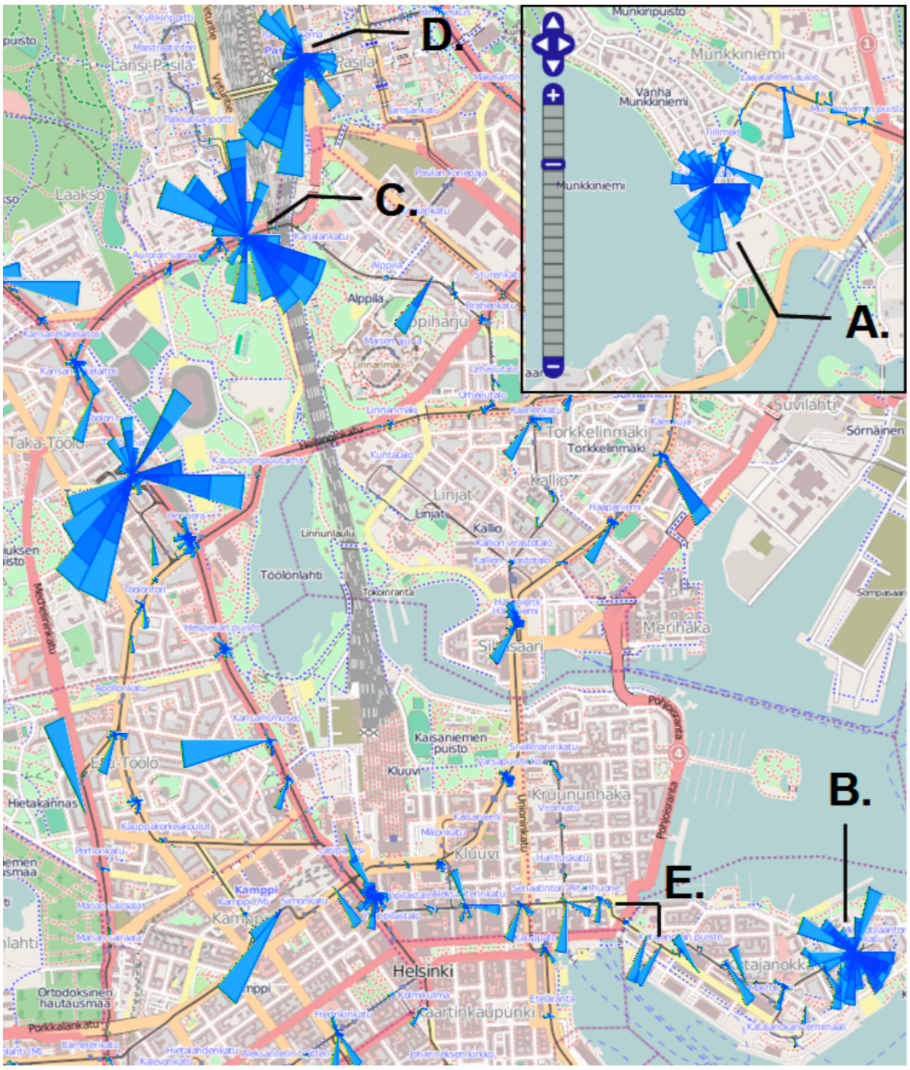


Figure 3.4: Rose diagrams provide an interesting visualization of stops and congestion analysis (extracted from [79])

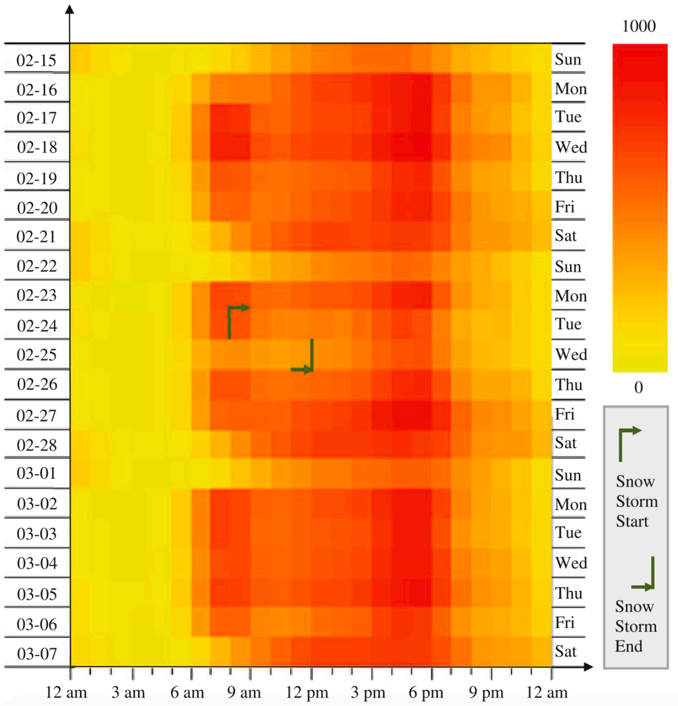


Figure 3.5: A heat map matrix visualization for traffic congestion analysis [80]

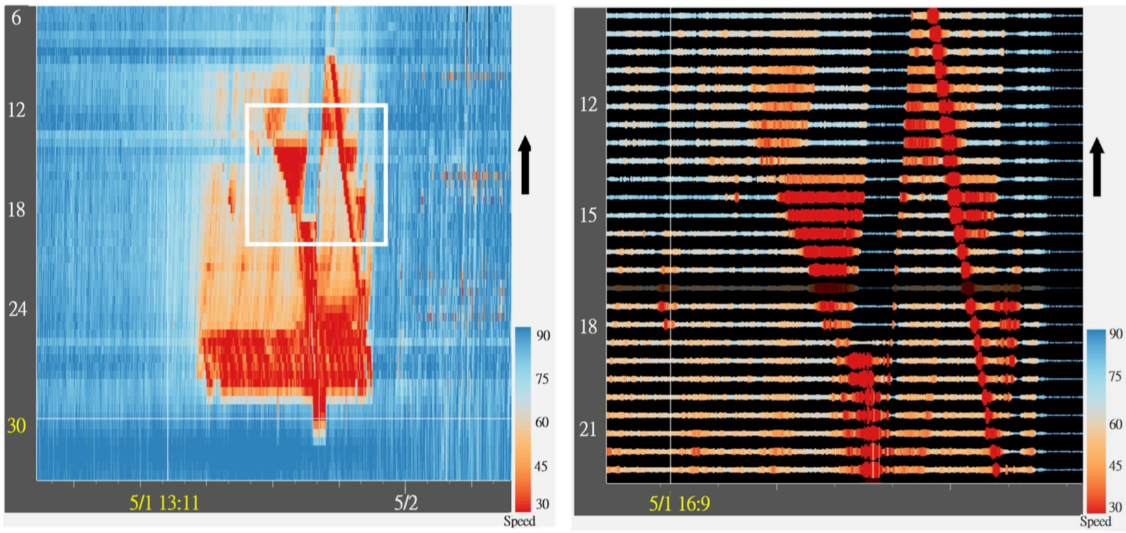
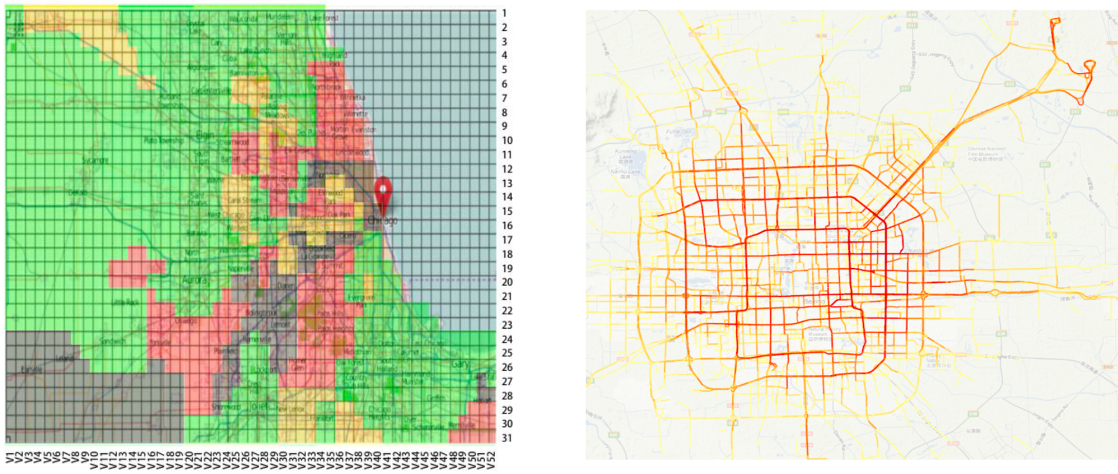


Figure 3.6: An example of semantic zoom for exploring traffic bottlenecks in different temporal granularities [83]

Other common, simple ways of depicting traffic flows using map-based techniques are overlaying heat maps on geographic maps [84–89] (see Figure 3.7a) or road segments [84, 86, 88, 89] (see Figure 3.7b), which are effective for detecting phenomena such as traffic jams.



(a) (b)

Figure 3.7: Simple ways of representing heat maps on geographic maps. (a) consists of a heat map overlay [85], while (b) provides colors to road segments according to a given scale [84]

Cheng et al. revisited the space-time cube proposed by Hägerstrand and applied three 3D visualization techniques to the exploration of congestion patterns: isosurface, network-constrained isosurface, and wall map [90]. An isosurface shows points of equal value on a 3D shape (Figure 3.8a). The network-constrained isosurface enhances the accuracy of the isosurface, as it assumes that congestion values will be interpolated between roads, i.e. where there are no cars (Figure 3.8b). A shortcoming of both methods is that they become less effective on analyzing particular road links. The wall map overcomes such limitations by reducing visual clutter and revealing congestion levels on road links. The authors demonstrated the effectiveness of those techniques with traffic data from London, extracted from ANPR systems.

Tanaka et al. combined map-based visualization and traditional techniques on a geospatial dashboard for winter road management using vehicle sensor and microblogging data [91]. The dashboard provides coordinated multiple views, through brush and linking interactions, and feature traditional visualization techniques such as bar charts, histograms, scatterplots and dendograms. The tool can be manipulated on touch-screen enabled devices. Wang et al. also proposed a dashboard for exploring real traffic situations, and features a map-based visualization technique overlaid with heat maps, with bar charts, histograms and scatterplots [92].

Hsieh et al. approached the problem of traffic flow analysis using video stream data [93]. The visualization system uses video streams from one location to depict traffic situation of other places.

Huang et al. proposed the *TrajGraph* system to analyze traffic flows using taxi trajectories data [94]. The interface provides multiple coordinate views with different visualization techniques. A map-based view was combined with a rose diagram overlay (similar to Figure 2.2 that is used to represent traffic information and network centralities. Line plots and an abstract, graph-based representation of the road network are used. The authors carried evaluation with one domain expert.

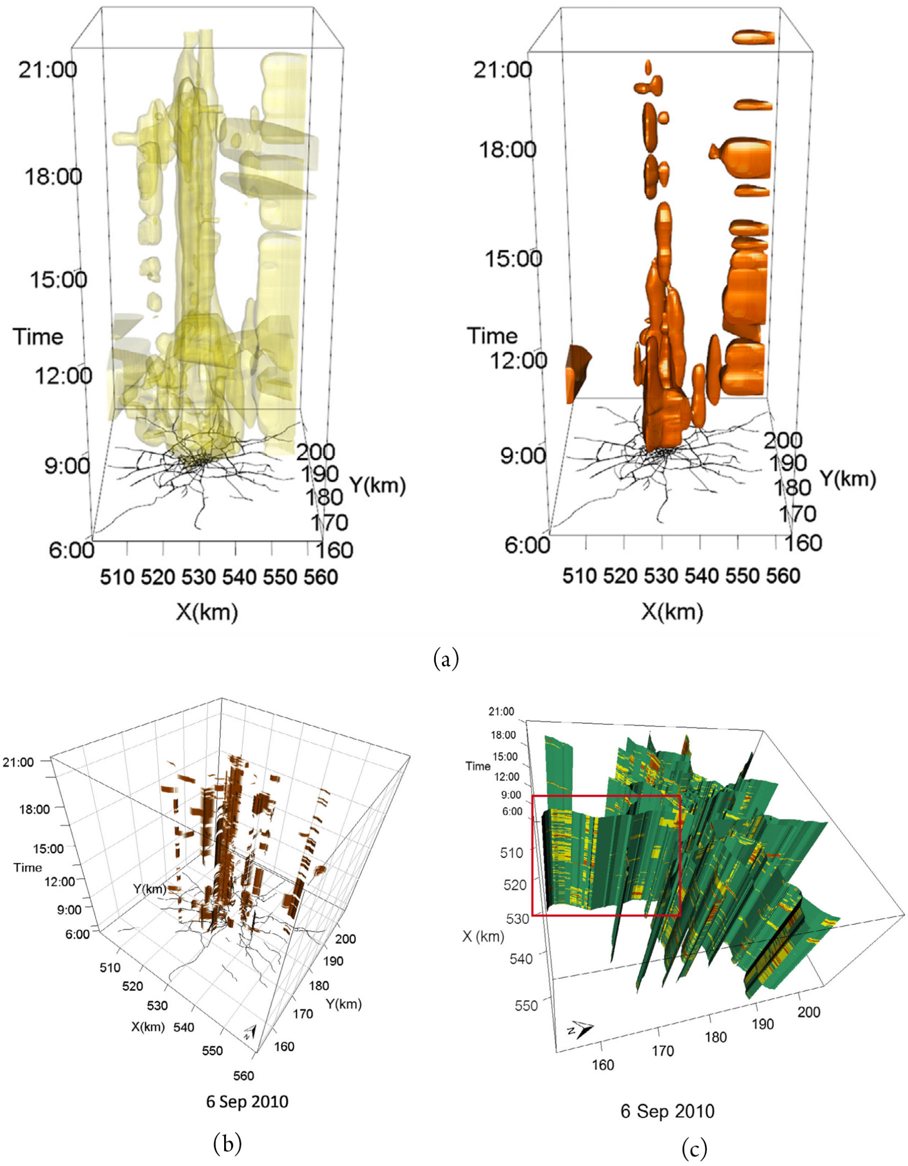


Figure 3.8: The isosurface, network-constrained isosurface, and the wall map visualization techniques [90]

Clustering techniques can be combined with categorical color scales to depict cluster membership. For instance, Andrienko et al. proposed a flow map visualization in which colors are given according to the cluster membership of the mean speeds on road links [95], as shown in Figure 3.9.

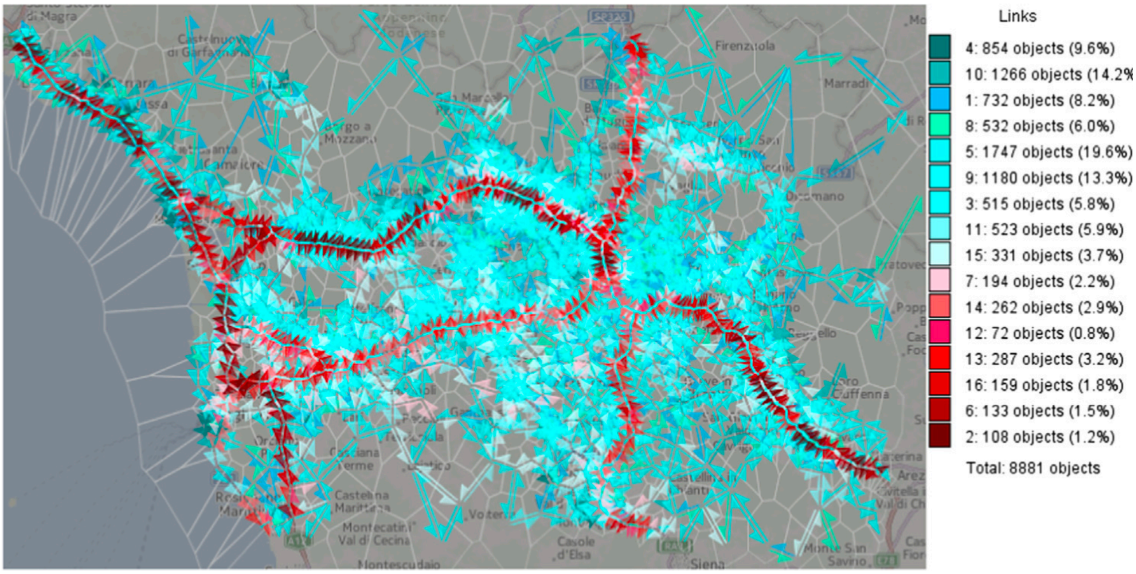


Figure 3.9: Flow map visualization color-coded in terms of mean speed in road links [95]

3.1.2 People dynamics in urban environments

The study of people dynamics has been mostly focused on detecting urban hotspots. A major data source that supports related works consist of mobile phone data [96–101]. Other data sources were socio-economic data [96], taxi GPS trajectories [102], travel diary survey data [103], model-generated OD matrices [103], vehicle sensor data [95], and microblogging data [101, 104].

Kang et al. used the space-time cube visualization to analyze aggregate mobility dynamics of people in urban settings [96]. Sagl et al. used 2D map-based and abstract visualizations for exploring mobility patterns in four Italian cities [97]. Map-based heat maps were used to estimate the spatial density of total mobility. Sparklines (see Figure 3.10) were used to analyze the temporal variation of total mobility and net migration flow on each urban center. Map-based heat maps and sparklines were used by Zuo and Zhang for the same purpose [98].

Demissie et al. analyzed cellular network handover information to test certain assumptions of mobility patterns in Lisbon, Portugal [99]. Visual exploration of data consisted of simple and effective map-based visualizations such as flow maps (Figure 3.11) and sized circles (Figure 3.12). The former was used to depict the direction and strength of the handover flow. The latter provided an effective comparison between incoming and outgoing handover on main road links.

Ferreira et al. proposed the *TaxiVis* system for exploratory visualization of taxi trips, using the city of New York, USA, as a case study [102]. The system provides multiple coordinated visualizations. For instance, map-based choroplets and heat maps are used for analyzing trip density within city regions. Line plots are combined with scatterplots and bar charts for visualizing temporal and thematic information, such as trip duration, fare amount and distance. Figure 3.13 shows the main system interface.

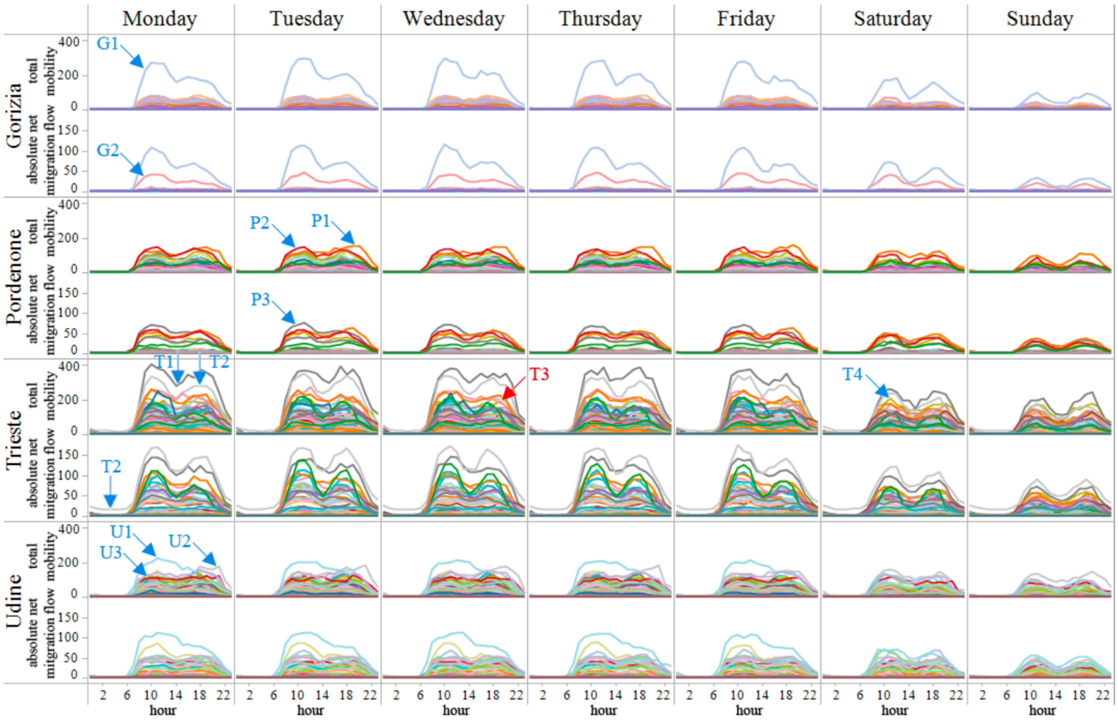


Figure 3.10: Sparklines visualization of temporal variation of total mobility and net migration flow on each urban center [97]

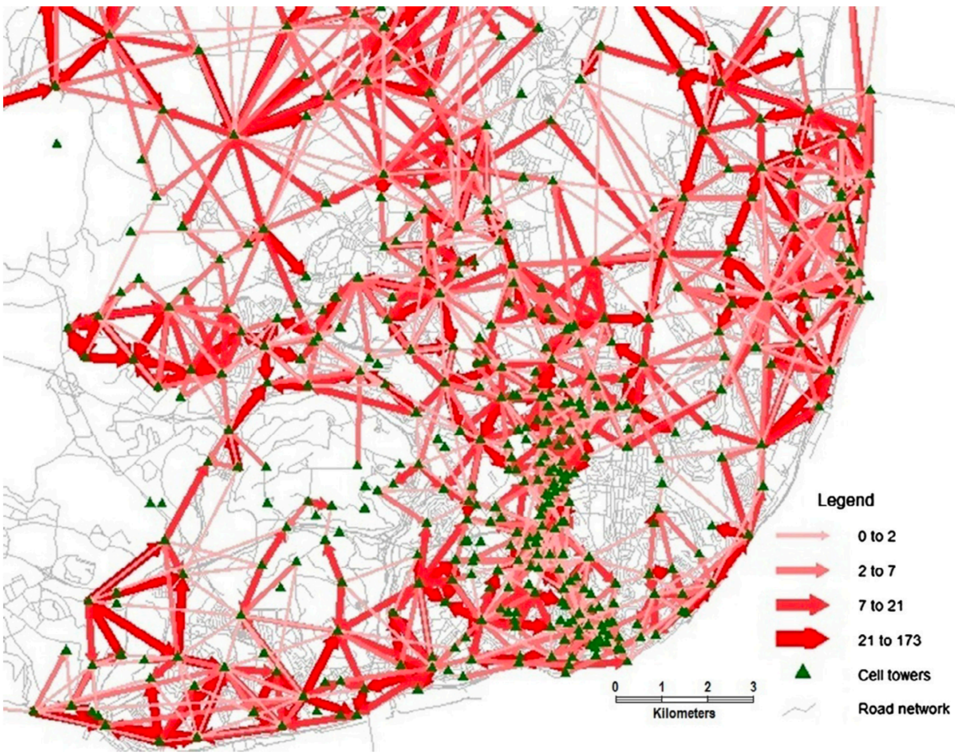


Figure 3.11: Map-based visualization of handover flows using flow maps [99]

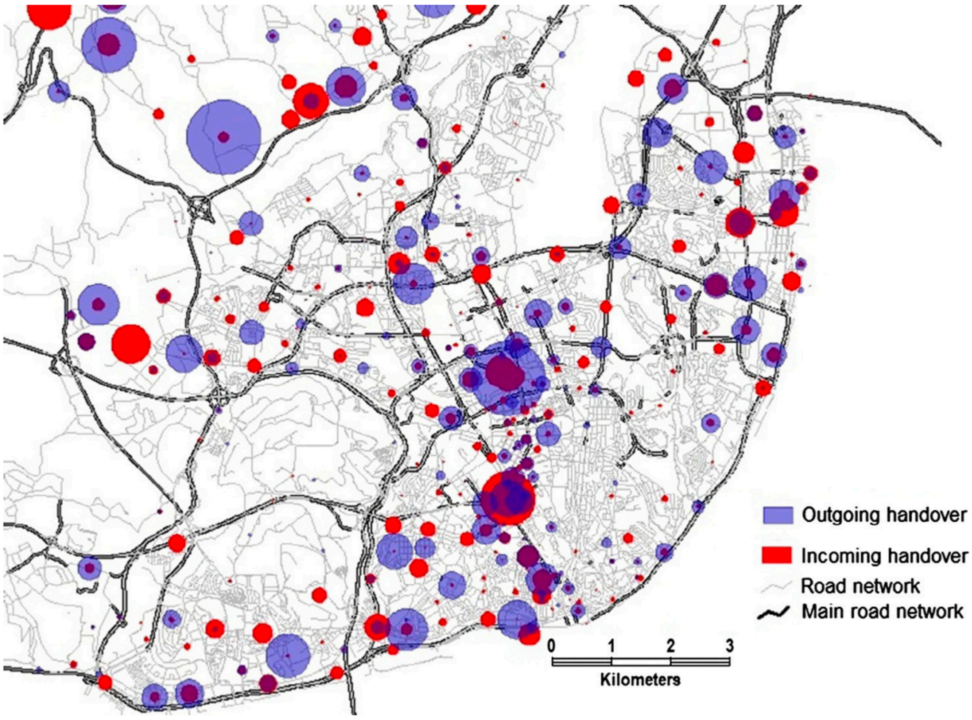


Figure 3.12: Map-based visualization of volume of incoming versus outgoing handover flows using sized circles [99]

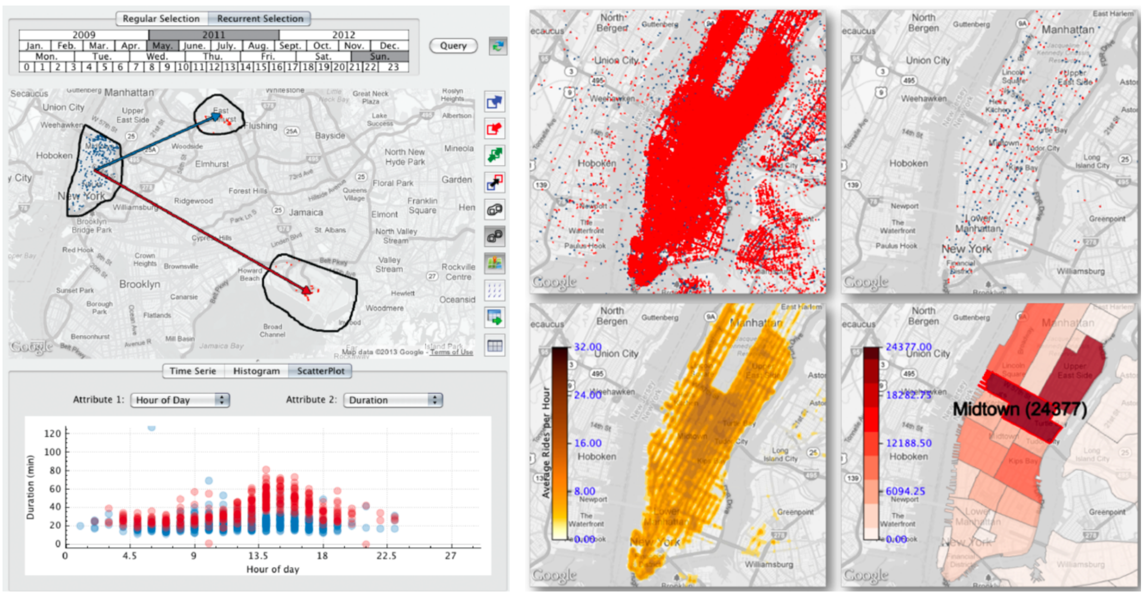


Figure 3.13: The TaxiView system for exploration of mobility patterns through taxi trips [102]

Andrienko et al. developed a visual analytics system for supporting mobility analysis from episodic data, while preserving citizen's privacy [100]. To the best of our knowledge, this work seems to be the only that is explicitly concerned with that matter. Despite being beyond the scope of this thesis, we argue that such factor should be taken into consideration in future works. Episodic data was retrieved for each individual, from which an algorithm was used to derive the most likely meaning of the places visited by users. Figure 3.14 shows an example of the semantic space map visualization technique, which represents flows between several place categories, combined with heat map matrices for each of those places. The widths and opacity of lines are proportional to the total moves between each origin-destination pair. The color scale correspond to temporal clusters of similar flows.

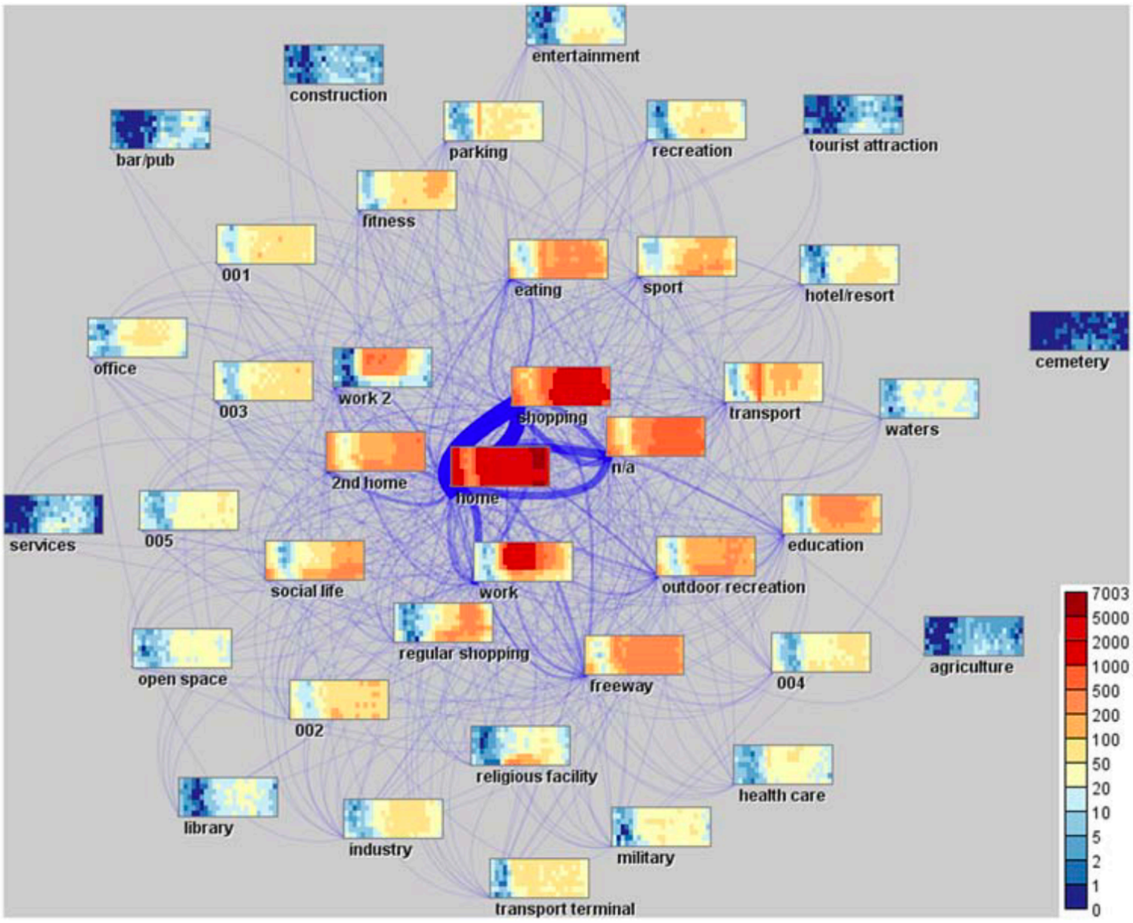


Figure 3.14: Semantic space map for visualization of mobility flows [95]

Von Landesberger et al. tackled the issue of visual clutter that may occur in flow maps [101] . By introducing spatial and temporal simplifications through cluster analysis, graph-based flow maps were combined with temporal cluster representations to give insights on regular daily and weekly patterns of the population, as shown in Figure 3.15.

Chen et al. proposed a visual analytics approach to address the shortcoming of microblogging data, which is typically sparse [104]. The system features several abstract visualization techniques, connected with brushing and linking mechanisms, which are combined with a map-based visualization for displaying aggregate spatiotemporal data (Figure 3.16b). The work makes a novel use of Sankey diagrams (Figure 3.16d) to represent pairwise movements in time. Heat map

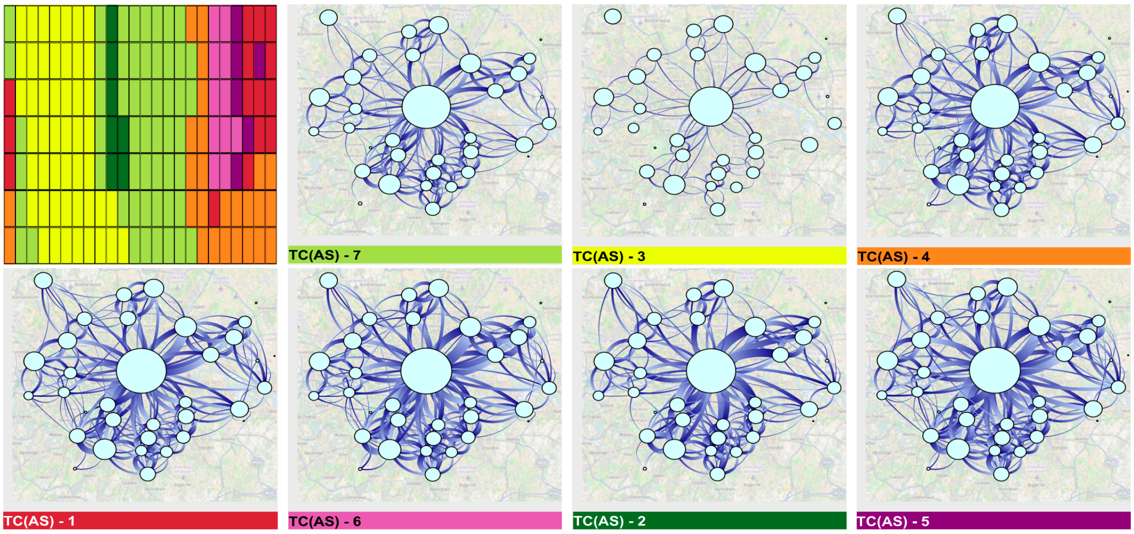


Figure 3.15: Graph-based flow maps visualization of regular mobility flows [101]

matrices (Figure 3.16c) and time plots (Figure 3.16a) were used to represent the distribution of movement in distance and time, and for temporal data filtering, respectively.

Nunes et al. [105] developed the Beanstalk platform for analysis of tourism dynamics, such as trip itineraries, based on passenger counts retrieved from activity-based data, e.g. passenger count from points of interest, and survey data retrieved from tourism authorities. The platform contains various types of interactive visualization techniques. Chord diagrams are used to display movement information between points of interest. Time-based occupancy rates in points of interest are displayed on a heat map matrix.

3.1.3 Road traffic incidents

Visualization of road traffic incidents has been supported by datasets related to car incident records [69, 106–109], and vehicle sensor data [109].

Li et al. used a 3D GIS-based visualization to represent potential crash risks on road links, by ranking and estimating segments with potential for vehicle crashes [106]. A 3D map with the road segments is overlaid with bar charts. The height of each bar represents the crash risk of a given location. Pack et al. proposed a visualization tool that combines multiple coordinated views [69]. A map-based visualization was used to display the location of each accident. A bar chart histogram showed the frequency of each accident property, e.g. fatality, injury, roadwork. Given that an accident can be related to multiple properties, parallel coordinate plots were used to explore the relationship between each property. Finally, scatterplots and heat map matrices allow exploring the pairwise relationship between variables. Map-based heat maps have also been used to represent vehicle incidents [107, 108]. Plug et al. evaluated the effectiveness of heat maps with domain experts and general public, and reported positive results [108].

Anwar et al. proposed a novel map-based visualization technique for exploration of road conditions under traffic incident conditions [109]. The *Traffic Origins* visualization, depicted in Figure 3.17, shows a red circle glyph whenever an accident occurs, and displays the road conditions in the surroundings of the accident location. After the accident, the glyph changes its color to represent the road conditions after the accident, e.g. heavy traffic or breakdowns.

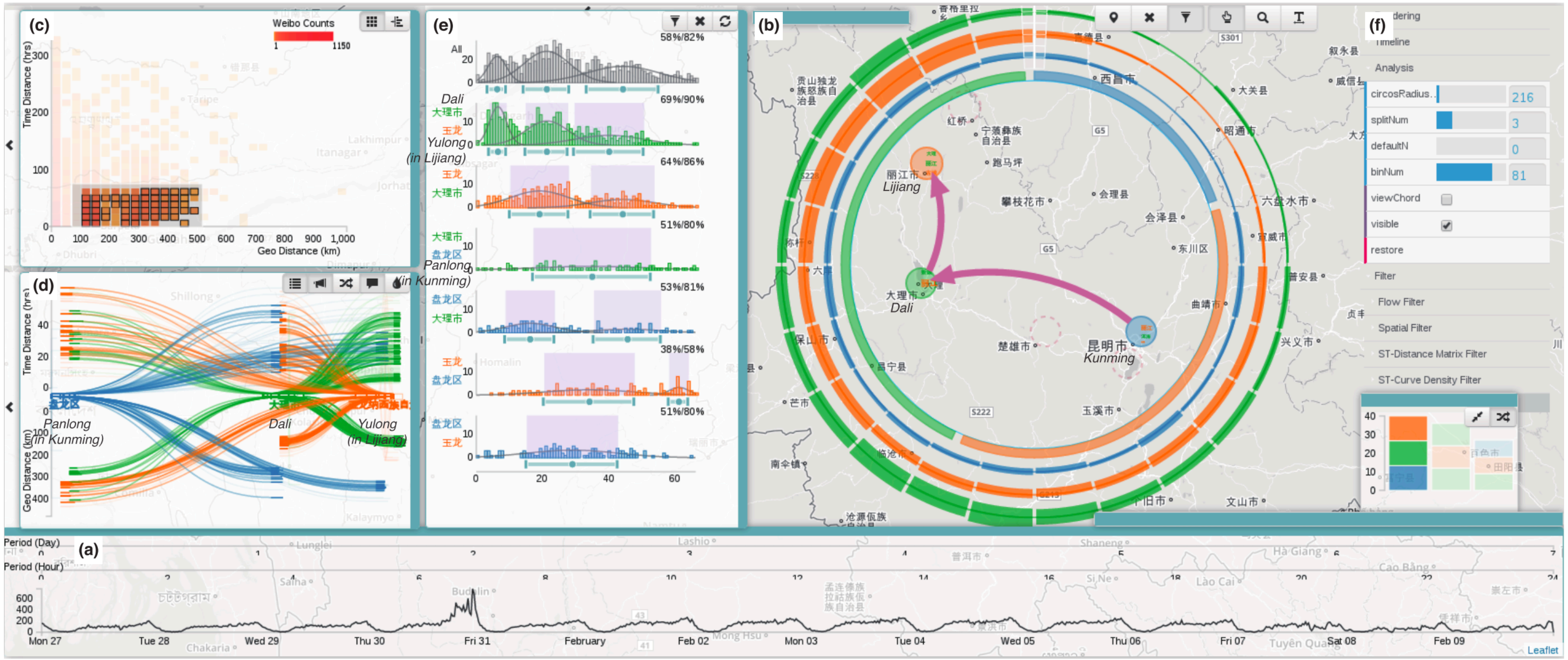


Figure 3.16: A visual analytics system for exploring sparse microblogging data. Several visualization techniques are interrelated [104]

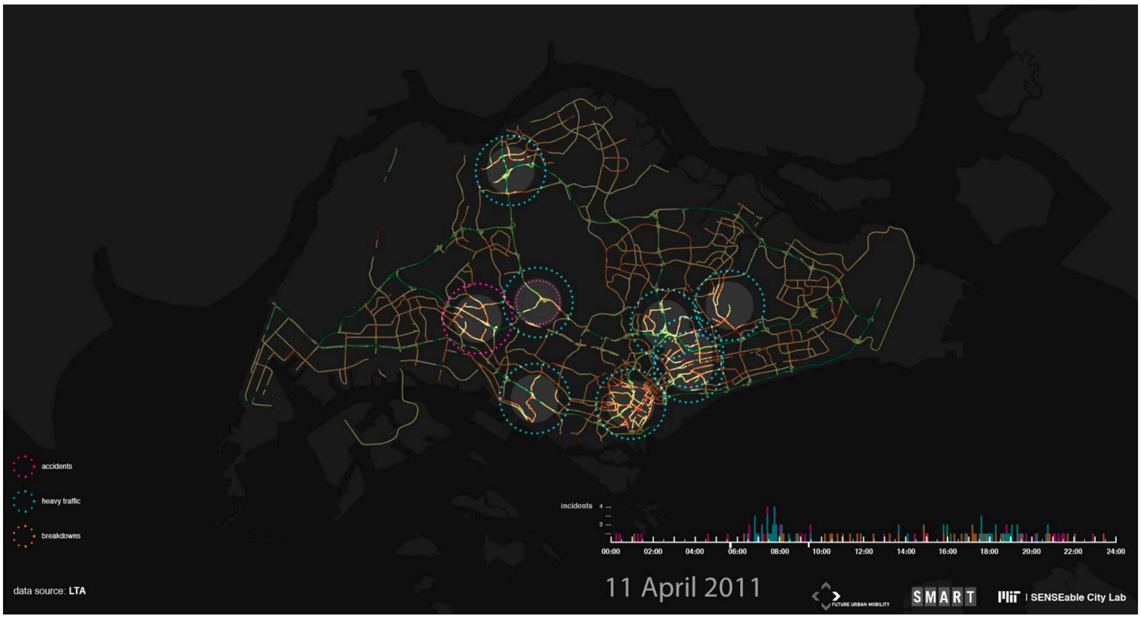


Figure 3.17: The Traffic Origins visualization technique for exploring road conditions under traffic incident conditions [109]

3.1.4 Air pollution

Visualization of air pollution uses data from vehicle sensors [110] and model-based estimations of emissions, dispersions or heat [75, 110, 111], model-based traffic flow data [75], bus AVL data and GPS trajectories [112], and video streams [113].

All surveyed studies used map-based visualizations with heat map based overlays. Rebolj et al. and Wang et al. used GIS and 3D maps in combination with bar charts to identify road links with high air pollution levels [75, 110]. Li et al. proposed a web-based visualization system for visualizing emissions of diesel buses on a microscopic scale, i.e. bus route segments [112]. Heat maps are applied to road segments to indicate the emissions rate along several bus routes. Morris et al. used video stream data to estimate traffic flows and emissions on highway segments [113]. The authors used simple yet effective representations of line plots to depict the evolution of emissions of pollutants over time.

Cristie et al. proposed an interactive visualization tool, *CityHeat*, for cellular automata based simulation and analysis of traffic heat in microscopic scale [111]. The tool, as shown in Figure 3.18 provides interaction tasks such as pan and zoom, filtering, and temporal querying. Heat cubes represent represent the temperatures of road sections according to simulated traffic intensity and vehicle types.

3.1.5 Travel behavior on PTS

Data for visualization of travel behavior on public transportation systems was retrieved from smart cards (AFC) [114–117], socioeconomic and travel diary surveys [118], traveler information systems and vehicle sensors [4].

Fuse et al. used smart card data from a Japanese city to analyze travel behavior under certain weather conditions [114]. Line plots and bar charts were effectively used simultaneously to analyze passenger ridership and precipitation amount. Aggregate time series data was repre-

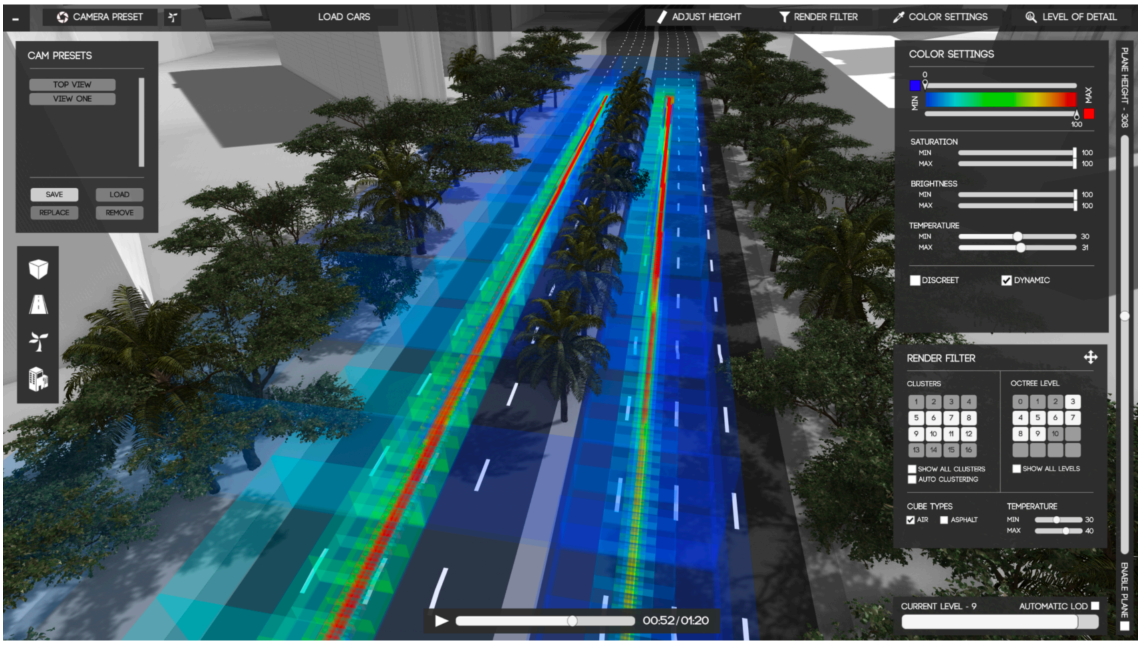


Figure 3.18: CityHeat visualization tool for microscopic simulation and analysis of traffic heat on a three-dimensional virtual city environment[111]

sented using stacked bar charts for analysis of public transportation use frequency under different weather conditions, e.g. sunny or rainy, and day type, e.g. weekday or holidays. Kamruzzaman used bar charts to analyze temporal trip patterns of students, and a GIS-based 3D map visualization to identify activity hotspots within the city, using heat surface maps [118]. Roux et al. provided a simple implementation of map-based heat maps to analyze passenger flows [115].

Tao et al. introduced the use of flow-comaps to visualize aggregate flow patterns of passengers at a network level [116]. Such technique proved to be useful to identify the major flows of bus passengers over a time period. Flow-comaps combine flow maps with conditional plots (see Figure 3.19. Once again, map-based heat maps and line plots were identified for spatial and temporal analysis of passenger flow patterns, respectively.

Zeng et al. stated that visualization techniques tend to focus on the network topology across stops, ignoring mobility factors such as riding and waiting times. They proposed three visualization techniques for tackling this gap, focusing on a variety of time-related factors that impact mobility in public transportation systems [117]. Such techniques are discussed in the following paragraphs.

The isochrone map-based visualization (see Figure 3.20) depicts a reachable spatial region within a given timespan. In this particular case, a bus station is chosen as a starting point. Dark and light blue represent a timespan of [0,30] and (30,60] minutes, respectively.

Figure 3.21 exemplifies the isotime flow map, which linearizes a flow map in a parallel isotime. It is possible to visualize the time efficiency of journeys that start at a certain stop (red circle on the left side of the picture), which is calculated in terms of standard deviations of the mean travel time. Each small node corresponds to a bus stop. The OD-pair journey view (see Figure 3.22) is based on the isotime visualization technique. Given an origin and destination, it is possible to visualize the transfer and waiting times, as well as round-the-clock variations with the mobility wheel glyph, which is used to encode such temporal information.

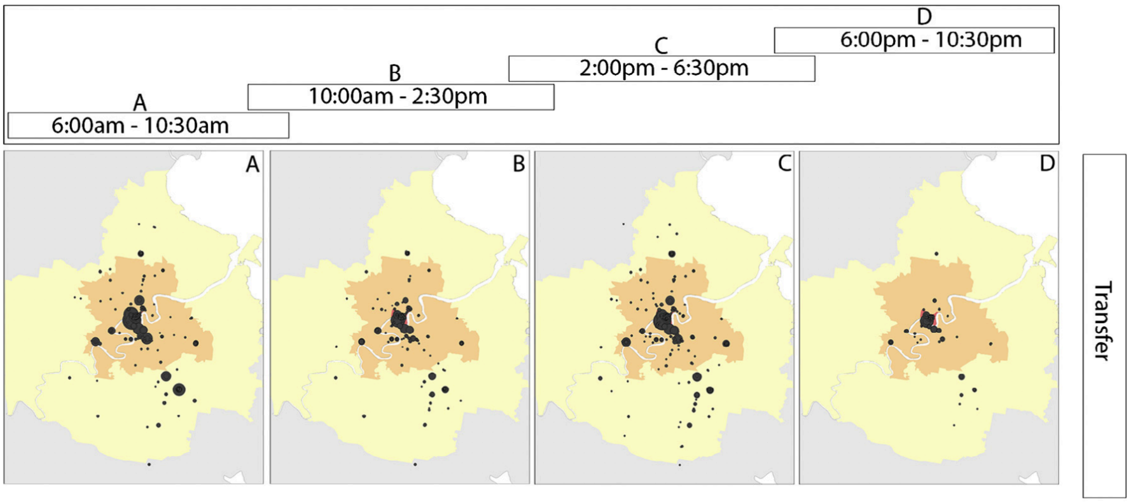


Figure 3.19: The flow-comap visualization technique for exploration of spatiotemporal mobility patterns [116]

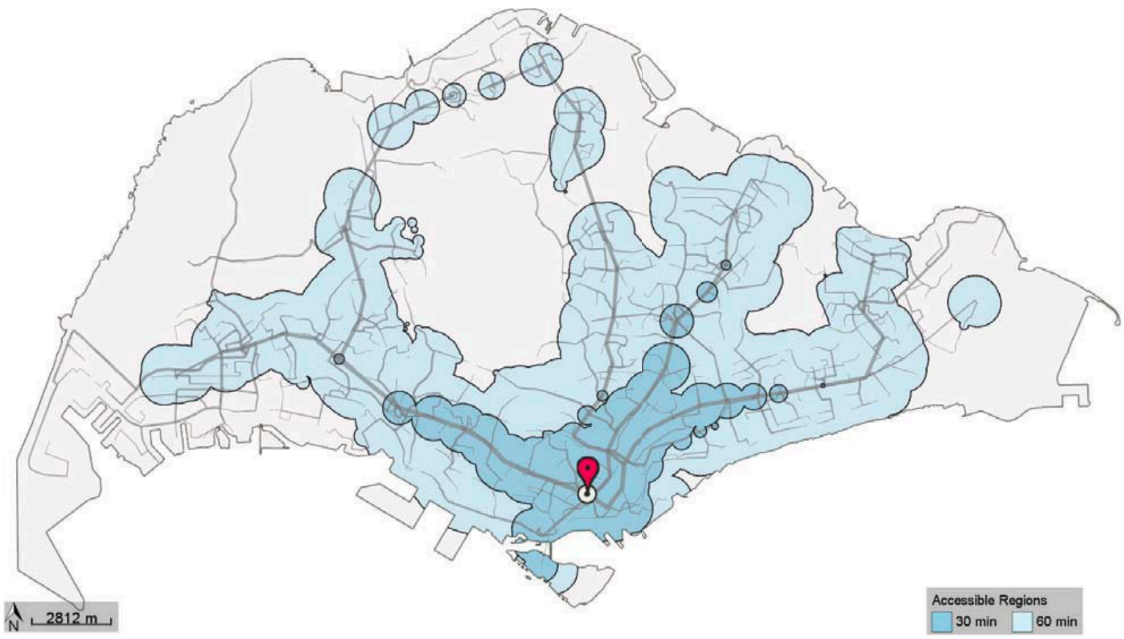


Figure 3.20: The isochrone visualization technique for exploration of reachability of a spatial region within a timespan [117]

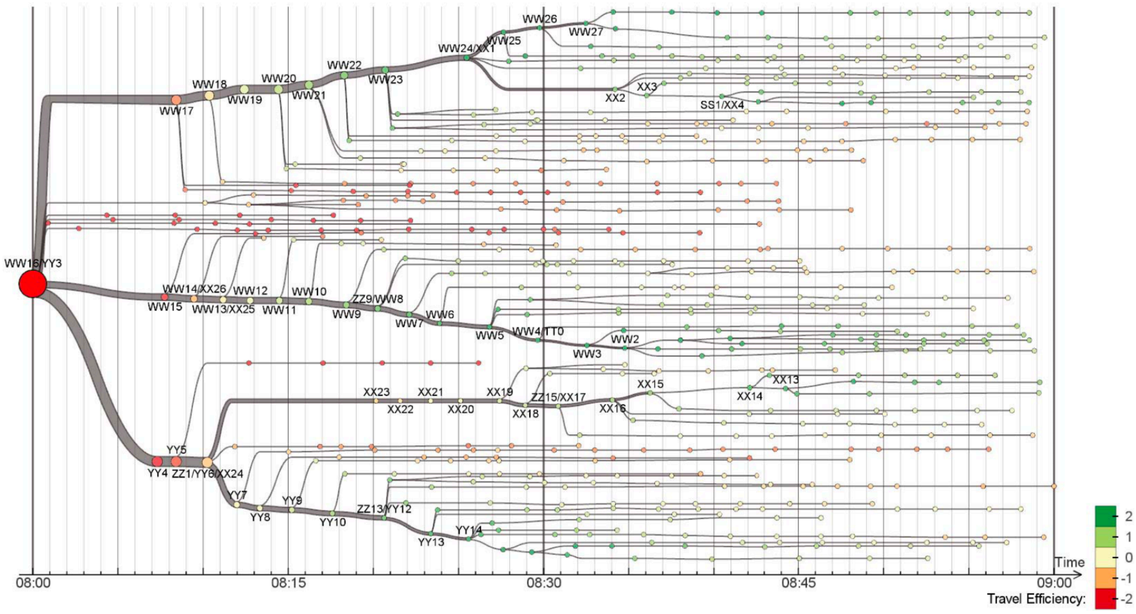


Figure 3.21: The isotime visualization technique for exploration of time efficiency of journeys based on a starting stop [117]

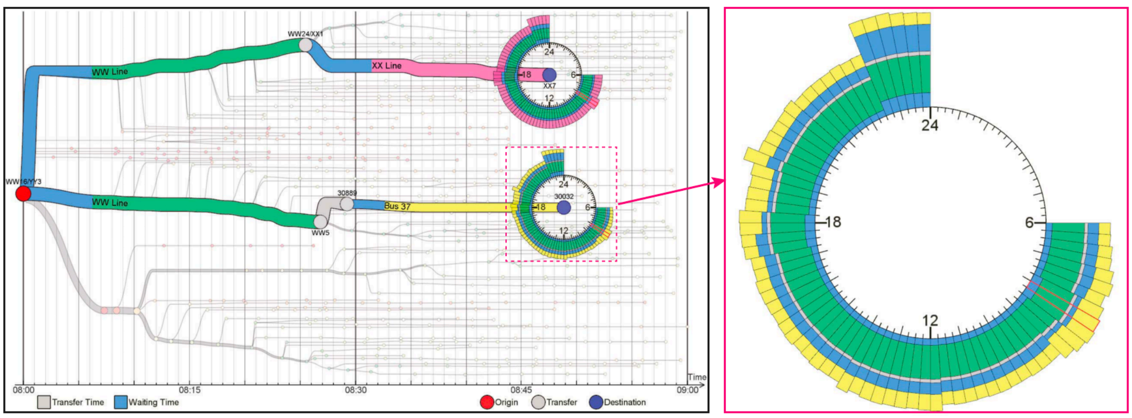


Figure 3.22: The OD-pair journey view uses the isotime visualization technique to analyze different route options for the same OD-pair [117]

3.1.6 Level of Service on PTS

Visualization of level of service on public transportation systems uses data from transit reports [119], tram AVL data [120, 121], subway AVL and schedule data [122].

Yu et al. used simple GIS-based map visualization to analyze bus schedule adherence, comparing static and realtime data for a set of stops [119]. Currie et al. and Mesbah et al. proposed a methodology for mining tram AVL data to support reliability analysis (actual versus scheduled travel times), and trend analysis of reliability [120, 121]. The resulting data was visualized with geographic heat maps.

Palomo et al. proposed an effective visualization tool, *TR-EX*, for transportation schedules [122]. The tool uses kernel density estimation techniques, and allows users to compare planned timetables against real service, to analyze speed profile at route segments level, and to assess delay, wait time and reliability at station level. The top of Figure 3.23 shows the visualization of average speed for inbound (left) and outbound trips. The bottom shows the visualization at stops level. Warmer colors indicate higher average delay.

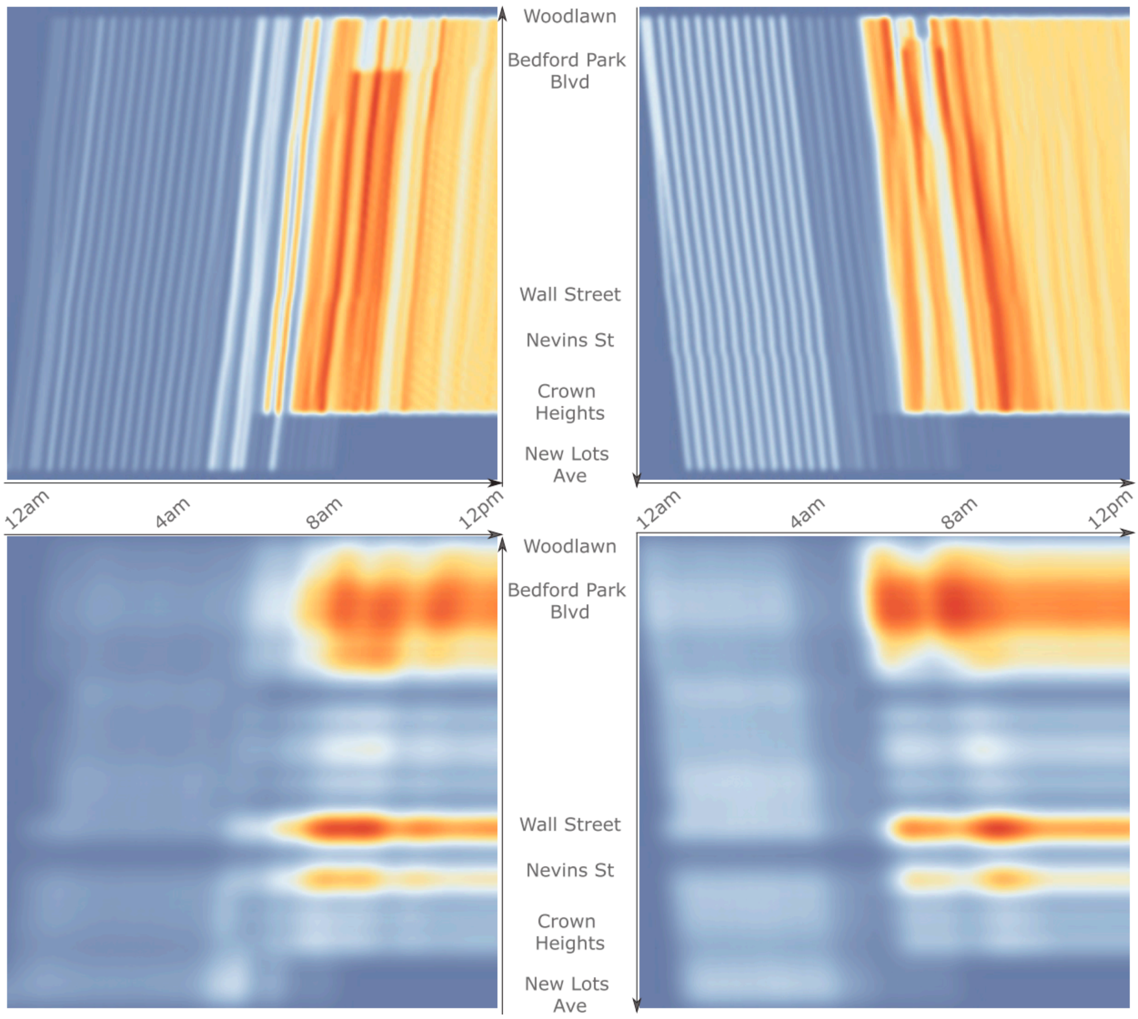


Figure 3.23: The TR-EX visualization system for exploration of transportation schedules at trips and stops levels. Trips visualization is shown on the top frames, while stops visualization is shown on the bottom [122]

3.1.7 Trip patterns

The few studies regarding analysis and visualization of trip patterns make use of taxi trajectory data, hence all of them are related to taxi trips [102, 123–125]. The *TaxiVis* visualization system of Ferreira et al. has already been featured in the topic People Dynamics in Urban Environments.

Liu et al. used geographic heat maps to analyze the spatial distribution of pick up and drop off points [123]. Mao et al. used GIS-based visualizations to analyze spatiotemporal trip patterns [125]. Map-based techniques such as choroplets and flow maps were used to analyze travel density and connectivity.

Chu et al. proposed a novel approach to trip patterns analysis [124]. Spatiotemporal information is transformed into contextual semantic information, which is used to drive hidden themes, named as taxi topics by the authors. Each topic, generally the name of a street or avenue, is related to a certain pattern. The visualization system that supports such analysis provides multiple coordinated visualization techniques, as in Figure 3.24. In (a), topics are represented on a map. Word clouds featuring street names are used in (b) along with sparklines to depict the representativeness of each street on a topic. Parallel coordinates view (c) is used to explore the relationship between topics. Temporal relationship between topics can be explored in (d).

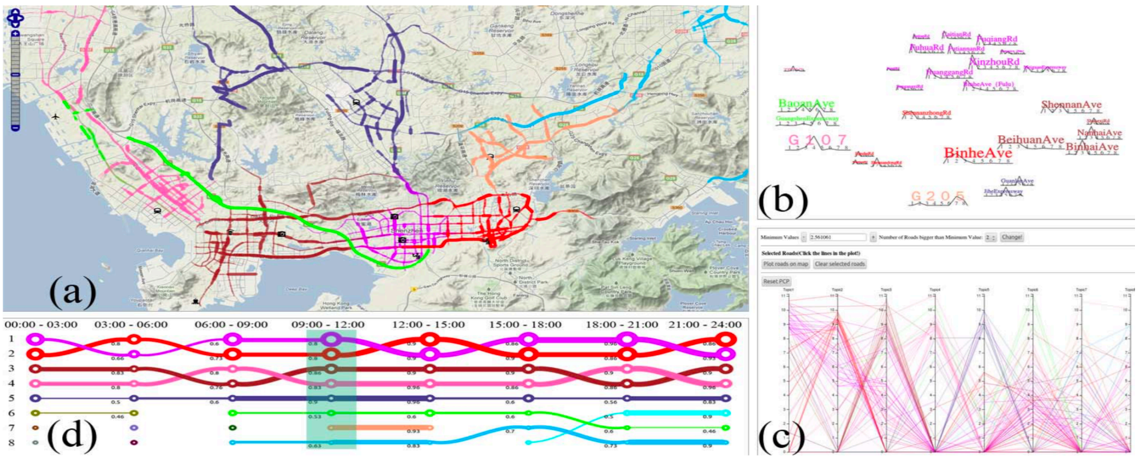


Figure 3.24: Multiple coordinated visualizations for analyzing hidden themes of taxi trip patterns, which are defined by the authors as "taxi topics" [124]

3.1.8 Other topics

In this section we discuss topics for which we have found three or less studies. Some of them may suggest future exploration by other researchers.

Some works proposed the concept of *big city data*, i.e. data from several systems for the purpose of gaining an holistic perspective of the dynamics of a city. In 2012, Corral-Soto proposed the *3DTown* system for real-time integration and visualization of 3D urban models, video streams, sensors and several real-time information sources [126]. Visualization techniques are mostly GIS-based to depict building and vehicle 3D models. Pedestrian tracking is also represented on maps using 2D glyphs and heat maps for analyzing pedestrian density. Lv et al. and Li et al. proposed a web browser-based VRGIS focused on 3D visualization of city dynamics [127, 128]. 3D building models are also used to facilitate the identification of the main city points, although several visualization techniques were combined for displaying different types of data. For instance, passenger flows on PTS stops were represented with 3D bar charts and overlaid 2D heat maps. Video stream data was also overlaid on the 3D map. Bar and pie charts were used to visualize socio-economic information. Line plots were used to show temporal information about passenger flows. Both studies conducted usability evaluation with domain users, although there was no description of adopted evaluation protocols.

Visualization of travel demand used data from mobile phone records [129], socio-economic and travel survey records [129], and taxi GPS trajectories [130, 131]. Toole et al. proposed a model for travel demand estimation and proposed an interactive visualization platform for engaging transportation stakeholders [129]. The tool shown in Figure 3.25 uses a map-based visualization for characterizing city regions that are attract (blue) and generate (red) trips.

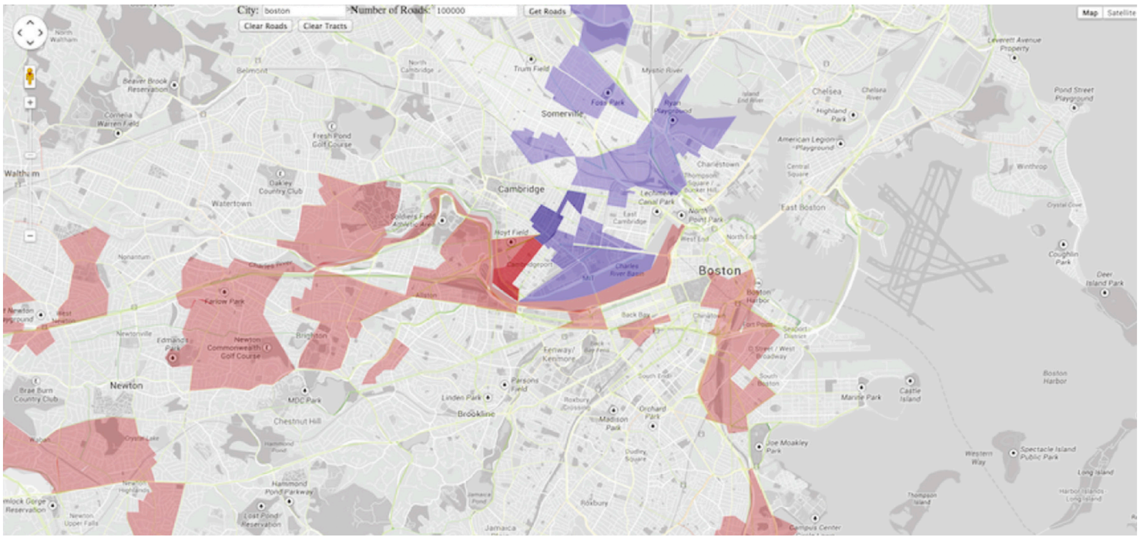


Figure 3.25: Map-based visualization of travel demand. City regions that attract trips are encoded in blue. Regions that generate trips are encoded in red [129]

Lu et al. proposed a novel visualization technique for exploring origin-destination patterns [130, 131], which was positively evaluated by domain experts. The technique was evaluated with taxi trajectory data, although it can be used for general trajectory data. The OD-Wheel features a linear and circular component. Origin and destination clusters are sorted in descending order according to traffic volume. The traffic volume within each cluster is shown with bar charts. The temporal axis is preserved for both linear and circular representations. Travel time can also be identified in the linear axis.

Commuting efficiency has been explored by Dewulf et al. using floating car and simulated travel demand [132]. The authors used map-based choroplets to visualize average time differences in commuting time during peak and off-peak hours.

Visualization accessibility measures have been identified in the works of Yin et al., and Stewart and Zegras [133, 134]. Both works aimed to identify what activities could be reached by city residents within a given timespan, and spatial (in)equities in terms of transportation availability. Data sources included land use data and transit data such as GTFS schedule data. Map-based visualizations were used to represent isochrones. Yin et al. used heat maps to represent travel time and choroplets to represent accessibility indexes [133], as shown in Figure 3.27. Stewart et al. proposed used polygon-based isochrones combined with bar charts to show access to job opportunities [134]. The interactive tool, CoAXs, allows stakeholders to compare two distinct

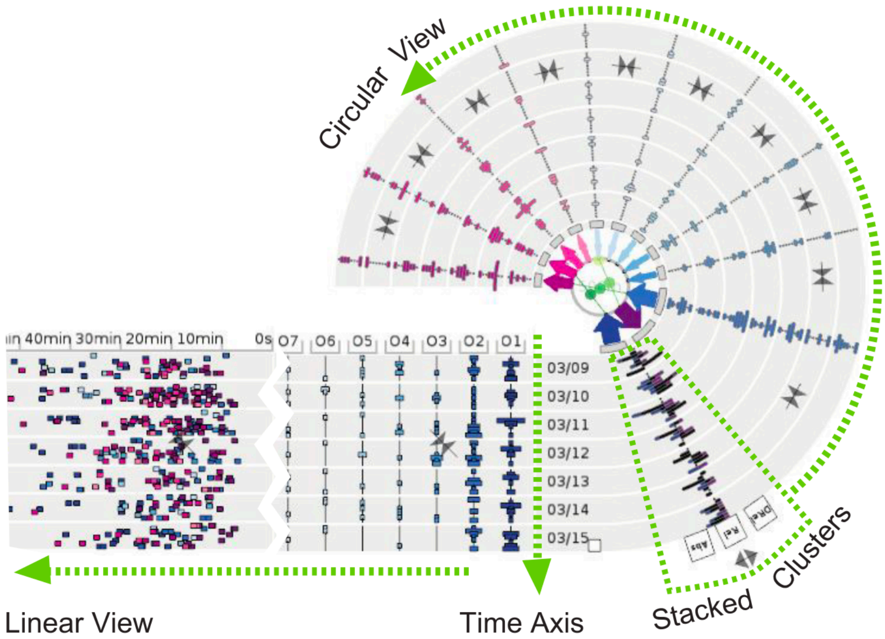


Figure 3.26: The OD-Wheel technique for exploring origin-destination patterns from trajectory data [130, 131]

transportation network scenarios (see Figure 3.28). It is possible to change route parameters, e.g. trajectory, headway and number of buses. In both works, it was possible to identify extensive use of stakeholders on evaluation studies, as both tools are concerned on engaging them on discussion about accessibility.

Visualization of PTS ridership has been explored in two studies. Data sources included non-APC passenger counts [135] and smart card data from AFC systems [136]. Polisciuc et al. [135] implemented the metaballs visualization technique for the analysis of anomalies on the number of passengers on bus stops, i.e. with significant deviations from the average number of passengers throughout the day. Metaballs were implemented in two ways, as shown in Figure 3.29: point-based metaballs provide a clear, although exaggerated view of stops with anomalies. Vertex-based metaballs preserves visibility of road network segments, and still allows the identification of areas in which such anomalies occur.

Du et al. used a combination of map-based and abstract technique for analyzing ridership [136]. An abstract calendar visualization was used to show ridership levels for each stop throughout a year. Brushing and linking interactions allow the user to select a specific day and visualize bar charts that show hourly information about ridership for a day. The authors evaluated the system with domain experts.

Cyclists' trip patterns can also reveal interesting mobility dynamics. The topic was studied by Wood et al., and Romanillos and Austwick [137, 138]. Wood et al. proposed a visual analytics system (see Figure 3.30) featuring a combination of coordinated map-based visualization of movements, bar and line plots to represent temporal and thematic information of trips, e.g. trip type, cyclist gender, etc [137]. Domain users were involved during the design and evaluation of the visualization system. Romanillos and Austwick proposed an online visualization platform for visualization of bicycle routes [138].

The problem of visualizing sparse trajectory data has been addressed by Wang et al. and Chen

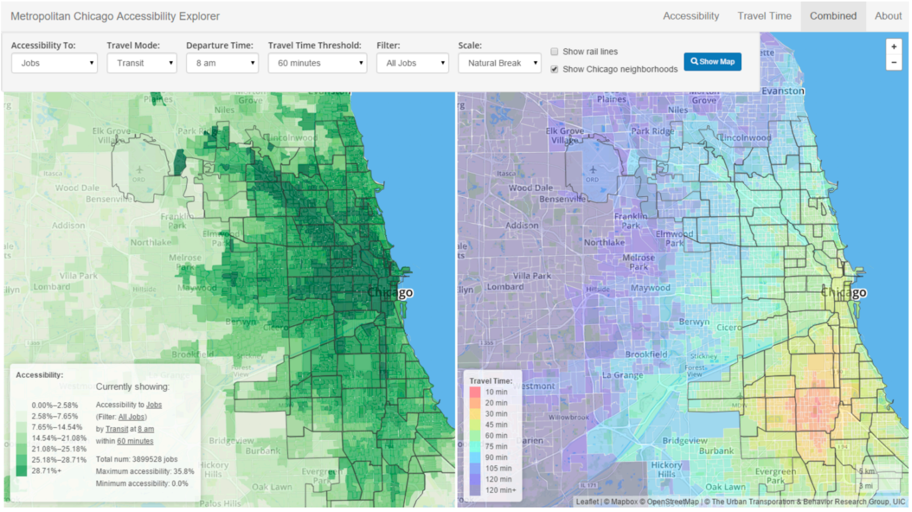


Figure 3.27: Use of geographic choroplets and heat maps to represent accessibility indexes and travel times [133]

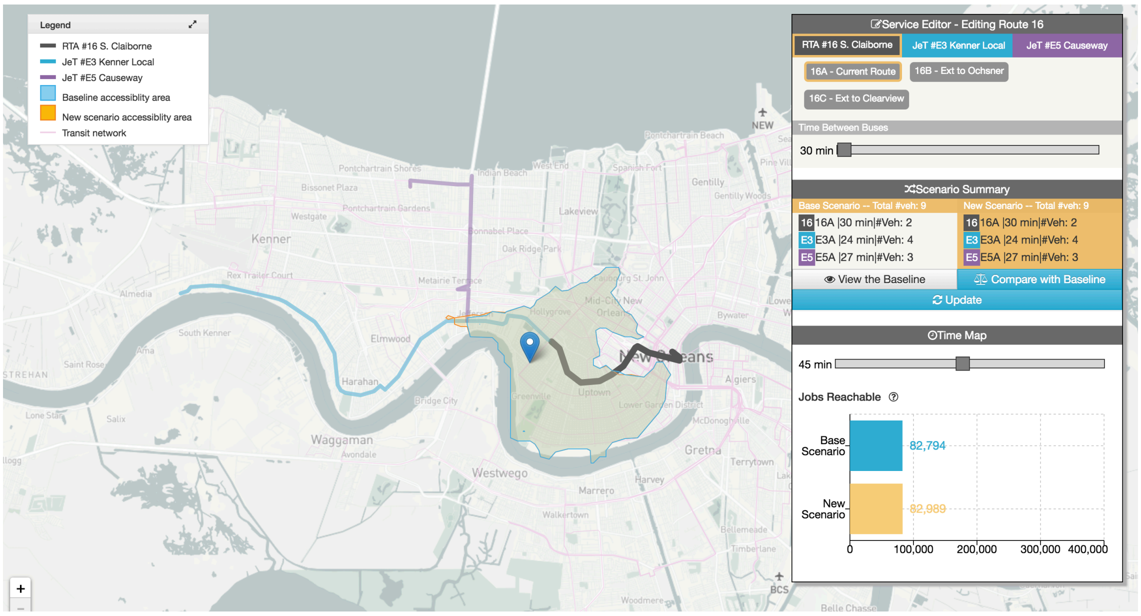
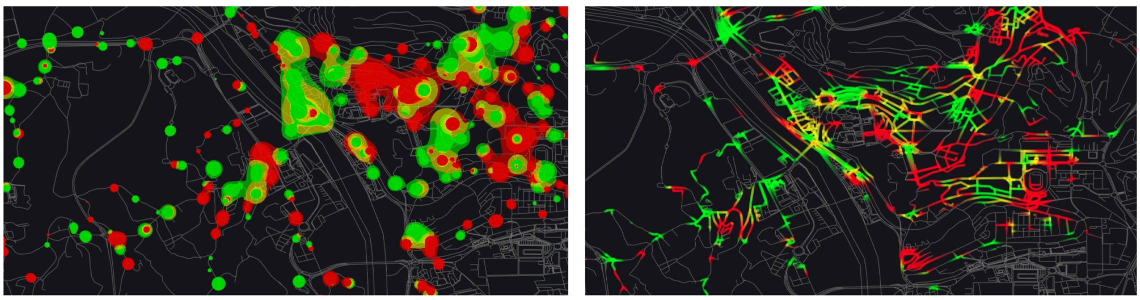


Figure 3.28: The CoAXs visualization tool for accessibility-based stakeholder engagement [134]



(a) (b)

Figure 3.29: Point and vertex-based metaballs visualization technique for visualizing anomalies on bus stops [135]

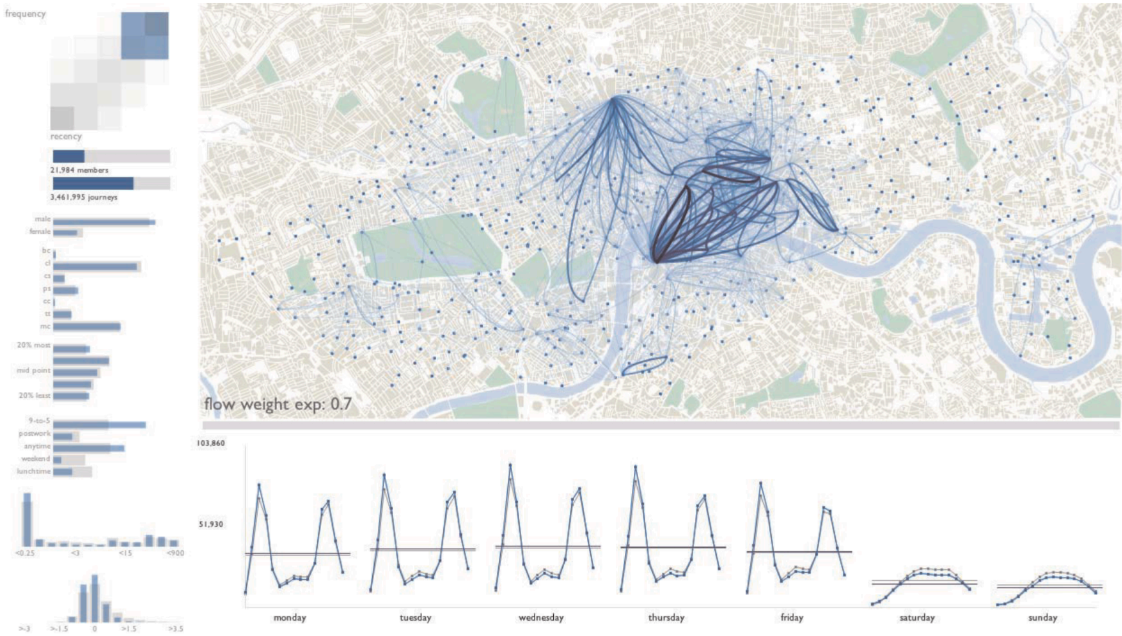


Figure 3.30: A visual analytics system for exploring cyclists' trip patterns in London [137]

et al [104, 139]. The latter has already been addressed in the topic People Dynamics in Urban Environments. Wang et al. proposed a visual analytics system based on video stream and vehicle sensors data from ANPR systems [139]. A map-based visualization provides the location of vehicle sensors and road links.

VanDaniker proposed the abstract Spiral Graph visualization for temporal transportation data [140]. The visualization tries to overcome the limitations of representing time on linear axes, such as scatter and line plots. Data is plotted on a circular temporal axis, which spirals outward at regular intervals. Figure 3.31 shows a prototypical visualization tool for collision data. It is possible to visualize the duration of a specific event throughout the circular axis.

Wu et al. proposed a prototypical visualization tool for exploring conversations about traffic using microblogging data [141], with focus on sentiment analysis and trending topics. The system provides abstract visualization techniques mostly based on variations of word clouds. The prototype was evaluated with potential users.

Zeng et al. proposed a change to the chord diagrams technique, to visualize interchange patterns in junction nodes, in order to reduce visual clutter (see Figure 3.32) [142]. Smart card data from AFC systems was used to support visualization development. Frame (a) depicts the original version of the Circos diagram. The junction node is represented as a ring on frame (b), and ribbons are bundled on frame (c), thus reducing visual clutter. Finally, additional statistics such as outgoing and incoming flow are added to facilitate analysis.

Krüger et al. proposed an interactive visualization system, *TrajectoryLens* for exploring long-term trajectory data [143]. The following interaction tasks are available: focus plus context, dynamic queries and filtering. The system provides multiple coordinate visualization techniques, including a map-based visualization of trajectories, which can be filtered and aggregated by the user. Hierarchical time sliders allow users to filter trajectories according to the desired timespan.

Polisciuc et al. proposed an interesting approach to visualize clusters of points of interest using polygons [144]. As shown in Figure 3.33, polygons are also used as containers for textual information. The authors used POI data from the city of Boston, USA.

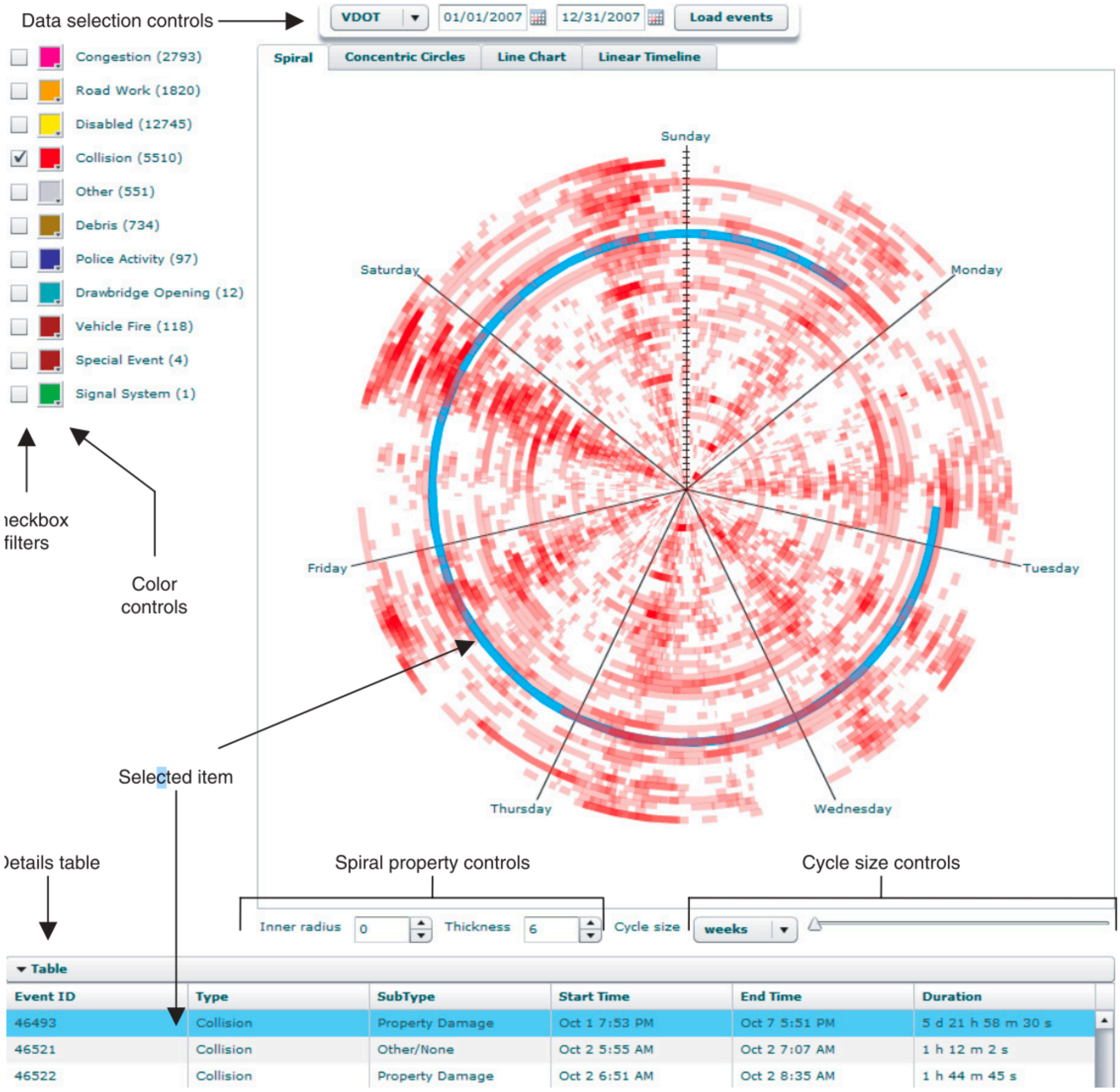


Figure 3.31: The Spiral Graphs tool for temporal transportation data [140]

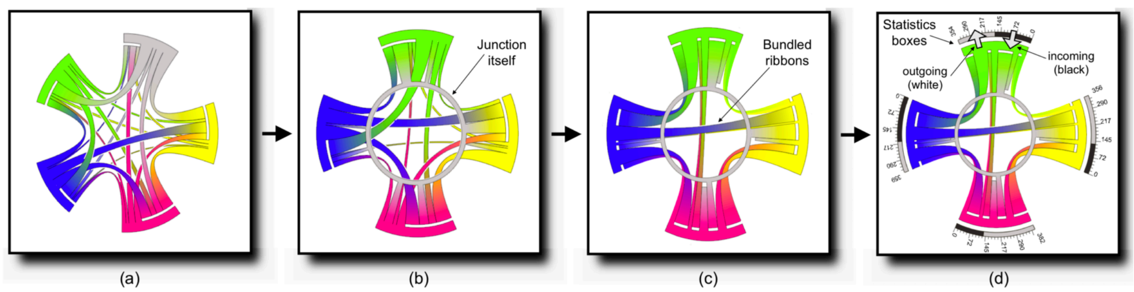


Figure 3.32: The interchange Circos diagram for visualizing interchange patterns in juction nodes [142]

Figure 3.33: Visualization of POI clusters as polygons [144]

Wu et al. proposed the *TelCoVis* visual analytics system for analyzing co-occurrence, i.e. when individuals from two regions visit an urban place during the same timespan. The authors used mobile phone data for supporting the analysis. Similar to other works shown in this section, the system presents multiple coordinated visualization techniques, as shown in Figure 3.34, two maps are used to show heat maps for analysis of incoming and outgoing mobility flows (a,b). Abstract visualizations such as matrix heat maps (c,f), contour-based tree map (d) and parallel coordinate plots (e) provide additional information about clusters and their correlations. Domain experts were interviewed for system evaluation.

3.2 Ontologies

The relevance of ontologies to transportation has been acknowledged by the research community. Applications can be found on topics such as, for example, urban planning, transportation network modeling, road traffic management, and content personalization for travelers. As in any model, the main challenge of ontology modeling is to simplify reality while ensuring validity and practical feasibility, from a computational perspective. It is acknowledged that most studies focus on integrating information from various sources [148, 149]. Furthermore, Valle et al. highlighted some benefits of ontologies and Semantic Web technologies for transportation research, in particular for Intelligent Transportation Systems [149]:

- A fraction of urban-related data is natively managed by Geographic Information Systems;
- Need for information integration from various heterogeneous sources.

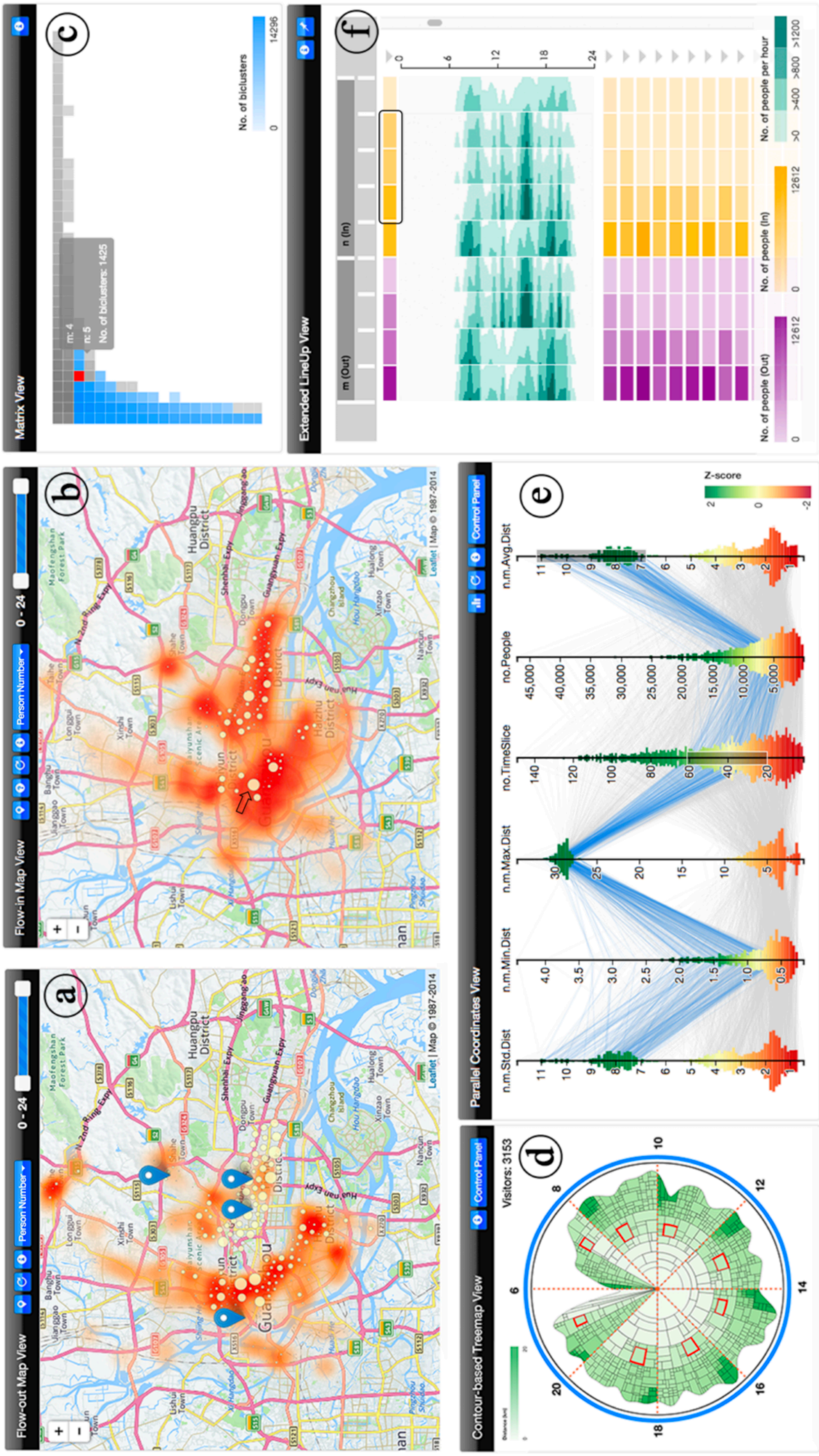


Figure 3.34: The TelCoVis visualization system for exploring co-occurrence [145]

Table 3.1: Topics of surveyed studies and their representatives

Topic	Representative studies
Urban traffic flows and monitoring	[73–77, 79–95]
People dynamics in Urban Environments	[95–104]
Road traffic incidents	[69, 106–109]
Air pollution	[75, 110–113]
Travel behavior on PTS	[114–118]
Level of Service on PTS	[119–122]
Trip patterns	[102, 123–125]
Big city data	[126–128]
Travel demand	[129–131]
Ridership	[135, 136]
Sparse trajectory data	[104, 139]
Cyclist behavior	[137, 138]
Temporal transportation data	[140, 143]
Commuting efficiency	[132]
Accessibility	[133]
Urban traffic conversations	[141]
Interchange patterns	[142]
Land use analysis	[144]
Co-occurrence	[145]

Table 3.2: Data types found in surveyed studies and their representatives

Group	Subgroup	Data type	Representative studies
Sensor	Activity-based	Floating car data	[132]
		Mobile phone data	[96–101, 129, 145]
		Smart card data (AFC)	[114–117, 128, 136, 142]
	Device-based	Bicycle trajectories data	[137, 138]
		Bus AVL data	[79, 112]
		Bus GPS Trajectories	[112]
		Vehicle sensor data	[74, 76, 77, 83, 85, 89, 91, 95, 109, 110, 126–128, 140, 146]
		Non-APC Passenger count data	[105, 135]
		Taxi GPS trajectories	[81, 82, 86, 87, 92, 94, 102, 123–125, 128, 130, 131]
		Subway AVL data	[122]
		Tram AVL data	[79, 120, 121]
	Location-based	Vehicle GPS trajectories	[84, 143]
		Video stream data (incl. ANPR)	[90, 93, 113, 126, 139]
Others	Survey-based	Household survey data	[127]
		Land use data	[133, 134, 144]
		Points of interest data	[144]
		Socio-economic data	[96, 118, 129]
		Travel diary survey data	[103, 105, 118, 129]
	Report-based	Car incident record data	[69, 106–109, 140]
		Transit data	[119]
		Schedule data	[112, 122, 133, 134]
	Social networks	Microblogging data	[91, 101, 104, 141]
	Model-based	Highway traffic flow data	[80]
		Origin-destination matrices (travel demand)	[103, 132]
		Urban traffic flow data (network capacity, travel times, etc.)	[73, 75, 88, 95, 147]
		Road traffic air pollution (emission and dispersion) or heat	[75, 110, 111]
	CAD-based	3D Building data	[126–128]

In this section, we address eleven studies that proposed either domain or foundational ontologies for transportation, with focus on urban mobility. In addition, we restrict the inclusion criteria to studies that formalize ontologies in OWL, which is the current standard for ontology modeling. We found that no ontologies directly address the issue of visualization for transportation. In addition, further research is required on semantic modeling of spatiotemporal urban mobility data, as only two studies could be identified [150, 151].

A recent literature review surveyed studies on various facets of the application of ontologies to transportation research, which highlights the relatively small number of robust studies that effectively address the use of ontologies in transportation [148]. The authors concluded that ontologies cover various concepts of the transportation domain, and there is no single ontology that attempts to provide a unified conceptualization. Furthermore, the authors suggest, as future research directions, works that attempt to semantically align the various proposed ontologies.

The Ontology for Transportation Networks (OTN) provides an extension and formalization to the Geographic Data Files (GDF) standard [150]. In essence, the ontology describes a transportation network using concepts from different levels, i.e. edges and nodes, as well as aggregate concepts such as roads and routes. However, most concepts are not sufficient to address spatiotemporal urban mobility data. The support to temporal references is also limited.

The Towntology project developed a software for construction and visualization of a semantic network of concepts [152]. The software uses various lightweight ontologies, hence it is not able to infer new knowledge from linked data. Concepts can be explored through the Towntology browser (see Figure 3.35). The software also allows images to be annotated with concepts; the project applied the software to the construction of some lightweight ontologies, i.e. vocabularies, related to road system description, urban renewal, and urban mobility. Domain experts validated the semantic model through inquiries and performed changes where needed. The authors pointed some difficulties during the ontology construction process:

- Some concepts still depend on the development of consensus, i.e. their meaning can vary depending on the context;

- Coherent choice of relevant and unambiguous set of relationships for concepts, i.e. too-specific or too-personal relationships compromise the utilization of the ontology by other people.

Zhang et al.proposed an approach for automatic transformation of UML transportation data models into OWL, under the justification that manual ontology building is a bottleneck in the ontology-acquisition process and is prone to human errors [153]. Such approach claimed to be a cost-effective way of generating ontologies. The authors found that transportation data models in OWL offer many advantages over UML such as facilitated data sharing and inference. The algorithm was applied to a case study of a real world object-oriented GIS data model for transit trip planning systems. The transformation algorithm was implemented using eXtensible Stylesheet Language Transformation (XLST) [154].

The Urban Large Knowledge Collider (LarKC) project used the city of Milan, Italy, as a case study [149]. A functional prototype was built to allow city tourists to plan their movements to their destinations by using a combination of transportation means. Ontologies were used to integrate several data sources, for example, city topology, points of interest and events. Figure 3.36 depicts the schematic representation of the prototype, which provided a SPARQL endpoint for query ingestion.

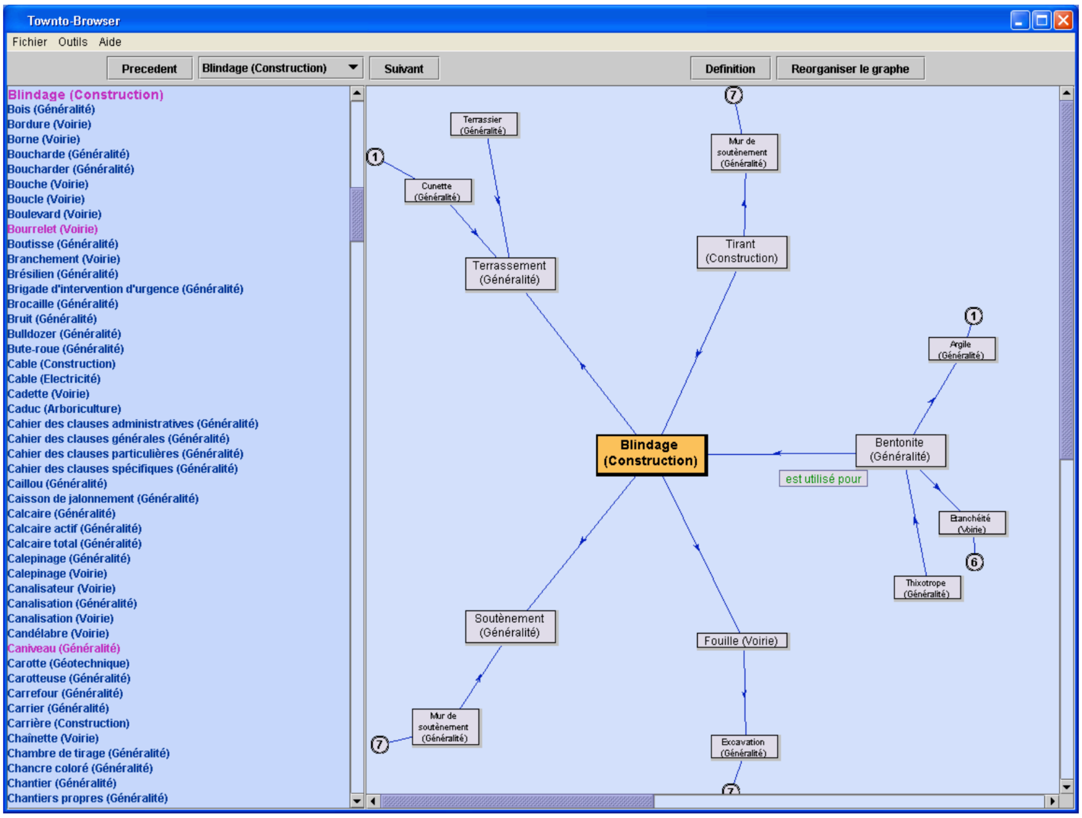


Figure 3.35: Towntology user interface [152]

Figure 3.36: Schematic representation of the Urban LarKC prototype [149]

A study by Plu et al. presented a workflow for publishing and linking transport data on the World Wide Web [155]. The goal was to provide a framework for the development of smart applications that take advantage of semantic data. The authors exemplify the application of the workflow with two data sets: one containing a French standard for describing a transportation line, and a directory containing information about transportation for all French cities. The proposed workflow, although it could be argued to be too generic, consists of the following phases: (i) ontology definition; (ii) conversion and alignment of a dataset to an ontology; (iii) publication of semantic data; (iv) interlinking with other semantic datasets.

Kathia et al. defined a transportation ontology oriented to the semi-automatic generation of personalized user interfaces (UI) of transportation interactive systems [156]. The ontology captures the information about the users and presents a customized UI based on their preferences. The study was focused on the problem of travel planning. The authors pointed the difficulty and need of having a deep knowledge about the application domain, in order to perform correct associations between each concept and its respective context element.

Bermejo et al. developed a decision support system (DSS) based on an ontology for road traffic management [157]. It extends the A3ME (Agent-based Middleware approach for Mixed Mode Environments) ontology proposed by Herzog et al. [158]. The goal of the DSS was to provide decision support to drivers, to clear an effective path for emergency vehicles.

Corsar et al. proposed the Transport Disruption Ontology to support integration of data related to events that may disrupt a transportation network [151]. The ontology was applied to two projects: the Social Journeys project investigated how social media can provide information to passengers. The TravelBot system provided travel advice to passengers based on information extracted from social media. The nature of this ontology implies the need of formally characterizing some type of events (see Figure 3.37), as it is expected to make use of spatiotemporal data.

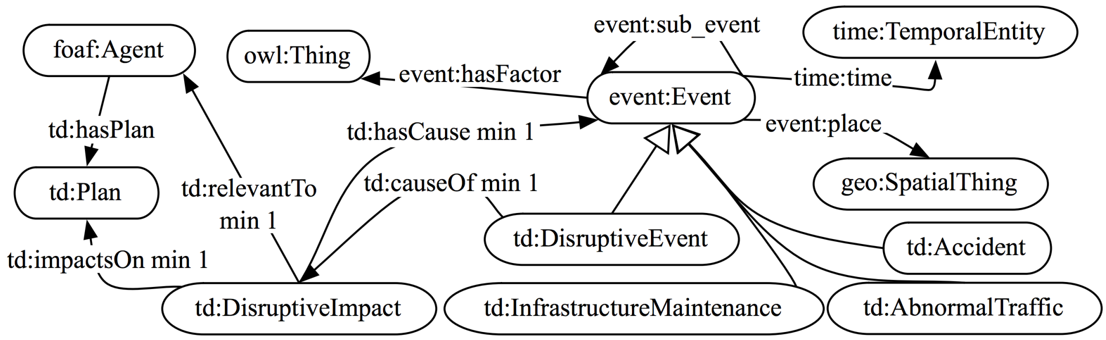


Figure 3.37: Excerpt of the Transport Disruption Ontology, with focus on the characterization of an event [151]

Seliverstov and Rossetti proposed an ontology-based approach for modeling spatiotemporal information in transportation [159]. The approach consists of an architecture capable of integrating several domain ontologies. The architecture is based on the R2DF[3] framework, which allows accessing relational databases as a virtual RDF graph. A prototype tool, *Trontegra*, was implemented to explore traffic analysis using data from inductive loops.

Benvenuti et al. developed the KPIOnto ontology to support public transportation systems monitoring [160]. The authors built the Transmodel ontology based on the Transmodel Data

[3]http://d2rq.org/

Model [161]. The KPIOnto ontology was built to represent KPIs. Both ontologies are part of a framework to support the design and analysis of a system for public transportation systems management. The framework provides a Prolog-based reasoning component that can be used to support basic design tasks of a monitoring system. The same reasoning component can be used for specification and calculation of KPIs.

3.3 Summary

This chapter presented the applications of Information Visualization, and Semantic Web Technologies and ontologies to urban mobility analysis. Regarding Information Visualization, most studies are related to traffic flows and monitoring, and the analysis of people dynamics and urban environments. Based on the surveyed studies, we summarized the topics in which applications of visualization have been found, and the data types that were used. We suggest that the role of the end user on the design, development and evaluation of visualization techniques should be increased. Few studies demonstrated some level of involvement with domain users.

Some relevant applications of ontologies and semantic web technologies could be found, although the number of studies is still small. The potential of ontologies for data integration has already been acknowledged by the research community. Studies proposed applications to urban planning, transportation network modeling, road traffic management, and content personalization for travelers. Nonetheless, to the best of our knowledge, the issue of visualization of heterogeneous, spatiotemporal urban mobility data remains unaddressed. The elements defined in the OTN ontology are not sufficient to address spatiotemporal urban mobility data, due to the broadness of concepts and very limited support to temporal data. Moreover, there are still no studies that focus on the development of semantically enriched, user-centered visualization systems, to facilitate users on the process of exploring urban mobility data.

Such limitations justify the need of developing an ontology that provides support to spatial and temporal concepts that define urban mobility data. To meet the requirements of data integration, such concepts should also be related to the different urban mobility topics and the types of data they require. Finally, to foster the involvement of domain users on the development of visualization techniques and tools, we acknowledge the relevance of addressing concepts related to domain users and their role on the design, development and evaluation of visualization techniques.

Part III

Research Contributions and Applications

Chapter 4

The VUMO Ontology

This chapter describes the Visualization-oriented Urban Mobility Ontology (VUMO). Practical examples are provided to facilitate the comprehension of the ontology's components, e.g. classes, properties, instances and rules, and to demonstrate some modeling approaches. VUMO implements a conceptual model that is proposed in Section 4.1.

As this thesis addresses the problem of visualizing spatiotemporal mobility data from various (un)related sources, VUMO provides a foundational knowledge representation model to support the development of semantically rich, knowledge-assisted visualization systems. Specifically, VUMO allows the following:

- Integration of multi-source heterogeneous urban mobility data related to spatial events, and their description in terms of transportation network elements;
- Specification of analytical tasks that users want to carry with data in the form of data transformations (queries);
- Annotation of visualization techniques implemented in a visualization systems, using concepts from Information Visualization theory, e.g. interaction tasks;
- Annotation of empirical domain user knowledge, e.g. user information and feedback about visualization techniques;
- Inference of implicit knowledge from instance data that is relevant for the data exploration and visualization process, e.g. implicit links between instances that come from distinct datasets, characteristics of data transformations, and compatibility with visualization techniques.

The structure of VUMO is modular, i.e. classes and properties were thought with the goal of having a well defined role on the development of semantically rich, knowledge-assisted visualization system, according to the following pipelines we defined:

1. *Data integration*: the system should be able to integrate data from multiple sources. The data structure not only maintains the original attributes of instance data; the structure is also used to infer visual attributes;

2. *Visualization technique design and development*: a visualization technique should be characterized in terms of its intrinsic attributes. Such attributes are expected to be used to evaluate the compatibility of visualization techniques with data transformations (defined in Subsection 4.1.4), and to aid users on the process of finding appropriate visualization techniques;

3. *Visualization technique evaluation and specification of system users*: user feedback about visualization techniques should be formally represented. The specification of system users allows the definition of their characteristics.

Section 4.1 introduces the conceptual model that precedes the implementation of the VUMO ontology. Section 4.2 provides an overview of the ontology and describes the development methodology. Section 4.3 introduces the upper classes of VUMO. Sections 4.4, 4.5, 4.6 and 4.7 thoroughly describe the ontology upper classes and their components. Section 4.8 describes the rules and functions embedded into VUMO that support inference of implicit knowledge. Section 4.9 specifies which existing ontologies were reused. Section 4.10 describes the evaluation of VUMO in terms of logical consistency. Section 4.11 describes how VUMO can support user-oriented methodologies for visualization techniques development, according to the aforementioned pipelines. Finally, Section 4.13 summarizes the main findings of this chapter.

4.1 Modelling Data, Visualizations and Expert Knowledge

In this section, we propose a conceptual formalization for spatiotemporal urban mobility data, visualization techniques and empirical knowledge derived from domain users. Such formalization is required to provide common, coherent semantics to the three components that will form a semantically enriched, user-centered visualization system. The resulting model is the starting point for building the VUMO ontology. Henceforth, urban mobility data will be simply regarded as *data*.

Subsections 4.1.1 and 4.1.4 formalize spatiotemporal urban mobility data and their transformations, respectively. Subsection 4.1.5 formalizes visualization techniques and their link to data transformations. Subsection 4.1.6 describes the formalization of empirical knowledge. Throughout the subsections, it is shown how the concepts (classes) of this conceptual model are interrelated. Finally, a UML Class Diagram provides a schematic representation of the model.

4.1.1 Spatiotemporal data

The formalization of data is built upon acknowledged frameworks for modeling geographic and movement data [18, 162–164]. We define every instance of spatiotemporal data as an *event*. An event is an action performed by one or more agents of a transportation network, and occurs at/in one or more spatial and temporal dimensions. As identified in Table 3.2, an event can be recorded by sources such as, for example, GPS devices, social networks and mathematical models, as in the case of origin-destination matrices.

The pyramid framework is a rigorous, non-domain specific conceptual framework for representing geographic phenomena. It allows their decomposition into three cognitive perspectives: *when* (time), *where* (space), and *what* (theme) [164].

Figure 4.1 provides a schema for the representation of data according to the pyramid framework. The cognitive perspective form the *data* component. The derived knowledge forms the *knowledge* component, and can be regarded as a semantic object: a conceptual entity. The term *knowledge* differs from *data* in the sense that the former consists of cumulative understanding of information found in data, and the latter simply consists of raw observational measurements. The semantic object may belong to a taxonomy (classification) and may have part-whole relationships with other semantic objects (partonomy).

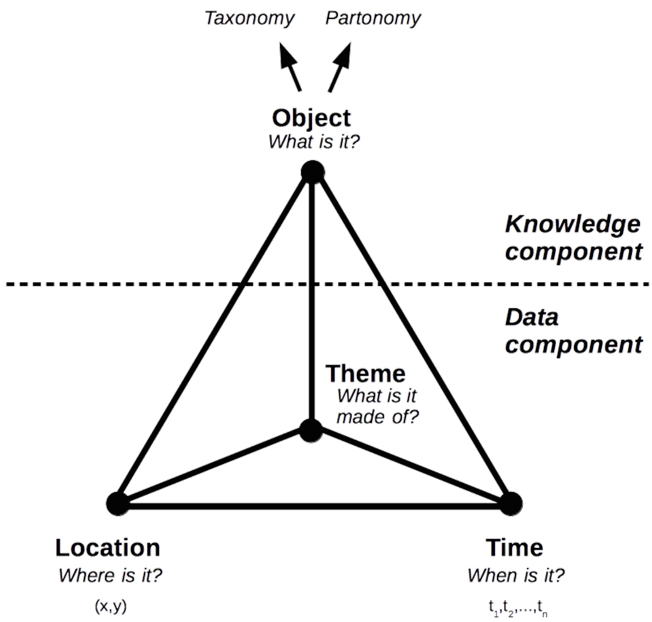


Figure 4.1: The pyramid framework (adapted from [164])

Based on this framework, Andrienko and Andrienko defined three fundamental sets for movement phenomena: space **S** (set of locations), time **T** (set of instants or intervals), and objects **O** [18]. Elements of each set have their intrinsic attributes. Attributes that are not related to space and time are regarded as *thematic*.

Another perspective is the *data-centric* perspective on Information Visualization, which defines two types of data components: *referential* and *characteristic* [18]. According to that perspective, a dataset is understood as a mapping from a set of references (independent variables) onto a set of characteristics (dependent variables). Let d be such mapping function. In mathematical terms,

$$d : \mathbf{R} \rightarrow \mathbf{C}$$

where **R** and **C** are the referential and characteristics sets, respectively. Both sets can be understood as cartesian products, where each R_i and C_k represent the set of all possible values for each referential and characteristic variables, respectively, with $1 \leq i \leq M$ and $1 \leq k \leq N$.

$$\mathbf{R} = \mathbf{R_1} \times \mathbf{R_2} \times ... \times \mathbf{R_M}$$
$$\mathbf{C} = \mathbf{C_1} \times \mathbf{C_2} \times ... \times \mathbf{C_N}$$

A hypothetical dataset record is then given by

$$d(r_1, r_2, ...r_m) = (v_1, v_2, ...v_n)$$

Where $r_i \in R_i \forall i$, and $v_k \in C_k \forall k$.

Space and Time

GIS literature defines two views of space and time: *absolute* and *relative* [18]. The *Absolute* view of space and time asserts that datasets have those dimensions as referrers, which can be generalized by a mapping $\mathbf{S} \times \mathbf{T} \rightarrow \mathbf{C}$, i.e. from space and time to a set of characteristics C. Space and time form an abstract container, where objects are placed. Figure 4.2 exemplifies a dataset related to car accidents. The characteristics of an accident can be fully identified if one knows the location and instant in time, except for the records that do not have complete spatiotemporal information.

DATE	TIME	BOROUGH	LATITUDE	LONGITUDE	INJURED	FATALITIES	CONTRIBUTING FACTOR VEHICLE 1	CONTRIBUTING FACTOR VEHICLE 2
08/10/2018	00:00	BROOKLYN	40,713074	-73,952095	0	0	Driver Inattention/Distraction	Unspecified
08/10/2018	00:00	BROOKLYN	40,577908	-74,00818	0	0	Driver Inattention/Distraction	Driver Inattention/Distraction
08/10/2018	00:00	MANHATTAN	40,75868	-73,98336	0	0	Other Vehicular	Unspecified
08/10/2018	00:00		40,708324	-73,84314	0	0	Driver Inattention/Distraction	Unspecified
08/10/2018	00:00		40,763428	-73,96522	0	0	Following Too Closely	Unspecified
08/10/2018	00:00	QUEENS	40,700768	-73,81019	1	0	Traffic Control Disregarded	Unsafe Speed
08/10/2018	00:01		40,704712	-73,727425	0	0	Unspecified	Unspecified
08/10/2018	00:10	MANHATTAN	40,724564	-74,00779	1	0	Failure to Yield Right-of-Way	Unspecified
08/10/2018	00:15	BRONX	40,87181	-73,85556	0	0	Driver Inexperience	Unspecified
08/10/2018	00:17				0	0	Other Vehicular	Other Vehicular
08/10/2018	00:17		40,706944	-73,91774	0	0	Turning Improperly	Unspecified
08/10/2018	00:25	MANHATTAN	40,710114	-74,012665	0	0	Unspecified	
08/10/2018	00:30	BRONX	40,831287	-73,88173	0	0	Passing or Lane Usage Improper	Unspecified
08/10/2018	00:30	MANHATTAN	40,80796	-73,94908	0	0	Driver Inattention/Distraction	Unspecified
08/10/2018	00:30		40,71603	-73,81637	0	0	Passing or Lane Usage Improper	Unspecified
08/10/2018	00:44	BROOKLYN	40,691586	-73,99927	0	0	Driver Inattention/Distraction	Unspecified
08/10/2018	00:44		40,753513	-73,98879	0	0	Driver Inattention/Distraction	Driver Inattention/Distraction
08/10/2018	00:45	QUEENS	40,740173	-73,811264	0	0	Unspecified	
08/10/2018	10:00		40,766434	-73,83767	0	0	Passing or Lane Usage Improper	Unspecified

Figure 4.2: Excerpt of a dataset about car accidents in New York city. Referential and characteristic components are depicted in blue and green, respectively

The *relative* view of space and time states that such dimensions may act as characteristics rather than referrers. Space and time are intrinsic properties of objects, given by the general mapping $\mathbf{O} \rightarrow \mathbf{S} \times \mathbf{T} \times \mathbf{C}$. In those studies, researchers found four different modeling approaches to spatiotemporal data [18]. Two of them are related to the spatial dimension: *location-based* and *entity-based* models. The first considers all information as characteristics of a given spatial referrer (e.g. pre-defined spatial units). The second considers spatial information as characteristics of a given entity. Based on those differences, a further classification is made according to the way time is considered: data may refer to temporal instances according to a universal time frame, or time is also a characteristic of such spatial units or entities.

Data Properties

Thematic attributes usually follow the classic categorization proposed by Stevens: *nominal*, *ordinal*, *interval* and *ratio* [165]. Some authors place *interval* and *ratio* into a same category called *quantitative* [13, 17]. Nominal attributes can be used to depict categorical information. Currently, datasets may contain other types of complex attributes, such as file attachments (e.g. photos, videos, worksheets) or data representation in specific formats (e.g. binary data structures, geometric shapes).

4.1.2 Fundamental Structures

The proposed data model defines fundamental structures that describe spatiotemporal and thematic attributes (characteristics). An entity-based perspective was adopted, thus space and time were regarded as attributes of every conceptual entity represented by the model. The choice for this perspective is justified by the characteristics of semantic data: every entity is an object that exists by itself, which is described in terms of other instances (resources or literals, as in Subsection 2.2.2). We present several definitions for each model structure.

Spatial References

An entity may have one or more spatial attributes depending on the nature of data. Two entity categories were defined: *Point* and *PointSet*.

Definition 4.1.1. A *Point* is described by latitude and longitude, following the WGS84 datum. This definition is flexible in the sense that other optional attributes may exist, e.g. elevation.

Points are divided into two disjoint subcategories:

Definition 4.1.2. A *Generic Point* is a *Point* that does not contain any thematic attribute that acts as an identifier, i.e. does not contain any property that assigns an identification, name, code, or any other textual attribute in a dataset.

Generic points are references that, in practice, do not require identification, e.g. the location of a citizen at a given time, retrieved from a GPS-assisted device.

Definition 4.1.3. A *Known Point* is a *Point* that contains at least one identifier attribute.

Known points are those for which the identification is relevant for some practical purpose. For instance, the bus stops of a network are KnownPoints, as it is possible to retrieve their spatial references by knowing their identification, e.g., `STCP_AEPT1`.

Definition 4.1.4. A *Point Set* is a (un)ordered collection of two or more *Point*s.

Point sets are divided into two subcategories:

Definition 4.1.5. An *OrderedPointSet* P is well-ordered, i.e. every non-empty subset of P has a least element in this ordering.

Definition 4.1.6. An *UnorderedPointSet* is a set of points that is not well-ordered.

An example of an ordered point set is a passenger route plan, formed by points which describe the departure and arrival locations. It is possible to infer the rank of any point. In contrast, shapes can be described in terms of unordered point sets, e.g., a set of Points which defines the boundaries of a polygon representing a public transportation system zone.

4.1.3 Spatial Distribution

Spatial references induce distinct visual arrangements, according to their type. Such arrangements are defined Spatial Distributions. Three primary arrangements are considered: Discrete, Quasi-continuous and Graph. Without loss of generality, spatial distributions are present in both geographic and abstract spaces. Spatial arrangements become apparent as the number of entities increase in visualization space, as shown in Figure 4.6. We defined the following axioms that link spatial reference types to distributions:

Definition 4.1.7. A *Generic Point* induces a *Quasi-Continuous* spatial distribution.

Generic points are related to entities that can be spatially located in any location within a geographic region. Hence, the visual pattern depicted by a number of those points induces the notion of smoothness (continuity).

Figure 4.3 shows a 2D Geographic Heat Map combined with data from citizens' locations while using a route planner mobile application, during their search for nearby stops based on their location. As a citizen can be located anywhere within the city, the visual pattern of data resembles continuity.

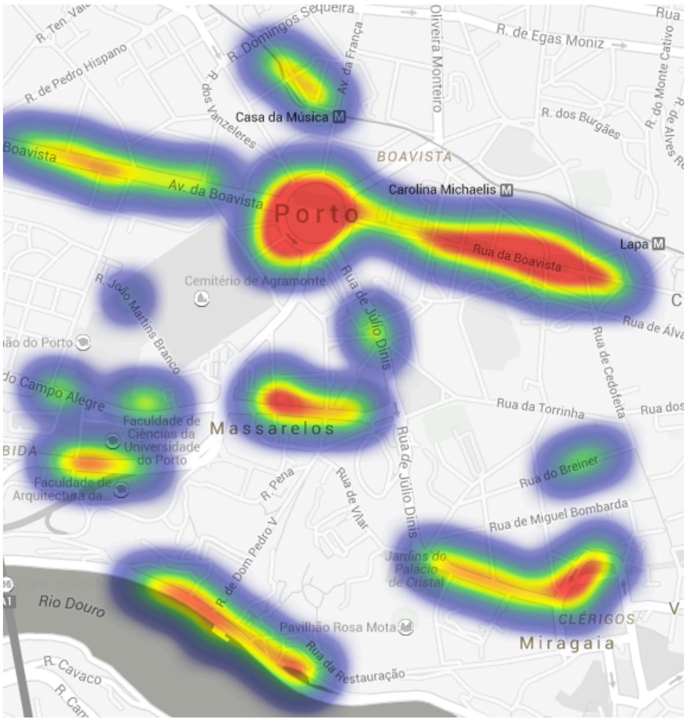


Figure 4.3: Example of quasi-continuous spatial distribution in a 2D Geographic Heat Map

Definition 4.1.8. A *Known Point* induces a *Discrete* spatial distribution.

The requirement of one or more identifier attributes suggests that known points represent entities that have a well-defined location within a region, and are less dense than generic points.

Figure 4.4 shows a 2D Geographic Heat Map combined with data from realtime schedule requests for bus stops while using a route planner mobile application.

Definition 4.1.9. An *UnorderedPointSet* induces a Discrete spatial distribution. Unordered point sets yield structures such as shapes and clusters, which are well defined in visualization space.

Definition 4.1.10. An *OrderedPointSet* induces a Graph spatial distribution. They form arrangements that resemble the notion of trajectory.

A PointSet inherits the spatial distributions induced by its Points. In other words, Point sets also induce Discrete and/or Quasi-continuous spatial distributions, according to the types of Points in a set. The practical purpose of this definition is to account for situations in which a

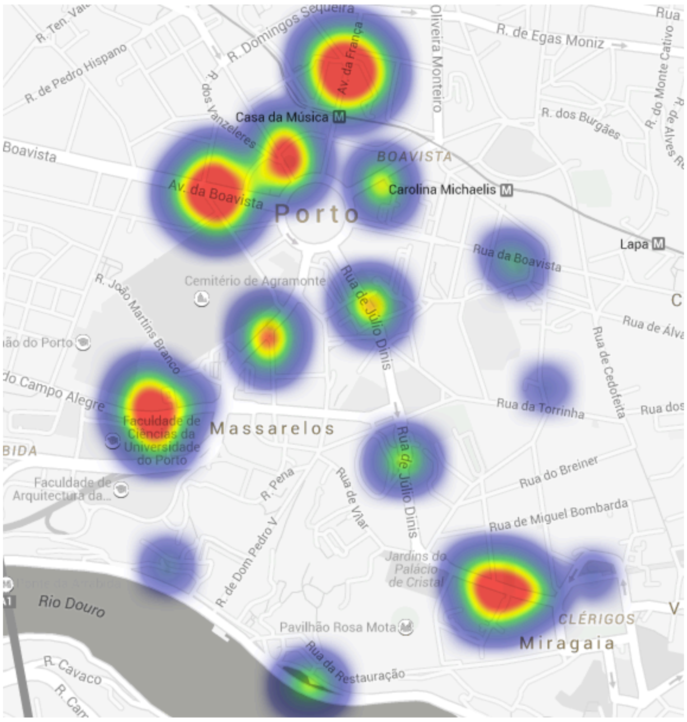


Figure 4.4: Example of discrete spatial distribution in a 2D Geographic Heat Map

visualization provides mechanisms to not only explore point sets as a whole, but to shift the perspective to its elements, using interaction mechanisms like semantic zoom. Point sets can be interpreted as paths in the mathematical sense, ordered or not. Figure 4.5 shows an abstract visualization technique that represents the volume of travel intentions between Metro stations in the city of Porto, also using data from a route planer mobile application. Each route plan consists of a point set with two points: origin and destination stations.

Definition 4.1.11. A *Point Set* inherits the spatial distribution(s) of its points.

This definition states that point sets also induce *Discrete* or *Quasi-Continuous* spatial distributions, depending on the categories of their points.

Temporal References

An entity may have one or more temporal attributes depending on the nature of data. Two types were defined: *Instant* and *Interval*.

Definition 4.1.12. An *Instant* is described by a timestamp, i.e. containing date and time, and an optional time zone description.

Definition 4.1.13. An *Interval* is described by two *timestamps*, corresponding to start and end.

Although an instant could be considered a zero-length interval in the mathematical sense, we argue that such consideration does not add practical value.

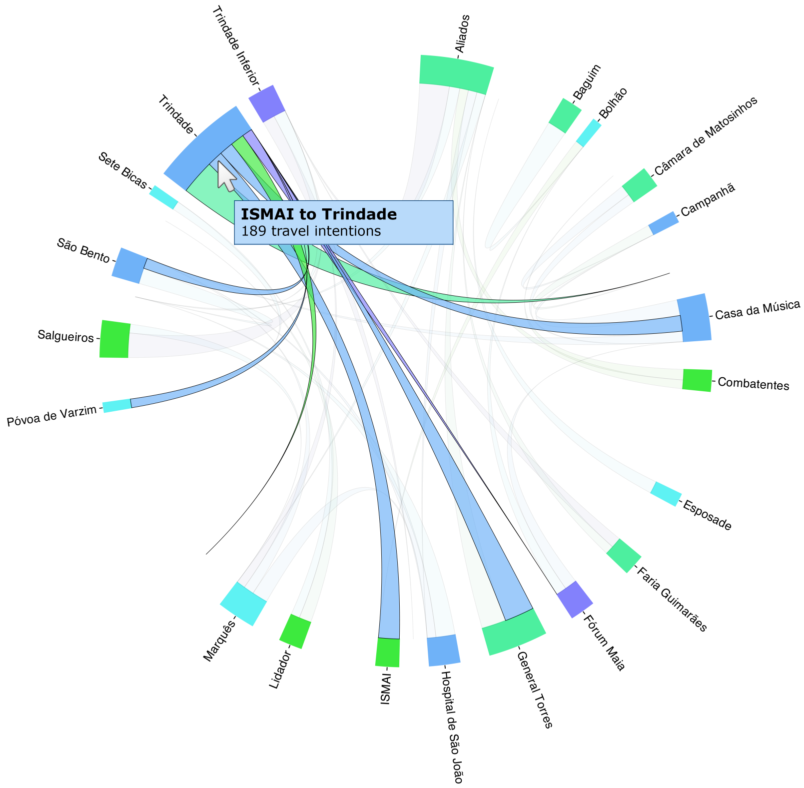


Figure 4.5: Example of graph spatial distribution in a chord diagram visualization technique

Thematic attributes

Thematic attributes can consist of a variety of types (e.g. strings, numerical values, binary file structures or geometric shapes). Due to our orientation towards Information Visualization, we introduce the concept of measures of an entity, and how that may appear in data.

Definition 4.1.14. A *Measure* is a certain quantity or degree of something, expressed by a quantitative, ordinal or categorical value.

An entity can have more than one measure. Recalling the car accidents dataset in Figure 4.2, the amounts of injured and killed are measures for an accident event. We now formalize an intuitive concept of measure, which is inherent to every entity described in the model.

Definition 4.1.15. A *Unitary Measure* corresponds to a numeric measure of "1" that is intrinsic to every entity.

In practical terms, this measure provides an existential indicator of an entity in a dataset. Such measure intuitively appears when performing simple actions like counting the number of entities in a dataset. For instance, the unitary measure is used in the car accidents dataset to count the total number of accidents during a given period.

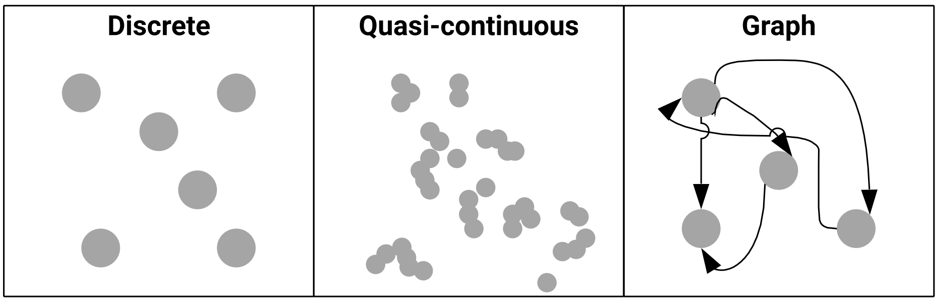


Figure 4.6: Schematic representation of spatial distributions proposed by the model

4.1.4 Transformations

Visualization of raw data is not sufficient for exploratory analysis. Frequently, domain users have questions which can be answered by performing data transformations (e.g. queries) or more complex operations (e.g. data mining techniques) in order to extract useful information. We define a *transformation* as a sequence of operations (analytical abstractions, as in Figure 2.10) that uses raw data as input, and produces a new dataset as output. This output may contain part of the original entities found in raw data, or yield completely different entities that result from such operations.

From a cognitive standpoint, we assume that a transformation can be seen as the formalization of an analytical task, which in turn yields questions about data. In this thesis, transformations are assumed to be queries. Figure 4.7 illustrates such assumption.

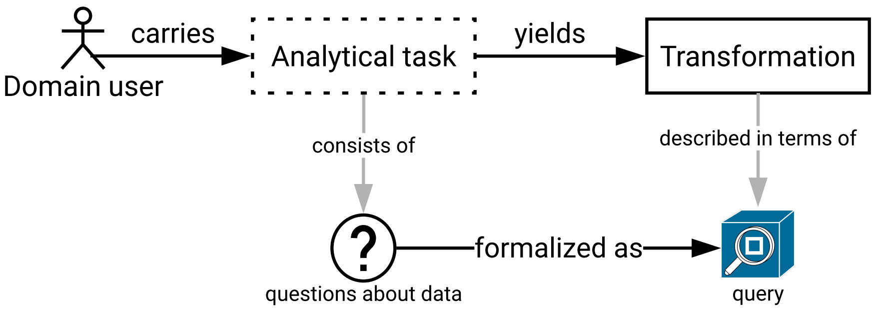


Figure 4.7: Representation of transformations as the formalization of analytical tasks in terms of queries

The output of a transformation, herein defined as *output variables*, may contain spatiotemporal references, depending on the output they generate. Likewise, it can generate new measures based on raw data, or use existing ones as output. The existence of spatial references in the output of a transformation implies that there are also visual spatial patterns associated with it. The following proposition demonstrates this statement.

Proposition 4.1.1. A *transformation* may have zero or more spatial distributions.

Proof. Let T be a transformation that yields $\{o_1, ..., o_n\}$ as output variables, where $O_s \subseteq O$ is a non-empty subset of O containing spatial references, with $1 \leq i \leq n$. By definition, a spatial reference a_i induces one or more spatial distributions, hence the output of T also induces one or more spatial distributions. In the null case, it suffices to consider that all outputs are attributes other than spatial references. □

For example, consider a sample dataset showing ticket validations for bus and metro routes in Porto, Portugal, in Figure 4.8. A simple transformation example is given, which groups the number of ticket validations by fare zones. The count of validations is a new measure built upon existing measures in raw data.

In this example, the transformation is given by the simple SQL query:

```
SELECT Zona, COUNT(*) as CountOfValidations
FROM table
GROUP BY Zona
```

Where $O = \{\texttt{Zona}, \texttt{CountOfValidations}\}$ and $O_s = \{\texttt{Zona}\}$.

4.1.5 Visualization techniques

A visualization technique has one or more features and provides one or more interaction tasks. Examples of interaction tasks can be found in Section 2.1.2. Visualization techniques features are the intrinsic components for data visualization. Our model specifies four features: *input variable*, *reference frame*, *spatial dimensionality* and *temporal arrangement*. The last three features are derived from a classification of visualization techniques proposed by [162].

Input variables are responsible for receiving the values from output variables (of a transformation) that will be mapped onto visual variables.

Reference frame describes the ability of a visualization technique to represent *geographic* (georeferenced) data, i.e. map-based visualizations, and *abstract* data.

Spatial dimensionality describes the number of dimensions used by the visualization canvas, e.g. 2D or 3D.

Temporal arrangement describes how the time dimension is represented on the visualization canvas, e.g. linear, cyclic.

As an example, consider the prototypical chord diagram visualization technique in Figure 4.5. It requires two input variables for source and destination, and one for weight. The prototype provides four interaction tasks: overview, filter, focus plus context, and dynamic queries. The technique has an abstract reference frame, and a linear temporal arrangement. Such conceptual specification is represented in Figure 4.9.

4.1.6 Empirical knowledge

Given the need of developing visualization systems that meet the needs and requirements of end users, empirical knowledge was divided into two facets: system user specification, and user feedback.

A system user has an analytical profile, which is specified according to the system's objectives. For instance, in our previous work, we worked with experts that belonged to strategical and operational profiles [3, 4]. Such characterization is not meant to be neither exhaustive nor unique. Each system can provide its own taxonomy of analytical profiles, as they are considered categorical values.

op	cod_cartao	Paragem	Linha	Sentido	Variante	Veiculo	zonamento	dhiniviagem	datahoravalidacao	Grupo	Perfil	zonas	Zona
...	...	...	...	...	...	...	...	...	...	...	...	...	...
STCP	20007188034	4207	907	1	1	3032	2	30-01-13 19:37	30-01-13 20:19	Assinatura	Normal	C1-2 S8-9	S9
STCP	20031228054	2211	907	2	1	3032	2	30-01-13 20:36	30-01-13 20:36	Assinatura	Normal	C1 S8-9	S9
STCP	20031312314	2085	907	2	1	3032	2	30-01-13 20:36	30-01-13 20:37	Assinatura	Estudante	C1 S8-9	S9
STCP	20031259162	2085	907	2	1	3032	2	30-01-13 20:36	30-01-13 20:37	Assinatura	Social+	C1 S8-9	S9
STCP	20032224996	2085	907	2	1	3032	2	30-01-13 20:36	30-01-13 20:37	Assinatura	Normal	C1 S8-9	S9
STCP	20032171602	2085	907	2	1	3032	2	30-01-13 20:36	30-01-13 20:37	Assinatura	Normal	C1 S8-9	S9
STCP	20007315826	2184	907	2	1	3032	2	30-01-13 20:36	30-01-13 20:38	Assinatura	Normal	C1 S8-9	S9
...	...	...	...	...	...	...	...	...	...	...	...	...	...

Zona	CountOfValidations
...	...
N10	1394
N11	1102
S1	805
S2	431
S3	499
S4	685
S5	593
...	...

Figure 4.8: Example of transformation. The number of ticket validations is aggregated into fare zones

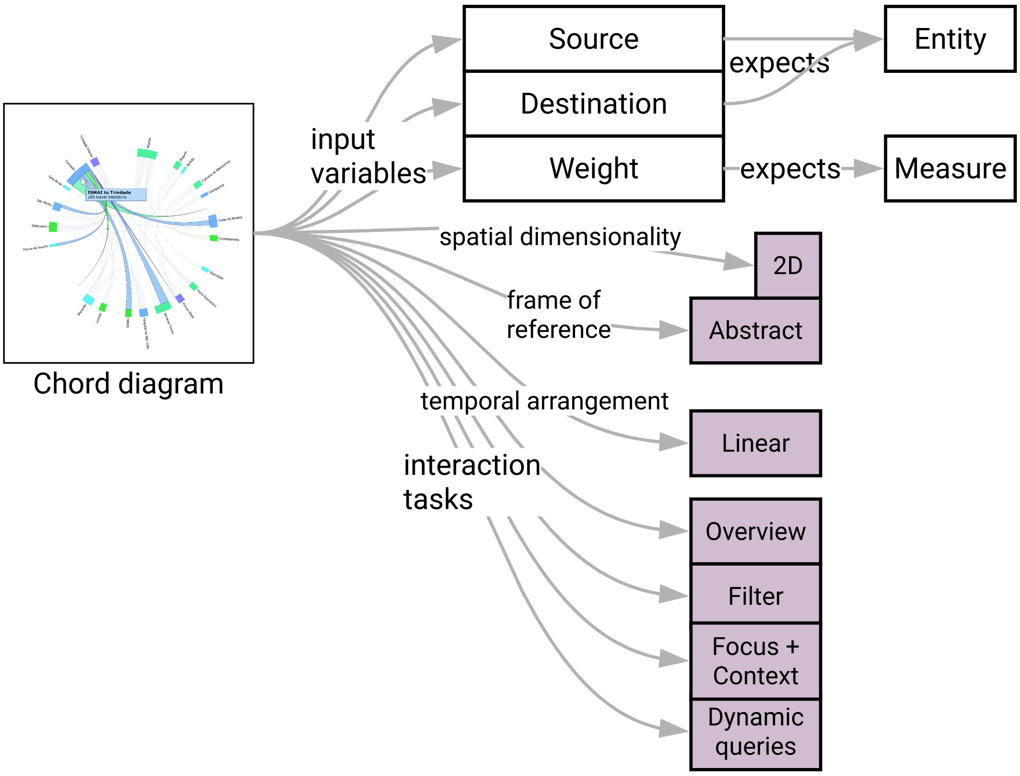


Figure 4.9: The conceptual description of a possible implementation of the chord diagram visualization technique

We define two perspectives for user feedback. The first consists of statements related only to a visualization technique, such as ratings about a certain property, e.g. complexity, or more subjective statements, e.g. *"this visualization is recommended for analyzing ticket validations"*.

The second facet allows users to provide specialized feedback about visualizations with respect to a transformation, which is defined as a *cross rating*. In cases in which the feedback is provided as a quantitative measure, a technique rating has a rating score related to a rating component. Qualitative feedback in given in terms of categorical values.

A quantitative rating component is subject to a scale and polarization, i.e. whether its value impacts positively or negatively in the overall rating.

Figure 4.10 provides a conceptual representation of empirical knowledge of two users about the chord diagram technique shown in Figure 4.5. User 1 rated the technique with respect to two components: "Visual Complexity", a *quantitative* rating component, and "Recommended analytical profile", a *qualitative* rating component. For instance, visual complexity is a component with negative polarization, in the sense that higher values will are likely to negatively impact the user experience. User 2 rated the technique with respect to the qualitative component "Recommended theme".

The UML Class Diagram on Figure 4.11 formalizes the fundamental structures that were defined for the conceptual data model, and describes how the structures are interrelated. The color gray indicate classes that refer to spatiotemporal data; blue refers to visualization techniques classes; green refer to expert knowledge classes.

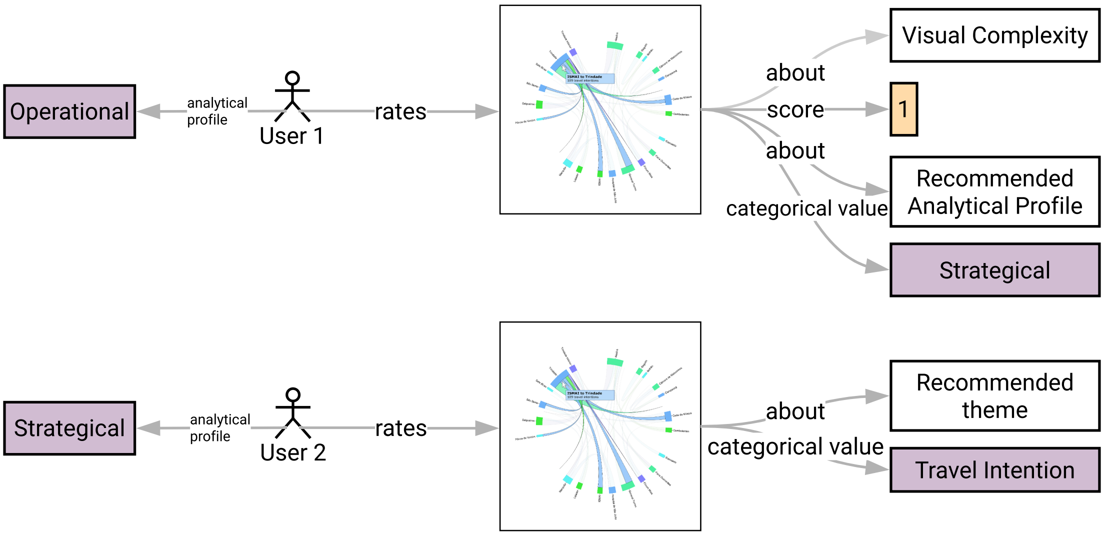


Figure 4.10: Conceptual representation of empirical knowledge of two users about the chord diagram visualization technique. Values in pink and orange depict qualitative and quantitative values, respectively

4.2 VUMO overview and development methodology

VUMO is an OWL ontology that conforms to the OWL 2 RL profile, which is a syntactic subset of OWL 2 that supports rule-based inference by trading some of its logical expressiveness [166], as explained in Section 2.2.4. The implementation of inference rules and functions uses the SPIN vocabulary and modeling language, hence they are expressed as queries in standard SPARQL language. Table 4.1 describes the main ontology characteristics. An exhaustive list of classes, properties and individuals can be found in Appendix A.

Table 4.1: Main characteristics of the VUMO ontology

Characteristic	Value	Description
URL	`http://purl.org/vumo#`	The URL for the VUMO ontology
Namespace	`vumo`	The adopted qname for shortening the aforementioned URL
OWL Profile	OWL RL	OWL profile for rule-based applications

Regarding the various perspectives for ontology classification (see Section 2.2.7), VUMO has the following characteristics:

- *Domain specific*: the ontology entails the domain of urban mobility;
- *Formal*: the ontology shall not contain ambiguous concepts. There should be consensus about its elements;
- *Heavyweight*: in order to support the various pipelines, an intensively axiomatized ontology is required. It would not be sufficient to provide a simple taxonomic structure;
- *Application*: the ontology is focused on supporting the development of semantically rich, knowledge-assisted visualization systems.

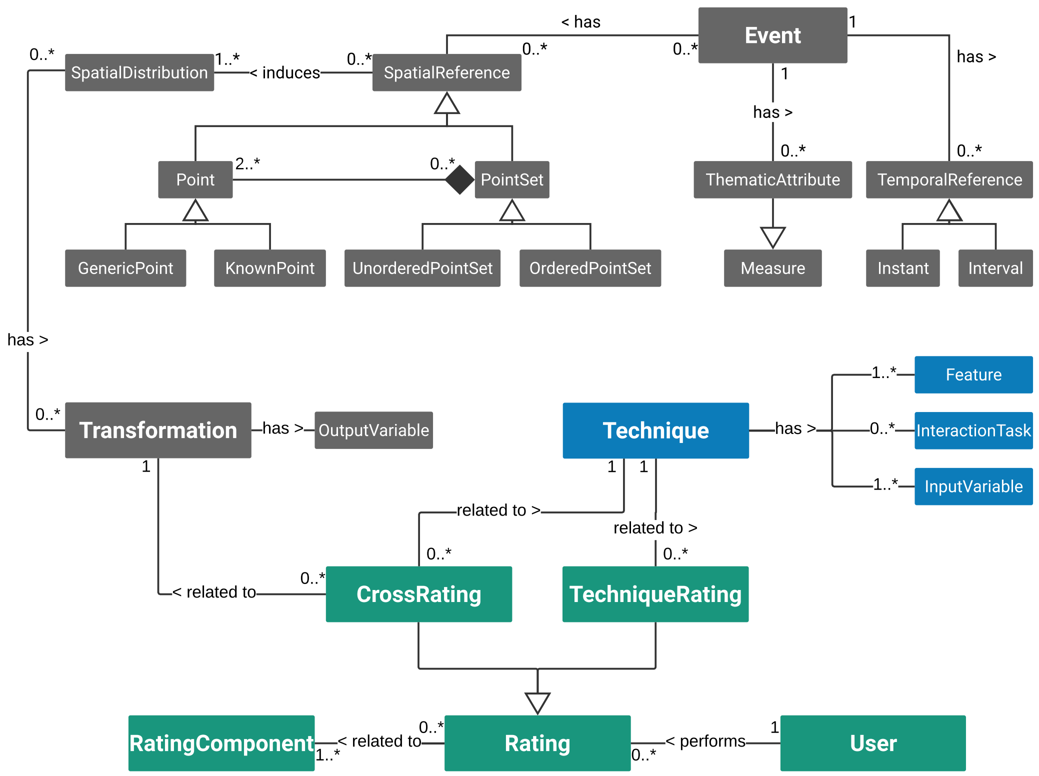


Figure 4.11: UML Class Diagram for the conceptual model, which serves as the basis for the implementation of the VUMO ontology

The conceptual model defined in Section 4.1 provides the foundation for the ontology. The process of ontology development considered the definition of competency questions, i.e. requirements in the form of structured questions that the ontology should answer. An example of competency question is ”How is a transportation journey characterized?” [156]. In the context of VUMO, we defined the following competency questions:

1. How to characterize an urban mobility event that occurs in a transportation network?
2. How to characterize the visual features of an event?
3. How to characterize a visualization technique?
4. How to characterize a user of a visualization system?
5. How to represent the analytical domain tasks of a system user in terms of data transformations?
6. How to represent empirical knowledge from system users?

We adopted, with some flexibility, the guidelines of the IDEF methodology as it is concerned not only with the creation of ontologies, but with their further modifications [167]. As we expect our ontology to be used in different contexts, we argue that the choice for this methodology is reasonable. In brief, the methodology comprises the following activities:

- *Organization and scope definition*: the identification of the purpose and context of the ontology;
- *Data collection*: The acquisition of the data needed for the development of the ontology;
- *Data analysis*: The definition of which elements of the data collection are necessary to be present in the ontology;
- *Initial ontology development*: a prototype of the ontology in which the preliminary validations are made;
- *Refinement and validation*: application of tests with real data.

The ontology was built according to a top-down approach, i.e. upper classes and properties were defined and further refined. Given that VUMO is strongly oriented to practical contexts, concepts were modeled after analyzing real data. We acquired data for the cities of Porto and Boston, which will be described in detail in Chapter 5 with practical cases. The structures of the datasets were analyzed in terms of their attributes. We also took advantage of the data sources types identified in the literature review (see Table 3.2).

Validation consisted of logical consistency tests using reasoners (see Section 4.10), and practical applicability with real data by carrying the case studies described in Chapter 5.

4.3 Upper classes

VUMO is divided into four upper classes:

- *UrbanMobilityConcept (UMC)*: a superclass for all classes that describe public transportation systems and events;
- *DataConcept (DC)*: a superclass for all classes of the conceptual model that are related to modeling of spatiotemporal data, and data transformations (see Sections 4.1.2 and 4.1.4);
- *VisualizationConcept (VC)*: a superclass for all classes of the conceptual model that characterize a visualization technique, its features and interaction tasks (see Section 4.1.5);
- *DomainUserConcept (DUC)*: a superclass for all classes of the conceptual model that characterize system users and their empirical knowledge about visualization techniques (see Section 4.1.6).

Table 4.2 shows the role of classes in each pipeline. A main role ($\star$) means that visualization systems developers (and users) will mostly use elements from that class on a specific pipeline. An auxiliary role ($\circ$) means that such class may be used directly - but not frequently - or indirectly (automatically) through rule-based inference. In the following sections, examples will demonstrate the role of each class across all pipelines.

Upper classes branch out into subclasses that represent more specific concepts. VUMO contains Object and datatype properties to relate instances from the aforementioned classes. Semantic data is herein represented using the standard form, i.e. subject-predicate-object triples, according to the Resource Description Framework (RDF) data model.

Table 4.2: Role of each superclass on the pipelines of a visualization system

Pipeline	UMC	DC	VC	DUC
Data integration	⋆	○		
Visualization design and development	○	○	⋆	○
Visualization evaluation and system user specification	○	⋆	⋆	⋆

Throughout the text, classes, properties and instances are written in monospaced font, e.g. `ns:semanticThing`, where the prefix `ns` indicates a namespace, i.e. the ontology in which the property is defined. For instance, `geo` is the prefix of the Basic Geo vocabulary, which is used to describe spatial coordinates [8]. No prefix was used to refer to VUMO components. Table 4.3 provides a natural language definition for each first-level subclass.

To facilitate understanding, we provide illustrative examples related to the context of Porto, Portugal. The namespace `porto` is used whenever we refer to instance data related to the city's public transportation system data. Depending on the complexity of the example, we also introduce a visual representation of the corresponding triples for the sake of readability.

4.4 UrbanMobilityConcept (UMC)

UMC concepts are fundamental to semantic integration of raw data, i.e. data is mapped onto its subclasses. The `Agent` and `InfrastructureComponent` subclasses describe structural concepts of a transportation system. An `Event` enables to describe distinct types of events related to urban mobility, e.g. ticket validation, accident, among others. Table 4.4 describes UMC and its components, which are important for understanding the examples of this section.

The example below describes two instances: one `Operator` and one `Vehicle`, along with some properties.

```
porto:STCP rdf:type :Operator .
  :hasName "Sociedade de Transportes Coletivos do Porto" .

porto:Bus3801 rdf:type :Vehicle .
  :ownedBy porto:STCP .
```

In VUMO, the property `ownedBy` is defined as the inverse equivalent of `owns`. Hence, the triple `porto:STCP :owns porto:Bus3801` could be automatically inferred. The next example shows the instantiation of a bus stop (a node). If applicable, an identification can be defined with `hasID` or its semantically equivalent subproperties, such as `hasInternalID` or `hasFriendlyID`. To illustrate the utility of multiple ID properties, consider an example of bus stop shown in Figure 4.12 from the city of Porto. `AEPT1` is a user friendly identification used by passengers to consult schedules using real time services. From the operator's perspective, one or more identifications can be used, such as `STCP_AEPT1` or 54. All IDs uniquely determine the stop within their distinct semantic contexts, yet they refer to the same entity.

```
porto:AEPT1 rdf:type :BusStop ;
  :hasFriendlyID "AEPT1" ;
  :hasInternalID "STCP_AEPT1" ;
  :hasInternalID "54" ;
```

Table 4.3: Upper classes of VUMO and their respective first-level subclasses

Upper class	Subclass	Definition
Urban Mobility Concept (UMC)	*Agent*	- An entity that is part of the urban mobility network, which can perform actions such as *Events*.
	InfrastructureComponent	- A physical or abstract entity. *Agents* can use it (in)directly trigger *Events*, or to provide them with context information.
	Event	- An action performed by one or more *Agents*, which may take place in multiple space and time dimensions.
Data Concept (DC)	*SpatialReferenceType*	- A type of spatial reference.
	TemporalReferenceType	- A type of temporal reference.
	SpatialDistribution	- A type of visual arrangement of spatial data.
	SpatialDistributionAxiom	- An abstract container for the spatial distribution axioms defined in the conceptual model.
	Transformation	- A sequence of operations (e.g. query) that yields new information from raw data.
Visualization Concept (VC)	*Technique*	- A visualization technique available in a VUMO-based visualization system.
	Feature	- An intrinsic characteristic of a *VTechnique*.
	InteractionTask	- An interaction task available in a *Technique*.
Domain User Concept (DUC)	*DomainUser*	- A user of a VUMO-based visualization system.
	DomainUserProfile	- The profile of a *DomainUser* with respect to the context of his/her activities, e.g. *Strategical, Operational*.
	TechniqueRating	- An abstract container that stores rating information made by a *DomainUser* with respect to a *Technique*.
	RatingComponent	- An abstract component evaluated in a rating that impacts the user experience, e.g. *Difficulty*, *Visual clutter*. Such components are divided into *Positive/NegativeRatingComponent*.
	CrossRating	- An abstract container that stores rating information about a *Technique* with respect to a *Transformation*. A *CrossRating* provides a specialized rating in comparison to a *TechniqueRating*, which does not depend on a *Transformation*.

Table 4.4: First-level and further subclasses of UrbanMobilityConcept (UMC)

Subclass	Second-level subclass	Definition and further subclasses (if applicable)
Agent	*Operator*	- A transportation operator, e.g. bus or subway companies, taxi agencies and shared bicycles operators).
	Passenger	- An individual who uses a public transportation system.
	Vehicle	- An action performed by one or more *Agent*s, which may take place in multiple space and time dimensions, e.g. buses, trains and bicycles.
Infrastructure Component	*Line*	- A PTS line that may consist of various *Route*s.
	Node	- A node of the transportation network graph, e.g. *BusStop*, *SubwayStation*, *Sensor*, *BicycleStation*.
	Zone	- A pre-defined geographic zone used for a specific purpose, e.g. to define fares.
	Route	- A path followed by a *Line*. Lines may consist of several routes
	RouteSegment	- An elementary part of a *Route*, generally defined by two *Node*s.
	Ticket	- A ticket that allows a passenger to travel within a public transportation network.
	TicketType	- The type of a ticket.
Event	*TravelEvent*	- A travel made by a passenger within a transportation network triggered by an action, e.g. *TicketValidation*, *BicycleTrip*.
	TravelIntention	- An intention of traveling within a transportation network, based on information requests, e.g. schedule requests or route plans.
	SensorReading	- A reading made by a *Sensor*.
	SocialMediaPost	- A post from social media regarding transportation information.
	TripEvent	- A trip made by a *Vehicle*, e.g. bus or taxi.
	UnexpectedEvent	- An unforeseen *Event*, often capable of (partly) disrupting a transportation network in some way.

Operator's perspective

id	alt_id	sign_id	label	lat	lon	operator	zone
…	…	…	…	…	…		…
54	STCP_AEPT1	AEPT1	Aeroporto	41.23709	-8.66953	STCP	N10
…	…	…	…	…	…		…

Figure 4.12: An illustration of a bus stop sign of STCP operator in Porto. The identification AEPT1 is meant to be used by passengers when checking schedules in a real-time service. From the operator's perspective, multiple internal identifications may exist for the same stop

```
 :hasName "Aeroporto" ;
 geo:lat "41.23709" ;
 geo:long "-8.66953" ;
 :operatedBy porto:STCP ;
 :locatedInZone porto:N10 .
```

Figure 4.13 provides a visual representation of the triples that describe the bus stop instance AEPT1. The property `hasName` is defined as a subproperty of `rdfs:label` to provide a user-readable version of a resource's name; in this case, `Aeroporto`. The property `operates` is an inverse property of `operatedBy`, thus it is possible to infer the triple

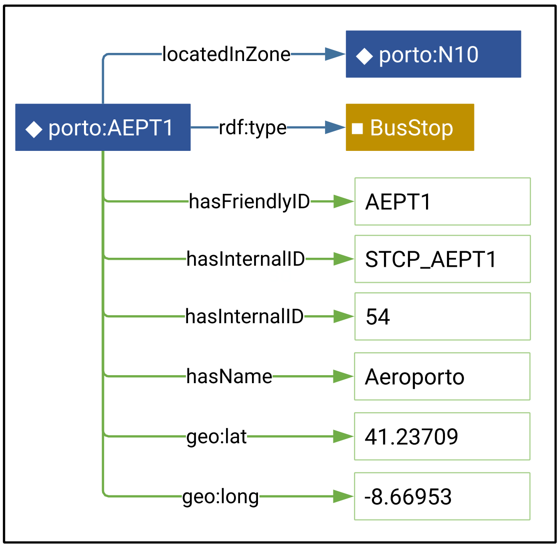


Figure 4.13: Visual representation of the RDF graph that describes the AEPT1 stop

```
porto:STCP :serves porto:AEPT1.
```

The same applies to the properties `locatedInZone` and `hasZoneElement`. A zone, for instance, can be described by asserting triples that indicate its boundary points.

```
porto:N10 :hasZoneBoundaryPoint porto:Point301 ;
  :hasZoneBoundaryPoint porto:Point302 .
  ...
```

The instantiation of a line can be described in terms of several routes and their respective route segments, as in the serialization below. Figure 4.14 provides a visual representation of the corresponding graph.

```
porto:Line300 rdf:type :Line .
  :hasDescription "300 - CIRC. HOSP S. JOAO X ALIADOS" ;
  :operatedBy porto:STCP ;
  :hasRoute porto:Line300_route1 .

porto:Line300_route1 rdf:type :Route ;
  :hasDirection porto:HSJ6 ;
  :hasRoutePath [ rdf:first portotip:Segment34 .
                  rdf:rest [ rdf:first portotip:Segment35 .
                             rdf:rest [ ... ] ] . ] .

porto:Segment34 rdf:type :RouteSegment ;
  :hasInitialNode porto:FEUP2 ;
  :hasFinalNode porto:FEP1 .
```

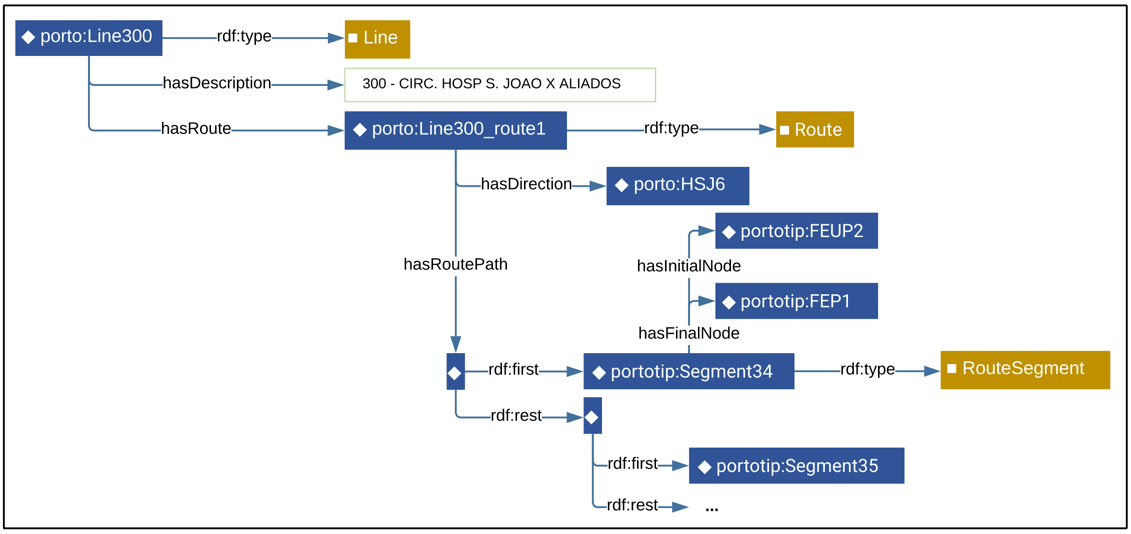


Figure 4.14: Visual representation of the RDF graph that describes the bus line 300

A ticket can be instantiated and described in terms of its ticket type, and other properties such as the zone in which it can be used.

```
porto:Ticket1201838 rdf:type :Ticket ;
  :hasTicketType porto:MonthlyPass_Regular ;
  :isValidInZone porto:N11 ;
  :isValidInZone porto:N10 .
```

```
porto:MonthlyPass_Regular rdf:type :TicketType ;
  :hasName "Assinatura Mensal - Tarifario Normal" .
```

The next three examples show possible modeling approaches for different event types. Firstly, a ticket validation was defined in terms of the trip (instance of `TripEvent`) in which the validation occurred. An important note is the distinction between the property `hasDateTime`, which represents instants, and the properties `hasStartDateTime` and `hasFinishDateTime`, which are meant to represent intervals.

```
porto:TicketValidation301 rdf:type :TicketValidation ;
  :hasDateTime "2013-05-13T08:00:01" ;
  :occursAtNode porto:AEPT1 ;
  :usesTicket porto:Ticket1201838 ;
  :occursInVehicle porto:Bus3801 ;
  :occursInTrip porto:Trip323042 .
```

```
porto:Trip323042 rdf:type :Trip ;
  :hasStartDateTime "2013-05-13T07:58:01" ;
  :hasFinishDateTime "2013-05-13T08:35:24" ;
  :madeByVehicle porto:Bus3801 ;
  :hasRoute porto:Line300_route1 .
```

Secondly, the following route plan was defined in terms of its origin and destination locations. Intermediate points (waypoints) can also be represented by using the property `:hasWaypoint`.

```
porto:RoutePlan75 rdf:type :RoutePlan ;
  :hasDateTime "2013-07-24T12:33:21" ;
  :hasPath [ :hasOrigin porto:FEUP2 ;
             :hasDestination porto:RFAR1 . ] .
```

```
porto:Reading944 rdf:type :SensorReading ;
  :hasDateTime "2013-01-12T00:05:00" ;
  :madeBySensor porto:S302 ;
  :hasSensorReading 350 .
```

Alternatively, the specification may use an interval temporal reference. The property `hasDuration` can be used to describe the duration of an event. Moreover, it is shown on Section 4.8 (rule R3) how a VUMO rule can infer the duration of events (a measure) when intervals are specified.

```
  :hasStartDateTime "2013-01-12T00:00:00" ;
  :hasFinishDateTime "2013-01-12T00:05:00" .
```

Thirdly, the following description of a car accident was defined in terms of a spatial reference that has no specific identification, i.e. it simply consists of geographical coordinates, and has the form of a blank node.

```
porto:CarAccident121 rdf:type :Accident ;
  :hasDateTime "2013-06-01T07:05:48" ;
  :occursAtLocation [ rdf:type geo:Point ;
```

```
                                geo:lat "41.178808" ;
                                geo:long "-8.594556" . ]
```

Measures are defined as subproperties of `hasMeasure`. Some examples of measures that are already implemented in the VUMO ontology are:

- `hasDuration`;
- `hasNumberOfInjuredPassengers`;
- `hasNumberOfAvailableBicycles`.

The latter two properties can be used to describe, for example, an `Accident` or the status of a `BicycleStation`. Due to the intrinsic semantics of OWL and RDF, it is possible to quickly identify all measures related to an event, as they are subproperties of `hasMeasure`.

Figure 4.15 shows the interconnection between the instances presented in the examples. As the volume of data grows, it is possible to visualize the complex interrelation between instances. A symbolic notation was adopted for representing data: classes (□) and their instances (◇). Object and Datatype properties are represented by blue and green edges. Solid and dashed edges indicate asserted (explicit) and inferred (implicit) triples, respectively.

4.5 DataConcept (DC)

DC describes structural properties of spatiotemporal data and transformations. The subclass `SpatialReferenceType` is refined into two subclasses:

- `Point`, with further subclasses `GenericPoint` and `KnownPoint`;
- `PointSet`, with further subclasses `UnorderedPointSet` and `OrderedPointSet`.

`TemporalReferenceType` contains two subclasses: `Instant` and `Interval`. The `SpatialDistribution` class contains three instances: `Discrete`, `Quasi-continuous` and `Graph`.

With the exception of `Transformation`, it is not expected that users directly manipulate the remaining classes defined in DC, given that VUMO rules are responsible for inferring data properties from instance data, i.e. data described in terms of UMC subclasses.

For instance, a reasoner can infer that the spatial references exemplified in UMC (Subsection 4.4) are a `KnownPoint`, `UnorderedPointSet`, `OrderedPointSet` and `GenericPoint` respectively, in accordance to the definitions established in Section 4.1. Hence, the graph of instance data would be semantically enriched with the triples below.

```
// Spatial reference 1: a bus stop
porto:AEPT1 rdf:type :KnownPoint .

// Spatial reference 3: a zone
porto:N10 rdf:type :UnorderedPointSet .

// Spatial reference 4: the origin and destination of a route plan
[ :hasOrigin porto:FEUP2 ;
  :hasDestination porto:RFAR1 . ] rdf:type :OrderedPointSet .
```

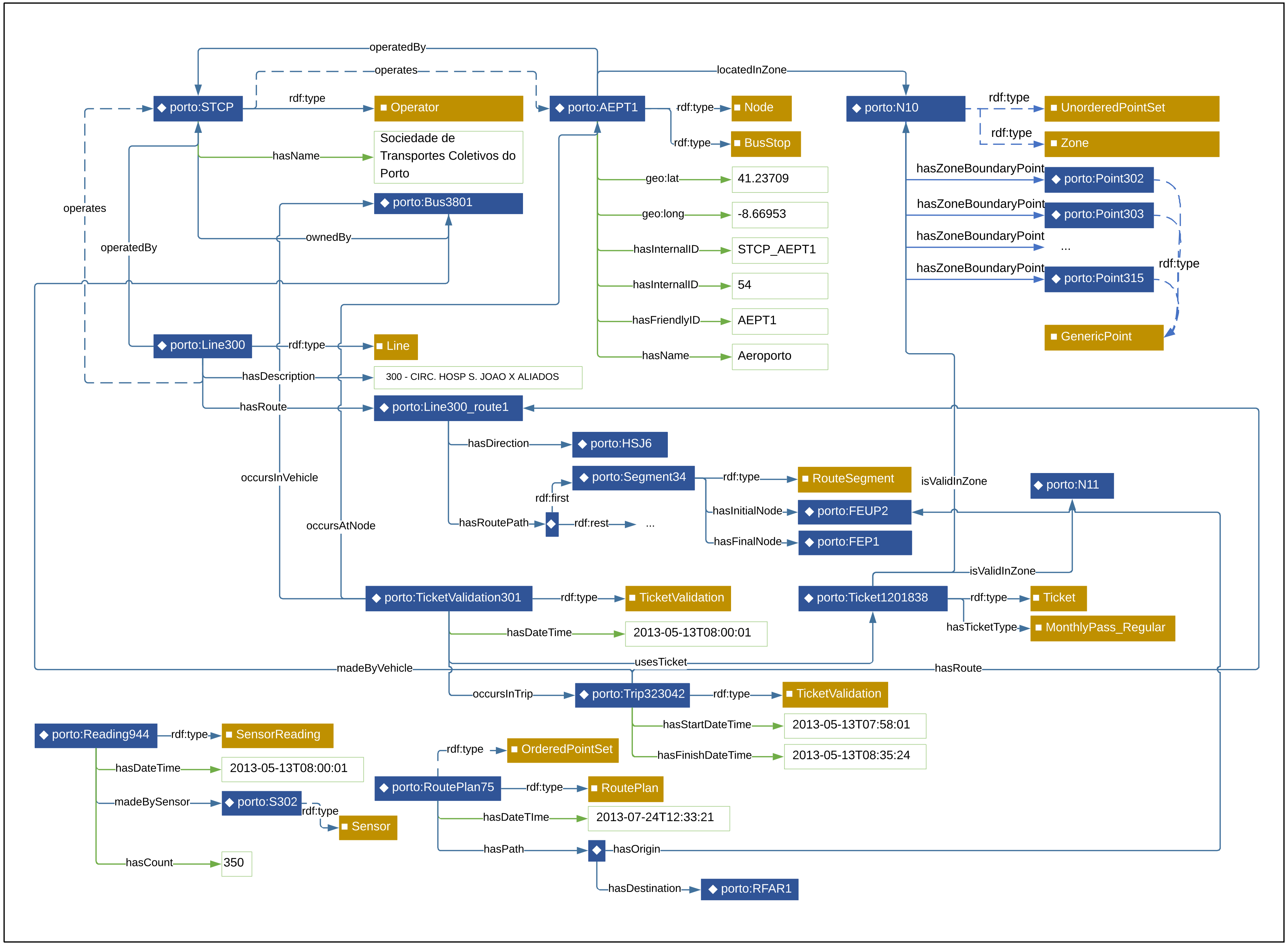


Figure 4.15: Illustrative example of instance urban mobility data described in Section 4.4

```
// Spatial reference 4: the location of an accident
[ geo:lat 41.178808 ;
  geo:long -8.594556 . ] rdf:type :GenericPoint .
```

The description of the subclasses of the `TemporalReferenceType` class is given in Section 4.9, in which we discuss the reuse of existing ontologies.

The definitions of spatial distributions (see Section 4.1.3) are represented as instances of `SpatialDistributionAxiom`. The goal of this subclass is to serve as an abstract container for the definitions about spatial distributions that are internally used by VUMO rules to infer the visual patterns of instance data and data transformations.

In practice, spatial distribution axioms are implemented as RDF statements, hence the standard properties `rdf:subject`, `rdf:predicate` and `rdf:object` are used. Notice that the definition of the spatial distribution axiom for a `PointSet` applies to both subclasses `UnorderedPointSet` and `OrderedPointSet`.

```
// A Known Point induces a Discrete spatial distribution.
:SD_KnownPoint rdf:type :SpatialDistributionAxiom .
  rdf:subject :KnownPoint .
  rdf:predicate :hasSpatialDistribution .
  rdf:object :Discrete .

// A Generic Point induces a Quasi-Continuous spatial distribution.
:SD_GenericPoint rdf:type :SpatialDistributionAxiom .
  rdf:subject :GenericPoint .
  rdf:predicate :hasSpatialDistribution .
  rdf:object :QuasiContinuous .

// A Point Set induces a Graph spatial distribution,
// and Discrete and Quasicontinuous as well (by inheritance)
:SD_PointSet rdf:type :SpatialDistributionAxiom .
  rdf:subject :PointSet .
  rdf:predicate :hasSpatialDistribution .
  rdf:object :Discrete .
  rdf:object :QuasiContinuous .
  rdf:object :Graph .
```

Data transformations are modeled as instances of `Transformation`, and their queries can be expressed in SPARQL. The SPIN vocabulary has the advantage of allowing the specification of a query as a graph. Such specification is transparent to the user, in the sense that a SPIN-compatible reasoner is capable of parsing a plain-text query in standard SPARQL into its respective graph, without user intervention. Moreover, we take benefit from this advantage to infer implicit knowledge from transformations, i.e. spatial distributions, tags (themes), and compatibility with visualization techniques.

Recalling the transformation illustrated in Figure 4.8, which groups the number of ticket validations by fare zones, a possible modeling approach is given by the following example:

```
:Query_ValidationsByFare rdf:type :Transformation
  rdfs:comment "This query aggregates ticket validations during a certain
      period by fare zones" .
  spin:query "SELECT ?zone COUNT(?ev) AS ?num_validations
              WHERE {
                   ?ev rdf:type :TicketValidation .
```

```
        ?ev :occursAtNode ?node .
        ?node :isLocatedInZone ?zone .
        FILTER (?time >= ?start && ?time <= ?finish)
    }
    GROUP BY ?zone" .
```

The SPARQL query is represented in plain-text form using the `spin:query` property from the SPIN vocabulary. Placeholders `?start` and `?finish` receive the values specified by users according to the desired time interval. On a technical note, the class `Transformation` is a subclass of the `spin:Template` from the SPIN vocabulary [49]. This class allows the specification of query templates which are meant to be reused by a system.

Figure 4.16 illustrates the graph structure that corresponds to this transformation, which is automatically inferred by the SPIN vocabulary. We found appropriate to represent it in a visual way to facilitate visualization, as the graph is not meant to be human-readable. Although it is out of scope to explain how SPIN works, we argue that such example will facilitate the understanding of VUMO rules that extract implicit knowledge from transformations.

The aforementioned query is represented as a blank node, (blank circle). The first two triples indicate the query type (`SELECT`) and the variable used for grouping the query results (`?zone`), respectively. Result variables are those that a query will yield as the result of data transformation. The term *result variables* belongs to the SPIN terminology. In this thesis, the term *output variables* is used to facilitate the comprehension of their relationship with the *input variables* of a visualization technique. The remaining blank nodes represent all conditions expressed inside the `WHERE` block. The `FILTER` operator is also expressed internally as a condition.

4.6 VisualizationConcept (VC)

VC allows for the annotation of visualization techniques, hence it corresponds to the visualization components of the conceptual model. The `Technique` class is used to create instances that represent the visualization techniques implemented in a visualization system. `InteractionTask` refers to interactive mechanisms that a technique provides. Some instances are already available in the ontology, e.g. `SemanticZoom` or `Filtering`. `Feature` comprises intrinsic components related to the graphical and data aspects of visualizations. The subclasses of `Feature` already have a number of pre-defined instances as shown below:

- `ReferenceFrame`: `Abstract`, `Geographic`;
- `SpatialDimensionality`: `2D`, `3D`;
- `TemporalArrangement`: `Linear`, `Cyclic`;
- `TemporalRepresentation`: `Static`, `Dynamic`;
- `InputVariable`.

We provide a simple example of a bar chart implementation that supports dynamic temporal querying (see Figure 4.17). For each record, this implementation requires three input variables. For the x- and y- axes, nominal and quantitative values are required, respectively. A timestamp is required. In Figure 4.17, a record is considered for visualization if the timestamp (`var3`) lies

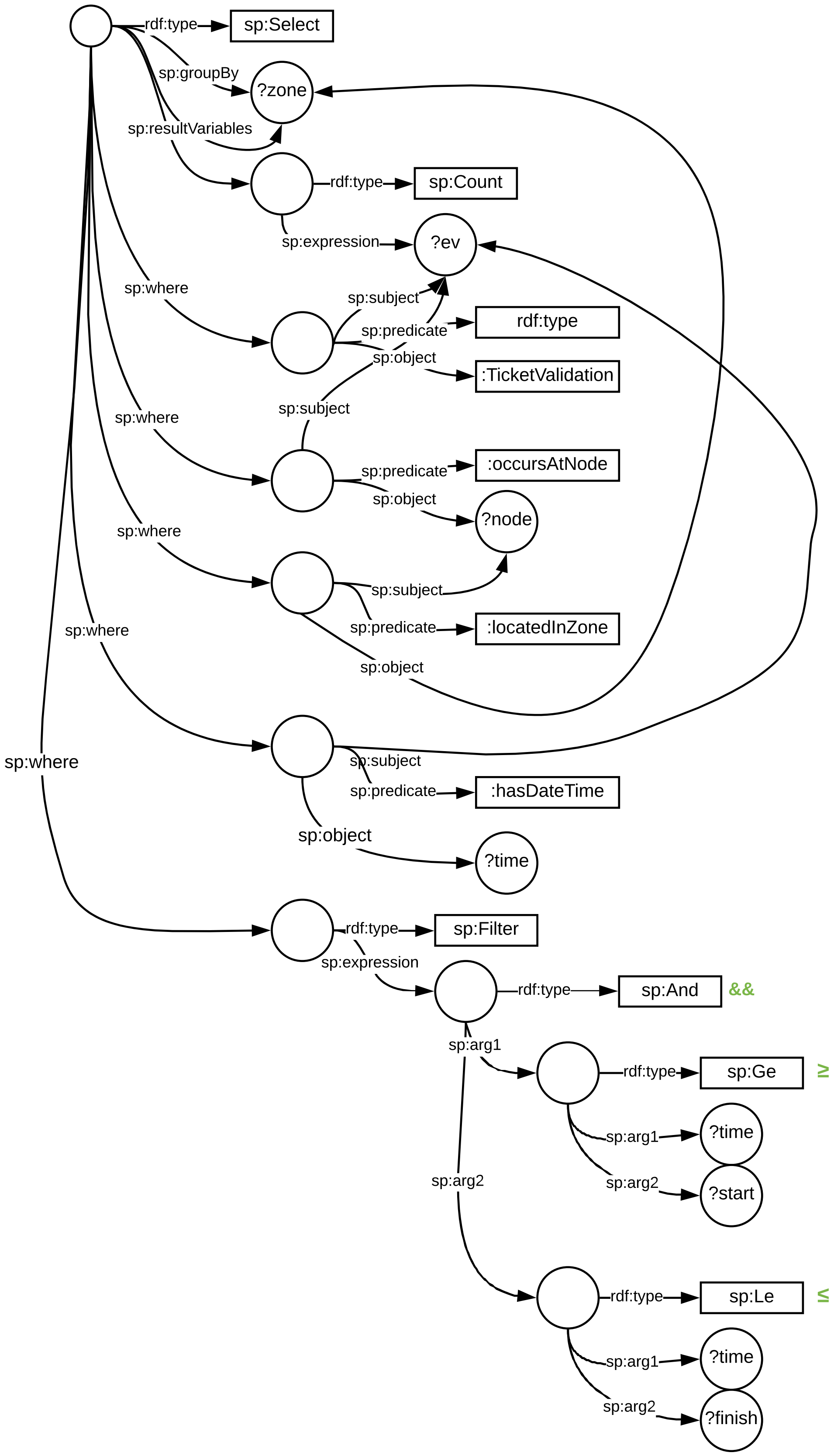


Figure 4.16: Visual representation of the graph corresponding to a data transformation example

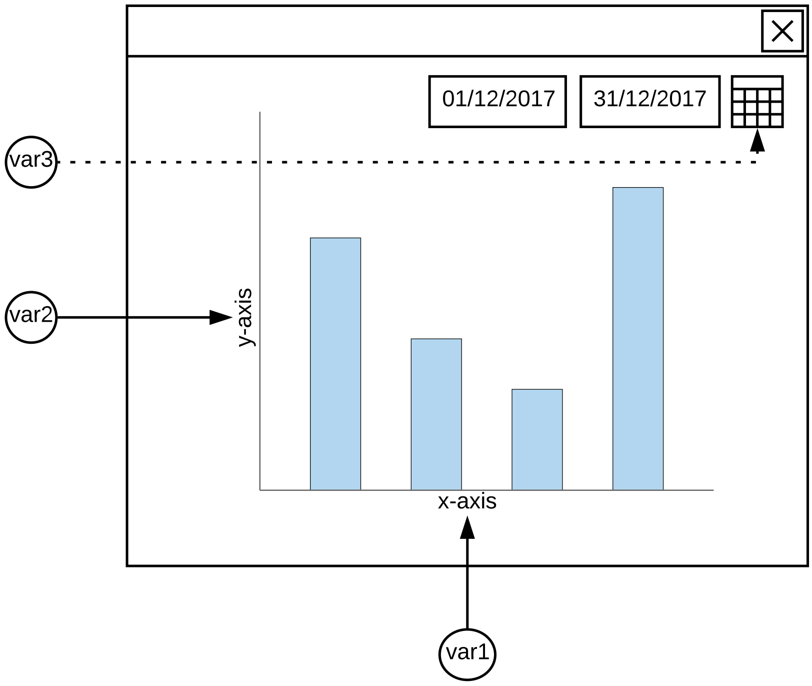


Figure 4.17: Mockup of a bar chart visualization technique

within the temporal interval selected by the user. A possible modeling approach is given by the serialization below.

```
vistool:BarChart_TimeFilter rdf:type :VisualizationTechnique ;
  rdfs:comment "An interactive bar chart with support to time filtering.";
  :hasReferenceFrame :Abstract ;
  :hasSpatialDimensionality :2D ;
  :hasTemporalRepresentation :Static ;

  :hasInputVariable vistool:var1 ;
  :hasInputVariable vistool:var2 ;
  :hasInputVariable vistool:var3 .

  // Variable for nominal values
  vistool:var1 rdf:type :InputVariable ;
    :hasCompatibleValueType xsd:string ;
    :isRequired true .

  // Variable for quantitative values
  vistool:var2 rdf:type :InputVariable ;
    :hasCompatibleValueType xsd:decimal ;
    :hasCompatibleValueType xsd:float ;
    :isRequired true .

  // Variable for timestamps
  vistool:var3 rdf:type :InputVariable ;
    :hasCompatibleValueType xsd:dateTime ;
    :isRequired true .
```

Instances for input variables can have any URI. In this case, the names `var1`, `var2` and `var3` were used for clarity. The semantics of the property `hasInputVariable` can automatically infer that such instances belong to the class `InputVariable`, as VUMO specifies the `rdfs:range` of this property to `InputVariable`.

The property `hasCompatibleValueType` allows the specification of several datatypes that are accepted by an input variable. We recommend, as good practice, the use of the XSD standard for data types [46]. The property `isRequired` expects a boolean value. It is used to specify whether an input variable is optional or not. This property is used by VUMO to evaluate compatibility of visualization techniques with data transformations.

The bar chart example provides two interaction tasks: dynamic queries and filtering, which can be represented as the following:

```
vistool:BarChart_TimeFilter :hasInteractionTask :DynamicQueries, :Filtering .
```

4.7 DomainUserConcept (DUC)

DUC allows for the annotation of empirical domain user knowledge. Such knowledge can be used to assess *appropriateness* of visualizations. Users are represented as instances of `DomainUser`, where each user has one or more `DomainUserProfile`. VUMO provides two pre-defined instances of user profiles: `Strategic` or `Operational`.

`TechniqueRatings` are statements made by `DomainUsers` about a `Technique`. A `TechniqueRating` contain one or more statements regarding `RatingComponents`.

VUMO allows for the annotation of specialized ratings. `CrossRatings` are used to rate a `Technique` with respect to a `Transformation`, according to one or more instances of `RatingComponents`. The following specification reflects the illustrative example of Figure 4.10.

```
vistool:User1 :hasUserProfile :Operational .
vistool:User2 :hasUserProfile :Strategical .

vistool:Rating1 rdf:type :TechniqueRating ;
  :isAboutVisualizationTechnique vistool:ChordDiagram ;
  :hasRatingStatement [ :hasRatingComponent :VisualComplexity ;
                        :hasRatingScore 1 . ] ;
  :hasRatingStatement [ :hasRatingComponent :RecommendedProfile ;
                        :hasRatingCategoricalValue :Strategical . ] .

vistool:Rating2 rdf:type :TechniqueRating ;
  :isAboutVisualizationTechnique vistool:ChordDiagram ;
  :hasRatingStatement [ :hasRatingComponent :RecommendedTheme ;
                        :hasRatingCategoricalValue :TravelIntention . ] .
```

In addition, a cross rating is represented as the following:

```
vistool:Rating3 rdf:type :CrossRating ;
  :isAboutVisualizationTechnique vistool:ChordDiagram ;
  :isAboutTransformation vistool:OD_byStations ;
  :hasRatingStatement [ :hasRatingComponent :Effectiveness ;
                        :hasRatingScore 4 . ] ;
  :hasRatingStatement [ :hasRatingComponent :VisualComplexity ;
                        :hasRatingScore 3 . ] ;
```

```
:hasRatingStatement [ :hasRatingComponent :RecommendedProfile ;
                      :hasRatingCategoricalValue :Strategical . ] ;
:hasRatingStatement [ :hasRatingComponent :RecommendedProfile ;
                      :hasRatingCategoricalValue :Operational . ] .
```

4.8 VUMO rules and functions

Rules and functions extend the capability of the VUMO ontology on regards to inference of new knowledge beyond the intrinsic semantics of OWL and RDF. We developed a set of rules and functions to automatically infer visualization-related properties from instance data, e.g. types of spatial references and spatial distributions, and to infer implicit knowledge from data transformations and visualization techniques. The proposed rules and functions are not meant to be exhaustive, but provide a solid starting point for the development of semantically rich, user-centered visualization tools.

4.8.1 Rules

Seven rules were defined, labeled from R1 to R7. They are independent in the sense that the execution of a rule during inference is made independently from the others.

Rules *R1* and *R2* detect spatial references within instance data and infer their type, i.e. points, point sets, and their subtypes. *R3* infers the duration of intervals if their start and finish times are specified. Rules *R4*, *R5* and *R6* infer characteristics of *Transformation* queries based on their structure, namely: spatial distribution (*R4*), themes (*R5*), use of aggregate functions (*R6*). *R7* infers the compatibility between transformations and visualization techniques.

Tables 4.5 and 4.6 provides the pseudocode representation of rules R1-3 and R4-8, respectively. Inferred triples are represented with a specific notation. For instance, $s \in$ *GenericPoint* is equivalent to "s is an instance of *GenericPoint*"; t *isCompatibleWith* v denotes a subject-predicate-object triple.

R4 analyzes conditional clauses for predicates containing equivalent subproperties of *hasSpatialReference*. If one or more clauses satisfy that condition, the range of such property is used to retrieve the spatial reference type. The corresponding axiom is then used to retrieve the spatial distribution.

R5 extracts themes, i.e. tags, that describe the urban mobility concepts related to a *Transformation*. The rule finds condition clauses whose properties' ranges are subclasses of *UrbanMobilityConcept*. Themes provide a natural language description of the contents of a *Transformation*.

R6 verifies if a *Transformation* returns aggregate data, i.e. if at least one *OutputVariable* contains an aggregate function. Such verification occurs while evaluating compatibility, as a *Technique* may expect disaggregate instance data to perform aggregations externally.

R7 evaluates the compatibility of a *Transformation* with respect to a *Technique*. Compatibility holds if the aggregate requirements (*R5*) match, and if there exists at least one bijective mapping m such that

$$
\begin{aligned}
m\colon O' \subseteq O &\rightarrow I \\
o_j &\longmapsto i_k
\end{aligned}
$$

Table 4.5: Pseudocode representation of VUMO rules related to data integration

Rule	Pseudocode

R1	// R1 infers *Point*s and their subtypes
	$s \leftarrow$ instance of *rdfs:Resource* // receives an instance of any class
	if *containsLatitudeLongitude*(s) **then**
	    if *containsIdentification*(s) **then**
	        $s \in$ *KnownPoint* // infers s as an instance of *KnownPoint*
	    else
	        $s \in$ *GenericPoint* // infers s as an instance of *GenericPoint*

R2	// R2 infers *PointSet*s and their subtypes
	$s \leftarrow$ instance of *rdfs:Resource* // receives an instance of any class
	$P \leftarrow \bigcup_s p$ // Points referred by s, if any
	if $\lvert P \rvert \geq 2$ **then** // P should have at least two Points
	    if *isOrdered*(P) **then**
	        $P \in$ *OrderedPointSet* // infers P is an *OrderedPointSet*
	    else
	        $P \in$ *UnorderedPointSet* // infers P is an *UnorderedPointSet*

R3	// R3 infers the duration of intervals, when applicable $e \leftarrow$ instance of *Event* $p_i, p_f \leftarrow$ // ordered temporal reference properties (initial and final) **if** $(e \quad p_i \quad t_i) \wedge (e \quad p_f \quad t_f)$ **then** // if triples exist for start and finish times e *hasDuration* $(t_f - t_i)$ // inferred triple

and $\theta(o_j) = \theta(i_k)\,\forall(o_j, i_k)$, where $o_j \in O'$ and $i_k \in I$ are the output and input variables, respectively.

O is the set of all output variables returned by a data transformation. I is the set of all input variables of a visualization technique. O' is a subset of O. The function θrepresents an operator that returns the type of an output or input variable, e.g. string, integer, resource.

4.8.2 Functions

Besides compatibility of evaluation of visualization techniques and data transformations, the evaluation of appropriateness is specific to the implementation of recommendation algorithms of each visualization system. To support recommendation, VUMO provides embedded functions (helpers) to assist methods on retrieving asserted empirical knowledge:

- `getTechniqueRating(t)`: returns all ratings given to a visualization technique `t`;
- `getCrossRating(t,v)`: returns all cross ratings assigned to the a transformation `t` and a visualization technique `v`;
- `getExpertInfo(e)`: returns asserted knowledged related to domain expert `e`.
- `getBroaderConcepts(c)`: returns concepts that are broader than `c`, based on the assertions made with the SKOS vocabulary;
- `getNarrowerConcepts(c)`: returns concepts that are narrower than `c`, based on the assertions made with the SKOS vocabulary;

Helpers are also stored as SPARQL queries, and take advantage of the SPIN vocabulary to be executed as functions.

4.9 Reuse of existing ontologies and vocabularies

The reuse of existing ontologies is acknowledged as a good practice in the context of Semantic Web technologies. Firstly, it reduces the effort of data integration, especially in situations in which data is already modeled in terms of established knowledge representation models. Secondly, it prevents semantic redundancy of similar concepts across various ontologies.

Given our orientation towards spatiotemporal data, VUMO imports elements from the WGS84 Geo Positioning vocabulary (`geo`) [8] and Time ontology (`time`) [168]. From the `geo` vocabulary, we make use of data properties `geo:lat` and `geo:long` for latitude and longitude, respectively, and a `vumo:Point` is declared as a subclass of `geo:Point`, which in turn is a subclass of `geo:SpatialThing`. All properties in VUMO that are used to indicate temporal instants in the form of literals are defined as subproperties of `time:inXSDDateTime` from the `time` ontology. Modeling of time periods, as in Subsection 5.3.2 (Chapter 5), are described in terms of the property `hasTimePeriod`, which is a subclass of `time:ProperInterval`. The `time` ontology provides other constructs for modeling time with varying levels of complexity.

The GTFS ontology (`gtfs`) is partially used to describe concepts related to public transportation system information, such as routes, stops, trips, among others. For instance, the classes `vumo:Zone` and `gtfs:Zone` are considered equivalent.

Table 4.6: Pseudocode representation of VUMO rules related to data transformations

Rule	Pseudocode

R4	// R4 infers *SpatialDistribution*s of a *Transformation*
	$t \leftarrow$ instance, $q_t \leftarrow$ query within t, such that $t \in$ *Transformation*
	$C \leftarrow \bigcup_{q_t} c$ // condition clauses of q_t
	for each $c \in C$ **do**
	  $p_c \leftarrow$ *property*(c) // receives the property (predicate) of c
	  if $p_c \equiv$ *hasSpatialReference* **then**
	    $r_{p_c} \leftarrow$ *range*(p_c) // receives the range of property p_c
	    $\sigma_{r_{p_c}} \leftarrow$ *getSpatialDistribution*(r_{p_c})
	    t *hasSpatialDistribution* $\sigma_{r_{p_c}}$ // inferred triple

R5	// R5 infers themes (tags) of a *Transformation*
	$t \leftarrow$ instance, $q_t \leftarrow$ query within t, such that $t \in$ *Transformation*
	$C \leftarrow \bigcup_{q_t} c$ // condition clauses in q_t
	for each $c \in C$ **do**
	  $p_c \leftarrow$ *property*(c) // receives the property (predicate) of c
	  if *range*$(p_c) \equiv$ *UrbanMobilityConcept* **then**
	    $r_{p_c} \leftarrow$ *range*(p_c) // receives the range of property p_c
	    t *hasTheme* r_{p_c} // inferred triple

R6	// R6 infers if the query of a *Transformation* returns aggregate results
	$t \leftarrow$ instance, $q_t \leftarrow$ query within t, such that $t \in$ *Transformation*
	$V \leftarrow \bigcup_{q_t} v$ // set of output variables of q_t
	for each $v \in V$ **do**
	  if *isAggregate*(v) **then**
	    t *returnsAggregateResults true* // inferred triple
	    break // one occurrence is sufficient
	  else
	    t *returnsAggregateResults false* // inferred triple

R7	// R7 infers compatibility of a *Transformation-Technique* pair
	$t, v \leftarrow$ instances, such that $t \in$ *Transformation*, $v \in$ *Technique*
	$O, I \leftarrow$ output and input variables sets of t, respectively
	$\xi \leftarrow \varnothing$ // result set of compatible mappings
	if *meetsAggregateRequirements*(t, v) **then**
	  $\xi \leftarrow$ *findMapping*(t, v) // stores compatible mappings in ξ
	  if $\lvert\xi\rvert \geq 1$ **then**
	    t *isCompatibleWith* v // inferred triple

The SKOS vocabulary (`skos`) is used to describe the level of generality or specialization of some VUMO concepts. For instance, the concept `Zone` is broader than `Node`. Analogously, `Strategical` is broader than `Operational`. We used the SKOS ontology to categorize several VUMO concepts, as shown in Figure 4.18. In Section 5.5, such approach yielded interesting applications on the development of a visualization tool. The properties `narrowerTransitive` and `broaderTransitive` include the following semantics: if B is narrower than A and C is narrower than B, then C is narrower than A. The properties `narrower` and `broader` do not include the semantics of transitivity.

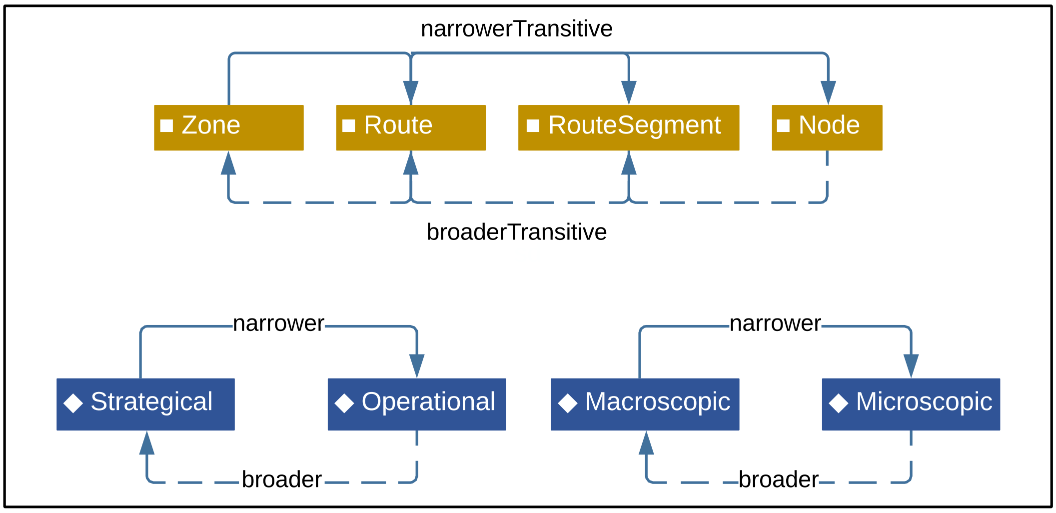


Figure 4.18: Hierarchy of some VUMO concepts defined with the SKOS vocabulary

The Schema.org vocabulary is used to declare some subclasses of UMC as subclasses from that vocabulary. For instance, `vumo:Vehicle` is a subclass of `schema:Vehicle`. `vumo:Zone` is a subclass of `schema:AdministrativeArea`.

Other standard ontologies and vocabularies are used to support the modeling of VUMO elements and their semantics, e.g. `owl2xml`, `rdfs`, `rdfs`, `spin`. Table 4.7 shows all ontologies imported in VUMO.

Table 4.7: Ontologies reused by VUMO

Name	Qname	URI
WGS84 Geo Positioning	`geo`	http://www.w3.org/2003/01/geo/wgs84_pos#
Time Ontology	`time`	https://www.w3.org/TR/owl-time#
General Transit Feed Specification	`gtfs`	http://vocab.gtfs.org/terms#
Simple Knowledge Organization System	`skos`	https://www.w3.org/2008/05/skos
Schema.org vocabulary	`schema`	https://schema.org#
Web Ontology Language	`owl2xml`	http://www.w3.org/TR/owl2-syntax/
Resource Description Framework	`rdf`	http://www.w3.org/1999/02/22-rdf-syntax-ns#
Resource Description Framework Schema	`rdfs`	http://www.w3.org/2000/01/rdf-schema#
Extended Markup Language vocabulary	`xml`	http://www.w3.org/XML/1998/namespace
Extensible Markup Language schema	`xsd`	http://www.w3.org/2001/XMLSchema#

4.10 Ontology evaluation

VUMO was evaluated for logical consistency using third-party reasoners that run several tests: Pellet, FaCT++ and TopSPIN [49]. The results indicated no inconsistency errors, e.g. logical contradictions. The OOPS! Pitfall Scanner [169] was also used to detect issues such as missing annotations, which were corrected. Purposely, some properties do not have information about their domains and ranges, in order to avoid undesired side-effects in terms of axiomatic classifications, as well as to reduce inference time. On regards to practical validity, VUMO was used in practical applications, as will be explained in Chapter 5. The ontology was revisited several times to refactor properties, classes and instances names, and their specifications.

4.11 Applications to user-oriented methodologies

This section describes how VUMO can also be applied to the user-centered design process of visualization techniques, in the context of a specific methodology chosen as an example, proposed by Koh et al. [35] (see Chapter 2, Subsection 2.1.5). We adapted this methodology by creating a new phase which is referred as *semantic annotation*, i.e. implementation of the outcomes of a phase by using VUMO and other ontologies that may be considered. Figure 4.19 illustrates the extension of the UCD process. Table 4.8 summarizes the artifacts from semantic annotation that derive from UCD phases, which are described in this section. In Chapter 5, we provide practical applications that are based on the proposed extension of the UCD process.

The data integration pipeline typically takes place during the actual use of the visualization system. Hence, this pipeline is not expected to be visited during UCD processes, except for ingestion of a minimal working dataset for testing purposes, which shall not involve domain users but developers.

In *Problem Domain Analysis*, the analysis of context and work domain provides a description of the potential users of a visualization system. In this phase, it is important to define the `AnalyticalProfile` of domain users, and the analytical tasks, i.e. instances of `Transformation` that the system should support. To formally support the description of empirical knowledge, actual users of the visualization system should be instantiated as instances of `DomainUser`. This ensures that any ratings provided by a user would be related to it during the system use, which is particularly important in case the tool provides (semi)-automatic means for suggesting visualization techniques.

After the *Conceptual Development and Prototyping* phases, artifacts consist of a set of visualization techniques prototypes, not necessarily implemented into the system. The semantic annotation of visualization techniques is expected to occur at the end of those phases, whenever revisited. It is fundamental that visualization techniques are minimally described in terms of their input variables, so that VUMO can evaluate the compatibility of a technique with all analytical tasks that have been already described during the *Problem Domain Analysis*. To make comprehensive use of available VUMO constructs, visualization techniques should also be described in terms of their frame of reference, available interaction tasks, spatial dimensionality and temporal dynamics.

During *Interaction and Usability Studies*, visualization techniques are likely to be revisited and corrected, to meet user requirements, which may require changes to the semantic annotation of visualization techniques, and even analytical tasks. More importantly, the first empirical knowledge is expected to be collected and formalized during this phase, in the form of instances of

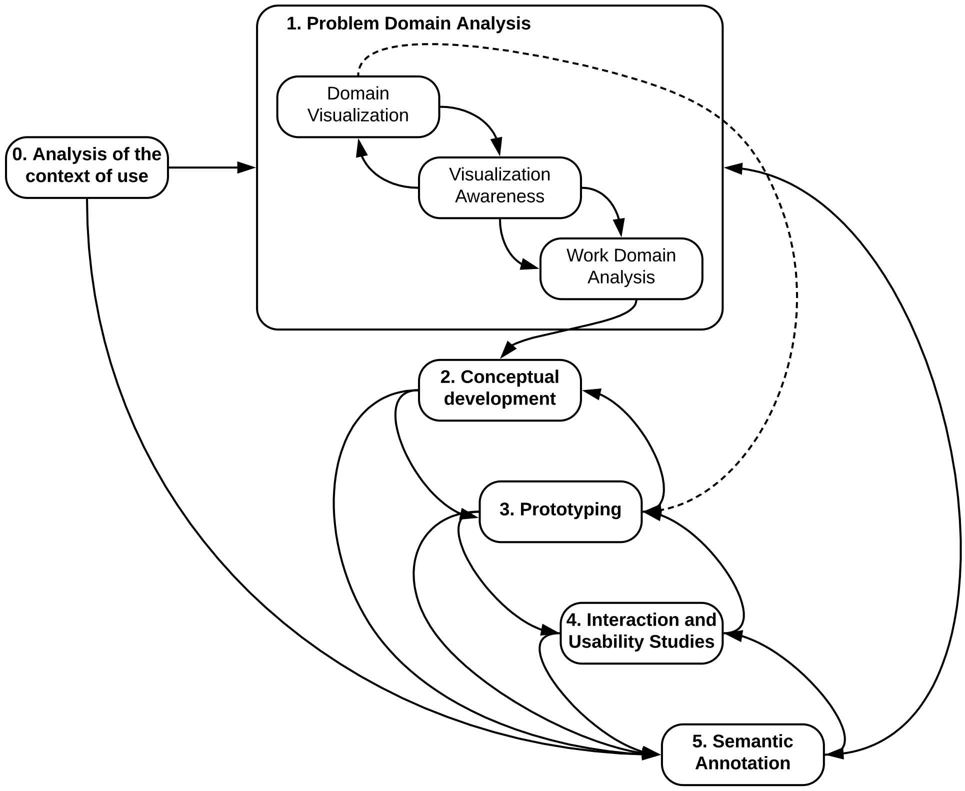


Figure 4.19: An example of user-centered design process based on the work of Koh et al. [35], with an extension of a new phase for semantic annotation of the artifacts that were generated in the preceding phases

`Rating` and `CrossRating`. Additional feedback can be included afterwards, provided that the tool allows users to add ratings and feedback over time.

As seen in the previous sections of this chapter, VUMO already provides a set of instances to describe analytical profiles, e.g. `Strategic` and `Operational`; interaction tasks, e.g. `SemanticZoom`, among others. It is not expected that developers are restricted to the available instances. For instance, a system may require the specification of users with a tactical profile. To meet that requirement, a new instance `ns:Tactical` can be created, or reused from another ontology, and related to the VUMO ontology by asserting the instance as belonging to the class `AnalyticalProfile`. The prefix `ns` was added to indicate that contents that are not originally declared in VUMO should not be declared as belonging to VUMO namespace, in accordance to good practice standards for RDF data. Preferably, a namespace should be created for storing data related to a specific visualization system implementation.

4.12 Discussion

This section provides a critical analysis of the VUMO ontology in terms of completeness and scalability. We discuss the proposed formalization of spatiotemporal data, visualization techniques and empirical domain user knowledge, in terms of the corresponding implementations of these concepts in the VUMO ontology.

Table 4.8: Summary of artifacts that are subject to semantic annotation, according to each phase of a UCD methodology and the three pipelines for the deveopment of semantically-rich, knowledge-assisted visualization systems

Pipeline	Phase	Artifacts for semantic annotation
1. Data integration	Transversal to all phases	Integrated urban mobility data
2. Visualization technique design and development	Problem domain analysis	Analytical profile Domain users Data transformations
	Conceptual development and prototyping	Visualization techniques
3. Visualization evaluation and specification of system users	Interaction and usability studies	Rating and cross ratings

The pyramid framework is the starting point for the formalization of spatiotemporal data, which can be decomposed in *where*, *when* and *what* perspectives (data component). In VUMO, the *where* perspective can be modeled with subproperties of `hasSpatialReference`. The *when* perspective can be modeled with subproperties of `hasTemporalReference` for literal values, such as timestamps, or `hasTemporalResourceReference`, for resources which hold more complex constructs, such as time intervals, or recurring periods, e.g. ”Weekdays” or ”Holidays”. Spatial references are based on different types of points and point sets. Temporal references are based on instant and interval granularities. The studies surveyed in the literature review, and the datasets which we analyzed prior building the ontology, show evidence that the proposed spatial and temporal constructs enable the effective modelling of spatiotemporal data that is used for visualization purposes. On regards to scalability, new subproperties can be instantiated depending on the practical context in which VUMO should be applied, provided that it conforms to the aforementioned superclasses. The thematic description of an event, which defines the *what* perspective, is possible by creating an instance of a subclass of `Event`. The proposed major classification of events is able to cover the most important types of data found in literature: travel events, travel intentions, and unexpected events. The event instance is a semantic object (knowledge component) that belongs to a taxonomy defined by the VUMO ontology classes. Partonomy relations are defined by properties that link an event instance to other instances, such as other events. Thematic attributes can also be defined as subproperties of `hasMeasure`.

The formal representation of data transformations in terms of SPARQL queries and the SPIN vocabulary provides a standards-based representation of analytical tasks as a graph, which is independent of the syntax and schema of source datasets. Hence, this allows the reuse of analytical tasks across systems. We exploited that representation by defining a set of rules that is able to identify the urban mobility themes (tags) related to data transformations, and features that are relevant for the visualization process, including the compatibility of transformations with visualization techniques. Additional rules can be built using the same rationale to meet specific system requirements.

The semantic annotation of visualization techniques and empirical knowledge is independent of a sytem's logic and technological stack. New instances and subclasses of `VisualizationConcept` can be created, for instance, to include new types of interaction tasks. On regards to empirical knowledge, the proposed formalization is able to feed several types of recommendation methods, including those based on collaborative filtering (CF) and content-based (CB) approaches. CF approaches can retrieve user information from instances of domain users and their analytical profiles, and item information from technique ratings; cross ratings; themes extracted from data transformations, and the features of visualization techniques. CB approaches can rely only strict information about users, data transformations and visualization techniques. The *cold start* limitation of such approaches can be reduced, as VUMO provides the necessary semantics to evaluate a set of compatible visualization techniques. The structure of ratings allows, by construction, for scalability, as new instances of `RatingComponent` can be created, according to the criteria of recommendation methods.

4.13 Conclusions

This chapter described the components of the VUMO ontology and their role on modeling information for the development of visualization tools that are semantically rich, as they are based on a formal knowledge representation model, and user-centered, given the ontology capabilities of representing empirical domain user knowledge. VUMO has, as its basis, a conceptual model - also proposed in this chapter - which supports the following pipelines: (i) data integration, (ii) visualization technique design and development, and (iii) visualization technique evaluation and specification of system users.

To support pipeline (i) data integration, we focused on the definition of an event, based on acknowledge frameworks for general spatiotemporal data. We then extended the fundamental structures found in those frameworks to provide a link with visual properties: spatial reference types were defined. The visual spatial patterns they induce on a visualization canvas was defined as spatial distribution. The concept of transformation was defined as a sequence of operations in raw data that yield new information. We showed that the output variables of a transformation can also induce spatial distributions. Our model allows transformations to be related to categorical attributes that can be used to describe them.

For pipeline (ii), visualization technique design and development, we formalized the general structure of visualization techniques. Techniques can have a collection of features and interaction tasks. Input variables are responsible for connecting the output variables of compatible visualization techniques.

Finally, to support pipeline (iii), visualization technique evaluation and specification of system users and their feedbacks. The latter was branched into two categories: visualization technique rating, and cross rating, which consists of a more specialized rating that relates a visualization technique and a transformation. Rating components can consist of quantitative or qualitative values.

VUMO is fully built upon *de facto* industry and web standards for semantic data. Besides the inherent semantics of OWL and RDF, we proposed a set of rules and functions that extend the ontology capabilities in terms of extracting implicit knowledge. Rules are used to infer relevant visual properties from instance data according to our proposed conceptual model, and to extract features from data transformations. Moreover, VUMO is capable of evaluating the compatibility of a data transformation with one or more visualization techniques. Embedded functions allows tools to retrieve information about users and their asserted ratings, which can be exploited by visualization tools to evaluate the appropriateness of visualization techniques

Finally, we showed how VUMO can support user-centered design process of visualization systems development, by aligning the artifacts generated by that process and their respective semantic annotation using VUMO classes and properties.

Chapter 5

Practical applications

This chapter describes practical applications of VUMO to the various phases of development and use of semantically rich, user-oriented visualization systems that support heterogeneous data sources. As described in Table 4.8 (see Section 4.11), each pipeline and phase generate certain artifacts. To demonstrate how such artifacts are generated and annotated, we present case studies involving the cities of Porto, Portugal, and Boston, USA. The studies use real public transportation system data collected from transportation agencies, a transportation consulting company and government bodies. We also involved domain users with various backgrounds from both countries. To support the case studies, we developed a visualization system prototype named *SUMVis* (Semantic Urban Mobility Visualization), which acts as a testbed platform for the three visualization pipelines. Section 5.1 describes the technological stack of *SUMVis*.

Each of the remaining sections focuses on specific visualization pipelines. In Sections 5.2 and 5.3, we focused on the data integration pipeline. We describe the datasets considered for each city, and demonstrate, with concrete examples, the process of mapping instance data to VUMO classes and properties. In addition, we also demonstrate how additional inference rules can be built to meet new domain user requirements. In particular, Section 5.3 exemplifies a modeling approach in which additional classes and properties had to be created, as they were not originally pre-defined in VUMO. This case is a useful reference for situations in which VUMO needs to be combined and related to concepts from other ontologies.

Section 5.4 describes the semantic annotation of prototypical visualization techniques, in the context of visualization of public transportation ridership, using data from the city of Porto. Data transformations are also defined to demonstrate the evaluation of visualization compatibility using inference rules. Moreover, the section also shows how integrated data can be annotated with additional knowledge found by domain users while manipulating *SUMVis*. The Sections 5.5 and 5.6 focus on the pipelines related to the design and evaluation of visualization techniques.

Section 5.5 describes a UCD study for visualization of OD flows in public transportation systems from various perspectives, e.g. stops, lines and fare zones. The aim was to evaluate a set of visualization prototypes against a set of analytical tasks (data transformations) that were built after discussion with potential users. Furthermore, we implemented an additional rule for inferring features from data transformations, based on the results of the context analysis phase (see Phase 0 on Figure 4.19). This section demonstrates the semantic annotation of data transformations, visualization techniques, and empirical knowledge in the form of ratings. We also demonstrate a prototypical VUMO-based recommendation method using empirical knowledge and functions.

Finally, Section 5.6 describes an exploratory study in which we developed a set of visualization techniques for emissions in bus corridors, in the context of CoAXs [134], a visualization tool developed by MIT to foster stakeholder engagement on transit planning, as part of our temporary collaboration on this project. This section demonstrates the semantic annotation of analytical tasks and visualization techniques. In particular, the analytical tasks differ from the ones shown in Section 5.5, as they contain elements that were not originally part of the VUMO ontology.

We argue that the presented examples oriented to specific pipelines corroborates to showing how VUMO is able to support all phases development phases, taking into account different contexts. Moreover, the examples that demonstrate the extension of VUMO ontology to meet new user requirements . Table 5.1 provides a high-level map for the case studies presented in this chapter, along with their contributions to each pipeline.

Table 5.1: Practical references within each pipeline, and the sections in which they can be found

Practical references within each pipeline	Sections
1. Data integration	
- Mapping data onto VUMO components	5.2, 5.3
- New rules	5.2
- New functions	5.3
- New classes and properties	5.2, 5.3
- Annotation of discovered knowledge after analytical task	5.4
2. Visualization technique design and development	
- Annotation of data transformations	5.4, 5.5, 5.6
- Annotation of visualization techniques	5.4, 5.5, 5.6
3. Visualization evaluation and specification of system users	
- Annotation of empirical knowledge (ratings)	5.5, 5.6
- Development of a recommendation method	5.5

5.1 *SUMVis* Architecture

This section describes the technological stack that was used to develop the *SUMVis* (Semantic Urban Mobility Visualization) tool prototype that supports the presented case studies. *SUMVis* is a browser-based tool that is coupled to a GraphDB[1] triple store engine as backend, for storing RDF data, and an OWL RL reasoner capable of executing SPIN rules. The triple store engine also provides a SPARQL endpoint for querying instance data, with extended support to geospatial queries (GeoSPARQL). The chosen technological stack is not restricted to commercial licenses. Figure 5.1 shows an overview of the *SUMVis* architecture.

External data sources (raw data) were mapped onto the RDF metadata model using classes and properties from VUMO and other ontologies. The process of mapping was supported by *ad hoc* parsers which were created for that purpose. The resulting linked data was then stored into the triple store database, in which rule-based inference was executed during runtime. The mapping process is described in Sections 5.2 and 5.3.

In the frontend, users are able to select, during their sessions, the desired analytical task. In Section 5.5, we describe a recommendation strategy that provides a rank of appropriate visualizations based on a globally calculated rating, i.e. all ratings made by domain users. Prototypes

[1]http://graphdb.ontotext.com

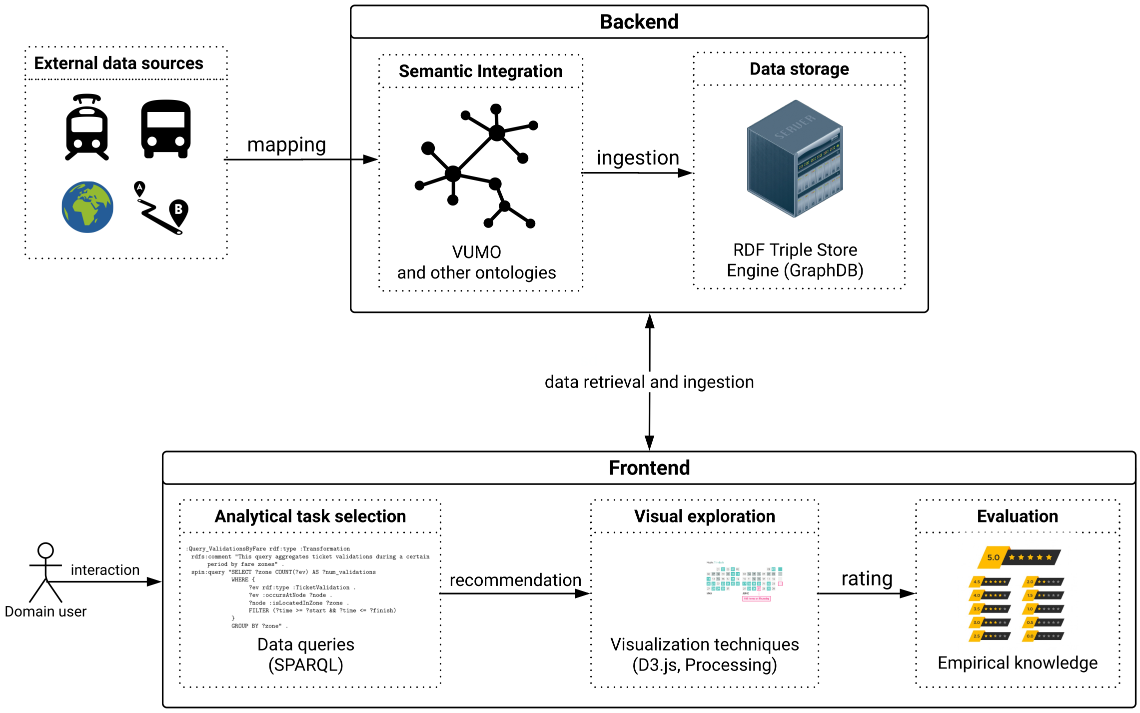


Figure 5.1: The architecture of the SUMVis prototype

of visualization techniques were developed with D3.js[2], Processing[3] and Leaflet[4]. After visual exploration, the user can evaluate the visualization technique according to various criterias. For that purpose, a simple interface was developed to facilitate collection of user feedback, which is then translated into instance data and inserted into the triple store engine. The following computer specifications were used: Intel Core i7 @ 3.1GHz, 32GB of RAM, and 2TB of hard disk space.

5.2 Semantic integration: the case of Porto, Portugal

The metropolitan area of Porto consists of 17 cities and approximately 1.76 million inhabitants, thus being the second largest urban area in Portugal. The metropolitan area is largely supported by the Andante intermodal system. The Andante system is an entry-only fare system consisting of public and private operators. Passengers can commute by bus, subway, train and tram. Passengers start their trip by validating their Andante paper ticket (occasional travels) or plastic card (monthly passes). Validators are located inside vehicles, and in subway and train stations. Citizens can retrieve real-time information about public transportation conditions and plan journeys. The main operators are Sociedade de Transportes Colectivos, S.A. (STCP), which is responsible for operating the buses in Porto region, and Metro do Porto, S.A. (MP), which operates the subway service.

The metropolitan area is divided into several fare zones, e.g. N6, in which the prefix denote its

[2] https://d3js.org
[3] https://processing.org
[4] https://leafletjs.com

geographic macroregion (N - North, C - Center, S - South). Passengers pay a fare in accordance to the zones they have to cross. Figure 5.2 shows a map of fare zones. For instance, the downtown area is located in zone C1. Passengers that subscribe to monthly passes have a pre-defined set of zones in which they can use public transportation.

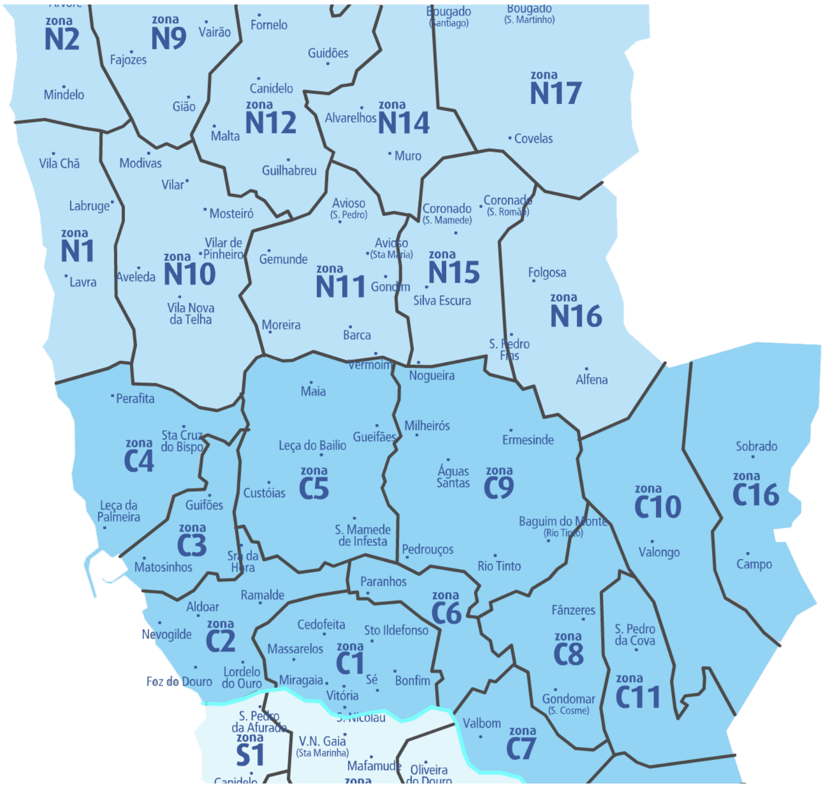


Figure 5.2: Zones of the Andante system in Porto, Portugal

Four datasets from different sources were collected for this study:

- PRT1: Ticket ridership from Andante system;
- PRT2: Information requests from a mobile application that provides real-time information about the public transportation system;
- PRT3: Estimated OD matrix from Andante system;
- PRT4: Shape files for all counties of the metropolitan area of Porto.

5.2.1 PRT1: Ticket ridership from Andante system

The ridership dataset (PRT1) is supported by auxiliary datasets that describe the Andante system:

- PRT1_TICKETS: Description of ticket categories;
- PRT1_ZONES: Description of fare zones, as shown in Figure 5.2;
- PRT1_STOPS: Descriptions of stops within each fare zone;

- PRT1_OPERATORS: Description of active operators.

These datasets were available as CSV files and were provided by *Transportes Intermodais do Porto* (TIP), which is the company that runs the Andante system. Every record in PRT1 is a row that describes a ticket validation. Each row introduces redundant information about the respective ticket, such as the allowed fare zones – in case it is a subscription ticket – and the ticket type. Figure 5.3 shows the non-relational schema of the main and auxiliary datasets.

The namespace `portotip` describes integrated data related to PRT1, in order to not overlap the default `vumo` namespace.

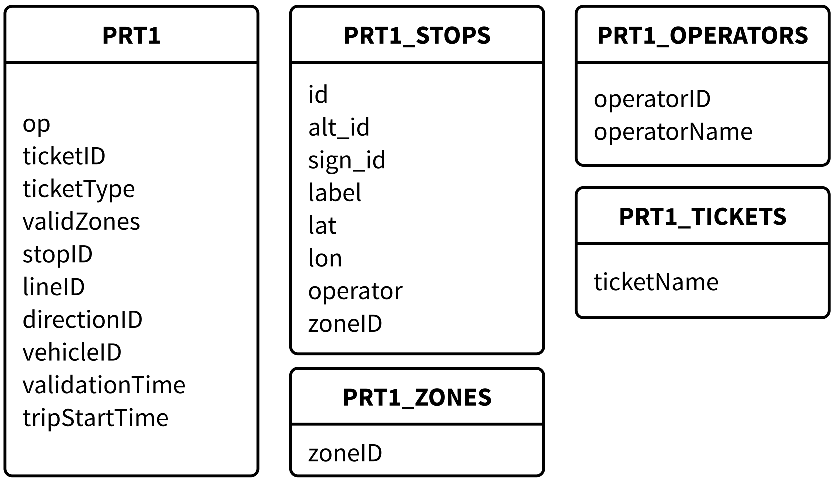


Figure 5.3: Schema of PRT1 and its auxiliary datasets

The Andante system offers several types of transportation tickets which belong to the following categories:

- ”Título ocasional” (Occasional travels): a paper ticket that can be used for several journeys within a given time limit, e.g. 1 hour;
- ”Assinatura Mensal” (Monthly subscription): a card for passengers that opt for a pre-paid monthly subscription. The subscription allows passengers to make unlimited travels within the set zones they paid for;
- ”Título Diário” (Day ticket): a card that allows passengers to make unlimited travels within all zones within a 24-hour period, which starts at the time of the first validation.

The PRT1_TICKETS dataset has a single column that describes all ticket types (`ticketType`). Some of the ticket types could be mapped onto the subclasses `Pass` and `OccasionalTicket`. Given that names could have white spaces and characters with diacritics, the parser defined a URI for each ticket type by removing leading spaces and removing the diacritics. The actual ticket type name is preserved and mapped onto the property `hasName`, as in the following example:

```
portotip:TituloOcasional rdf:type :OccasionalTicket ;
  :hasName "Título Ocasional" .
```

The PRT1_ZONES dataset has a single column that describes the code of each zone, e.g. N10. This dataset has no geographical information about a zone's boundary points. Therefore, each zone becomes an instance of `Zone` by asserting a triple of the form below.

`portotip:N10 rdf:type :Zone .`

In PRT1_STOPS, stops are uniquely identified by two alternative attributes: internal and external identifiers. The former is meant to be used by the operator itself and the Andante system. The latter provides a user-friendly identification to passengers, as described in Section 4.4. Table 5.2 shows a possible mapping between the source attributes and target properties.

Table 5.2: Mapping between attributes of PRT1_STOPS and VUMO properties

Source attribute	Target property	Property type
`id`	`hasInternalID`	Datatype
`alt_id`	`hasInternalID`	Datatype
`sign_id`	`hasFriendlyID`	Datatype
`label`	`hasName`	Datatype
`lat`	`geo:lat`	Datatype
`lon`	`geo:long`	Datatype
`operator`	`operatedBy`	Object
`zoneID`	`isLocatedInZone`	Object

The mapping of the attributes `operator` and `zoneID` required special attention, as the values of these attributes are translated into instances of VUMO classes `Operator` and `Zone`, respectively. The parser's output for those attributes consists of an attribute's value preceded by the namespace, e.g. `portotip:STCP` and `portotip:N10`. The URI of bus stops and subway stations are given after their `sign_id` source attribute.

The PRT1_OPERATORS also share the same simple structure, hence the resulting triples have the form `portotip:STCP rdf:type :Operator`.

```
portotip:STCP rdf:type :Operator ;
  :hasName "Sociedade de Transportes Coletivos do Porto" .

portotip:MP rdf:type :Operator ;
  :hasName "Metro do Porto" .
```

In case these triples have not been asserted, it would still be possible to infer that instances `portotip:STCP` and `portotip:MP` belong to the class `Operator`, due to the range of the property `operatedBy`.

The main PRT1 dataset of validation records include information about the respective operator and the vehicle that registered the validation (`op` and `vehicleID`), ticket's ID and the zones in which it can be used (`ticketID` and `validZones`), the IDs of the line, its direction and stop in which the validation occurred (`lineID`, `directionID` and `stopID`), and the timestamp of the validation (`validationTime`). For validations that occurred in buses, the timestamp related to the trip start is also recorded (`tripStartTime`). Table 5.3 provides an example of a validation record.

Two additional auxiliary datasets were manually created. PRT1_LINES and PRT1_ROUTES datasets were created by scraping the websites of STCP and MP to retrieve information about

Table 5.3: Example of ticket validation

Source dataset attributes	Value
`op`	STCP
`ticketID`	20023201283
`ticketType`	Assinatura Mensal
`validZones`	C1 C2 C6 C9
`stopID`	14
`lineID`	805
`directionID`	2
`vehicleID`	2107
`validationTime`	2013-01-22 07:30:56
`tripStartTime`	2013-01-22 07:23:34

lines and their respective routes. In this context, we use the VUMO definition of lines and routes (see Table 4.4), i.e. a line may be formed by one or more routes (e.g. inbound, outbound). Each route is defined by a route path, which is formed by route segments that connects two nodes.

Our particular approach to modeling lines and routes consisted of providing granular information about them, i.e. modeling all route paths as an ordered list, taking advantage of the available VUMO components and the standard RDF semantics for representing ordered lists, i.e. with `rdf:first` and `rdf:rest` properties. The path of a route - an ordered list - is represented using the standard RDF properties: a chain of blank nodes where the first element (segment) is declared, along with the rest of the chain. The serialization below exemplifies the modeling of lines and routes. Figure 5.4 shows the visual representation for the same serialization.

```
portotip:Line204 rdf:type :Line ;
  :operatedBy portotip:STCP ;
  :hasDescription "204 - HOSP S. JOAO X FOZ" ;
  :hasInboundRoute portotip:Line204_IB ;
  :hasOutboundRoute portotip:Line204_OB .

portotip:Line204_IB rdf:type :Route ;
  :hasRoutePath [ rdf:first portotip:Segment120 ;
                  rdf:rest [ rdf:first portotip:Segment121 ;
                             rdf:rest [ ... ] ] . ] .
```

The previous serializations illustrated the mapping from auxiliary datasets onto VUMO properties and classes, thus yielding semantic data about the infrastructure components of the Porto public transportation system. Table 5.4 shows the proposed mapping for PRT1.

Each row on the source dataset yields nine triples, as in the example below. Differently from the illustrative example of Figure 4.15, we did not instantiate vehicle trips as objects, as in our particular case, the dataset did not contain additional information about the trip, e.g. finish time. The chosen URI for each ticket validation was `portotip:TV_N`, where `N` is the row number corresponding to the validation record in the source dataset. Tickets were instantiated as members of the `Ticket` class, and their type was defined with the `hasTicketType` property, marked with an asterisk in Table 5.4.

```
portotip:TV_101 rdf:type :TicketValidation ;
```

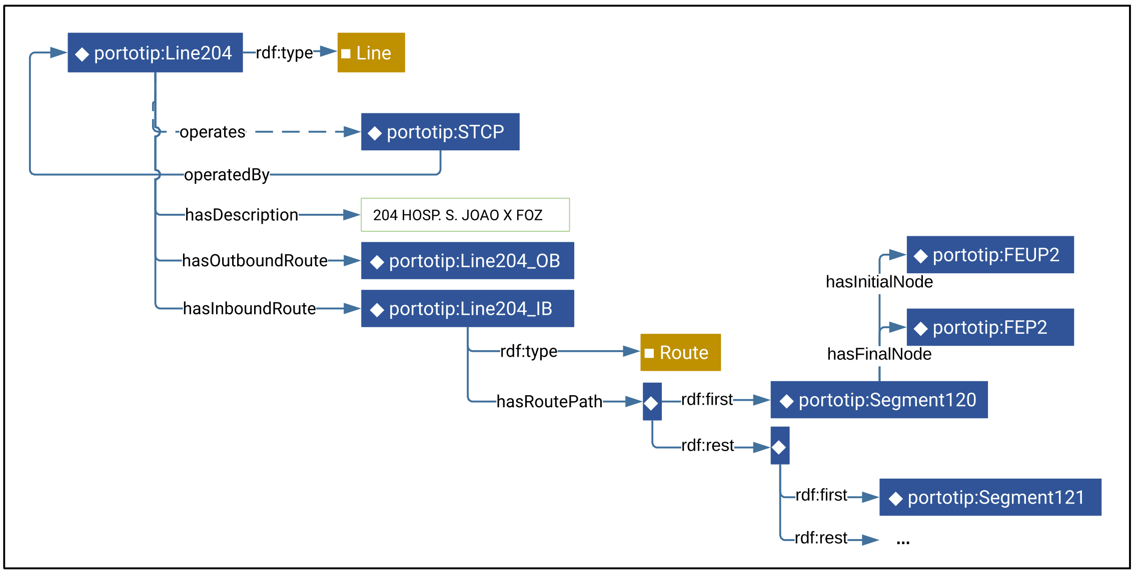


Figure 5.4: Visual representation of the triples that describe a line and its routes. A route path is described as an ordered list, using the standard RDF properties `rdf:first` and `rdf:rest`

Table 5.4: Mapping between attributes of PRT1 and VUMO properties

Source attribute	Target property	Property type
`op`	`hasOperator`	Object
`ticketID`	`hasTicket`	Object
`ticketType*`	`hasTicketType`	Object
`stopID`	`occursAtNode`	Object
`lineID`	`occursInLine`	Object
`directionID`	`occursInRoute`	Object
`vehicleID`	`occursInVehicle`	Object
`validationTime`	`hasDateTime`	Datatype
`tripstartTime`	`hasTripStartDateTime`	Datatype

```
  :hasOperator portotip:STCP ;
  :hasTicket portotip:Ticket118 ;
  :occursAtNode portotip:FEUP2 ;
  :occursInLine portotip:Line204 ;
  :occursInRoute portotip:Line204_IB ;
  :occursInVehicle portotip:2211 ;
  :hasDateTime "2013-01-05T20:36:01" ;
  :hasTripStartDateTime "2013-01-05T20:33:21" .

portotip:Ticket118 :hasTicketType portotip:TituloOcasional .
```

Given that we did not have a dataset containing information about the STCP fleet but we still desired to keep track of such information, we implemented an additional rule that infers that a vehicle belongs to a certain operator, whenever an event has triples that indicate the operator (`hasOperator`) and vehicle (`occursInVehicle`) simultaneously. Formally, the rule can be represented as:

$$\texttt{hasOperator}(e, o) \wedge \texttt{hasVehicle}(e, v) \rightarrow \texttt{owns}(o, v)$$

Where $e \in$ `Event`, $o \in$ `Operator` and $v \in$ `Vehicle`.

In the subway system of Porto, ticket validation checkpoints are located in several locations inside a station. As stations are usually served by more than one line, the recorded information about those validations contains less information about the trip, as illustrated in the following example.

```
portotip:TV_211 rdf:type :TicketValidation ;
  :hasOperator portotip:MP ;
  :hasTicket portotip:Ticket594 ;
  :occursAtNode portotip:TRD ;
  :hasDateTime "2013-01-18T19:01:48" .
```

The resulting graph for one month of ticket validations yielded approximately 140 million triples, totaling 8GB of disk space.

5.2.2 PRT2: Information requests from a journey planner mobile application

Move-me[5] is a mobile application developed by Optimização e Planeamento de Transportes S.A. (OPT)[6] that provides real time information about public transportation. Given that the application is a means of providing information about transportation, it is unknown whether the user who requested information actually travelled or not. Move-me currently provides three types of information:

- *Next departures*: provides the next bus/train/subway departures for a specific stop or station selected by the user;
- *Route finder*: for a given origin and destination selected by the user, not necessarily a predefined stop or station, the app provides a sequence of public transportation routes that can be taken in order to reach the destination;

[5] http://www.move-me.mobi/
[6] http://www.opt.pt/

- *Nearby stops*: provides the nearby station or stops based on the location provided by the user (assisted by the user's mobile phone GPS). It is also possible to select a search radius.

The dataset was extracted from a relational database management system (RDBMS) as a table dump. It consists of a single table that contains users' information requests, where each request was conceptually regarded as a travel intention, i.e. an intention to use public transport. Every travel intention is tied to a timestamp. Each record contains information about the type of request, according to desired service.

The namespace `portoopt` was used for semantic integration of this dataset. Table 5.5 lists the relevant attributes of PRT2. Each record contains a user's input that depends on the type of information. The attribute `requestDateTime` holds the timestamp corresponding to the time of the request. The attribute `requestType` holds a string that encodes the type of service that the user requested. It is used by the application's logic to determine which `requestInput` is expected. The latter attribute also holds a string that encodes the input parameters required by each service. The *Next Departures* service requests the code of a stop.

Table 5.5: Source attributes from PRT2 dataset

Attribute	Type
`requestDateTime`	Timestamp
`requestType`	String
`requestInput`	String input depends on `requestType`: • *Next departures* Stop/station ID • *Route finder* Origin Destination • *Nearby stops* Latitude Longitude Search radius

For an information request about next departures, the input is the internal ID of a stop. For example: "TRD3". It is important to note that this ID does not necessarily match the ones from PRT1_STOPS, in which the respective ID is "391", as the datasets were retrieved from different sources.

For a request about a route plan, the input is a pair of origin and destination points. Given that this input type is more complex than the other two input types, the original dataset encodes the records in XML, as depicted in the following serialization, which describes a travel plan between the "São Bento" subway station and the "Francelos" train station.


```
<Request>
  <StartTime>2013-11-11T18:50:22.16+00:00</StartTime>
  <EndTime>2013-11-11T23:50:22.16Z</EndTime>
  <Type>RouteFinder</Type>
  <Route>
<Track> <In>
```

```
      &lt;Name&gt;Sao Bento&lt;/Name&gt;
      <Code>Sao Bento</Code>
      &lt;Type&gt;Stop&lt;/Type&gt;
     &lt;Provider&gt;METRO DO PORTO&lt;/Provider&gt;
    &lt;/In&gt;
    &lt;Out&gt;
      &lt;Name&gt;Francelos&lt;/Name&gt;
      <Code>CP_Francelos</Code>
      &lt;Type&gt;Stop&lt;/Type&gt;
     &lt;Provider&gt;CP&lt;/Provider&gt;
    &lt;/Out&gt;
    &lt;VisitTime&gt;0&lt;/VisitTime&gt;
    &lt;ProviderCode /&gt;
  &lt;/Track&gt;
 &lt;/Route&gt;
&lt;MaxResultTrips&gt;3&lt;/MaxResultTrips&gt;
&lt;/Request&gt;
```

For an information request about nearby stops, the input consists of two elements: a pair of geographic coordinates in WGS84 format, which represents the current user's location or any other location selected by the user, and the desired search radius in meters, e.g. `(41.1306, -8.5867, 500)`.

Data primitives from the "Near stops" service already depict the spatial dimension as geographic coordinates. We used an auxiliary Move-me dataset, named PRT2_STOPS, which contained the geographic coordinates and information about stops and stations. They were mapped in a similar way as in Table 5.2.

Datasets PRT1 and PRT2 share similarities on regards to information about the public transportation system. In fact, both describe stops, stations and operators. For instance, the subway station `portoopt:MP30` is semantically equivalent to `portotip:TRD`, even if they are described in different ways. OWL allows the semantic alignment of instances by using the property `owl:sameAs`. Hence, the statement `portoopt:MP30 owl:sameAs portotip:TRD` ensures that both instances are interpreted as being equivalent, which allows, for example, to infer that validations and requests related to those instances actually refer to the same location. Such possibility would not be possible, at least with difficult technical workarounds, on systems that make use of relational databases. The following rule was built to infer that instances are equivalent if they share at least one equivalent ID. If n_1 and n_2 are instances of `Node`, and if the string literals corresponding to the IDs are equivalent. Formally:

$n_1, n_2 \leftarrow$ instances from `portotip` and `portoopt`, respectively
$x_1, x_2 \leftarrow$ IDs asserted with property `hasInternalID`
if $(n_1, n_2 \in \texttt{Node}) \wedge (x_1 = x_2)$ **then** $\texttt{owl:sameAs}(n_1, n_2)$

Tables 5.6, 5.7 and 5.8 show the proposed mappings for each request type. Figure 5.5 shows three examples of the resulting RDF graph; each of them belongs to a specific travel intention.

5.2.3 PRT3: Estimated origin-destination flows in the public transportation system

This dataset contains information about estimated origin-destination flows between stops and stations operated by STCP and Metro do Porto, using validation data extracted from PRT1. The

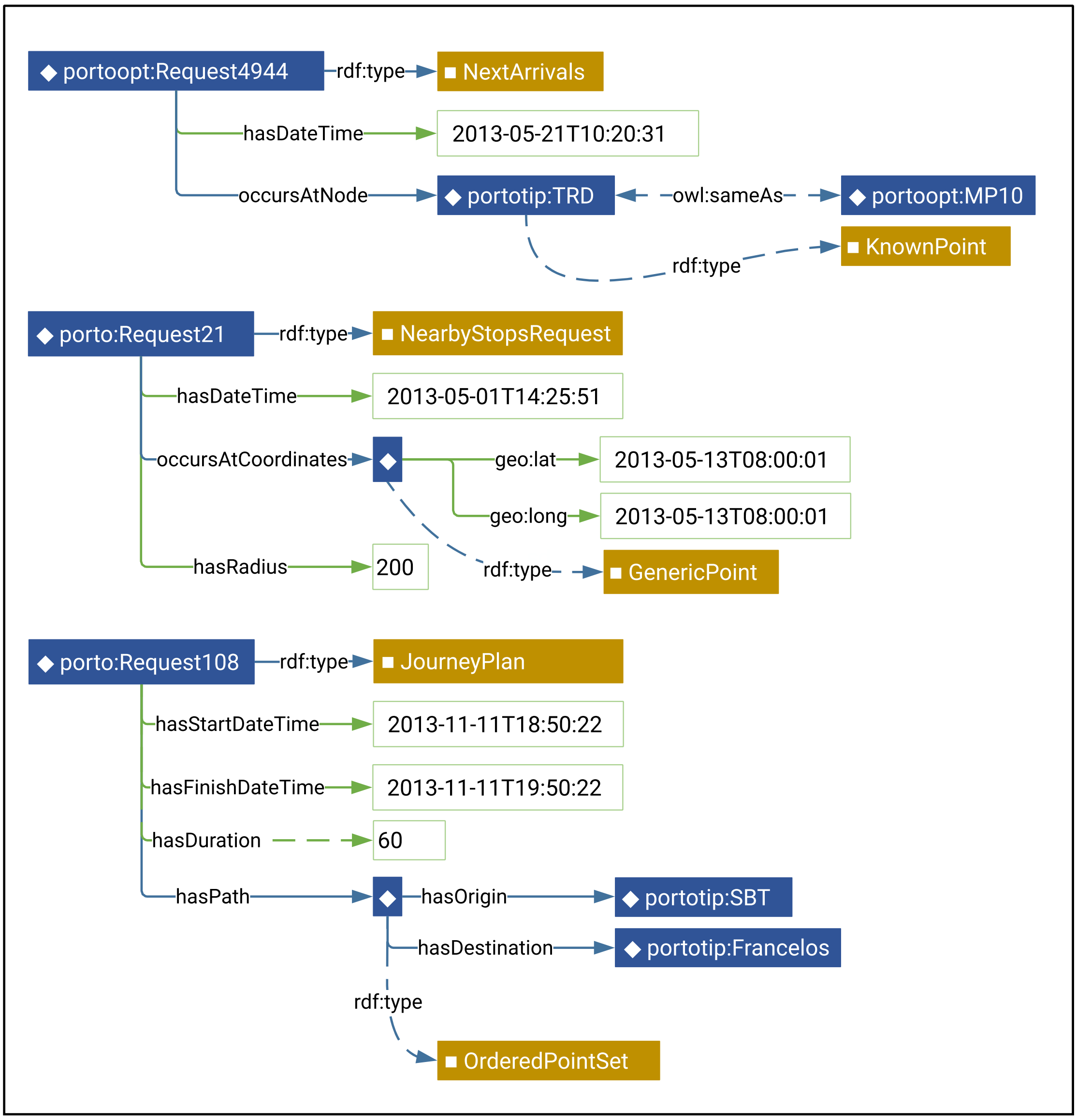


Figure 5.5: RDF graph containing instances from PRT2

Table 5.6: Mapping between attributes of PRT2 (Next departures) and VUMO properties

Source attribute	Target property	Property type
`requestDateTime`	`hasDateTime`	Datatype
`requestInput`	`occursAtNode`	Object

Table 5.7: Mapping between attributes of PRT2 (Route finder) and VUMO properties

Source attribute	Target property	Property type
`requestDateTime`	`hasStartDateTime`	Datatype
	`hasFinishDateTime`	Datatype
`requestInput`	`hasPath`	Object
	`hasOrigin`	Object
	`hasDestination`	Object

flows were calculated according to the algorithm proposed by Hora et al, in which the author of this work was involved [170]. The matrix is based on one week of ticket validations. The structure of the dataset consists of an Excel XLS sheet containing a square matrix M, in which rows and columns represent all stops and stations, thus $M(i, j)$ represents the number of expected passengers between i and j, which may be different from $M(j, i)$. The namespace `portood` was used for this dataset. As this dataset was built upon existing information from PRT1, we benefit from using spatial references that belong to the `portotip` namespace.

Each cell $M(i, j)$ was mapped onto an instance of `ODFlow`, described by the property `hasPath`, which refers to a blank node described by the properties `hasOrigin` and `hasDestination`. The chosen URI for each flow record was defined as `portotip:ODFlow_N`, where `N` is the number corresponding to a cell in M. The following example shows an OD flow of two passengers for two stops:

```
portood:ODFlow_431 rdf:type :ODFlow ;
  :hasPath [ :hasOrigin portotip:FEUP1 .
             :hasDestination portotip:RFAR2 ] ;
  :hasFlow 2 .
```

In this example, it is possible to infer that such OD flow goes from fare zone `portotip:C6` to `portotip:C2`, given the existing asserted information about the zones in which each stop is located. The same applies to all instances of `ODFlow`.

Table 5.8: Mapping between attributes of PRT2 (Nearby stops) and VUMO properties

Source attribute	Target property	Property type
`requestDateTime`	`hasDateTime`	Datatype
`requestInput`	`occursAtCoordinates`	Object
	`hasRadius`	Object

5.2.4 PRT4: Geographic information of Porto's boroughs

We collected shape data from all boroughs of the city of Porto, from Direção-Geral de Território, an official Portuguese government body[7]. A preliminary data conversion consisted of transforming ArcGIS shape files into Well-known Text (WKT) format, which is an open standard for describing shapes in plain-text form. Figure 5.6 shows a visualization of Porto boroughs. The namespace `portogeo` was used for this dataset.

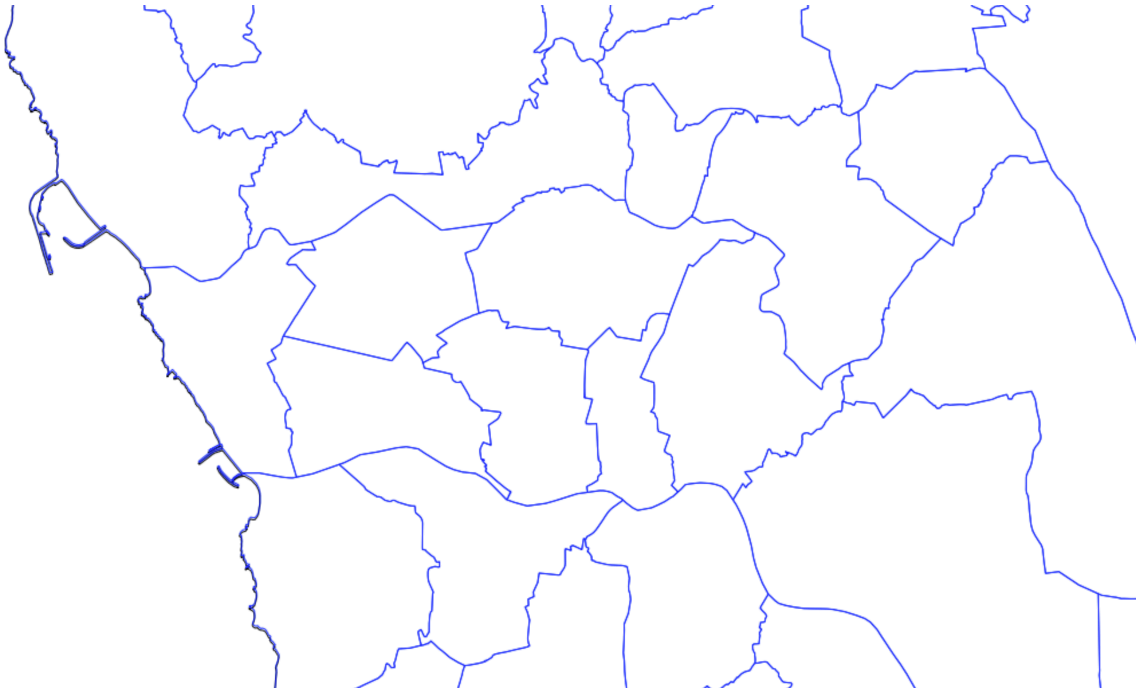


Figure 5.6: Spatial data representation of Porto boroughs

The typical structure of a polygon in WKT is given by the following example:

```
POLYGON ((30 10, 40 40, 20 40, 10 20, 30 10))
```

Every borough was instantiated as an instance of `schema:AdministrativeArea`, which is asserted in VUMO to be a subclass of `Zone`. The VUMO datatype property `hasWKTGeometry` was used to store the plain-text information about shapes in WKT format, which can be processed with GeoSPARQL. Given that spatial references are treated as objects (instances) according to VUMO semantics, the implemented rules would not be able to infer that boroughs are, in fact, point sets. Furthermore, points that form the shape would not be recognized as instances of `Point`. Although RDF databases support the execution of spatial queries, GeoSPARQL is still not standardized, but its use is heavily encouraged. Therefore, a feasible, standard-compliant workaround consisted of implementing a parser capable of translating WKT into standard RDF representation of ordered lists, thus providing an alternative representation besides the use of `hasWKTGeometry` with the usual VUMO components: the property `hasPath` was used to represent the polygon. As a result, `portogeo:Paranhos` is inferred as an `OrderedPointSet`. The `assignedBy` property is used to identify the entity that defined the zone.

[7]http://snig.dgterritorio.pt/portal/

The example below describes both representations for the same borough. An excerpt of the values was shown due to size restrictions.

```
portogeo:Paranhos rdf:type schema:AdministrativeArea ;
  :assignedBy portogeo:DGT ;
  :hasID "Paranhos" ;
  :hasName "Freguesia de Paranhos" ;
  :hasWKTGeometry "POLYGON (( -8.6528 41.1726,
   -8.6242 41.1710..."^^geosparql:wktLiteral ;
  :hasPath [ rdf:first [ geo:long -8.6528 . geo:lat 41.1726 . ] ;
             rdf:rest [ rdf:first [ geo:long -8.6242 41.1710 ; ] .
                        rdf:rest ... ] . ] .
```

We implemented a persistent SPIN rule that assigns every spatial reference of type `Point` to its respective administrative area. The SPIN rule takes advantage of GeoSPARQL operators such as `geosparql:within`, which evaluates if a spatial entity is located within the limits of a polygon. This rule reveals implicit knowledge from stops, stations, sensors and any other node about their location, including any others that could be integrated in the future. The rule is formally represented as the following:

for each $p \in$ `Point`
  for each $z \in$ `schema:AdministrativeArea`
    if `geosparql:within`(p, z) **then** `isLocatedInZone`(p, z)

5.3 Semantic integration: the case of Boston, USA

The metropolitan area of Boston has approximately 5.8 million inhabitants, being the most populous of the state of Massachusetts. The public transportation system, known as the T, is intermodal, consisting of bus, subway, train and ferry services. The T is operated by the Massachusetts Bay Transit Authority (MBTA).

We retrieved four datasets for this study:

- BOS1: Description of the MBTA network, e.g. lines, routes and stops;
- BOS2: Estimated OD data for bus and subway services;
- BOS3: Estimated average speeds of route segments in bus corridors;
- BOS4: CO_2 emission profiles for various bus engine technologies.

5.3.1 BOS1: GTFS data

The descriptive information about the MBTA network is described by TXT files, in accordance to the General Transit Feed Specification [5]. It includes information about lines, routes, stops and stations. We took advantage of the GTFS ontology to build a direct mapping between source data and its terms. VUMO provides the required semantics to align the ontology's classes and properties to those in the GTFS ontology, thus data becomes described in terms of VUMO's constructs. The namespace `bostongtfs` was used for this dataset.

Figure 5.7 shows an RDF graph that describes the MBTA operator, a stop, and a line in terms of a route, analogously to the PRT1 dataset.

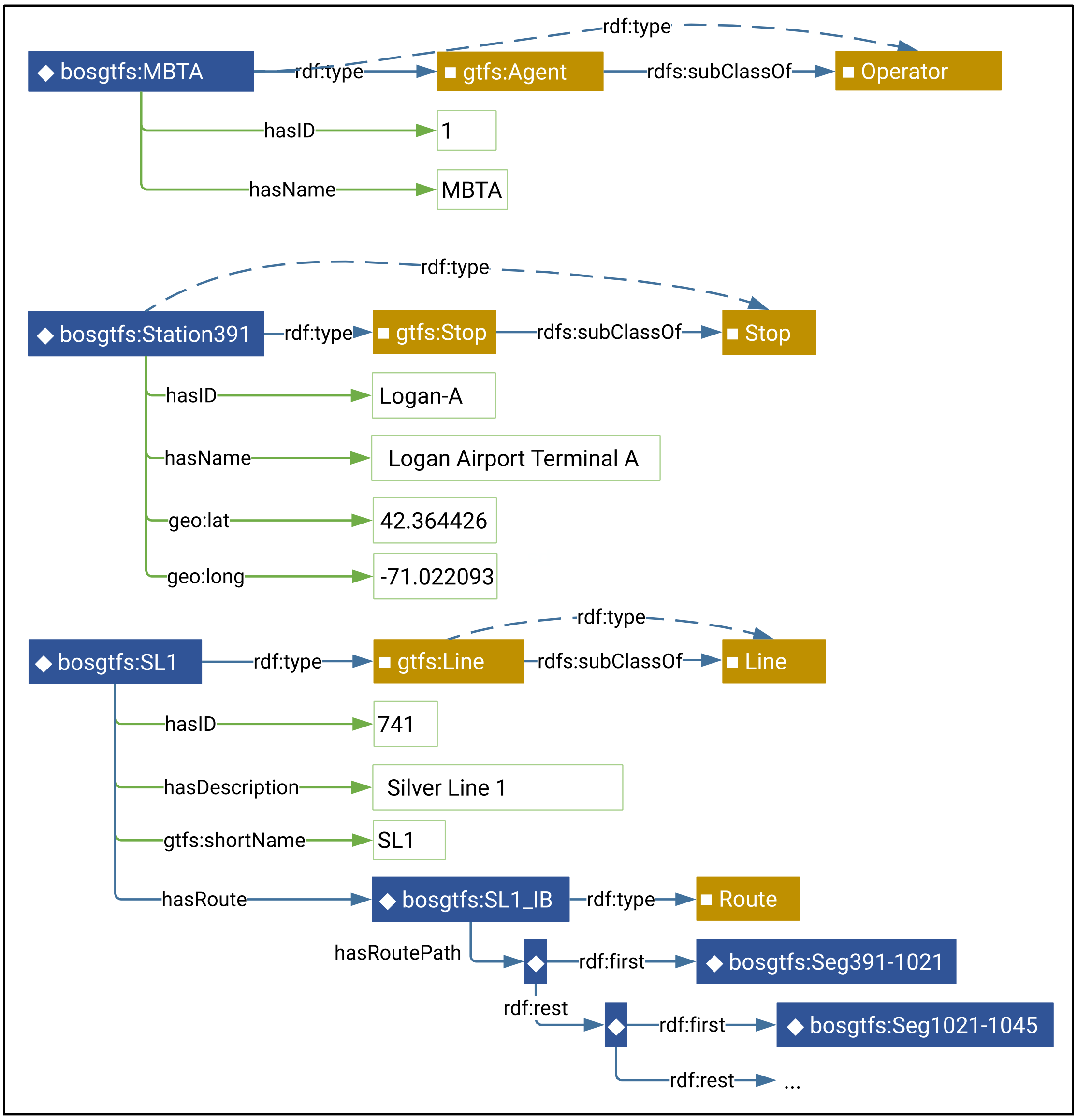


Figure 5.7: RDF graph of some elements of the MBTA network

5.3.2 BOS2: Estimated origin-destination data

This dataset provides estimated origin-destination flows based on a dynamic programming model proposed by Sánchez-Martinez, which produced more reliable results than previous models that have been tested against the MBTA network [171]. In comparison to the PRT3 dataset, BOS2 provides more granular information about OD flows, as it covers different time intervals: several periods for weekdays, e.g. AM and PM peaks, and school hours, and two periods for weekends: Saturdays and Sundays. The dataset contains flows for a week of September 2016, between stops and stations of the T network, which were coded in accordance to GTFS data from BOS1. Table

5.9 exemplifies a record of BOS2, which describes an OD flow between the subway stations of Davis (Red Line) and Fenway (Green Line). The `timePeriod` 11 corresponds to a Saturday.

Table 5.9: Example of a record related to an origin-destination flow between two subway stations of the T network

`origin`	place-davis
`destination`	place-fenway
`timePeriod`	11
`flow`	116

The namespace `bostonod` was chosen for this dataset. The `timePeriod` source attribute was transformed into instances of `Interval` an interval according to the original classification of time periods.

```
bostonod:saturday rdf:type :Interval;
  time:dayOfWeek time:Saturday ;
  :hasStartTime "00:00:00" ;
  :hasFinishTime "23:59:59" .
```

The chosen URI for each flow was defined as `bostonod:ODFlow_N`, where `N` corresponds to the row number in the file. During data conversion, `date` and `timePeriod` were merged into a single structure, so that each `ODFlow` could be characterized in terms of start and finish datetimes. The following example shows an instance of `ODFlow`:

```
bostonod:ODFlow_30 rdf:type :ODFlow ;
  :hasTimePeriod bostonod:saturdays ;
  :hasStartDateTime "2016-09-11T00:00:00" ;
  :hasFinishDateTime "2016-09-11T23:59:59" ;
  :hasPath [ :hasOrigin bostongtfs:place-davis ;
             :hasDestination bostongtfs:place-fenway ] ;
  :hasFlow 116 .
```

From VUMO semantics follows that the blank node referred in property `hasPath` is an `OrderedPointSet`.

5.3.3 BOS3: Estimated per-segment bus speeds

This dataset was retrieved from MBTA based on collected real-time information from buses with AVL system. For each route segment, the average speed was calculated based on all trips of the AM peak, which corresponds to the period between 7 a.m. and 8:59 a.m.. For the purposes of this case study, the data is related to bus routes of the Forest Hills corridor in South Boston, which consists of the following lines: 30, 34, 35, 36, 37, 40, 50 and 51.

The source schema is a plain-text CSV file. For each line, route segments are specified with two columns that indicate their start and finish stops. Stops are specified by their IDs, according to BOS1. Table 5.10 shows a record of average speed for line 30, inbound, between a route segment. `direction` is used to identify whether a route is inbound (1) or outbound (0); `stopID` and `nextstopID` correspond to a segment's start and finish points, respectively.

Table 5.10: Example of average bus speed of line 30, inbound, between two stops

line	30
direction	1
stopID	185
nextstopID	16458
avgmph	38.7899

The mapping strategy consisted of creating instances of SpeedReading, where each instance corresponds to a row in the source dataset, thus the spatial and temporal references correspond to the route segment and time interval (morning peak), respectively. Given that the designation morning peak corresponds to the same period used for BOS2 dataset, the same instance was used. Figure 5.8 shows the RDF graph corresponding to the example of Table 5.10. Instead relating the properties hasStartTime and hasFinishTime directly to SpeedReading30, we relate it to the temporal entity bosod:ampeak that already describes that information.

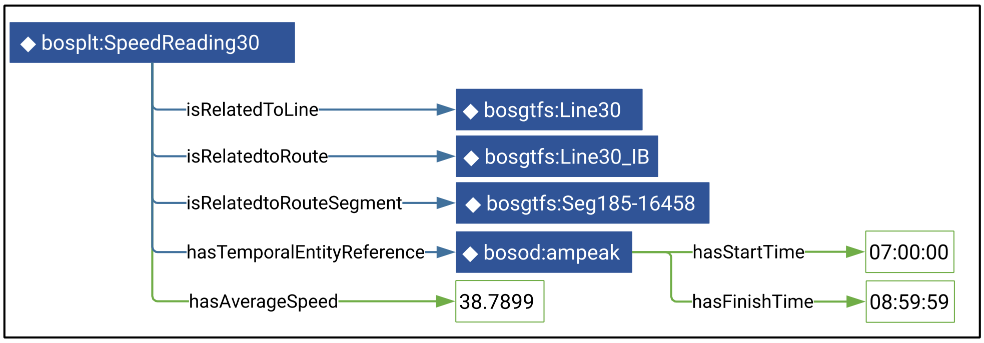


Figure 5.8: RDF graph of a speed reading based on the example of Table 5.10

5.3.4 BOS4: Emission rates for various bus engine technologies

This dataset was developed by the CoAXs team, and provides interpolated emission rates by speed bins for various pollutants, for the following bus engine technologies: diesel, diesel hybrid, CNG, electric, and hybrid. Pollutants include CO, CO_2, NO_x, VOC, PM10 and PM2.5. The information in BOS4 can be related to BOS3 to provide estimates of per-segment emissions, allowing one to compare the impacts of different technologies. The bosplt namespace was used for this dataset. The source schema consists of as an Excel XLS worksheet with the attributes described below in Table 5.11.

A speed bin of 15mph corresponds to a closed interval of the form [15,16). For instance, according to Table 5.11, a Diesel engine at a speed bin of 15mph emits the aforementioned pollutants in g/mile units. Emission rates are available for speed bins between 2 and 50mph.

As we could not find an ontology that could thoroughly describe the concepts from this dataset, we extended VUMO for this purpose. The following subclasses of UrbanMobilityConcept were created:

- EmissionProfile;

Table 5.11: Example of an emission profile from the BOS4 dataset

engine	Diesel
speedbin	15
co	0.314144
co2	1108.17
nox	0.847415
pm10	0.3404848
pm25	0.0525447
voc	0.05829

- `EngineTechnology`;
- `Pollutant`.

Object and datatype properties were also created to interrelate instances of these classes. Source attributes corresponding to a pollutant were instantiated as members of the class `Pollutant`. An instance of `EmissionProfile` describes the emission rates of a certain pollutant, for a given instance of `EngineTechnology`, for a speed bin. Figure 5.9 illustrates the example shown in Table 5.11.

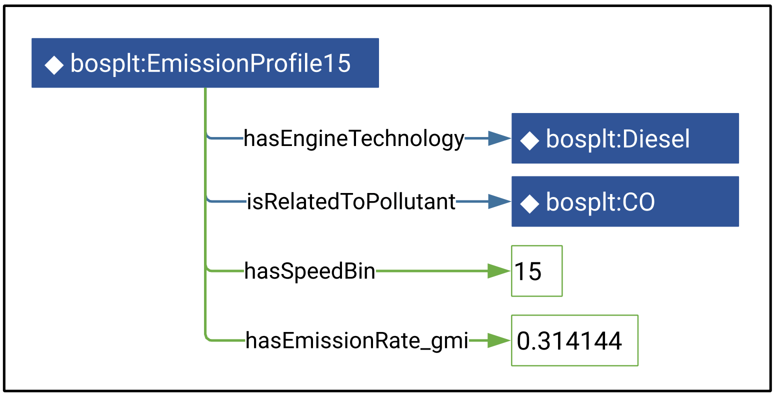


Figure 5.9: RDF graph of an emission profile based on the example of Table 5.11

To facilitate the retrieval of emission rates for a given speed, a SPIN function was created. The function is not part of the default VUMO implementation, and was developed for this particular case, thus belonging to the `bosplt` namespace. The function `getEmissionsRate` expects three arguments in the following order: engine technology, pollutant and speed. To find the appropriate speed bin, the integer part of the speed is retrieve by using the `FLOOR()` function. The query corresponding to this function is given by the following serialization.

```
SELECT ?rate
WHERE {
  ?emissionprofile rdf:type :EmissionProfile ;
  ?emissionprofile :hasEngineTechnology ?arg1 ;
  ?emissionprofile :isRelatedToPollutant ?arg2 ;
  ?emissionprofile :hasSpeedBin FLOOR(?arg3) ;
  ?emissionprofile :hasEmissionsRate_gmi ?rate . }
```

Based on the example of Figure 5.9, the `getEmissionsRate` function can be used to estimate the emissions rate of CO for Diesel engines running at 15.7 mph, yieliding 0.314144 g/mi.

```
getEmissionsRate(bosplt:Diesel,bosplt:CO,15.7)
```

5.4 Visualization of public transportation ridership and annotation of extracted knowledge

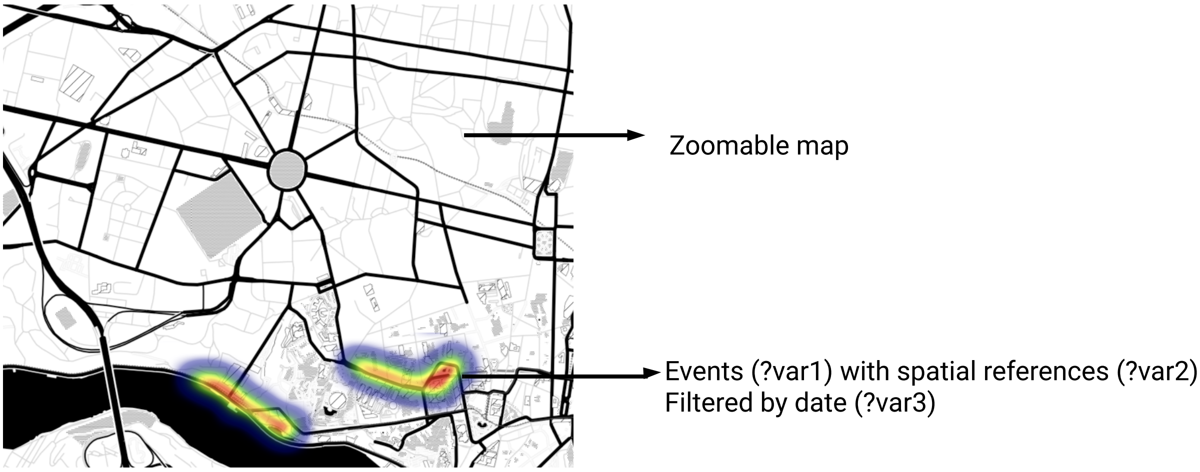
This section demonstrates the development of visualization prototypes for exploring actual and potential public transportation ridership in the context of Porto, based on information from datasets PRT1 (ticket validations) and PRT2 (travel intentions from Move-me). We focus on the semantic annotation of visualization techniques, and showing how VUMO can support the annotation of knowledge that was extracted during exploratory data analysis, so it could be further used to support other analytical tasks and enrich the original instance data graph.

A map-based and an abstract visualization techniques were developed: geographic (`GeoHeatMap`) and calendar (`CalHeatMap`) heat maps, respectively. The former depicts the density of instance data in geographic space. The latter depicts density in a daily arrangement. Each day is assigned to a color in a pre-defined color scale, according to the number of occurrences. Figure 5.10 provides the schematic representation of both visualization techniques.

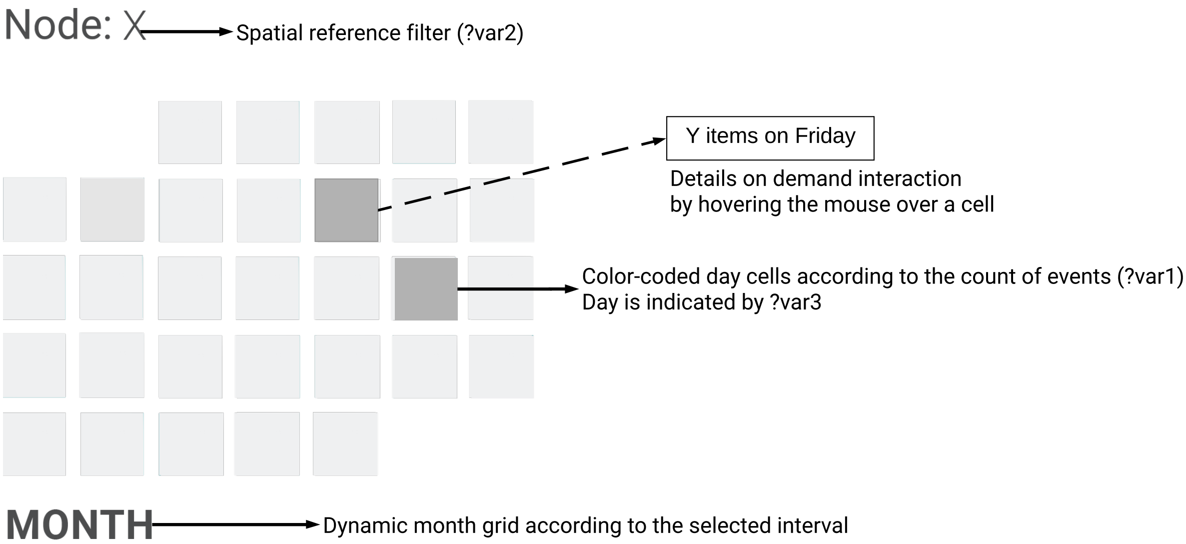
`GeoHeatMap` requires the following input variables: (i) an event instance; (ii) spatial and (iii) temporal references (instant). Variable (i) receives the entity to be plotted, (ii) provides the instance coordinates on a map, and (iii) is used for temporal filtering. `CalHeatMap` requires variables (i) and (iii). Overview, semantic zoom and dynamic queries are available in both techniques. By construction, `CalHeatMap` requires aggregate data, while `GeoHeatMap` does not.

The serialization below provides the semantic annotation for both techniques. The annotation of the `CalHeatMap` technique makes use of the input variables defined for the `GeoHeatMap` technique.

```
// Semantic annotation of the geographic heat map visualization technique
sumvis:GeoHeatMap rdf:type :VisualizationTechnique ;
  rdf:label "Geographic heat map" ;
  :hasFrameOfReference :Geographic ;
  :hasSpatialDimensionality :2D ;
  :hasTemporalRepresentation :Static ;
  :hasTemporalArrangement :Linear ;
  :hasInteractionTask :Overview :DetailsDemand :DynamicQueries ;
  :hasInputVariable sumvis:var1 sumvis:var2 sumvis:var3.

  // Variable for each event instance
  sumvis:var1 rdf:type :InputVariable ;
    :hasCompatibleValueType rdfs:Resource ;
    :isRequired true .

  // Variable for spatial reference
  sumvis:var2 rdf:type :InputVariable ;
    :hasCompatibleValueType rdfs:Resource ;
    :isRequired true .

  // Variable for temporal reference
```



(a) Schematic representation of the Geographic heat map visualization technique



(b) Schematic representation of the Calendar heat map visualization technique

Figure 5.10: Geographical and calendar heat maps implemented in SUMVis

```
  sumvis:var3 rdf:type :InputVariable ;
    :hasCompatibleValueType rdfs:Resource ;
    :hasCompatibleValueType xsd:dateTime ;
    :isRequired true .
```

```
// Semantic annotation of the calendar heat map visualization technique
sumvis:CalHeatMap rdf:type :VisualizationTechnique ;
  rdf:label "Calendar heat map" ;
  :hasFrameOfReference :Abstract ;
  :hasSpatialDimensionality :2D ;
  :hasTemporalRepresentation :Static ;
  :hasTemporalArrangement :Linear ;
  :hasInteractionTask :Overview :DetailsDemand :Filtering :DynamicQueries ;

  :hasInputVariable sumvis:var1 sumvis:var2 sumvis:var3.
```

Let `AT0` be an analytical task represented by an instance of `Transformation`, which provides information about "Actual and potential public transportation ridership". The data transformation has five output variables:

```
sumvis:AT0 rdf:type :Transformation .
  rdf:label "Actual and potential public transportation ridership" .
  spin:query "SELECT ?ev ?lat ?long ?time COUNT (?node) AS ?total
  WHERE {
    ?event rdf:type (:TicketValidation OR :ScheduleRequest) .
    ?event :hasDateTime ?time ;
        :occursAtNode ?node .
    ?node geo:lat ?lat ;
       geo:long ?long .
    FILTER (?time >= ?arg1 && ?time <= ?arg2) } .
  GROUP BY ?node" .
```

The parameters `arg1` and `arg2` are placeholder values that can be changed by the user.

From R3 and axiom A1, it follows that `AT0` contains a `Discrete` spatial distribution, as the property `occursAtNode` has `Node` as its range, which is defined as semantically equivalent to a `KnownPoint`. R4 yields the themes (tag) `TicketValidation` and `NextDeparturesRequest`. It follows from R6 that `AT0` returns aggregate results due to the COUNT function.

It follows from R7 that `AT0` is compatible with `CalHeatMap`, but not `GeoHeatMap`, due to the aggregate results requirement. The inference yields the following triples:

```
sumvis:AT0 :isCompatibleWith sumvis:CalHeatMap ;
    :isNotCompatibleWith sumvis:GeoHeatMap .
```

Consider another instance of `Transformation`, `AT0_non_agg`, that implements the same query as `AT0` except for the aggregate function. The corresponding serialization is given below.

```
sumvis:AT0_non_agg rdf:type :Transformation .
  rdf:label "Actual and potential public transportation ridership, with no
      aggregation" .
  spin:query "SELECT ?ev ?lat ?long ?time ?node
  WHERE {
```

```
?event rdf:type (:TicketValidation OR :NextDeparturesRequest) .
?event :hasDateTime ?time ;
    :occursAtNode ?node .
?node geo:lat ?lat ;
    geo:long ?long .
FILTER (?time >= ?arg1 && ?time <= ?arg2) } ."
```

Hence, R7 yields that `ATO_non_agg` is compatible with `GeoHeatMap`, but not `CalHeatMap`:

```
sumvis:ATO_non_agg :isCompatibleWith sumvis:GeoHeatMap .
    :isNotCompatibleWith sumvis:CalHeatMap .
```

Figure 5.11 demonstrates the `GeoHeatMap` technique showing data retrieved with the analytical task `ATO_non_agg`, with focus on the Boavista region of the city of Porto during a weekday.

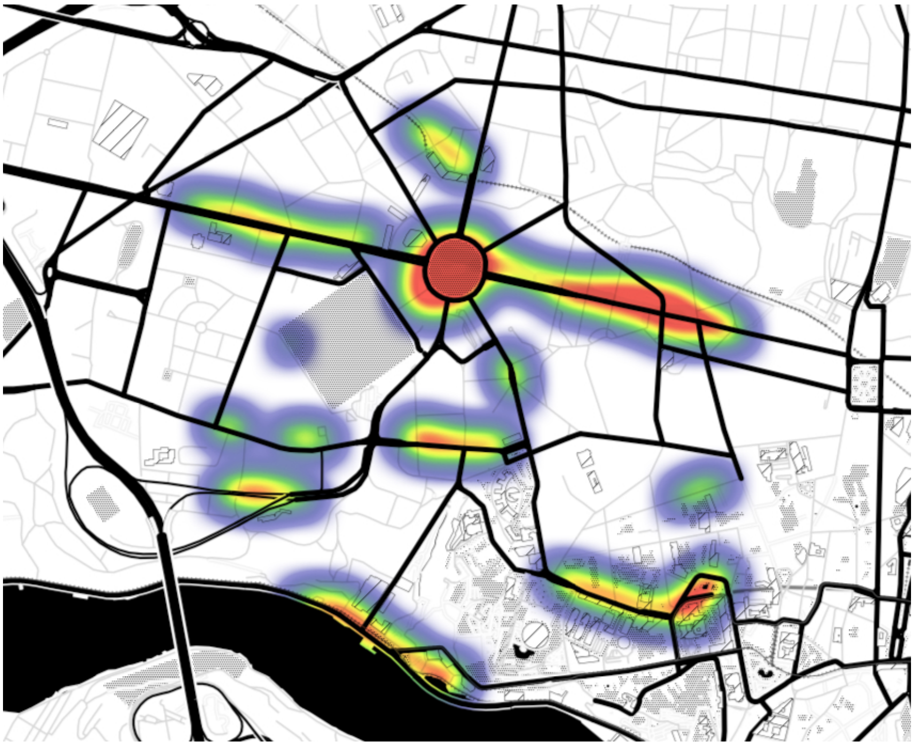


Figure 5.11: The geographic heat map visualization technique coupled with an analytical task related to public transportation ridership in Porto

Figure 5.12 demonstrates the `CalHeatmap` technique coupled with analytical task `ATO`, for a weekday in the Trindade subway station, which is the main subway hub of Porto.

As users explored data using the *CalHeatmap* technique, it was possible to identify an abnormal amount of information requests for next departures, by hovering the mouse on that particular day. The number of requests is related to the public transportation strike that occurred on June 27th, 2013. The *CalHeatMap* prototype was used to relate all schedule requrest instances on that date to a new event asserted as `Strike27Jun13`, which is an instance of *UnexpectedEvent*. Figure 5.13 shows the resulting graph.

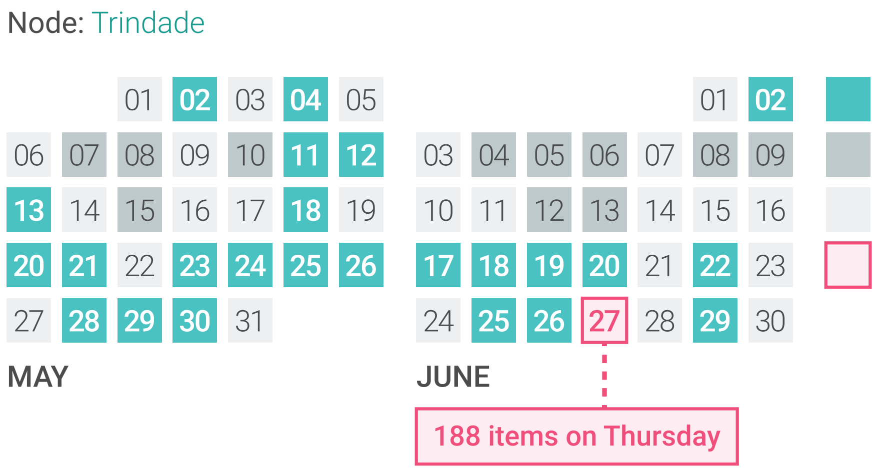


Figure 5.12: The calendar heat map visualization technique showing ridership information for the Trindade subway station in Porto

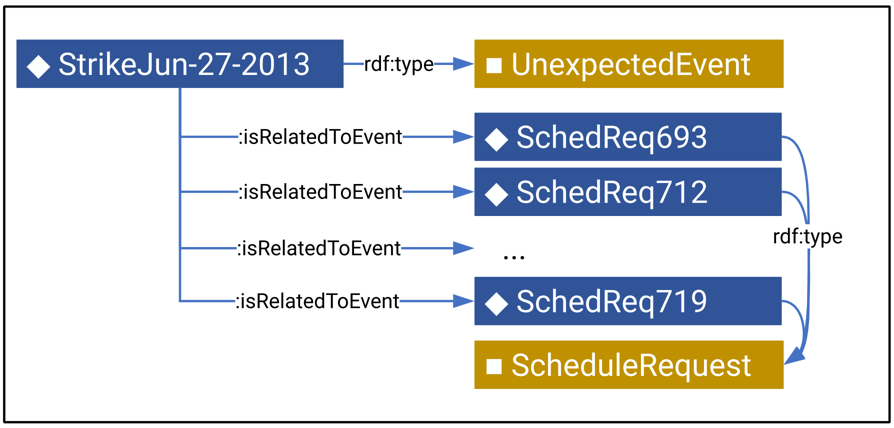


Figure 5.13: An instance of *UnexpectedEvent* is used to represent a public transportation system strike in Porto, which is related to several *ScheduleRequests*

5.5 Visualization of multi-level Origin-Destination flows

The goal of this section is to provide a comprehensive description of the semantic annotation of the proposed visualization techniques, and the empirical knowledge collected from domain users through exploratory usability tests. Furthermore, we provide an illustrative example of recommendation strategy, which was derived from the feedback obtained during the usability tests. This strategy required the design of an additional rule that attempts to derive analytical profile of a data transformation based on its structure. Finally, the rule was implemented in VUMO.

We carried a user-centered design process for the visualization of origin-destination flows, to take advantage of the collected data from both cities, and empirical knowledge from domain users from distinct contexts and backgrounds. We adopted the process described in Section 4.11.

For this study, we involved five academic researchers from Porto and Boston, a former transportation analyst from MassDOT, and four master students in Transportation from MIT. Due to the number of participants and their availability for the experiments, we considered appropriate to carry exploratory usability tests combined with qualitative interviews, as it yielded positive results from our past experience [3]. Most participants were not acquainted with semantic web technologies or ontologies, although they were familiar with visualization and databases.

The exploratory usability tests with experts followed the coaching method proposed by Mack and Robinson [172] combined with a semi-structured interview given during the session. The combination of different usability methods is well acknowledged in usability research [172]. The most distinguishing feature of the coaching method in comparison to other evaluation methods relies on the active involvement of the mediator during the test, who can actually steer users in the right direction while using a system. We argue that the exploratory nature of our study is appropriate the coaching method was considered more adequate.

Exploratory usability tests consisted of four phases:

- Briefing: Presentation of the goals for the exploratory usability test. When applicable, we presented an overview of the main findings and decisions from the previous meeting;
- Exhibition of visualizations: Presentation of the visualization prototypes and their features;
- Free exploration: Users were invited to explore the visualization prototypes and interact with their functionalities;
- Debriefing: Final considerations and review of requirements, ideas and potential suggestions.

Finally, we asked users to provide ratings and cross ratings according to one or more criterion available in *SUMVis*, e.g. recommended analytical profile, visual clutter, recommended theme. The system provided a simple visual interface that facilitated the input of empirical knowledge and their conversion to triples, as shown in Figure 5.14. The selection of the rating component automatically filters the feasible entries. In Figure 5.14, when the user chooses to provide feedback about recommended analytical profile (1), the available values, i.e. instances of `AnalyticalProfile` are shown (2). The checkbox transforms the visualization technique rating into a cross rating, by linking it to the analytical task that is actually being shown by the visualization technique. In Figure 5.15, focus is given to a quantitative rating component (1). The corresponding scale for that rating component is automatically shown, according to its polarization, i.e. whether the component is an instance of `NegativeRatingComponent` or `Positive-RatingComponent`.

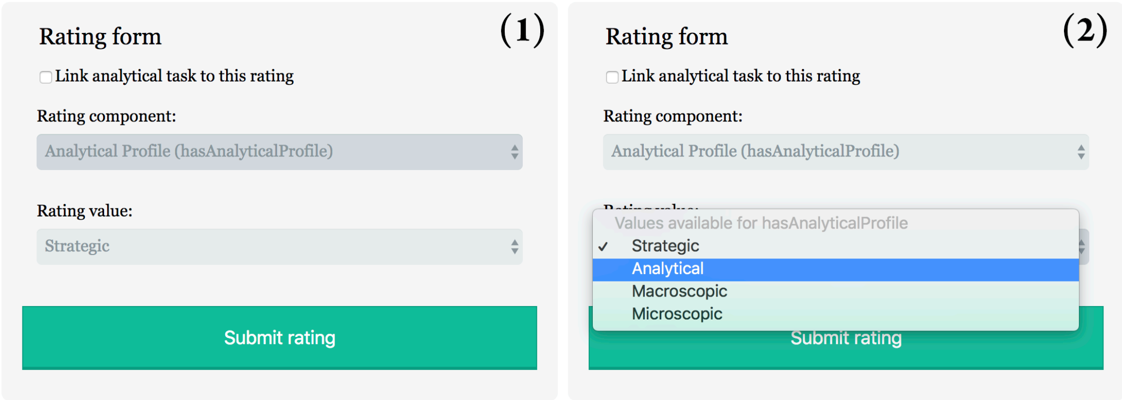


Figure 5.14: Rating form for visualization techniques and data transformations, with focus on a qualitative rating component

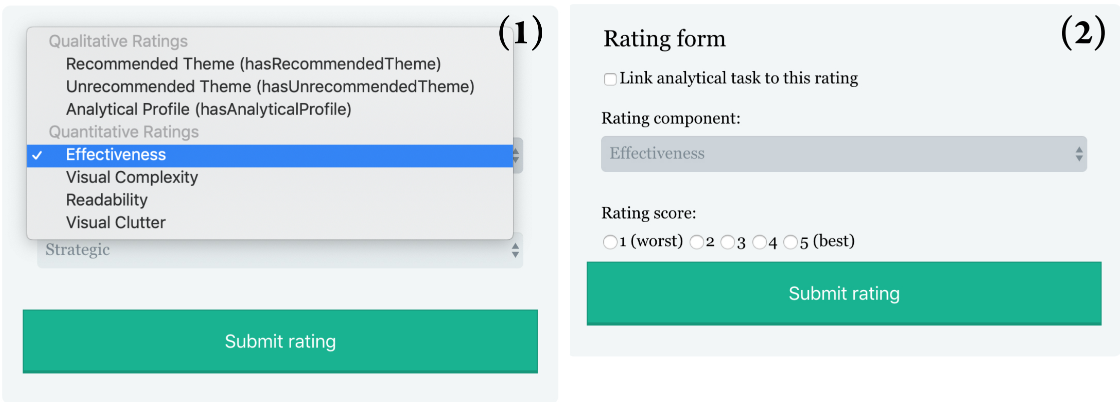


Figure 5.15: Rating form for visualization techniques and data transformations, with focus on a quantitative rating component

5.5.1 Context of analysis

During the context of analysis phase, we asked users to elicit possible and relevant analytical tasks to be done with origin-destination flows data. To avoid steering users to typical answers, no detail was provided about the structure of datasets, except for their nature, i.e. ridership data. The users' inputs could be categorized into three analytical tasks:

- AT1: Aggregate flows between subway lines;
- AT2: Aggregate flows between fare and geographic zones;
- AT3: Granular flows between stops.

Such tasks were instantiated in *SUMVis* as data transformations according to the following serialization.

```
// Analytical task AT1
sumvis:AT1 rdf:type :Transformation .
  rdf:label "Aggregate flows between subway lines" .
  spin:query "SELECT ?line1 ?line2 ?time SUM(?flow) AS ?totflow
              WHERE {
                   ?flow rdf:type :ODFlow .
                   ?flow :hasDateTime ?time .
                   ?flow :hasPath ?path .
                   ?path :hasOrigin ?origin .
                   ?path :hasDestination ?destination .
                   ?origin :isPartOfRoute ?route1 .
                   ?destination :isPartOfRoute ?route2 .
                   ?route1 :isPartOfLine ?line1 .
                   ?route2 :isPartOfLine ?line2 .
                   FILTER (?time >= ?arg1 && ?time <= ?arg2) . }
              GROUP BY ?line1 ?line2" .

// Analytical task AT2
// ?arg3 should be :Zone or schema:AdministrativeArea
// in case of a fare zones or boroughs, respectively.
sumvis:AT2 rdf:type :Transformation .
  rdf:label "Aggregate flows between Porto fare zones or boroughs" .
  spin:query "SELECT ?zone1 ?zone2 ?time SUM(?flow) AS ?totflow
              WHERE {
                   ?flow rdf:type :ODFlow .
                   ?flow :hasDateTime ?time .
                   ?flow :hasPath ?path .
                   ?path :hasOrigin ?origin .
                   ?path :hasDestination ?destination .
                   ?origin :isLocatedInZone ?zone1 .
                   ?destination :isLocatedInZone ?zone2 .
                   ?zone1/?zone2 rdf:type ?arg3 .
                   FILTER (?time >= ?arg1 && ?time <= ?arg2) . }
              GROUP BY ?zone1 ?zone2" .

// Analytical task AT3
sumvis:AT3 rdf:type :Transformation .
  rdf:label "Granular flows between stops" .
  spin:query "SELECT ?origin ?destination ?time SUM(?flow) AS ?totflow
              WHERE {
                   ?flow rdf:type :ODFlow .
                   ?flow :hasDateTime ?time .
                   ?flow :hasPath ?path .
                   ?path :hasOrigin ?origin .
                   ?path :hasDestination ?destination .
                   ?origin/?destination rdf:type :Node.
                   FILTER (?time >= ?arg1 && ?time <= ?arg2) . }
              GROUP BY ?origin ?destination" .
```

In accordance to SPIN logic, placeholders of the form `?argn` are provided by the user, as transformations are, in fact, query templates, i.e. subclasses of `spin:Template`.

It is inferred from R6 that all transformations return aggregate results, due to the `SUM` operator. From R4, it follows that all transformations induce a Graph spatial distribution. The inferred tags (themes) of each analytical task are described in Table 5.12.

Table 5.12: Inferred tags for analytical tasks

Analytical task	Themes
AT1	ODFlow, Node, Route, Line
AT2	ODFlow, Node, Zone
AT3	ODFlow, Node

Still during this phase, some users spontaneously pointed that it would be valuable for the tool to infer the analytical profile of a task based on its structure, i.e. if a transformation provides information that best suits strategical or operational contexts, or macroscopic or microscopic analysis. Based on this requirement and on the ontology structure, we empirically formulated a strategy to suggest the analytical profile of a transformation, which consists of the following high-level steps:

1. Evaluate how broad/narrow are the output variables from a query body that return instances of `UrbanMobilityConcept`;
2. Infer the profile (strategic vs. operational, and macroscopic vs. microscopic) based on a pre-established calculation.

The evaluation of broadness/narrowness of a concept considered the number of concepts that are broader/narrower, based on asserted triples that contain the predicates `skos:broader` or `skos:narrower`, or its transitive counterparts. The calculation involves an arithmetic average of the assigned measures for each output variable o_i. A measure of -1 is assigned to o_i if the cardinality of the set of classes that are broader than the class of o_i is greater than the cardinality of the set of classes that are narrower. Conversely, a measure of 1 is assigned. If the cardinality is the same, 0 is assigned.

1. For each output variable o_i of a transformation, with $o_i \in O$;
2. If variable o_i returns instances of `UrbanMobilityConcept`;
3. Retrieve the number of concepts broader than o_i and store in set B;
4. Retrieve the number of concepts narrower than o_i and store in set N;
5. If $\#B > \#N$, assign $M(o_i) = -1$;
6. If $\#B < \#N$, assign $M(o_i) = 1$;
7. If $\#B = \#N$, assign $M(o_i) = 0$;
8. Calculate $\alpha = \sum_{i=1}^{n} \frac{M(o_i)}{\#O}$;
9. If $\alpha > \tau, \tau > 0$, then the transformation is suggested to have strategic and macroscopic profiles, where τ is a pre-defined threshold;
10. If $\alpha < \tau$, then the transformation is suggested to have operational and microscopic profiles;

11. If $\alpha \leq \|\tau\|$, both analytical profiles are suggested.

For this study, the threshold $\tau = 0.3$ was chosen. The strategy was intuitively described to domain users, who agreed with the proposed approach and threshold definition after a number of iterations. To verify if an output variable is an instance of `UrbanMobilityConcept`, the rule retrieves the domain/range from conditions within the `WHERE` statement in which such variable is either a subject or object. The retrieval of concepts broader/narrower than the output variable benefits from the implemented functions `getBroaderConcepts` and `getNarrowerConcepts` described in Section 4.8. For instance, the proposed rule infers that `AT1` has a strategic profile. Both output variables return instances of `Line`. Hence, $B = \{\texttt{Zone}\}$, and $N = \{\texttt{Route}, \texttt{Route Segment}, \texttt{Node}\}$, yielding $\alpha_{\texttt{AT1}} = 1$. Analogously, transformations `AT2` and `AT3` have $\alpha_{\texttt{AT2}} = 1$ and $\alpha_{\texttt{AT3}} = -1$, respectively.

5.5.2 Conceptual development and prototyping

We proposed two visualization techniques typically used, according to the literature review, to depict origin-destination flows: heat map matrices and chord diagrams. Both techniques have been used in other domains for visualization of data that follows a source-destination structure.

During conceptual development, we asked users about the desired type of interactivity. At this point, we did not mention the formal terms that are usually referred to interaction tasks (see Subsection 2.1.2). Answers consisted of mechanisms that could allow an overview of all flows, and detailed information for specific origin-destination pairs. Users considered relevant to provide a means of filtering origins/destinations. Such interactions correspond to the following interaction tasks: overview, details on demand, filtering, and dynamic queries.

In *SUMVis*, the heat map matrix allows the user to know the exact flow between a pair of locations by placing the mouse cursor over a matrix cell. Monochromatic, grayscale tones are automatically adjusted according to the magnitude of flows. As a design limitation, it is not possible to filter one or more origin-destination pairs, although it is possible to filter a specific line or column.

The structure of the visualization technique requires four input variables, as illustrated in Figure 5.16. Variables $var1$ and $var2$ expect instances or string literals for source and destination. $var3$ expects a numeric value for the flow between $var1$ and $var2$. $var4$ expects a timestamp.

The serialization below corresponds to the semantic annotation of the heat map matrix.

```
// Semantic annotation of the heat map matrix visualization technique
sumvis:heatmapmatrix rdf:type :VisualizationTechnique ;
  rdf:label "Heat map matrix" ;
  :hasFrameOfReference :Abstract ;
  :hasSpatialDimensionality :2D ;
  :hasTemporalRepresentation :Static ;
  :hasInteractionTask :Overview :DetailsDemand :Filtering :DynamicQueries ;

  :hasInputVariable sumvis:var1 sumvis:var2 sumvis:var3 sumvis:var4 .

  // Variable for source
  sumvis:var1 rdf:type :InputVariable ;
    :hasCompatibleValueType rdfs:Resource ;
    :hasCompatibleValueType xsd:string ;
```

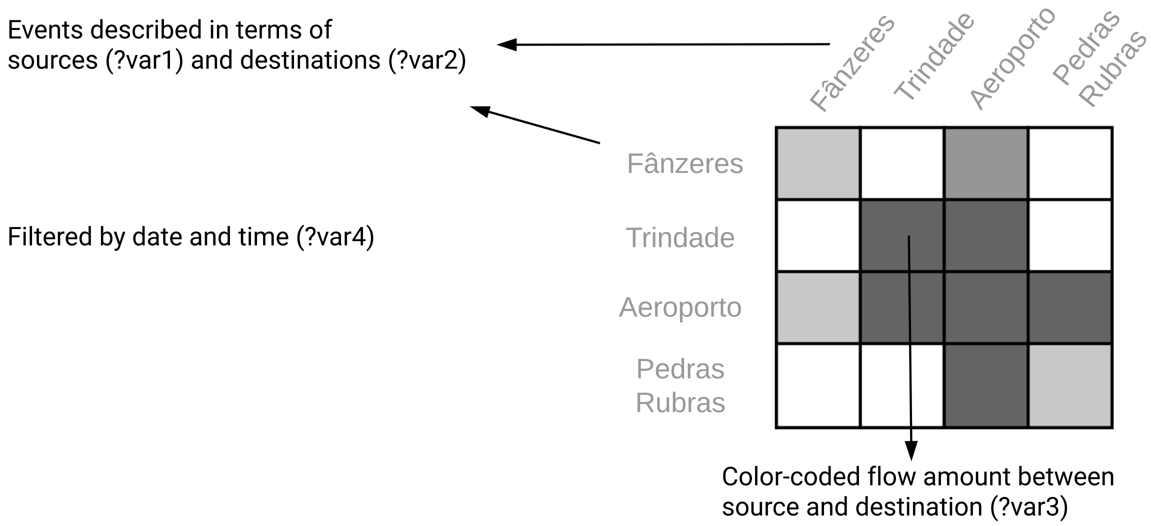


Figure 5.16: Schematic representation of the heat map matrix visualization technique

```
  :isRequired true .

// Variable for destination
sumvis:var2 rdf:type :InputVariable ;
  :hasCompatibleValueType rdfs:Resource ;
  :hasCompatibleValueType xsd:string ;
  :isRequired true .

// Variable for flows (quantitative values)
sumvis:var3 rdf:type :InputVariable ;
  :hasCompatibleValueType xsd:decimal ;
  :hasCompatibleValueType xsd:float ;
  :isRequired true .

// Variable for timestamps
sumvis:var4 rdf:type :InputVariable ;
  :hasCompatibleValueType xsd:dateTime ;
  :isRequired true .
```

The chord diagram technique, also abstract, provides information about the exact flow between origin and destination, and vice-versa, by placing the mouse over a chord. In addition, the remaining chords are occluded to reduce visual clutter and provide focus, while keeping the user aware of other flows (focus plus context). The width of a chord is given by the highest flow between a origin-destination pair. The arc size for each location is given by the sum of its chords widths. It is possible to filter locations by clicking on a location name. This procedure recalculates the chord diagram, as show in Figure 5.17.

Its interaction tasks are overview, details on demand, focus plus context, filtering and dynamic queries. The chord diagram structure requires the same input variables as the heat map matrix, as shown in Figure 5.18. Therefore, they were omitted from the annotation herein described.

The technique contains the following semantic annotation:

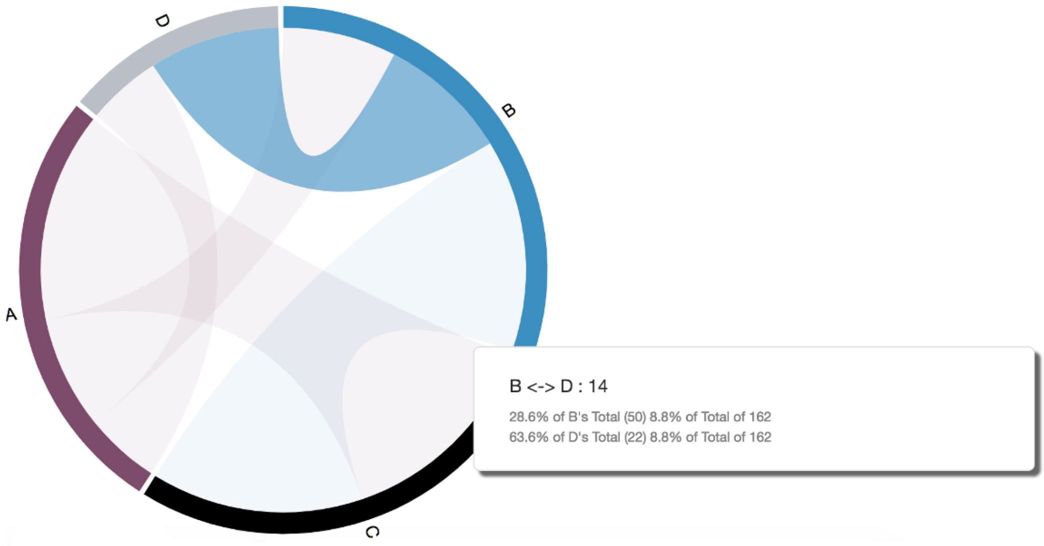


Figure 5.17: Focus plus context interaction on the chord diagram visualization technique

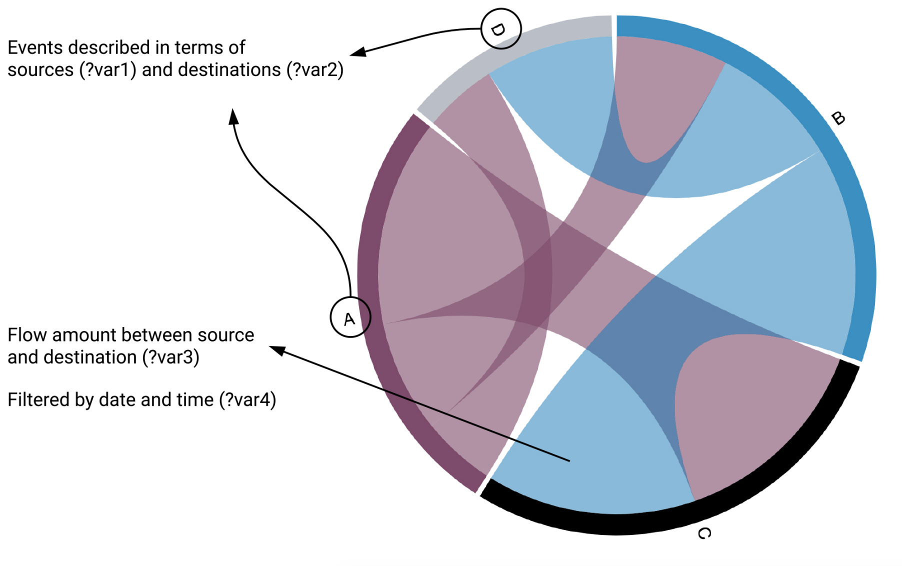


Figure 5.18: Schematic representation of the chord diagram visualization technique

```
// Semantic annotation of the chord diagram visualization technique
sumvis:chorddiagram rdf:type :VisualizationTechnique ;
  rdf:label "Heat map matrix" ;
  :hasFrameOfReference :Abstract ;
  :hasSpatialDimensionality :2D ;
  :hasTemporalRepresentation :Static ;
  :hasInteractionTask :Overview :DetailsDemand :FocusContext ;
  :hasInteractionTask :Filtering :DynamicQueries .
```

The rule R7 infers that the proposed visualization techniques are compatible with transformations AT1, AT2 and AT3, as it is possible to provide a mapping between their output and input variables, as illustrated in Figure 5.19 for the particular case of AT1, which is analogous to the remaining transformations, with the exception of them having zones and nodes as output variables.

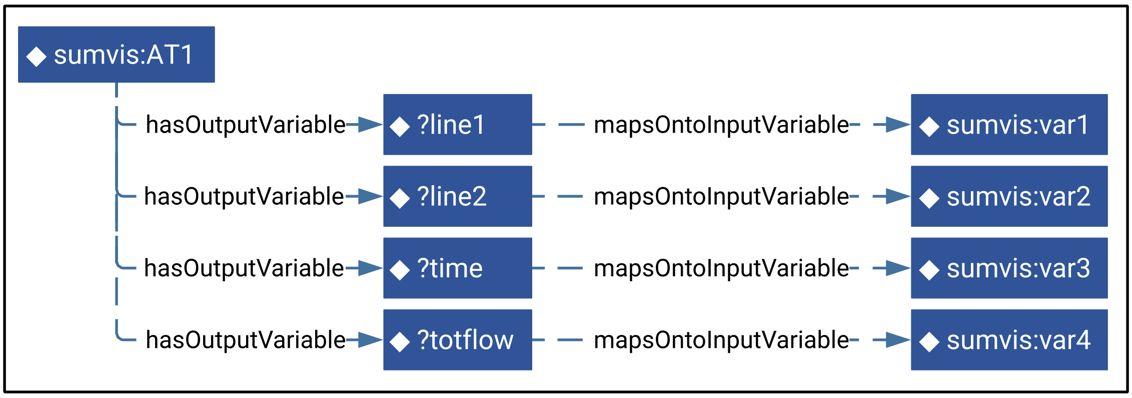


Figure 5.19: Compatibility inference between the heat map matrix and chord diagram techniques with an analytical transformation

5.5.3 Evaluation

Heat map matrix

All users were familiar with the heat map matrix, which was considered useful to support the three analytical tasks, and OD flows in general. However, users from Porto context were more inclined to use apply it to AT1 and AT2 instead of AT3, as they were more interested on the macroscopic flows between zones and lines. Moreover, they stated that the visualization technique becomes less effective when the number of items is too large, as in the case of AT3. Users from Boston context with an operational background considered the technique appropriate to visualize AT3, as it was still capable of revealing clusters of significant flows. Figure 5.20 shows the technique combined with analytical task AT2, using data from Porto. In Figure 5.21, we provide a graphical representation of some of the ratings that were described in terms of VUMO components.

Chord diagram

Half experts were already acquainted with the concept of chord diagrams, although the majority (8 out of 10) were initially confused about the visual representation, due to the *hairball* visual clutter, as all chords are shown simultaneously. After showing the interaction mechanisms, all users stated that the visualization became useful.

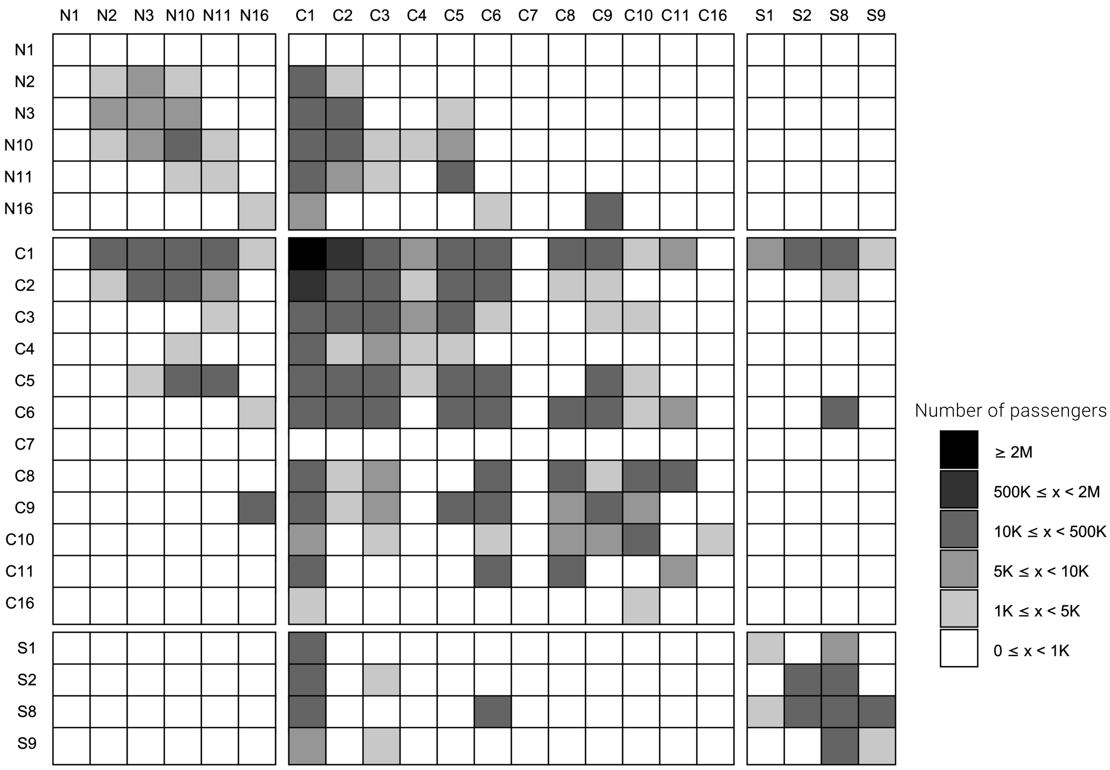


Figure 5.20: Heat map matrix applied to an analytical task that shows OD flows between Porto fare zones (AT2)

On regards to specific analytical tasks, the data transformation `AT3` was considered unfeasible by users from Porto context, due to the massive amount of stops that should be represented. Conversely, users from Boston context considered that the visualization provides interesting insights about flows within the MBTA subway network, at the lines and stops level. The visualization was considered appropriate for `AT1` (Figure 5.22) and `AT3` (Figure 5.23), as the number of locations is ideal to prevent excessive visual clutter, which can also be tackled with the filtering capabilities. Users from Boston context stated that color coding would significantly increase readability of `AT1` and `AT3`. In this case, colors would refer to subway lines.

5.5.4 Building a VUMO-assisted recommendation method

After the evaluation phase, we approached some researchers from our sample to collect some insights on building a recommendation method based on empirical knowledge that can be described with VUMO components. The goal was not to propose a one-size-fits-all recommendation method, but to evaluate to which extent VUMO components can, in fact, retrieve such empirical knowledge to feed various recommendation approaches. In this subsection, we show the resulting recommendation algorithm and exemplify its execution with two domain users. The output of the method is a score for each visualization technique, and a rank based on all assigned scores. The method is based on three premises. Given an analytical task and a technique:

1. Feedback about any criteria given by a domain user has a higher weight than feedbacks given by other domain users about the same criteria;

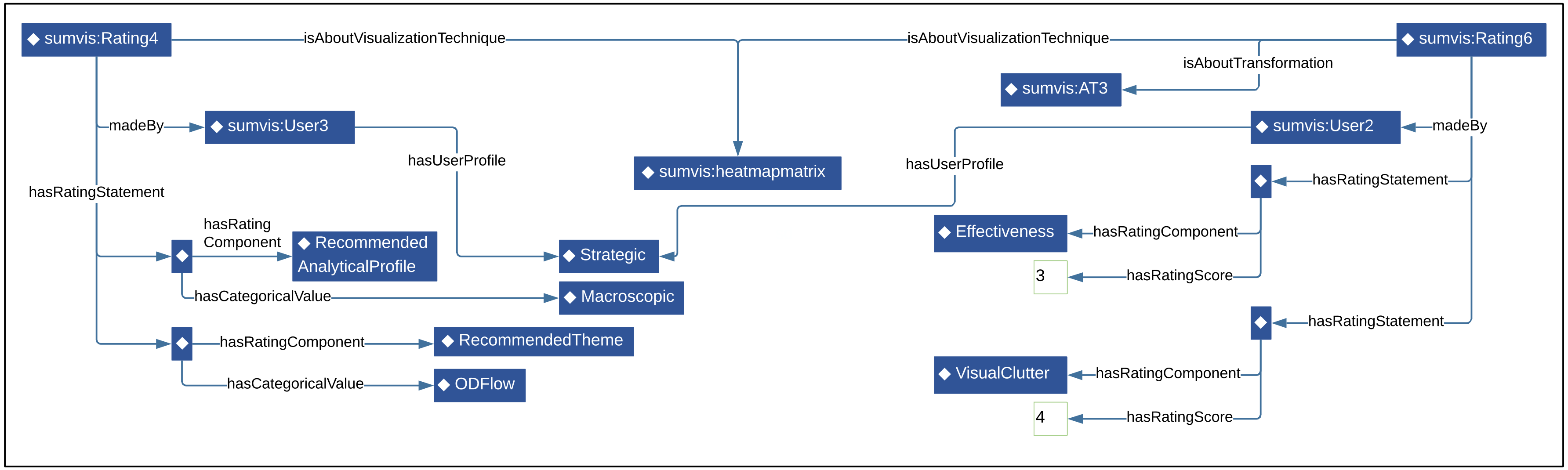


Figure 5.21: RDF graph containing two ratings about the heat map matrix visualization technique

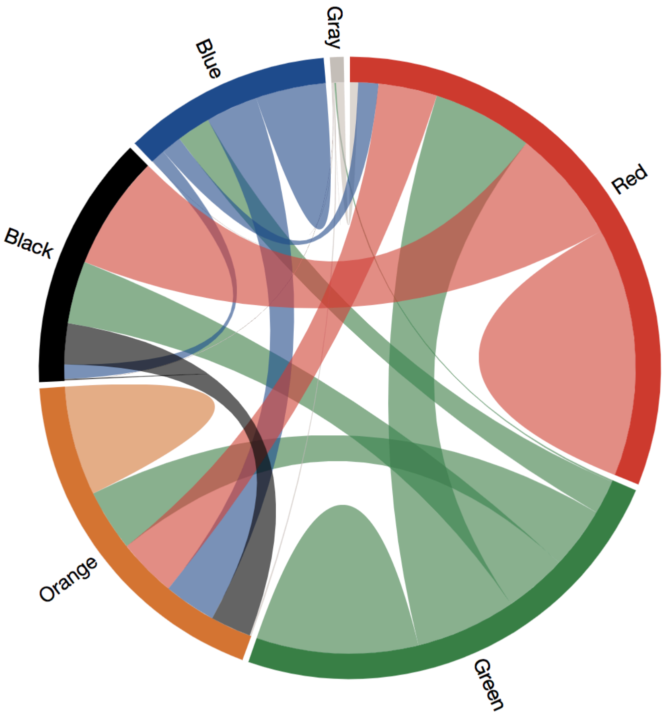


Figure 5.22: Chord diagram applied to an analytical task that shows OD flows between MBTA subway lines (AT1)

2. Feedback given by domain users with the same analytical profile has a higher weight than the ones given by users with different profiles;

3. Cross ratings have a higher weight on score calculation than visualization technique ratings.

In this particular implementation, let U be the set of users of the visualization tool, and u^* a user who has an active session on the tool. The weight of a user w_u is defined as follows:

$$\begin{cases} 2 & u = u^* \\ 1 & \text{otherwise} \end{cases}$$

Given a rating r, the weight w_r of cross ratings and visualization technique ratings are the following:

$$\begin{cases} 2 & r \in \texttt{CrossRating} \\ 1 & r \in \texttt{VisualizationTechniqueRating} \end{cases}$$

After the selection of an analytical task t, the tool retrieves the list of compatible visualization techniques $V_t = \{v_i\}$.

The method requires the following inputs:

- u^*: user with active session on the visualization tool;

- U: set of all domain users u such that $u \in$ `DomainUser`;

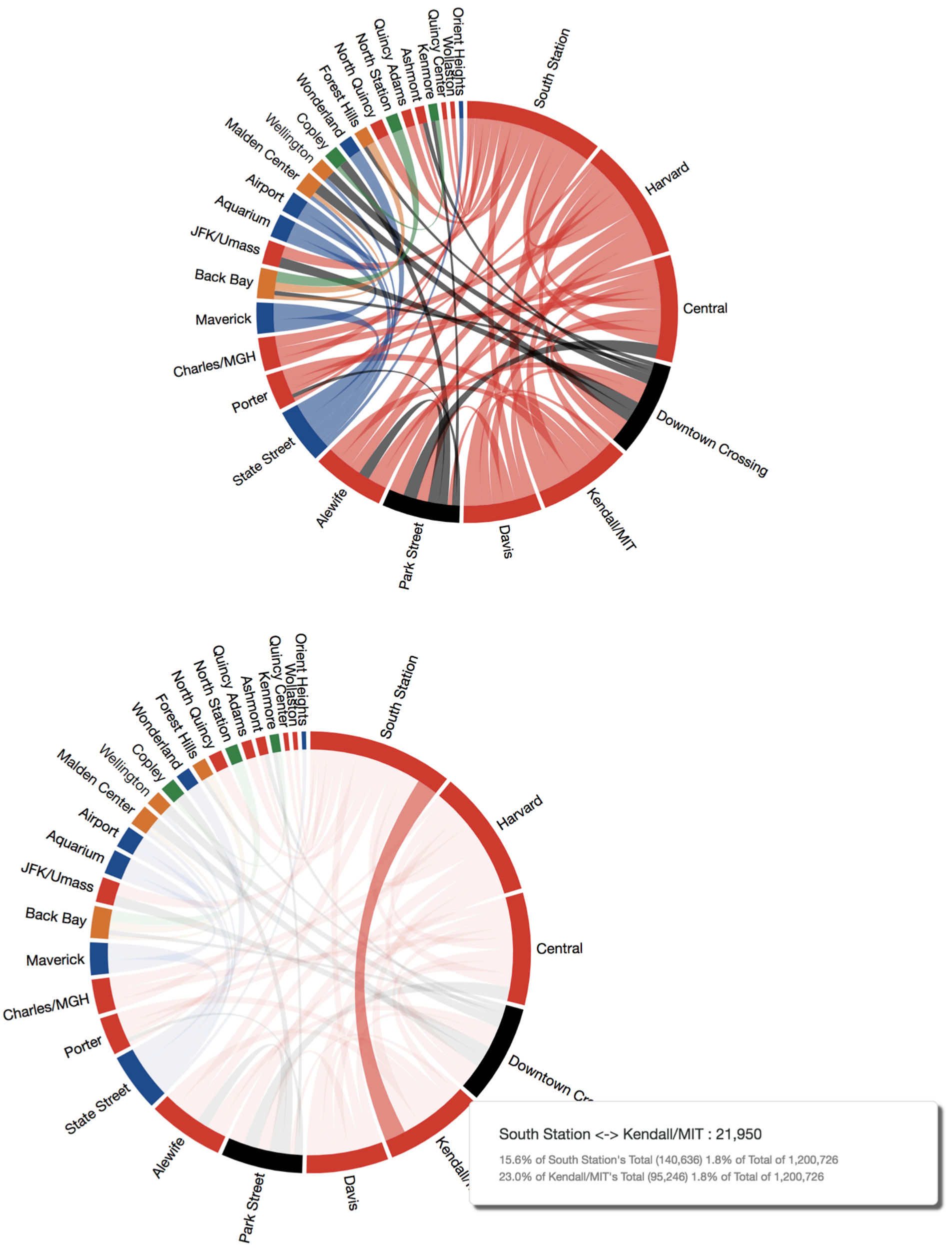


Figure 5.23: Chord diagram applied to an analytical task that shows OD flows between MBTA stops (AT3), hence with higher granularity than AT1 (top). By hovering the mouse over a chord, the visualization presents detailed information about the flow between two stations (bottom)

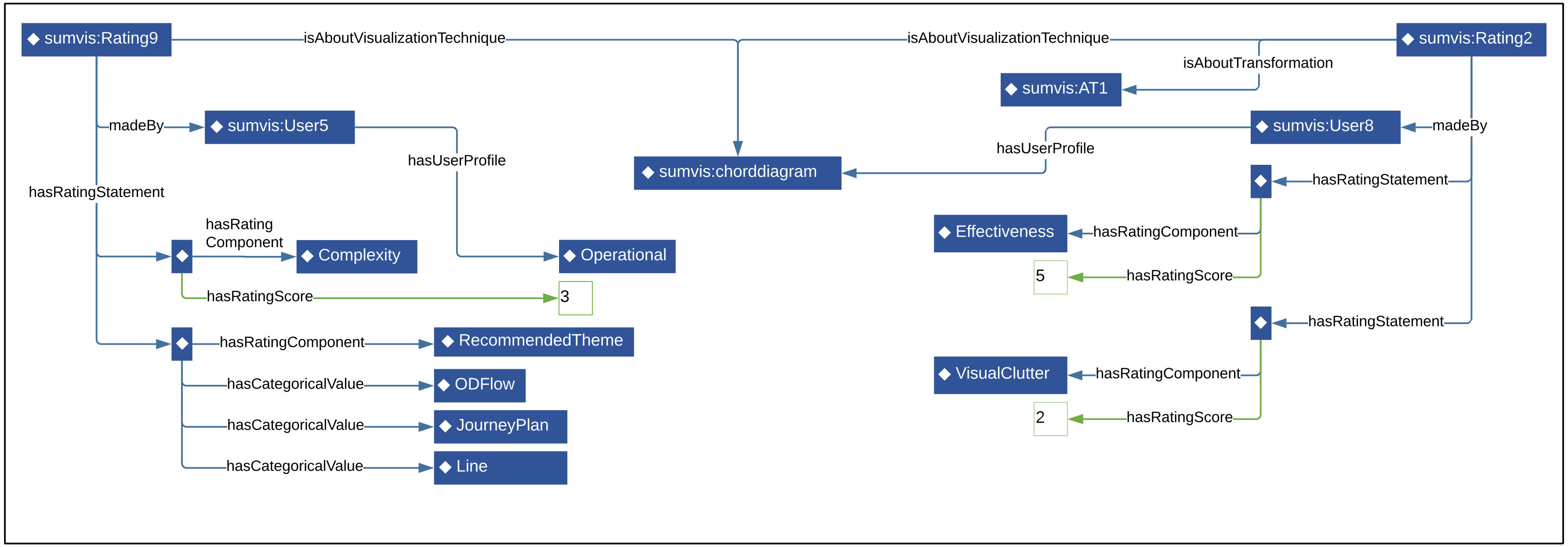


Figure 5.24: RDF graph containing two ratings about the chord diagram visualization technique

- t: selected analytical task;
- $V^t = \{v^t\}$: set of visualization techniques that are compatible with t.

The method calculates, for each visualization, an aggregate rating based on the scores of all users. For each user, the aggregate rating, for cross ratings or visualization technique ratings, is given by the arithmetic average:

$$\rho(v^t, u) = \frac{\sum_{i}^{M} m_i}{M}$$

where m_i is the contribution of a numeric score for a rating component. On regards to quantitative rating components, the contribution is calculated as follows:

- In case the rating component is a `PositiveRatingComponent`, the actual score given by the user is used, e.g. 4 out of 5;
- If the rating component is a `NegativeRatingComponent`, the score is subtracted from the maximum value of the implemented scale, e.g. a score of 3 out of 5 yields a score of 2.

The generalized formula for the aggregate rating for all users is given by:

$$\rho(v^t) = \frac{w_{CR}\frac{\sum_{i=1}^{U} w_{u_i}\rho_{CR}(v^t,u_i)}{\sum_{i=1}^{U} w_{u_i}} + w_{TR}\frac{\sum_{i=1}^{U'} w_{u_j}\rho_{TR}(v^t,u_j)}{\sum_{i=1}^{U'} w_{u_j}}}{w_{CR} + w_{TR}}$$

where w_u is the weight of a user.

For positive or negative qualitative rating components, the maximum or minimum value of the implemented scale is assigned when the value matches the respective feature of an analytical task. For instance, if a user stated that a visualization technique is recommended for visualizing `TicketValidation`, and this tag is also part of the analytical task (inferred by rule R5), a score of 5 is assigned to this rating component. If the rating component has a negative nature, a score of 1 is assigned.

A practical implementation consisted of creating several SPIN functions, where each function is responsible for performing part of the calculation process. Although the formula could be completely implemented in one query, we opted to fragment it into several functions for the sake of modularity. We started by defining a function that calculates the contribution of quantitative scores, according to the aforementioned definition. Here, we omitted the namespace `sumvis` in which those functions were declared.

The function `getScoreContribution(`c, s`)` expects an instance of `RatingComponent`, c, and a numeric score s. The following query is encapsulated into the function:

```
SELECT ?contribution
WHERE {
  c rdf:type :RatingComponent .
  c :hasMaximumScore ?maxscore .
  BIND(
    IF(c rdf:type :PositiveRatingComponent, s, ?maxscore - s)
  ) AS ?contribution }
```

The function `getValueContribution`(c) expects an instance of `RatingComponent`, c. The following query is encapsulated into the function:

```
SELECT ?contribution
WHERE {
  c rdf:type :RatingComponent .
  BIND(
    IF(c rdf:type :PositiveRatingComponent, 5, 1)
  ) AS ?contribution }
```

The function `getAggUserVisTechniqueRating`(u, v) returns the global rating given by user u about a visualization technique v. The following query is encapsulated into the function:

```
SELECT AVG(?contribution) AS ?aggrating
WHERE {
  ?rating rdf:type :VisualizationTechniqueRating .
  ?rating :isAboutVisualizationTechnique v .
  ?rating :madeByUser u .
  ?rating :hasRatingStatement ?stmt .
  { ?stmt :hasRatingComponent ?component .
  ?stmt :hasRatingScore ?score .
  BIND( getScoreContribution(?score) AS ?contribution ) . }
  UNION
  { ?stmt :hasRatingComponent ?component .
  ?stmt :hasRatingCategoricalValue ?value .
  BIND( getCategoricalValueContribution(?value) AS ?contribution ) . }
```

The function `getAggUserCrossRating`(u, v, t) returns the global cross rating given by user u to a visualization technique v and transformation t. The query is similar to the one for visualization technique ratings, only differing by an extra condition related to a transformation t.

```
SELECT AVG(?contribution) AS ?aggrating
WHERE {
  ?rating rdf:type :VisualizationTechniqueRating .
  ?rating :isAboutVisualizationTechnique v .
  ?rating :isAboutTransformation t .
  ?rating :madeByUser u .
  ?rating :hasRatingStatement ?stmt .
  { ?stmt :hasRatingComponent ?component .
  ?stmt :hasRatingScore ?score .
  BIND( getScoreContribution(?score) AS ?contribution ) . }
  UNION
  { ?stmt :hasRatingComponent ?component .
  ?stmt :hasRatingCategoricalValue ?value .
  BIND( getCategoricalValueContribution(?value) AS ?contribution ) . }
```

The function `getTotalUserRating`(u, v, t, w_{CR}, w_{TR}) returns the user rating for a visualization technique v.

```
SELECT ?totuserrating
WHERE {
  u rdf:type :DomainUser .
```

```
v rdf:type :VisualizationTechnique .
t rdf:type :Transformation .
{ BIND ( getAggUserVisTechniqueRating(u,v) ) AS ?techniquerating . }
UNION {
  BIND ( wTR * getAggUserVisTechniqueRating(u,v) ) AS ?techniquerating .
  BIND ( wCR * getAggUserCrossRating(u,v,t) ) AS ?crossrating .
  BIND ( (?aggvisrating + ?aggcrossrating)/(wTR + wCR) AS ?totuserrating . }}
```

Finally, the function `getTotalRating`(v, t, w_{CR}, w_{TR}) returns the global rating for a visualization technique v.

```
SELECT SUM(?userrating)/?usercount AS ?globalrating
WHERE {
  u rdf:type :DomainUser .
  v rdf:type :VisualizationTechnique .
  t rdf:type :Transformation .
  BIND ( IF(u = u*, 2, 1) ) AS wU ) // Weight of user
  BIND ( wU * getTotalUserRating(u,v,t,w_CR,w_TR) AS ?userrating . )
  BIND ( wU * getTotalUserRatingCount(u,v,t,w_CR,w_TR) AS ?userratingcount . )
    }
```

Example As an illustration, consider the ratings for the heat map matrix represented in Figure 5.21. Assuming that there are no other ratings on the system, the aggregate total cross rating given by `sumvis:User2` is:

$$\frac{3 + (5 - 4)}{2} = 2$$

Given that `VisualClutter` is an instance of `NegativeRatingComponent`, the score contribution to the average is 1, assuming 5 as the maximum allowed value. The visualization technique rating given by `sumvis:User3` is given by the expression:

$$\frac{5 + 5}{2} = 5$$

As AT2 has the tag `ODFlow` and a `Strategic` profile, a contribution of 5 is assigned for each match.

The global average rating for the heat map matrix, with respect to AT2, is then given by the expression:

$$\frac{2 \cdot 2 + 1 \cdot 5}{3} = 3$$

Where 2 and 1 are the weights assigned for w_{CR} and w_{TR}, respectively.

As a result, the aforementioned functions allows one to easily implement a personalized rank of visualization techniques for a given analytical task, even in situations where the number of ratings is small. In the case of a cold start, visualization techniques can still be recommended just by evaluating their *compatibility* with analytical tasks.

5.6 Visualization of emissions in bus corridors

This section describes a user-centered design process for the development of a proof-of-concept visualization prototype for emissions in bus corridors, which is expected to be implemented into an existing visualization system for urban mobility planning. The first goal is to demonstrate how analytical tasks can combine constructs that are not originally present in VUMO, in accordance to some of the datasets presented in Section 5.3, namely BOS3 and BOS4. The second goal is to demonstrate the role of domain user feedback on the definition and modification of analytical tasks.

5.6.1 Context of use analysis

CoAXs is an interactive planning and visualization tool developed at the Department of Urban Studies and Planning of MIT. The tool allows users to evaluate the impacts of different transportation scenarios on regards to accessibility metrics, i.e. travel time and number of job opportunities that can be reached. Some examples of scenario characteristics are bus speeds or frequencies, and alternative route paths for a same corridor. Ultimately, the goal is to foster stakeholder engagement and collaborative involvement in transportation decision making. The tool combines a map-based visualization of accessibility metrics and an abstract visualization technique such as a bar chart. The tool can be customized to meet the requirements of each city, and has been implemented in cities like Boston, New Orleans, Atlanta and San Francisco. Figure 5.25 shows CoAXs' interface customized for the city of New Orleans. Isochrones reflect the travel time for baseline (blue) and new (orange) scenarios. The bar chart effectively shows the variation on the access to job opportunities.

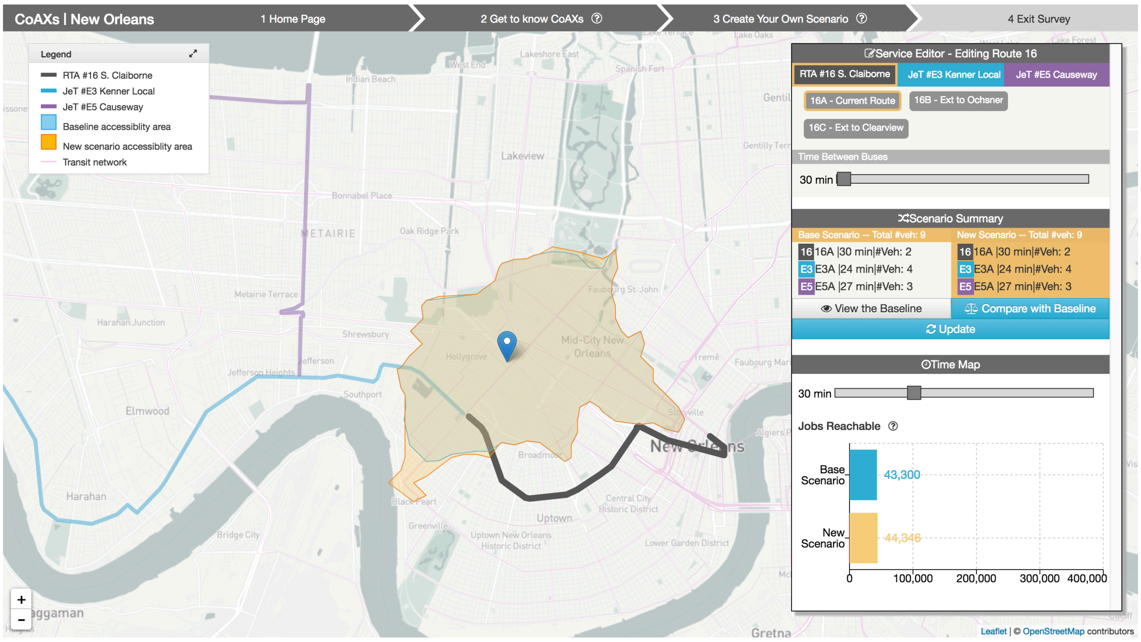


Figure 5.25: CoAXs customized for New Orleans

In addition to accessibility metrics, one of the project's goals is to allow stakeholders to visualize and compare the impact of scenarios on emissions in bus corridors, in terms of parameters such as route frequency or engine technology. Given that CoAXs aims to involve stakeholders from various perspectives and backgrounds, including citizens, one of the requirements was that visualizations should be simple and intuitive.

Initially, a fundamental decision consisted of the level of aggregation for analyzing emissions: from route or route segment levels. Although the analysis by route segments can be effective for showing critical hotspots, it may induce incorrect reasoning on regards to the actual concentration of pollutants. It was then decided that the visualization should intentionally err on the side of being general, and provide a caveat on regards to emissions visualization: the tool does not aim to depict the dispersion of pollutants, but to provide the big picture of the impact of emissions at each segment for a certain bus technology and route characteristics, e.g. headway.

The second decision consisted of choosing which variable should be prioritized on the main visualization interface, i.e. map interface: average speeds or total amount of emissions. Given that polluted hotspots are related to segments with lower average speeds, such variables are directly related, although it was stated that the latter could be more difficult to be understood by stakeholders such as general public. An initial analytical task was defined:

- AT4: Average speed by route segment for a given engine technology.

The annotation of the analytical task is given by the following transformation:

```
// Analytical task AT4
sumvis:AT4 rdf:type :Transformation .
  rdf:label "Average speed on route segments for Boston during AM peak" .
SELECT ?avgspeed ?routesegment ?line
WHERE {
  ?reading rdf:type :Reading .
  ?reading :isRelatedToLine ?line .
    ?reading :isRelatedToRoute ?route .
    ?reading :isRelatedToRouteSegment ?routeseg .
    ?reading :hasTemporalEntityReference bosod:ampeak .
    ?reading :hasAverageSpeed ?avgspeed . }
```

From inference follows that AT4 has a Graph spatial distribution, as it refers to route segments. AT4 has the tags `Reading`, `Line`, `Route` and `RouteSegment`. The suggested analytical profiles are `Strategic` and `Operational`, as $\alpha_{AT4} = 0$.

5.6.2 Prototyping

Although the visualization techniques were developed on top of *SUMVis*, the proposed prototypes were expected to be implementable on CoAXs. Therefore, it was important to ensure that the visualization techniques could be reproduced using the same technological stack. We initially proposed a line-based heat map for the exploration of average speeds of route segments (AT4). The visualization technique had interaction mechanisms for filtering by specific bus lines. When a particular route segment was selected, additional details were revealed. By rule R6, the technique is compatible with AT4. The semantic annotation of the visualization technique is given by the following serialization.

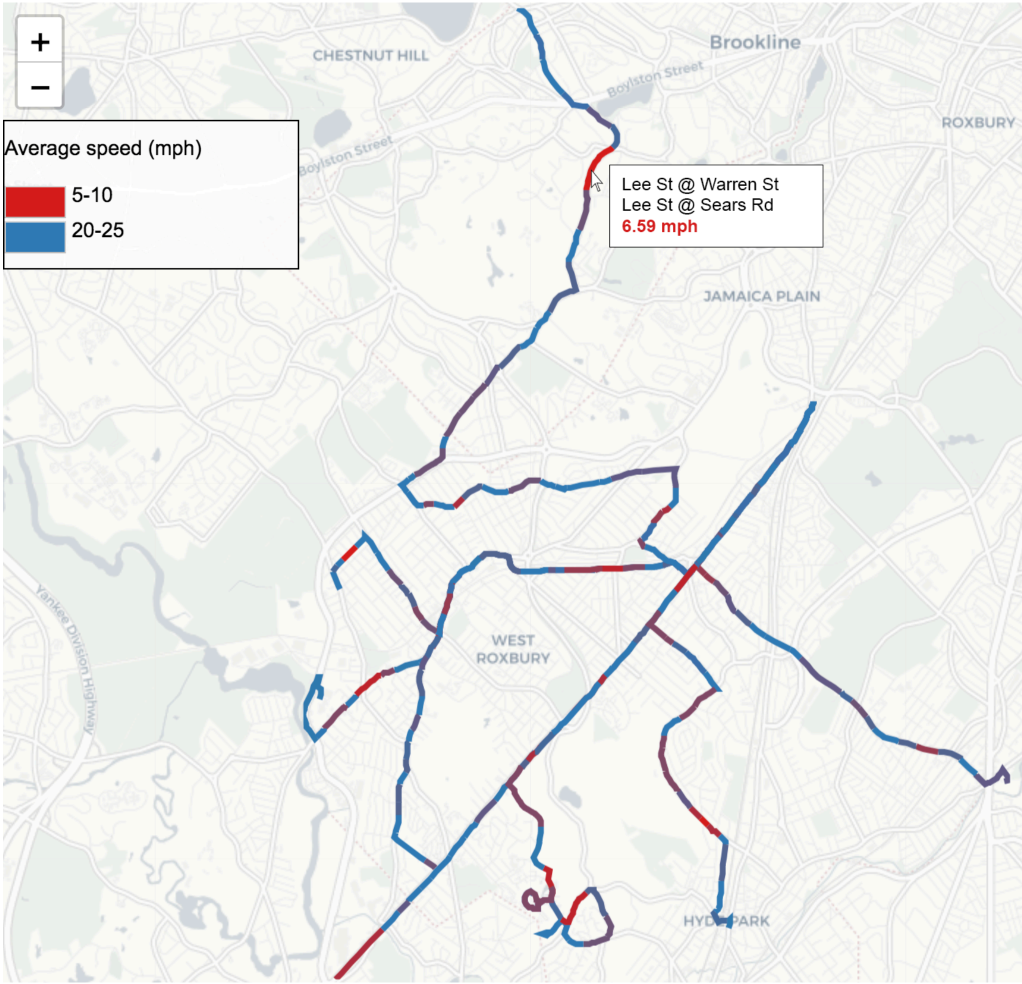


Figure 5.26: Interactive line-based heat map for visualization of average speed on route segments. Details are provided when the user selects a route segment

```
// Semantic annotation of the line-based heat map visualization technique
sumvis:lineheatmap rdf:type :VisualizationTechnique ;
  rdf:label "Line-based heat map" ;
  :hasFrameOfReference :Geographic ;
  :hasSpatialDimensionality :2D ;
  :hasTemporalRepresentation :Static ;
  :hasInteractionTask :Overview :DetailsDemand :Filtering .

  :hasInputVariable sumvis:var_a sumvis:var_b sumvis:var_c ;


  // Variable that receives instances of route segments
  sumvis:var_a rdf:type :InputVariable ;
    :hasCompatibleValueType rdfs:Resource ;
    :isRequired true .

  // Variable for numeric values (quantitative values)
  sumvis:var_b rdf:type :InputVariable ;
    :hasCompatibleValueType xsd:decimal ;
    :hasCompatibleValueType xsd:float ;
    :isRequired true .

  // Variable for time periods or timestamps
  sumvis:var_c rdf:type :InputVariable ;
    :hasCompatibleValueType rdfs:Resource ;
    :hasCompatibleValueType xsd:dateTime ;
    :isRequired true .

    // Variable for time periods or timestamps
    sumvis:var_d rdf:type :InputVariable ;
      :hasCompatibleValueType xsd:dateTime ;
      :isRequired true .
```

5.6.3 Evaluation and second iteration of prototyping

The evaluation took place in two moments: with CoAXs team members, and members involved in the application of CoAXs instances to their respective cities in countries such as Colombia, Chile and South Africa, during a workshop session held at MIT. Users agreed with the proposed technique, and added that the consistency between current CoAXs visualization techniques facilitates interaction and understanding of the visual representations. However, users were interested on more detailed information besides average speed. Concretely, it was required a way of exploring route-level figures of total emissions for various pollutants, and per-segment information of emissions. Finally, as per the original feature of CoAXs, users wanted to create a scenario in which a hypothetical engine technology could be selected. To meet the requirements, AT4 was modified to retrieve information about emissions for each segment. The transformation makes use of the function `getEmissionsRate()` defined in Section 5.3.4.

```
// Analytical task AT4
sumvis:AT4 rdf:type :Transformation .
  rdf:label "Average speed on route segments for Boston during AM peak" .
SELECT ?avgspeed ?routesegment ?line ?emissionsRate ?pollutantType
WHERE {
```

```
?reading rdf:type :Reading .
?reading :isRelatedToLine ?line .
  ?reading :isRelatedToRoute ?route .
  ?reading :isRelatedToRouteSegment ?routeseg .
  ?reading :hasTemporalEntityReference bosod:ampeak .
  ?reading :hasAverageSpeed ?avgspeed .
?engTech rdf:type :EngineTechnology .
?polType rdf:type :Pollutant .
LET (?emissionsRate := :getEmissionsRate(?engTech,?polType,?avgspeed)) . }
```

A bar chart was coupled with the line heat map to provide additional information on route-level figures for total emissions (see Figure 5.27). By rolling the cursor over a specific bar, the visualization shows the total amount of pollutants emitted. We opted to not change the original semantic annotation of the visualization technique, as its characteristics remained unchanged (the input variables in particular).

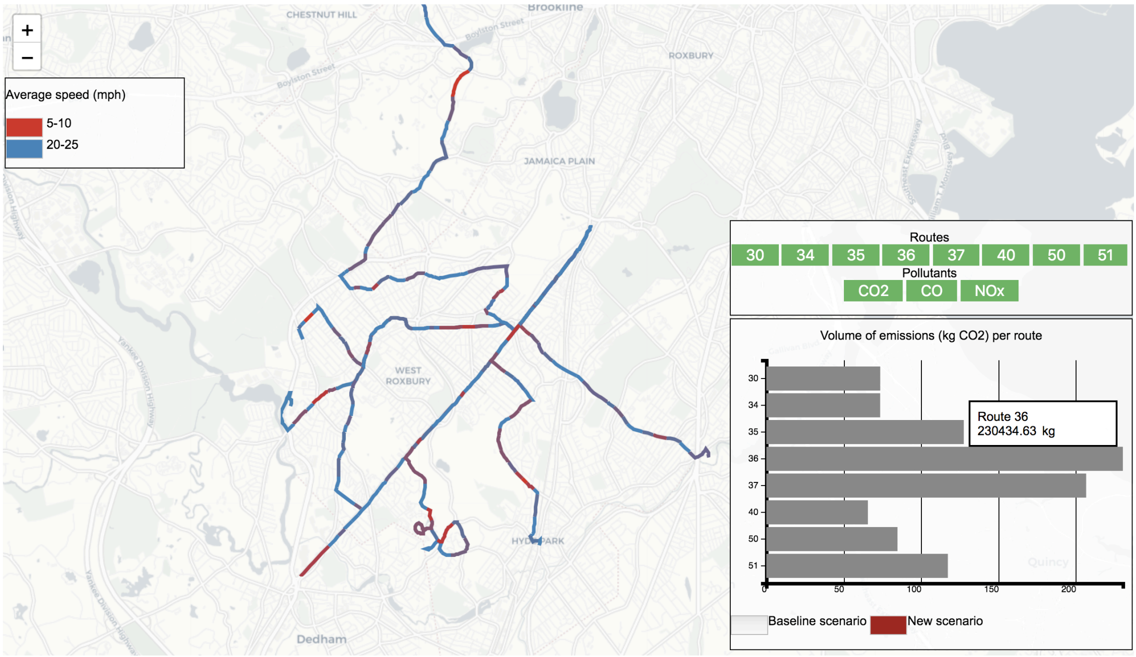


Figure 5.27: Interactive line-based heat map coupled with bar chart to provide route-level figures of emissions. Focus on CO_2 emissions

In Figure 5.28, the bar chart allows for the comparison between current engine technology (Diesel), and a hypothetical scenario in which only Diesel Hybrid buses were considered. Users considered that the overlapping results allowed one to easily recognize the impact of the new scenario.

5.7 Discussion

We argue that the presented case studies provide a comprehensive representation of the possible applications of VUMO to each of the pipelines of knowledge-assisted visualization systems, and are aligned with the practical challenges of real world applications of visualization to urban mobility studies. We also argue that the proposed approach is scalable, given the various requirements we identified throughout the case studies.

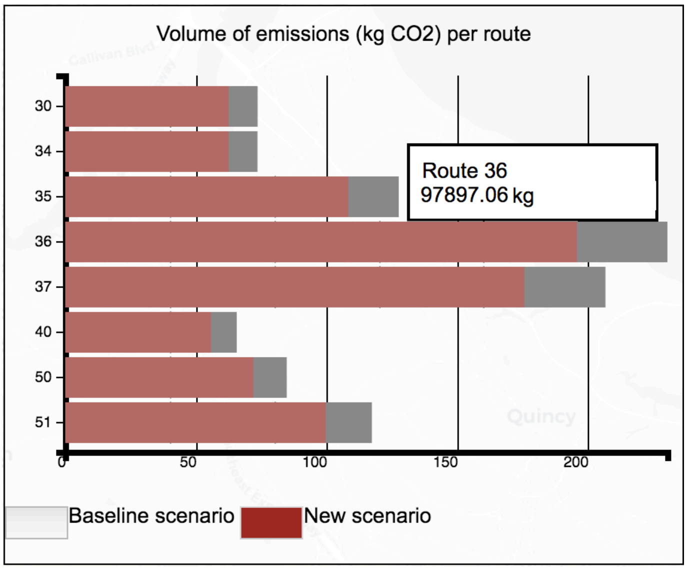


Figure 5.28: Bar chart comparing route-level emission figures for two scenarios. The new scenario with Diesel Hybrid buses (red) considerably reduces the amount of emissions from the baseline scenario, which consists of Diesel buses

We demonstrated the significance of data integration for the cities of Porto and Boston. In Porto, datasets did not conform to standard schemas for urban mobility data. Recalling the example of Section 5.2.2, the Trindade subway station has two distinct identifications in datasets PRT1 (MP30) and PRT2 (TRD). Systems would be unable to associate events to the same spatial reference, if they were characterized according to the first or second dataset. The semantic alignment of the corresponding resources `portotip:MP30` and `portoopt:TRD` overcomes such limitation, while preserving the inherent semantics of the identification in each context. We showed that it is possible to define new standard-based rules to enrich instance data, e.g. for associating stops and stations to city boroughs, or calculation of emissions rate. For example, by combining the Porto boroughs' dataset with other Porto datasets, it could be automatically inferred to which zone they belonged. This new knowledge can be applied to analytical tasks that involve multiple spatial granularities, which only depend on the level of detail of the available data. In Section 5.3, we showed that it is possible to integrate standard-compliant datasets (GTFS) and datasets with various schemas. Moreover, we demonstrated that VUMO is able to be extended with new concepts, while remaining consistent with the conceptual model which it is based on. The ontologies surveyed in the literature review are not sufficient to support the activities we carried out in the case studies, as their constructs do not cover all urban mobility themes we considered for this thesis, and they do not seem to apply rule-based inferencing to enrich the instance data graph, and support the process of data visualization.

We consider that the involvement of domain users offered interesting and challenging tasks, as we strived to meet their requirements using only ontologies and Semantic Web technologies, in order to provide examples that are independent of any proprietary, non-standard technological stack. For instance, the rules and recommendation method that were described in this chapter can be adapted and implemented in other systems, provided that they make use of the VUMO ontology.

The case studies covered relevant urban mobility topics, as identified in our literature review. The visualization techniques that were used as practical examples are relevant, as some of their variations are often found in surveyed studies (for example, see Figures 3.5, 3.6 and 3.7 for heat maps, and 3.32 and [105] for chord diagrams).

5.8 Conclusions

This chapter described practical applications of VUMO to semantic integration of urban mobility data, and user-centered design processes for visualization techniques. The case studies involved public transportation systems from two distinct cities, using data retrieved from various sources.

In Sections 5.2 and 5.3, we demonstrated the semantic integration of various datasets from Porto and Boston, in accordance to pipeline (i), by mapping between source data attributes and VUMO elements. We also demonstrated that new concepts can be introduced and semantically related to VUMO, as shown in Section 5.3, when integrating data from the emission rates dataset (BOS4). *Ad hoc* parsers were then implemented to translate source data onto RDF instance data.

Section 5.4 provided the examples of two visualization technique prototypes built for *SUMVis*. We also built a transformation for analyzing public transportation ridership, and showed which implicit knowledge could be retrieved from VUMO rules, including compatibility with the two visualization techniques. We also showed how VUMO can support the annotation of new knowledge that is derived from exploratory data analysis. As users could identify an unusual behavior in the transportation system due to a strike, all the underlying events were related to a new event that was created to represent the strike.

In Section 5.5, we carried a UCD study for developing visualization techniques to OD flows, using data from Section 5.2 and 5.3. Those studies intended to evaluate if VUMO components allowed comprehensive annotation of visualization techniques and empirical knowledge, in accordance to pipelines (ii) and (iii). We also showed how VUMO components can be used to create recommendation methods and new rules.

Section 5.6 described an exploratory study for the incorporation of emissions visualization in bus corridors into CoAXs, using data from Section 5.3. We demonstrated how analytical tasks could change throughout the involvement with domain users, which implied the modification of the respective data transformations.

Part IV

Conclusions, References and Supplementary Material

Chapter 6

Conclusions

This research aimed at solving the problem of visualizing heterogeneous urban mobility data. The proposed solution was centered on the development of the Visualization Urban Mobility Ontology (VUMO). The ontology provides the necessary semantic foundation for (i) integration of spatiotemporal mobility data from multiple sources, and (ii) formal representation of visualization and empirical knowledge. We also demonstrated how VUMO can support the development of semantically-rich, user-oriented visualization tools, in tasks such as finding compatible and appropriate visualization techniques based on context-specific analytical tasks, and semantic annotation of artifacts from UCD processes.

Chapter 2 introduced the theoretical background for our research. The literature review presented in Chapter 3 showed how visualization and ontologies have been applied to several topics of urban mobility analysis. The chapter highlighted the small involvement of domain users during the development and evaluation of visualization tools. It also showed that the current state of ontologies for integration of urban mobility data still asks for further research regarding spatiotemporal data. Moreover, the application of knowledge-assisted visualization to Transporation, in particular to urban mobility studies, is still subject to exploration, thus becoming one of the motivations for this research. Chapter 4 introduced the VUMO ontology, which implements a novel conceptual model that formalizes spatiotemporal urban mobility data, visualization techniques and empirical knowledge. Finally, Chapter 5 described practical applications of the VUMO ontology supported by a user-centered design methodology, based on the *SUMVis* visualization tool prototype.

6.1 Research Questions Revisited

In this section we revisit the research questions proposed in Section 1.3. The first research objective consisted of developing a visualization-oriented domain ontology for spatiotemporal mobility data, capable of integrating data and supporting knowledge-assisted visualization systems.

Q1. How to conceptualize the fundamental structure of spatiotemporal mobility data, visualization techniques and empirical knowledge?

Based on the theoretical background proposed in Chapter 2, we defined a novel conceptual model that interrelates the aforementioned facets, as shown in the UML Diagram in Figure 4.11.

Regarding spatiotemporal mobility data, an entity (e.g. event) may be characterized in terms of spatial and temporal references, and thematic attributes. Depending on the type of spatial references (e.g. generic points) distinct types of visual arrangements (distributions) emerge (e.g. quasi-continuous). An immediate result is that various types of mobility events can have their spatial distribution inferred in terms of the type of spatial references they require. For instance, ticket validations occur in known points (e.g. bus stops), hence a set of ticket validations induce a discrete spatial distribution. In cases which VUMO should be extended to include new event types, their spatial distributions can still be inferred.

The features of a visualization technique are the intrinsic components that characterize it. The conceptual model specifies four main features categories: input variables, frames of reference, spatial dimensionality and temporal arrangement. The model also specifies the interaction tasks that a technique provides.

Transformations represent the possible operations that can be performed on data, e.g. queries. The results of a transformation are expressed in terms of output variables. A transformation may have output variables related to spatial references, hence we demonstrated that transformations also have intrinsic spatial distributions.

Empirical knowledge concepts allow the modeling of the domain users of a system in terms of their analytical profiles (e.g. strategical). Domain users can provide multiple statements about visualization techniques which are specified in the form of ratings. We introducd two types of ratings: visualization ratings describe feedback that is strictly related to visualization techniques. For instance, a domain user may state that a particular technique is recommended for visualizing data with quasi-continuous spatial arrangement. Cross ratings provide specialized feedback about a visualization technique with respect to a transformation. For instance, a domain user can state that a given visualization technique is recommended to represent ”ticket validations at bus stops”.

Cross ratings can be exploited by recommendation methods to provide even more accurate suggestions for domain users, as described in Chapter 6. The conceptual model provides flexibility for specification of quantitative and qualitative rating criteria. For instance, it is possible to define rating scales for subjective criteria such as ”Visual Complexity” or ”Effectiveness”, or qualitative criteria using concepts from other parts of the conceptual model. As an illustration, the model allows the definition of criteria such as ”Recommended Analytical Profile” or ”Unrecommended Spatial Distributions”. VUMO implements various criteria which provide a solid starting point for the development of recommendation methods. Additional criteria can be created in order to account for different system needs.

Q2. Which concepts should the ontology contain?

The conceptual model that answers Q1 is the basis of the VUMO ontology. After analyzing several datasets from transportation systems and Information Visualization theory, VUMO allows the specification of the infrastructure of a transportation system and the events that occur in it (members of `UrbanMobilityConcept` class). Transformations and structural components of data are specified as in the conceptual model (members of `DataConcept` class). Visualization techniques can be annotated in terms of their input variables, which receive the values from output variables in transformations, and other features (members of `VisualizationConcept`). The annotation of domain users and their empirical knowledge are given in terms of their analytical profile and ratings about visualization techniques and transformations (members of `DomainUserConcept` class).

Finally, a novel contribution consisted on the use of a rule language (SPIN) for reasoning over instance data that has been semantically annotated with VUMO. We built a set of rules that allows the inference of implicit knowledge. The most notable capabilities are the extraction of features from transformations (expressed in queries) and the evaluation of compatible visualizations for a given transformation. Such set is not meant to be exhaustive; but shows that other rules can be built under the same rationale.

The second objective consists of proposing an ontology-based foundational framework for supporting semantically-rich, knowledge-assisted visualization systems.

Q3. How to comprehensively use the ontology on the development of visualization tools for spatiotemporal urban mobility data?

In Section 5.10, we showed how VUMO could be applied to user-centered design processes, as summarized in Figure 4.19. All phases are linked to the semantic annotation phase, in which the outcomes are annotated using VUMO components. The results of the context of Context of Use Analysis and Problem Domain Analysis (Phases 0 and 1) yield the annotation of domain users, their types of analytical profiles and the analytical tasks (transformations) they intend to carry. After iterating the Conceptual Development and Prototyping (Phases 2 and 3), visualization techniques can be described in terms of input variables, interaction tasks and other features. Interaction and Usability Studies (Phase 4) yield empirical knowledge that domain users produce. Such artifacts can be annotated in the form of ratings and cross ratings.

The third objective consisted of evaluating the ontology in practical contexts to validate the aforementioned objectives.

Q4. Does our approach succeed in tackling issues I2 (limited interoperability of visualizations) and I3 (infrequent involvement of domain experts on the design, development and evaluation of visualization techniques)?

Chapter 6 answers the last proposed question. The empirical case studies provide evidence of the successful application of VUMO to the development of semantically-rich, knowledge-assisted visualization tools. The case studies were supported by the *SUMVis* visualization tool, which was developed for the purposes of this thesis. We used real data from Porto and Boston, and involved domain users from both contexts with various academic backgrounds and practical expertise. We showed how VUMO can be applied to the pipelines of (i) data integration, (ii) annotation of visualization techniques, and (iii) annotation of empirical knowledge.

For the first pipeline, we detailed the integration of different types of urban mobility datasets from both cities. After the specification of two visualization techniques and a transformation, we showed how VUMO could evaluate their compatibility. Moreover, we showed how VUMO could support the semantic annotation of knowledge extracted from the exploratory analysis performed by domain experts, after they identified an unusual behavior in the transportation system of Porto at a specific time period.

For the remaining pipelines, we carried UCD processes with domain users, and described the semantic annotation of the resulting artifacts. Those iterations revealed interesting opportunites that highlight how VUMO can be extended new domain user requirements. For example, we showed how a new rule could be built in order to identify the analytical profile of a transformation based on the themes that compose it (see Section 6.4.1).

We described the implementation of a customized, VUMO-assisted recommendation method based on user requirements (see Section 6.4.4), which was also supported by rules. Moreover, it was also shown that, in cases where the ontology did not contain the required classes and properties for data integration (see Section 5.3), VUMO could be extended to account for new types of mobility-related datasets, thus showing its scalability potential.

To facilitate reading, we reintroduce the illustration of the thesis outline (see Figure 6.1), which indicates the research questions answered by each chapter.

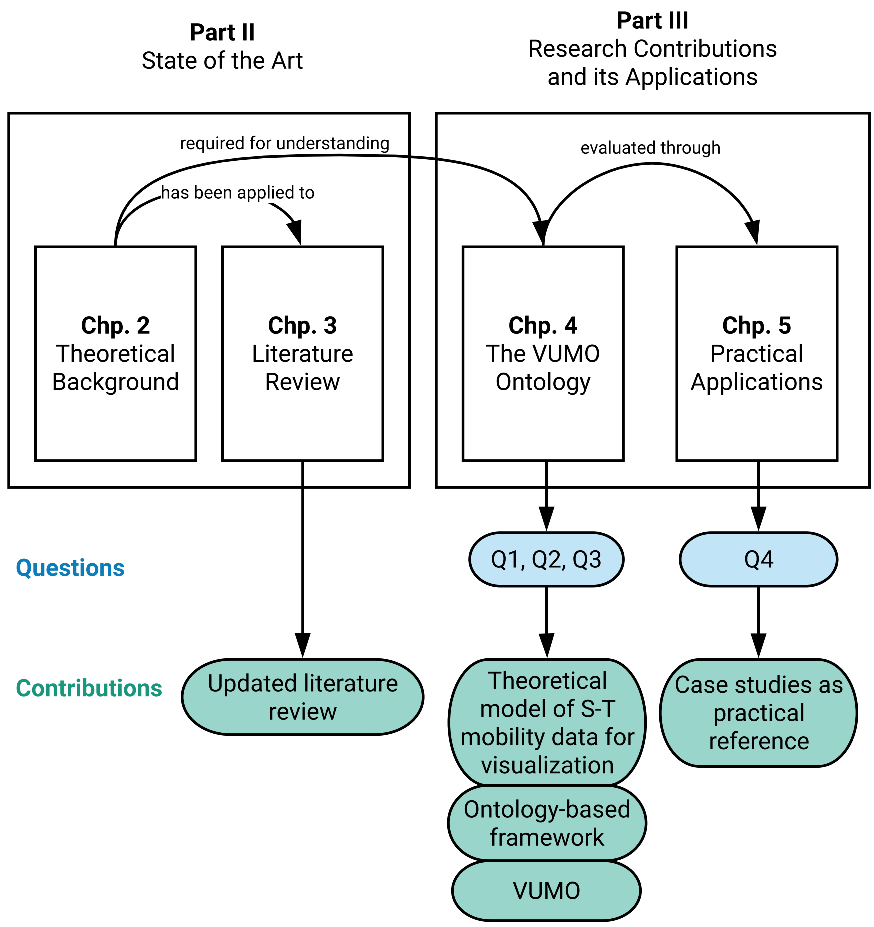


Figure 6.1: Illustration of the thesis outline. Repetition was done on purpose to facilitate reading

6.2 Contributions and Research Limitations

This thesis provides an original contribution to the state of the art by introducing an approach to visualization of heterogeneous urban mobility data from multiple sources, by exploring the potential of ontologies and Semantic Web technologies. Our approach values the involvement of domain users throughout the pipelines that form development cycle of visualization systems, as we found that such involvement is infrequent in scientific literature. The VUMO ontology addresses the shortcomings of other existing ontologies, such as OTN [150], which provide limited support to spatiotemporal urban mobility data. We demonstrated how inference rules built upon *de facto* standards can enhance the utility of integrated data and support the visualization process. We believe that our contributions serves a relevant reference to other domains of knowledge that also require integration of spatiotemporal data.

Regarding research limitations, the development and evaluation of the VUMO ontology did not have direct involvement of domain users. Firstly, few of the domain users we approached had some knowledge of Semantic Web technologies and ontologies; it would be unfeasible. Moreover, the limited availability of domain users led us to consider it would be reasonable to approach them while carrying the practical case studies.

In Section 4.1.4, we assumed that an analytical task can be formalized as a data transformation (query). While one may argue that analytical tasks can be sufficiently scaled to require one or more queries, we posit that our one-to-one assumption does not reduce the utility and effectiveness of the data visualization process, as demonstrated in the case studies, in which we successfully formalized a number of relevant analytical tasks. Still, we acknowledge that an interesting direction for further work should investigate how semantically-rich visualization systems can formalize and support analytical tasks that are formalized in terms of several data transformations.

On regards to the type of urban mobility events that VUMO supports, we showed that our formalization of spatiotemporal data allows to model various types of events, including soft modes, e.g. bicycle trips and pedestrian trajectories. The presented case studies were oriented towards data related to public transportation systems, given the availability of datasets. We argue that such fact does not compromise the generality of our approach, although it presents a direction we are willing to evaluate.

Finally, in Section 4.9, we highlighted the limited use of the resources provided by the `time` ontology. The reason is twofold. Firstly, the temporal representation in the datasets we posessed consisted of simple timestamps or textual descriptions of time intervals. We argue that adding unnecessary complexity to temporal modeling could be a factor that would hinder researchers and practitioners on the implementation VUMO. Secondly, we argue that using all constructs available in the `time` may still be unnecessary given the simple way that datasets often use to express time. Moreover, to the best of our knowledge, inference engines still provide limited support to the `time` ontology.

6.3 Looking ahead

Future work can be divided into three perspectives. The first, mainly technical, consists of improvements to the features and capabilities of the VUMO ontology described in Chapter 5. The second are improvements and further evaluation of VUMO-based UCD applications, as described in Chapter 6. Finally, the third perspective consists of ideas that we considered relevant for exploration in the long term.

Regarding the first perspective, possible improvements consist of extending the capabilities of semantic annotation of analytical tasks (data transformations) to not only SPARQL queries, but complex algorithms such as data mining methods. Moreover, as discussed in the research limitations, it would be interesting to explore how to address analytical tasks as a chain of multiple data transformations. While the current classes and properties in VUMO provide support to the majority of types of mobility modes, we aim to extend it in order to consider other types of data sources that we believe to be widely available in the near future. For example, the ubiquity of Internet of Things devices can provide new types of sensorial measurements. VUMO still does not provide the necessary components to semantically annotate non-text-based data such as video streams.

During the conclusion of this thesis, the SPIN rule language has been thoroughly standardized into SHACL (Shapes Constraint Language) by W3C; the language is considered to be an evolution of OWL. We plan to translate VUMO into SHACL, so that researchers and practitioners can opt for OWL or SHACL, in accordance to their requirements. Nonetheless, we alert the reader that, by the time of this thesis, few Semantic Web based systems provide thorough support to SHACL.

Regarding the second perspective, we intend to carry additional case studies with the SUMVis tool. As discussed in the research limitations, most of our datasets were related to public transportation systems. We aim to use VUMO with datasets containing information with other types of mobility traces, e.g. bicycle, private car trajectories or phone-based activity. In Chapter 3 we evidenced that visualization studies in urban mobility made use of several types of datasets, as summarized in Table 3.2. As an ongoing extension of our research, future work also consists of using VUMO as the semantic foundation for a decision support system for Porto's intermodal public transportation system.

Regarding the third perspective, we plan to provide an online shared repository for reuse of empirical knowledge. This would allow researchers and practitioners from various contexts to reuse knowledge made public from other studies or actual practical contexts. This idea is aligned with our orientation towards the involvement of domain users throughout the whole UCD process. Moreover, the idea is also aligned with the Semantic Web principle of reusing existing knowledge, in order to enhance the capability of other systems that also make use of such technologies. We also seek to facilitate the integration of data that already comply with standards such as SIRI and Transmodel, and data related to factors that are known to influence urban mobility dynamics, such as weather conditions.

Bibliography

[1] United Nations. World Urbanization Prospects: The 2014 Revision. 2014. ISSN 1435-9871. doi: 10.4054/DemRes.2005.12.9.

[2] Ta Chiraphadhanakul. A Network Analysis of Hubway: How does Hubway complement MBTA services?, 2014. URL http://hubwaydatachallenge.org/submission/45/. Last accessed: April 24, 2018.

[3] Thiago Sobral. *Developing Visualisations for Urban Mobility Data: A User-Centred Approach*. Master's thesis, Faculty of Engineering of the University of Porto, 2014.

[4] Thiago Sobral, Teresa Galvão Dias, and José Luís Borges. Towards a conceptual framework for classifying visualisations of data from urban mobility services. In *Lecture Notes in Business Information Processing*, volume 201, pages 228–242, 2015. ISBN 9783319149790. doi: 10.1007/978-3-319-14980-6_18.

[5] Google. General Transit Feed Specification (GTFS). URL https://developers.google.com/transit/. Last accessed: December 31, 2017.

[6] European Committee for Standardization. SIRI Standard Specification. URL https://www.vdv.de/siri.aspx. Last accessed: June 3, 2016.

[7] Wei Chen, Fangzhou Guo, and Fei-yue Wang. A Survey of Traffic Data Visualization. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):2970–2984, 2015. ISSN 1524-9050. doi: 10.1109/TITS.2015.2436897.

[8] Dan Brickley. Basic Geo (WGS84 lat/long) Vocabulary, 2003. URL https://www.w3.org/2003/01/geo/.

[9] Colin Ware. *Information visualization : perception for design*. Morgan Kaufmann, 2012. ISBN 9780123814647. URL https://www.sciencedirect.com/science/book/9780123814647.

[10] William Playfair. *The Commercial and Political Atlas and Statistical Breviary*. 1786. URL https://en.wikipedia.org/wiki/William%5C_Playfair.

[11] Florence Nightingale. Notes on matters affecting the health, efficiency, and hospital administration of the British Army, 1858. URL https://en.wikipedia.org/wiki/Florence%5C_Nightingale%5C#/media/File:Nightingale-mortality.jpg.

[12] John Corbett. Charles Joseph Minard: Mapping Napoleon's March, 1861. URL http://www.csiss.org/classics/content/58. Last accessed: September 20, 2014.

[13] Stuart K. Card, Jock D. Mackinlay, and Ben. Shneiderman, editors. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999. ISBN 1-55860-533-9.

[14] Robert Spence. *Information Visualization: Design for Interaction*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2nd edition, 2007. ISBN 0132065509.

[15] San Francisco County Transportation Authority. TNCs Today - Uber and Lyft in San Francisco, 2017. URL http://tncstoday.sfcta.org/. Last accessed: September 18, 2018.

[16] Bastian Nebenfuhr. *OpenFOAM: A tool for predicting automotive relevant flow fields*. Master's thesis, Chalmers University of Technology, Sweden, 2010.

[17] Jacques Bertin. *Semiology of Graphics*. University of Wisconsin Press, 1983. ISBN 0299090604.

[18] Natalia Andrienko and Gennady Andrienko. *Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach*. Springer Berlin Heidelberg, 2006.

[19] Ben Shneiderman. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *Proceedings of the 1996 IEEE Symposium on Visual Languages*, VL '96, pages 336–343, Washington, DC, USA, 1996. IEEE Computer Society. ISBN 0-8186-7508-X. URL http://dl.acm.org/citation.cfm?id=832277.834354.

[20] Matthew Bloch and Robert Gebeloff. Immigration Explorer - The New York Times, 2009. URL http://www.nytimes.com/interactive/2009/03/10/us/20090310-immigration-explorer.html. Last accessed: March 10, 2016.

[21] Andy Cockburn, Amy Karlson, and Benjamin B Bederson. A Review of Overview+Detail, Zooming, and Focus+Context Interfaces. *ACM Computing Surveys*, 41(1):2:1–2:31, jan 2009. ISSN 0360-0300. doi: 10.1145/1456650.1456652.

[22] Marti A Hearst. *Search User Interfaces*. Cambridge University Press, New York, NY, USA, 1st edition, 2009. ISBN 0521113792, 9780521113793.

[23] Robert Voigt. *An Extended Scatterplot Matrix and Case Studies in Information Visualization*. PhD thesis, Hochschule Magdeburg-Stendal, Germany, 2002.

[24] University York. Fish Eye view of Central Washington, D.C., 2010. Last accessed: April 17, 2018.

[25] Maged N Kamel Boulos. The use of interactive graphical maps for browsing medical/health Internet information resources. *International Journal of Health Geographics*, 2 (1):1, jan 2003. ISSN 1476-072X. doi: 10.1186/1476-072X-2-1.

[26] L Bartram. Can motion increase user interface bandwidth in complex systems? In *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, volume 2, pages 1686–1692 vol.2, oct 1997. doi: 10.1109/ICSMC.1997.638254.

[27] Peter R. Keller and Mary M. Keller. *Visual Cues: Practical Data Visualization*. IEEE Computer Society Press, Los Alamitos, CA, USA, 1994. ISBN 0818631023.

[28] Ji Soo Yi, Youn ah Kang, John Stasko, and Julie Jacko. Toward a Deeper Understanding of the Role of Interaction in Information Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1224–1231, nov 2007. ISSN 1077-2626. doi: 10.1109/TVCG.2007.70515.

[29] Kawa Nazemi. *Adaptive Semantics Visualization*. Springer, Cham, 2016.

[30] Ed H. Chi. A Taxonomy of Visualization Techniques Using the Data State Reference Model. In *Proceedings of the IEEE Symposium on Information Vizualization 2000*, INFOVIS '00, pages 69–75, Washington, DC, USA, 2000. IEEE Computer Society. ISBN 0-7695-0804-9. URL `http://dl.acm.org/citation.cfm?id=857190.857691`.

[31] Daniel A. Keim. Information Visualization: Scope, Techniques and Opportunities for Geovisualization. In *Exploring Geovisualization*, chapter 2, pages 21–52. Elsevier, 2005. ISBN 9780080445311. doi: 10.1016/B978-008044531-1/50420-6.

[32] Torsten Hägerstrand. What about people in Regional Science? *Papers of the Regional Science Association*, 24(1):6–21, 1970. ISSN 0486-2902. doi: 10.1007/BF01936872.

[33] Donna Peuquet and Menno-Jan Kraak. Geobrowsing: Creative Thinking and Knowledge Discovery Using Geographic Visualization. *Information Visualization*, 1(1):80–91, 2002.

[34] Natalie Adrienko, Genaldi Adrienko, and Peter Gatalsky. Impact of Data and Task Characteristics on Design of Spatio-Temporal Data Visualization Tools. In J Dykes, A M MacEachren, and M.-J. Kraak, editors, *Exploring Geovisualization*. Elsevier Ltd., 2005.

[35] Lian Chee Koh, Aidan Slingsby, Jason Dykes, and Tin Seong Kam. Developing and applying a user-centered model for the design and implementation of information visualization tools. In *Information Visualisation (IV), 2011 15th International Conference on*, pages 90–95. IEEE, 2011.

[36] Robert Kosara Christopher, Christopher G Healey, Victoria Interrante, David H Laidlaw, and Colin Ware. Thoughts on User Studies: Why, How, and When. *IEEE Computer Graphics and Applications*, 23(1):2003, 2003.

[37] Anthony C. Robinson, Jin Chen, Eugene J. Lengerich, Hans G. Meyer, and Alan M. Maceachren. Combining usability techniques to design geovisualization tools for epidemiology. *Information Science*, 32:243–255, 2005.

[38] Robert E. Roth, Kevin S. Ross, Benjamin G. Finch, Wei. Luo, and Alan M. MacEachren. A user-centered approach for designing and developing spatiotemporal crime analysis tools. *Zurich, Switzerland*, 2010.

[39] Terry A. Slocum, Daniel C. Cliburn, Johannes J. Feddema, and James R. Miller. Evaluating the Usability of a Tool for Visualizing the Uncertainty of the Future Global Water Balance. *Cartography and Geographic Information Science*, 30(4):299–317, 2003. doi: 10.1559/152304003322606210.

[40] Melanie Tory and Thorsten Moller. Evaluating visualizations: do expert reviews work? *IEEE Transactions on Computer Graphics and Applications*, 25(5):8–11, sep 2005. ISSN 0272-1716. doi: 10.1109/MCG.2005.102.

[41] Geoffrey Ellis and Alan Dix. An Explorative Analysis of User Evaluation Studies in Information Visualisation. In *Proceedings of the 2006 AVI Workshop on Beyond Time and Errors: Novel Evaluation Methods for Information Visualization*, BELIV '06, pages 1–7, New York, NY, USA, 2006. ACM. ISBN 1-59593-562-2. doi: 10.1145/1168149.1168152.

[42] John Hebeler, Matthew Fisher, Ryan Blace, and Andrew Perez-Lopez. *Semantic Web Programming*. Wiley Publishing, 2009. ISBN 047041801X, 9780470418017.

[43] Jonathan B L Bard and Seung Y Rhee. Ontologies in biology: design, applications and future challenges. *Nature Reviews Genetics*, page 213. doi: 10.1038/nrg1295.

[44] H Wache, T Všgele, U Visser, H Stuckenschmidt, G Schuster, H Neumann, and S HŸbner. Ontology-Based Integration of Information - A Survey of Existing Approaches. In *In IJCAI'01 Workshop on Ontologies and Information Sharing*, 2001.

[45] Jeremy Pitt and Abe Mamdani. A protocol-based semantics for an agent communication language. In *Proceedings of the 16th International Joint Conference on Artifical Intelligence*, volume 1 of *IJCAI'99*, pages 486–491, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc. URL `http://dl.acm.org/citation.cfm?id=1624218.1624288`.

[46] W3C. XML Schema Part 2: Datatypes Second Edition, 2004. URL `https://www.w3.org/TR/xmlschema-2/`. Last accessed: December 24, 2017.

[47] W3C. SPARQL 1.1 Query Language, 2003. URL `https://www.w3.org/TR/sparql11-query/`. Last accessed: January 1, 2018.

[48] W3C. SWRL: A Semantic Web Rule Language Combining OWL and RuleML. URL `https://www.w3.org/Submission/SWRL/`. Last accessed: October 1, 2016.

[49] Holger Knublauch. SPARQL Inference Notation (SPIN). URL `http://spinrdf.org/`.

[50] University of Manchester. FaCT++ Reasoner, 2011. URL `http://owl.man.ac.uk/factplusplus/`. Last accessed: January 2, 2018.

[51] Thomas R. Gruber. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2):199–220, jun 1993. ISSN 1042-8143. doi: 10.1006/knac.1993.1008.

[52] G. van Heijst, A. Th. Schreiber, and B. J. Wielinga. Using Explicit Ontologies in KBS Development. *International Journal on Human-Computer Studies*, 46(2-3):183–292, mar 1997. ISSN 1071-5819. doi: 10.1006/ijhc.1996.0090.

[53] W. N. Borst. *Construction of Engineering Ontologies*. PhD thesis, University of Twente, 1997.

[54] Mike Uschold and Michael Gruninger. Ontologies: Principles, methods and applications. *Knowledge Engineering Review*, 1996.

[55] Petar Kramaric. *Adopt: A Context-Aware Domain Ontology-Based Framework for Public Transportation*. PhD thesis, Ryerson University, 2012.

[56] Paulo Cesar G. da Costa. *Bayesian Semantics for the Semantic Web*. PhD thesis, George Mason University, 2005.

[57] Dean Allemang and James Hendler. *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2nd edition, 2011. ISBN 9780123859655, 9780123859662.

[58] Gio Wiederhold. Mediators in the architecture of future information systems. *Computer*, 25(3):38–49, mar 1992. ISSN 0018-9162. doi: 10.1109/2.121508.

[59] Gio Wiederhold. Intelligent Integration of Information. *SIGMOD Records*, 22(2):434–437, jun 1993. ISSN 0163-5808. doi: 10.1145/170036.170118.

[60] G Wiederhold and M Genesereth. The conceptual basis for mediation services. *IEEE Expert*, 12(5):38–47, sep 1997. ISSN 0885-9000. doi: 10.1109/64.621227.

[61] Holger Wache. Towards Rule-Based Context Transformation in Mediators. In *in International Workshop on Engineering Federated Information Systems (EFIS)*, pages 107–122. Infix-Verlag, 1999.

[62] Josep Maria Brunetti, Sören Auer, Roberto García, Jakub Klímek, and Martinh9 Nečaský. Formal Linked Data Visualization Model. In *Proceedings of International Conference on Information Integration and Web-based Applications*, IIWAS '13, pages 309–318, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2113-6. doi: 10.1145/2539150.2539162.

[63] Belgin Mutlu, Eduardo Veas, and Christoph Trattner. VizRec: Recommending Personalized Visualizations. *ACM Transactions on Interactive Intelligent Systems*, 6(4):1:39, nov 2016. ISSN 2160-6455. doi: 10.1145/2983923.

[64] Chris Stolte and Pat Hanrahan. Polaris: a system for query, analysis and visualization of multi-dimensional relational databases. In *IEEE Symposium on Information Visualization 2000*, pages 5–14, 2000. ISBN 1522-404X. doi: 10.1109/INFVIS.2000.885086.

[65] Jock D. Mackinlay, Pat Hanrahan, and Chris Stolte. Show Me: Automatic Presentation for Visual Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 13(6): 1137–1144, 2007. ISSN 1077-2626. doi: 10.1109/TVCG.2007.70594.

[66] Martin Voigt, Martin Franke, and Klaus Meissner. Using expert and empirical knowledge for context-aware recommendation of visualization components. *International Journal on Advances in Life Sciences*, 5:27–41, 2013.

[67] Chien Chin Chen, Yu-Hao Wan, Meng-Chieh Chung, and Yu-Chun Sun. An effective recommendation method for cold start new users using trust and distrust networks. *Information Sciences*, 224:19–36, 2013. ISSN 0020-0255. doi: 10.1016/j.ins.2012.10.037.

[68] Ronald Hughes. Visualization in Transportation: Current Practice and Future Directions. *Transportation Research Record: Journal of the Transportation Research Board*, 1899:167–174, jan 2004. ISSN 0361-1981. doi: 10.3141/1899-21.

[69] Michael L. Pack, Krist Wongsuphasawat, Michael VanDaniker, and Darya Filippova. ICE - Visual analytics for transportation incident datasets. In *IEEE International Conference on Information Reuse and Integration*, 2009. ISBN 9781424441167. doi: 10.1109/IRI.2009.5211551.

[70] Ronald Hughes. Research Agenda for the Application of Visualization to Transportation Systems. *Transportation Research Record: Journal of the Transportation Research Board*, 1937:145–151, jan 2005. ISSN 0361-1981. doi: 10.3141/1937-20.

[71] Junping Zhang, Fei-Yue Wang, Kunfeng Wang, Wei-Hua Lin, Xin Xu, and Cheng Chen. Data-Driven Intelligent Transportation Systems: A Survey. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1624–1639, 2011. ISSN 1524-9050. doi: 10.1109/TITS.2011.2158001.

[72] Genady Andrienko, Natalia Andrienko, Wei Chen, Ross Maciejewski, and Ye Zhao. Visual Analytics of Mobility and Transportation: State of the Art and Further Research Directions. *IEEE Transactions on Intelligent Transportation Systems*, 18(8):2232–2249, aug 2017. ISSN 1524-9050. doi: 10.1109/TITS.2017.2683539.

[73] C Claramunt, B Jiang, and A Bargiela. A new framework for the integration, analysis and visualisation of urban traffic data within geographic information systems. *Transportation Research Part C: Emerging Technologies*, 8(1):167–184, 2000. ISSN 0968090X. doi: 10.1016/S0968-090X(00)00009-7.

[74] Shashi Shekhar, Charlie Tsung-hsing Lu, Ren Ping Liu, and Chen Zhou. Cube view: A system for traffic data visualization. In *IEEE Conference on Intelligent Transportation Systems*, 2002. ISBN 0780373898. doi: 10.1109/ITSC.2002.1041299.

[75] Xinhao Wang. Integrating GIS, simulation models, and visualization in traffic impact analysis. *Computers, Environment and Urban Systems*, 29(4):471–496, 2005. ISSN 01989715. doi: 10.1016/j.compenvurbsys.2004.01.002.

[76] Jason Sewall, Jur Van Den Berg, Ming Lin, and Dinesh Manocha. Virtualized traffic: Reconstructing traffic flows from discrete spatiotemporal data. *IEEE Transactions on Visualization and Computer Graphics*, 17(1):26–37, 2011. ISSN 10772626. doi: 10.1109/TVCG.2010.27.

[77] Hanqi Guo, Zuchao Wang, Bowen Yu, Huijing Zhao, and Xiaoru Yuan. TripVista: Triple Perspective Visual Trajectory Analytics and its application on microscopic traffic data at a road intersection. In *2011 IEEE Pacific Visualization Symposium*, pages 163–170, mar 2011. doi: 10.1109/PACIFICVIS.2011.5742386.

[78] Susan Havre, Elizabeth Hetzler, Paul Whitney, and Lucy Nowell. ThemeRiver: Visualizing Thematic Changes in Large Document Collections. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):9–20, jan 2002. ISSN 1077-2626. doi: 10.1109/2945.981848.

[79] Peter Bak, Eli Packer, Harold J. Ship, and Dolev Dotan. Algorithmic and visual analysis of spatiotemporal stops in movement data. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems - SIGSPATIAL '12*, page 462, New

York, New York, USA, 2012. ACM Press. ISBN 9781450316910. doi: 10.1145/2424321.2424390.

[80] Ying Song and Harvey J. Miller. Exploring traffic flow databases using space-time plots and data cubes. *Transportation*, 39(2):215–234, 2012. ISSN 00494488. doi: 10.1007/s11116-011-9343-z.

[81] Siyuan Liu, Jiansu Pu, Qiong Luo, Huamin Qu, Lionel M. Ni, and Ramayya Krishnan. VAIT: A visual analytics system for metropolitan transportation. *IEEE Transactions on Intelligent Transportation Systems*, 14(4):1586–1596, 2013. ISSN 15249050. doi: 10.1109/TITS.2013.2263225.

[82] Jiansu Pu, Siyuan Liu, Ye Ding, Huamin Qu, and Lionel Ni. T-Watcher: A New Visual Analytic System for Effective Traffic Surveillance. In *2013 IEEE 14th International Conference on Mobile Data Management*, pages 127–136. IEEE, jun 2013. ISBN 978-0-7695-4973-6. doi: 10.1109/MDM.2013.23.

[83] Yi-Cheng Chen, Yu-Shuen Wang, Wen-Chieh Lin, Wei-Xiang Huang, and I-Chen Lin. Interactive Visual Analysis for Vehicle Detector Data. *Computer Graphics Forum*, 34(3):171–180, 2015. ISSN 01677055. doi: 10.1111/cgf.12629.

[84] Zuchao Wang, Min Lu, Xiaoru Yuan, Junping Zhang, and Huub Van De Wetering. Visual traffic jam analysis based on trajectory data. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2159–2168, 2013. ISSN 10772626. doi: 10.1109/TVCG.2013.228.

[85] Anna Izabel J. Tostes, Fátima de L. P. Duarte-Figueiredo, Renato Assunção, Juliana Salles, and Antonio A. F. Loureiro. From data to knowledge: City-wide Traffic Flows Analysis and Prediction Using Bing Maps. *Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing - UrbComp '13*, page 1, 2013. doi: 10.1145/2505821.2505831.

[86] Jorge Poco, Harish Doraiswamy, Huy T. Vo, João L D Comba, Juliana Freire, and Cláudio T. Silva. Exploring Traffic Dynamics in Urban Environments Using Vector-Valued Functions. *Computer Graphics Forum*, 34(3):161–170, 2015. ISSN 14678659. doi: 10.1111/cgf.12628.

[87] Chengkun Liu, Kun Qin, and Chaogui Kang. Exploring time-dependent traffic congestion patterns from taxi trajectory data. In *2nd IEEE International Conference on Spatial Data Mining and Geographical Knowledge Services (ICSDM)*, pages 39–44. IEEE, jul 2015. ISBN 978-1-4799-7748-2. doi: 10.1109/ICSDM.2015.7298022.

[88] Natasha Petrovska and Aleksandar Stevanovic. Traffic Congestion Analysis Visualisation Tool. In *IEEE Conference on Intelligent Transportation Systems*, pages 1489–1494, 2015. ISBN 9781467365956. doi: 10.1109/ITSC.2015.243.

[89] Ari Wibisono, Wisnu Jatmiko, Hanief Arief Wisesa, Benny Hardjono, and Petrus Mursanto. Traffic big data prediction and visualization using Fast Incremental Model Trees-Drift Detection (FIMT-DD). *Knowledge-Based Systems*, 93:33–46, 2016. ISSN 09507051. doi: 10.1016/j.knosys.2015.10.028.

[90] Tao Cheng, Garavig Tanaksaranond, Chris Brunsdon, and James Haworth. Exploratory visualisation of congestion evolutions on urban transport networks. *Transportation Research Part C: Emerging Technologies*, 36(1):296–306, 2013. ISSN 0968090X. doi: 10.1016/j.trc.2013.09.001.

[91] Yuzuru Tanaka, Hajime Imura, and Jonas Sjöbergh. Exploratory Visual Analytics for Winter Road Management Using Statistically Preprocessed Probe-Car Data. In *Smart Sensors and Systems*, pages 109–128. Springer International Publishing, Cham, 2015. doi: 10.1007/978-3-319-14711-6_6.

[92] Fei Wang, Wei Chen, Feiran Wu, Ye Zhao, Han Hong, Tianyu Gu, Long Wang, Ronghua Liang, and Hujun Bao. A visual reasoning approach for data-driven transport assessment on urban roads. In *IEEE Conference on Visual Analytics Science and Technology*, pages 103–112, 2015. ISBN 9781479962273. doi: 10.1109/VAST.2014.7042486.

[93] Cheng-you Hsieh and Yu-shuen Wang. Traffic situation visualization based on video composition. *Computers and Graphics*, 54:1–7, 2016. ISSN 0097-8493. doi: 10.1016/j.cag.2015.07.007.

[94] Xiaoke Huang, Ye Zhao, Chao Ma, Jing Yang, Xinyue Ye, and Chong Zhang. TrajGraph: A Graph-Based Visual Analytics Approach to Studying Urban Network Centralities Using Taxi Trajectory Data. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):160–169, 2016. ISSN 1077-2626. doi: 10.1109/TVCG.2015.2467771.

[95] Natalia Andrienko, Gennady Andrienko, and Salvatore Rinzivillo. Leveraging spatial abstraction in traffic analysis and forecasting with visual analytics. *Information Systems*, 57:172–194, 2016. ISSN 03064379. doi: 10.1016/j.is.2015.08.007.

[96] Chaogui Kang, Song Gao, Xing Lin, Yu Xiao, Yihong Yuan, Yu Liu, and Xiujun Ma. Analyzing and geo-visualizing individual human mobility patterns using mobile call records. In *18th International Conference on Geoinformatics*, pages 1–7. IEEE, jun 2010. ISBN 978-1-4244-7301-4. doi: 10.1109/GEOINFORMATICS.2010.5567857.

[97] Günther Sagl, Martin Loidl, and Euro Beinat. A Visual Analytics Approach for Extracting Spatio-Temporal Urban Mobility Information from Mobile Network Traffic. *ISPRS International Journal of Geo-Information*, 1(3):256–271, 2012. ISSN 2220-9964. doi: 10.3390/ijgi1030256.

[98] Xiaoqing Zuo and Yongchuan Zhang. Detection and analysis of urban area hotspots based on cell phone traffic. *Journal of Computers*, 7(7):1753–1760, 2012. URL `http://www.scopus.com/inward/record.url?eid=2-s2.0-84864492849&partnerID=40&md5=05913101923b4d0a0708e6a3847b92a8`.

[99] Merkebe Getachew Demissie, Gonçalo Homem de Almeida Correia, and Carlos Bento. Exploring cellular network handover information for urban mobility analysis. *Journal of Transport Geography*, 31:164–170, 2013. ISSN 09666923. doi: 10.1016/j.jtrangeo.2013.06.016.

[100] Natalia Andrienko, Genady Andrienko, Georg Fuchs, and Piotr Jankowski. Scalable and privacy-respectful interactive discovery of place semantics from human mobility traces. *Information Visualization*, 15(2):117–153, apr 2016. ISSN 1473-8716. doi: 10.1177/1473871615581216.

[101] Tatiana von Landesberger, Felix Brodkorb, Philipp Roskosch, Natalia Andrienko, Gennady Andrienko, and Andreas Kerren. MobilityGraphs: Visual Analysis of Mass Mobility Dynamics via Spatio-Temporal Graphs and Clustering. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):11–20, 2016. ISSN 1077-2626. doi: 10.1109/TVCG.2015.2468111.

[102] Nivan Ferreira, Jorge Poco, Huy T. Vo, Juliana Freire, and Claudio T. Silva. Visual exploration of big spatio-temporal urban data: A study of New York city taxi trips. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2149–2158, 2013. ISSN 10772626. doi: 10.1109/TVCG.2013.226.

[103] Ko Ko Lwin and Yuji Murayama. Person trip data browser, analyzer and space-time visualizer. In *ICSDM 2015 - Proceedings 2015 2nd IEEE International Conference on Spatial Data Mining and Geographical Knowledge Services*, pages 12–17, 2015. ISBN 9781479977482. doi: 10.1109/ICSDM.2015.7298017.

[104] Siming Chen, Xiaoru Yuan, Zhenhuang Wang, Cong Guo, Jie Liang, Zuchao Wang, Xiaolong Luke Zhang, and Jiawan Zhang. Interactive Visual Discovering of Movement Patterns from Sparsely Sampled Geo-tagged Social Media Data. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):270–279, 2016. ISSN 10772626. doi: 10.1109/TVCG.2015.2467619.

[105] Nuno Nunes, Miguel Ribeiro, Catia Prandi, and Valentina Nisi. Beanstalk: A Community Based Passive Wi-fi Tracking System for Analysing Tourism Dynamics. In *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '17, pages 93–98, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-5083-9. doi: 10.1145/3102113.3102142.

[106] Linhua Li, Li Zhu, and Daniel Z. Sui. A GIS-based Bayesian approach for analyzing spatial–temporal patterns of intra-city motor vehicle crashes. *Journal of Transport Geography*, 15(4):274–285, 2007. ISSN 09666923. doi: 10.1016/j.jtrangeo.2006.08.005.

[107] Brian Hilton, Thomas Horan, Richard Burkhard, and Benjamin Schooley. SafeRoadMaps: Communication of location and density of traffic fatalities through spatial visualization and heat map analysis. *Information Visualization*, 10(1):82–96, dec 2010. ISSN 1473-8716. doi: 10.1057/ivs.2010.14.

[108] Charlotte Plug, Jianhong (Cecilia) Xia, and Craig Caulfield. Spatial and temporal visualisation techniques for crash analysis. *Accident Analysis and Prevention*, 43(6):1937–1946, 2011. ISSN 00014575. doi: 10.1016/j.aap.2011.05.007.

[109] Afian Anwar, Till Nagel, and Carlo Ratti. Traffic origins: A simple visualization technique to support traffic incident analysis. In *IEEE Pacific Visualization Symposium*, pages 316–319, 2014. ISBN 9781479928736. doi: 10.1109/PacificVis.2014.35.

[110] Danijel Rebolj and Peter J. Sturm. A GIS based component-oriented integrated system for estimation, visualization and analysis of road traffic air pollution. *Environmental Modelling and Software*, 14(6):531–539, 1999. ISSN 13648152. doi: 10.1016/S1364-8152(99)00017-1.

[111] Verina Cristie, Matthias Berger, Peter Bus, Ashwani Kumar, and Bernhard Klein. CityHeat: visualizing cellular automata-based traffic heat in Unity3D. In *SIGGRAPH Asia 2015 Visualization in High Performance Computing*, pages 1–4, New York, New York, USA, 2015. ACM Press. ISBN 9781450339292. doi: 10.1145/2818517.2818527.

[112] Jing-Quan Li, Wei-Bin Zhang, and Liping Zhang. A web-based support system for estimating and visualizing the emissions of diesel transit buses. *Transportation Research Part D: Transport and Environment*, 14(8):533–540, 2009. ISSN 13619209. doi: 10.1016/j.trd.2009.07.009.

[113] Brendan Tran Morris, Cuong Tran, George Scora, Mohan Manubhai Trivedi, and Matthew J. Barth. Real-Time Video-Based Traffic Measurement and Visualization System for Energy/Emissions. *IEEE Transactions on Intelligent Transportation Systems*, 13(4):1667–1678, dec 2012. ISSN 1524-9050. doi: 10.1109/TITS.2012.2208222.

[114] T. Fuse, K. Makimura, and T. Nakamura. Observation of travel behavior by ic card data and application to transportation planning. In *Special Joint Symposium of ISPRS Commission IV*, volume 2010, 2010.

[115] Pascal Roux, Stéphanie Depraz-Depland, Guillaume Bouchard, Frederic Roulland, Luis Ulloa, Pascal Valobra, and Victor Ciriza. Fare collection data analytics and visualization for public transportation. In *19th Intelligent Transport Systems World Congress, ITS 2012*, 2012. URL `http://www.scopus.com/inward/record.url?eid=2-s2.0-84896993498&partnerID=tZOtx3y1`.

[116] Sui Tao, David Rohde, and Jonathan Corcoran. Examining the spatial–temporal dynamics of bus passenger travel behaviour using smart card data and the flow-comap. *Journal of Transport Geography*, 41:21–36, 2014. ISSN 09666923. doi: 10.1016/j.jtrangeo.2014.08.006.

[117] Wei Zeng, Chi Wing Fu, Stefan Muller Arisona, Alexander Erath, and Huamin Qu. Visualizing mobility of public transportation system. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1833–1842, 2014. ISSN 10772626. doi: 10.1109/TVCG.2014.2346893.

[118] Md. Kamruzzaman, Julian Hine, Banihan Gunay, and Neale Blair. Using GIS to visualise and evaluate student travel behaviour. *Journal of Transport Geography*, 19(1):13–32, 2011. ISSN 09666923. doi: 10.1016/j.jtrangeo.2009.09.004.

[119] Dan Yu, Santosh Mishra, and Jie Lin. Visualization of Bus Schedule Adherence Using GIS. In *Applications of Advanced Technology in Transportation*, pages 159–164, Reston, VA, aug 2006. American Society of Civil Engineers. ISBN 978-0-7844-0799-8. doi: 10.1061/40799(213)27.

[120] Graham Currie and Mahmoud Mesbah. Visualization of Geographical Information System and Automatic Vehicle Location Data to Explore Transit Performance. *Transportation Research Record: Journal of the Transportation Research Board*, 2216:59–66, dec 2011. ISSN 0361-1981. doi: 10.3141/2216-07.

[121] Mahmoud Mesbah, Graham Currie, Claudia Lennon, and Trevor Northcott. Spatial and temporal visualization of transit operations performance data at a network level. *Journal of Transport Geography*, 25:15–26, 2012. ISSN 09666923. doi: 10.1016/j.jtrangeo.2012.07.005.

[122] Cesar Palomo, Zhan Guo, Cláudio T. Silva, and Juliana Freire. Visually Exploring Transportation Schedules. *IEEE Transactions on Visualization and Computer Graphics*, 22(1): 170–179, 2016. ISSN 10772626. doi: 10.1109/TVCG.2015.2467592.

[123] Yu Liu, Chaogui Kang, Song Gao, Yu Xiao, and Yuan Tian. Understanding intra-urban trip patterns from taxi trajectory data. *Journal of Geographical Systems*, 14(4):463–483, oct 2012. ISSN 1435-5949. doi: 10.1007/s10109-012-0166-z.

[124] Ding Chu, David A. Sheets, Ye Zhao, Yingyu Wu, Jing Yang, Maogong Zheng, and George Chen. Visualizing hidden themes of taxi movement with semantic transformation. In *IEEE Pacific Visualization Symposium*, pages 137–144, 2014. ISBN 9781479928736. doi: 10.1109/PacificVis.2014.50.

[125] Feng Mao, Minhe Ji, and Ting Liu. Mining spatiotemporal patterns of urban dwellers from taxi trajectory data. *Frontiers of Earth Science*, 10(2):205–221, 2016. ISSN 20950209. doi: 10.1007/s11707-015-0525-4.

[126] Eduardo R. Corral-Soto, Ron Tal, Langyue Wang, Ravi Persad, Luo Chao, Chan Solomon, Bob Hou, Gunho Sohn, and James H. Elder. 3DTown: The automatic urban awareness project. In *9th Conference on Computer and Robot Vision*, pages 433–440, 2012. ISBN 9780769546834. doi: 10.1109/CRV.2012.64.

[127] Zhihan Lv, Xiaoming Li, Baoyun Zhang, Weixi Wang, Yuanyuan Zhu, Jinxing Hu, and Shengzhong Feng. Managing Big City Information Based on WebVRGIS. *IEEE Access*, 4: 407–415, 2016. ISSN 21693536. doi: 10.1109/ACCESS.2016.2517076.

[128] Xiaoming Li, Zhihan Lv, Weixi Wang, Baoyun Zhang, Jinxing Hu, Ling Yin, and Shengzhong Feng. WebVRGIS based traffic analysis and visualization system. *Advances in Engineering Software*, 93:1–8, 2016. ISSN 18735339. doi: 10.1016/j.advengsoft.2015.11.003.

[129] Jameson L. Toole, Serdar Colak, Bradley Sturt, Lauren P. Alexander, Alexandre Evsukoff, and Marta C. González. The path most traveled: Travel demand estimation using big data resources. *Transportation Research Part C: Emerging Technologies*, 58:162–177, 2015. ISSN 0968090X. doi: 10.1016/j.trc.2015.04.022.

[130] Min Lu, Zuchao Wang, Jie Liang, and Xiaoru Yuan. OD-Wheel: Visual design to explore OD patterns of a central region. In *IEEE Pacific Visualization Symposium*, volume 7, pages 87–91, 2015. ISBN 9781467368797. doi: 10.1109/PACIFICVIS.2015.7156361.

[131] Min Lu, Jie Liang, Zuchao Wang, and Xiaoru Yuan. Exploring OD patterns of interested region based on taxi trajectories. *Journal of Visualization*, 19(4):811–821, 2016. ISSN 18758975. doi: 10.1007/s12650-016-0357-7.

[132] Bart Dewulf, Tijs Neutens, Mario Vanlommel, Steven Logghe, Philippe De Maeyer, Frank Witlox, Yves De Weerdt, and Nico Van de Weghe. Examining commuting patterns using Floating Car Data and circular statistics: Exploring the use of new methods and visualizations to study travel times. *Journal of Transport Geography*, 48:41–51, 2015. ISSN 09666923. doi: 10.1016/j.jtrangeo.2015.08.006.

[133] Shi Yin, Moyin Li, Nebiyou Tilahun, Angus Forbes, and Andrew Johnson. Understanding Transportation Accessibility of Metropolitan Chicago Through Interactive Visualization. In *Proceedings of the 1st International ACM SIGSPATIAL Workshop on Smart Cities and Urban Analytics - UrbanGIS'15*, pages 77–84, New York, New York, USA, 2015. ACM Press. ISBN 9781450339735. doi: 10.1145/2835022.2835036.

[134] Anson Stewart and Christopher Zegras. CoAXs: A Collaborative Accessibility-based Stakeholder Engagement System for communicating transport impacts. *Research in Transportation Economics*, 59(C):423–433, 2016. doi: 10.1016/j.retrec.2016.07.016.

[135] Evgheni Polisciuc, Ana Alves, Carlos Bento, and Penousal Machado. Visualizing urban mobility. In *ACM SIGGRAPH 2013*, page 1, New York, New York, USA, 2013. ACM Press. ISBN 9781450323420. doi: 10.1145/2503385.2503511.

[136] Fan Du, Joshua Brulé, Peter Enns, Varun Manjunatha, and Yoav Segev. MetroViz: Visual Analysis of Public Transportation Data. jul 2015. URL `http://arxiv.org/abs/1507.05215`.

[137] Jo Wood, Roger Beecham, and Jason Dykes. Moving beyond sequential design: Reflections on a rich multi-channel approach to data visualization. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2171–2180, 2014. ISSN 10772626. doi: 10.1109/TVCG.2014.2346323.

[138] Gustavo Romanillos and Martin Zaltz Austwick. Madrid cycle track: visualizing the cyclable city. *Journal of Maps*, pages 1–9, oct 2015. ISSN 1744-5647. doi: 10.1080/17445647.2015.1088901.

[139] Zuchao Wang, Tangzhi Ye, Min Lu, Xiaoru Yuan, Huamin Qu, Jacky Yuan, and Qianliang Wu. Visual exploration of sparse traffic trajectory data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1813–1822, 2014. ISSN 10772626. doi: 10.1109/TVCG.2014.2346746.

[140] Michael VanDaniker. Leverage of Spiral Graph for Transportation System Data Visualization. *Transportation Research Record: Journal of the Transportation Research Board*, 2165: 79–88, dec 2010. ISSN 0361-1981. doi: 10.3141/2165-09.

[141] Jiayu Wu, Zhiyong Fu, Zhiyuan Liu, Jiajia Pan, Huiling Long, Xu Lin, Haoqing He, Xinxiong Chen, and Jiayu Tang. City flow: Prototype exploration for visualizing urban traffic conversations. In *IEEE International Conference on Social Computing*, pages 481–489, 2012. ISBN 9780769548487. doi: 10.1109/SocialCom-PASSAT.2012.40.

[142] Wei Zeng, Chi Wing Fu, Stefan Muller Arisona, and Huamin Qu. Visualizing interchange patterns in massive movement data. *Computer Graphics Forum*, 32(3):271–280, 2013. ISSN 01677055. doi: 10.1111/cgf.12114.

[143] Robert Krüger, Dennis Thom, Michael Wörner, Harald Bosch, and Thomas Ertl. TrajectoryLenses - A set-based filtering and exploration technique for long-term trajectory data. *Computer Graphics Forum*, 32(3 PART4):451–460, 2013. ISSN 01677055. doi: 10.1111/cgf.12132.

[144] Evgheni Polisciuc, Ana Alves, and Penousal Machado. Understanding Urban Land Use through the Visualization of Points of Interest. In *Proceedings of the Fourth Workshop on Vision and Language*, pages 51–59, Stroudsburg, PA, USA, 2015. Association for Computational Linguistics. doi: 10.18653/v1/W15-2810.

[145] Wenchao Wu, Jiayi Xu, Haipeng Zeng, Yixian Zheng, Huamin Qu, Bing Ni, Mingxuan Yuan, and Lionel M. Ni. TelCoVis: Visual Exploration of Co-occurrence in Urban Human Mobility Based on Telco Data. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):935–944, jan 2016. ISSN 1077-2626. doi: 10.1109/TVCG.2015.2467194.

[146] Yudong Chen, Yi Zhang, and Jianming Hu. Multi-Dimensional Traffic Flow Time Series Analysis with Self-Organizing Maps. *Tsinghua Science and Technology*, 13(2):220–228, 2008. ISSN 10070214. doi: 10.1016/S1007-0214(08)70036-1.

[147] Nobbir Ahmed and Harvey J. Miller. Time–space transformations of geographic space for exploring, analyzing and visualizing transportation systems. *Journal of Transport Geography*, 15(1):2–17, 2007. ISSN 09666923. doi: 10.1016/j.jtrangeo.2005.11.004.

[148] Megan Katsumi and Mark Fox. Ontologies for transportation research: A survey. *Transportation Research Part C: Emerging Technologies*, 89:53–82, apr 2018. ISSN 0968-090X. doi: 10.1016/J.TRC.2018.01.023.

[149] Emanuele Della Valle, Irene Celino, and Daniele Dell'Aglio. The Experience of Realizing a Semantic Web Urban Computing Application. *Transactions in GIS*, 14(2):163–181, apr 2010. ISSN 13611682. doi: 10.1111/j.1467-9671.2010.01189.x.

[150] Bernhard Lorenz, Hans Jürgen Ohlbach, and Laibing Yang. Ontology of transportation networks. Technical report, Reasoning on the Web with Rules and Semantics, 2005.

[151] David Corsar, Milan Markovic, Peter Edwards, and John D Nelson. The Transport Disruption Ontology. In Marcelo Arenas, Oscar Corcho, Elena Simperl, Markus Strohmaier, Mathieu D'Aquin, Kavitha Srinivas, Paul Groth, Michel Dumontier, Jeff Heflin, Krishnaprasad Thirunarayan, and Steffen Staab, editors, *The Semantic Web - ISWC 2015*, pages 329–336, Cham, 2015. Springer International Publishing. ISBN 978-3-319-25010-6.

[152] Chantal Berdier and Catherine Roussey. Urban Ontologies: the Towntology Prototype towards Case Studies. In Jacques Teller, John R. Lee, and Catherine Roussey, editors, *Ontologies for Urban Development*, volume 61 of *Studies in Computational Intelligence*, pages 143–155. Springer Berlin Heidelberg, 2007. ISBN 978-3-540-71975-5. doi: 10.1007/978-3-540-71976-2_13.

[153] Chuanrong Zhang, Zhong-Ren Peng, Tian Zhao, and Weidong Li. Transformation of Transportation Data Models from Unified Modeling Language to Web Ontology Language. *Transportation Research Record: Journal of the Transportation Research Board*, 2064: 81–89, 2008. doi: 10.3141/2064-11.

[154] W3C. eXtensible Stylesheet Language Transformation (XLST) Specification, 2017. URL `https://www.w3.org/TR/xslt/all/`. Last accessed: April 1, 2017.

[155] Julien Plu and François Scharffe. Publishing and Linking Transport Data on the Web: Extended Version. In *Proceedings of the First International Workshop on Open Data*, WOD '12, pages 62–69, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1404-6. doi: 10.1145/2422604.2422614.

[156] Káthia Marçal Oliveira, Firas Bacha, Houda Mnasser, and Mourad Abed. Transportation Ontology Definition and Application for the Content Personalization of User Interfaces. *Expert Systems with Applications*, 40(8):3145–3159, jun 2013. ISSN 0957-4174. doi: 10.1016/j.eswa.2012.12.028.

[157] Julita Bermejo-Alonso, Jorge Salvador, and Ricardo Sanz. Towards an Ontology for Task and Planning in Autonomous Systems: An Emergency Scenario. In Anibal Ollero, Alberto Sanfeliu, Luis Montano, Nuno Lau, and Carlos Cardeira, editors, *ROBOT 2017: Third Iberian Robotics Conference*, pages 429–440, Cham, 2018. Springer International Publishing. ISBN 978-3-319-70833-1.

[158] Arthur Herzog and Alejandro Buchmann. A3ME generic middleware for information exchange in heterogeneous environments. In *9th International Conference on Networked Sensing (INSS)*, pages 1–4, jun 2012. doi: 10.1109/INSS.2012.6240554.

[159] Alexey Seliverstov and Rosaldo J. F. Rossetti. An ontological approach to spatio-temporal information modelling in transportation. In *IEEE First International Smart Cities Conference (ISC2)*, pages 1–6, 2015. doi: 10.1109/ISC2.2015.7366160.

[160] Filippo Benvenuti, Claudia Diamantini, Domenico Potena, and Emanuele Storti. An ontology-based framework to support performance monitoring in public transport systems. *Transportation Research Part C: Emerging Technologies*, 81:188–208, aug 2017. ISSN 0968-090X. doi: 10.1016/J.TRC.2017.06.001.

[161] European Committee for Standardization. Transmodel (Reference Data Model For Public Transport). Technical report, European Committee for Standardization, 2005.

[162] Wolfgang Aigner, Silvia Miksch, Heidrun Schumann, and Christian Tominski. *Visualization of Time-Oriented Data*. Springer Publishing Company, Incorporated, 1st edition, 2011. ISBN 0857290789, 9780857290786.

[163] Nicholas Chrisman. *Exploring Geographic Information Systems*. Wiley, 1997.

[164] Donna Peuquet. *Representations of Space and Time*. Guilford Press, 2002.

[165] S. S. Stevens. On the Theory of Scales of Measurement. *Science*, 103(2684):677–680, 1946. ISSN 0036-8075. doi: 10.1126/science.103.2684.677.

[166] W3C OWL Working Group. OWL 2 Web Ontology Language Document Overview, 2012. URL https://www.w3.org/TR/owl2-overview/.

[167] Knowledge Based Systems Inc. IDEF: Integrated Definition Methods, 2014. URL http://www.idef.com/idef3.htm. Last accessed: February 25, 2015.

[168] W3C. Time Ontology in OWL, 2017. URL https://www.w3.org/TR/owl-time/. Last accessed: March 27, 2017.

[169] Ontology Engineering Group. OOPS! - Ontology Pitfall Scanner, 2016. URL http://oops.linkeddata.es/. Last accessed: January 2, 2018.

[170] Joana Hora, Teresa Galvão Dias, Ana Camanho, and Thiago Sobral. Estimation of Origin-Destination matrices under Automatic Fare Collection: the case study of Porto transportation system. *Transportation Research Procedia*, 27:664–671, jan 2017. ISSN 2352-1465. doi: 10.1016/J.TRPRO.2017.12.103.

[171] Gabriel Sánchez-Martínez. Inference of Public Transportation Trip Destinations by Using Fare Transaction and Vehicle Location Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2652:1–7, jan 2017. ISSN 0361-1981. doi: 10.3141/2652-01.

[172] Jakob Nielsen. *Usability Engineering*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993. ISBN 0125184050.

Appendix A

VUMO Ontology Reference

This appendix describes the elements of the VUMO ontology. It is intended to be a technical reference.

A.1 Classes

`vumo:VUMOThing`
A superclass for all classes declared in the VUMO ontology.
Subclass of: `owl:Thing`

A.1.1 UrbanMobilityConcept (UMC) and its subclasses

`vumo:UrbanMobilityConcept`
A superclass for urban mobility concepts, e.g., line, bus stop.
Subclass of: `vumo:VUMOThing`

`vumo:Agent`
An entity capable of triggering events.
Subclass of: `vumo:UrbanMobilityConcept`

`vumo:Operator`
A public, private or public-private operator that services a transportation system.
Subclass of: `vumo:Agent`

`vumo:Vehicle`
A private or public vehicle.
Subclass of: `vumo:Agent`

`vumo:Sensor`
A device that provides a measurable readings of something.
Subclass of: `vumo:Agent`

`vumo:EmissionsProfile`
The emissions profile of an entity e.g., vehicle, with respect to one or more pollutants and parameters e.g., speed, engine technology.
Subclass of: `vumo:UrbanMobilityConcept`

`vumo:EngineTechnology`
The engine technology of a vehicle, e.g., bus.
Subclass of: `vumo:UrbanMobilityConcept`

`vumo:Event`
A rating component that regards a negative aspect. When the component is rated quantitatively, higher ratings correspond to a higher negative impact.
Subclass of: `vumo:UrbanMobilityConcept`

`vumo:ODFlow`
A flow of something, e.g. vehicles, between a given origin and destination.
Subclass of: `vumo:Event`

`vumo:Reading`
A reading made by a source, e.g., sensor.
Subclass of: `vumo:Event`

`vumo:SensorReading`
A reading made by a sensor.
Subclass of: `vumo:Reading`

`vumo:TravelEvent`
An event that corresponds to an actual travel made by an entity in the transportation network.
Subclass of: `vumo:Event`

`vumo:TicketValidation`
A public transportation ticket validation detected by a validation checkpoint.
Subclass of: `vumo:TravelEvent`

`vumo:Trip`
A trip made by an entity, e.g. vehicle, person.
Subclass of: `vumo:TravelEvent`

`vumo:BicycleTrip`
A trip made by a bicycle.
Subclass of: `vumo:Trip`

`vumo:TravelIntention`
An event corresponding to an intention of traveling in the transportation network. Such intention may be inferred by route plans or consultations to a traveler information system.
Subclass of: `vumo:Event`

`vumo:NearbyStopsRequest`
A request that returns information about public transportation stops that are within a certain radius of a given location, usually expressed by geographical coordinates.
Subclass of: `vumo:TravelIntention`

`vumo:NextDeparturesRequest`
A request that returns information about the next departures for a stop or station. Typically, this type of request is made available by Traveler Information Systems.
Subclass of: `vumo:TravelIntention`

`vumo:RoutePlanRequest`
A request that returns information about a route plan between an origin and destination. Typically, this type of request is made available by Traveler Information Systems.
Subclass of: `vumo:TravelIntention`

`vumo:UnexpectedEvent`
An event that can (partially) disrupt the normal conditions of a transportation network, e.g. accident or public transportation strike.
Subclass of: `vumo:Event`

`vumo:Accident`
An accident which involves one or more entities.
Subclass of: `vumo:UnexpectedEvent`

`vumo:Strike`
A strike which affects the transportation network in some way, e.g. reduction on the number of buses or trains.
Subclass of: `vumo:UnexpectedEvent`

`vumo:Pollutant`
A substance that pollutes something.
Subclass of: `vumo:UrbanMobilityConcept`

`vumo:InfrastructureComponent`
A superclass for concepts defined in the conceptual data model.
Subclass of: `vumo:VUMOThing`

`vumo:BicycleStation`
A station in which bicycles can be hired and dropped out.
Subclass of: `vumo:InfrastructureComponent`

`vumo:Fare`
A price category applied by a transportation system authority to charge for some service, e.g. toll, ticket.
Subclass of: `vumo:InfrastructureComponent`

`vumo:Line`
A service that is formed by one or more routes.
Subclass of: `vumo:InfrastructureComponent`

`vumo:Route`
A trajectory that is part of a line

`vumo:Node`
A node that is part of a route. Nodes are the fundamental elements of route segments.
Subclass of: `vumo:InfrastructureComponent`

`vumo:Station`
A transportation hub that can be served by many operators and lines. A hub typically contains many stops.
Subclass of: `vumo:Node`

`vumo:Stop`
A stop that is served by one or more lines.
Subclass of: `vumo:Node`

`vumo:RouteSegment`
A segment of a route that consists of an initial and final node.
Subclass of: `vumo:InfrastructureComponent`

`vumo:Ticket`
A ticket or pass that allows passengers to travel in the transportation system.
Subclass of: `vumo:InfrastructureComponent`

`vumo:TicketType`
A category for tickets.
Subclass of: `vumo:InfrastructureComponent`

`vumo:Zone`
A zone of the transportation network.
Subclass of: `vumo:InfrastructureComponent`, `schema:AdministrativeArea`

A.1.2 DataConcept (DC) and its subclasses

`vumo:DataConcept`
A superclass for concepts defined in the conceptual data model.
Subclass of: `vumo:VUMOThing`

`vumo:OutputVariable`
A variable that holds values that are part of the output of a transformation.
Subclass of: `vumo:DataConcept`

`vumo:SpatialDistribution`
The visual pattern induced by a spatial reference type.
Subclass of: `vumo:DataConcept`

`vumo:SpatialDistributionAxiom`
An abstract class that materializes the spatial distribution axioms defined in the conceptual model.
Subclass of: `vumo:DataConcept`

`vumo:SpatialReferenceType`
A type of spatial reference that can be used to characterize an entity.
Subclass of: `vumo:DataConcept`

`vumo:Point`
An 1-dimensional type of spatial reference consisting of a pair of geographic coordinates, i.e. latitude and longitude, according to WGS84.
Subclass of: `vumo:SpatialReferenceType`

`vumo:GenericPoint`
A point that does not have an identification in terms of the property `hasID` or any of its subproperties.
Subclass of: `vumo:Point`

`vumo:KnownPoint`
A point that has at least one identification in terms of the property `hasID`, its equivalent, or any of its subproperties.
Subclass of: `vumo:Point`

`vumo:PointSet`
A set made of two or more points.
Subclass of: `vumo:SpatialReferenceType`

`vumo:OrderedPointSet`
A point set which is well-ordered, i.e. every non-empty subset of P has a least element in this ordering.
Subclass of: `vumo:PointSet`

`vumo:UnorderedPointSet`
A point set which is not well-ordered.
Subclass of: `vumo:PointSet`

`vumo:TemporalReferenceType`
A type of temporal reference that can be used to characterize an entity.
Subclass of: `vumo:DataConcept`

`vumo:Instant`
An instant in time, or an interval with zero duration.
Subclass of: `vumo:TemporalReferenceType`

`vumo:Interval`
A time interval defined by start and finish instants.
Subclass of: `vumo:TemporalReferenceType`

`vumo:Transformation`
A SPARQL query that yields new information from instance data. A number of SPIN rules are encapsulated into VUMO to infer information about transformations (see documentation).
Subclass of: `vumo:DataConcept`, `spin:Templates`

A.1.3 VisualizationConcept (VC) and its subclasses

`vumo:ReferenceFrame`
The type of data required by a visualization technique, as defined by Aigner et al. [162].
Subclass of: `vumo:VisualizationConcept`

`vumo:InputVariable`
A variable that receives values from a result variable of an analytical task (transformation). A visualization technique should then transform, programatically, an input variable into a visual variable or feature.
Subclass of: `vumo:VisualizationConcept`

`vumo:InteractionTask`
An interaction mechanism that enhances the user experience and knowledge extraction capabilities of a visualization technique.
Subclass of: `vumo:VisualizationConcept`

`vumo:SpatialDimensionality`
The dimensionality of space in the canvas of a visualization technique.
Subclass of: `vumo:VisualizationConcept`

`vumo:Technique`
A technique implemented into a visualization tool.
Subclass of: `vumo:VisualizationConcept`

`vumo:TemporalDimensionality`
The representation of the time dimension used by a visualization technique, e.g. linear, cyclic.
Subclass of: `vumo:VisualizationConcept`

A.1.4 DomainUserConcept (DUC) and its subclasses

`vumo:DomainUserConcept`
A superclass for all concepts related to empirical knowledge about domain users.
Subclass of: `vumo:DataConcept`

`vumo:AnalyticalProfile`
The analytical profile of a domain user, e.g., strategic, operational.
Subclass of: `vumo:DomainUserConcept`

`vumo:DomainUser`
The user of a visualization tool
Subclass of: `vumo:DomainUserConcept`

`vumo:Rating`
A feedback given by a domain user with respect to a visualization technique, which may also relate an analytical task (transformation). Ratings are described in terms of rating components.
Subclass of: `vumo:DomainUserConcept`

`vumo:TechniqueRating`
A rating about a visualization technique *per se*, i.e. not related to an analytical task (transformation).
Subclass of: `vumo:Rating`

`vumo:CrossRating`
A rating that relates a visualization technique and an analytical task (transformation).
Subclass of: `vumo:Rating`

`vumo:RatingComponent`
The topic (criterion) of a rating statement, e.g., effectiveness, recommended themes.
Subclass of: `vumo:Rating`

`vumo:PositiveRatingComponent`
A rating component that regards a positive aspect. When the component is rated quantitatively, higher ratings correspond to a higher positive impact.
Subclass of: `vumo:RatingComponent`

`vumo:NegativeRatingComponent`
A rating component that regards a negative aspect. When the component is rated quantitatively, higher ratings correspond to a higher negative impact.
Subclass of: `vumo:RatingComponent`

A.2 Properties

A.2.1 Datatype properties

`vumo:hasDateTime`
A property used to express the date and time of a resource, e.g. event.
Range: `xsd:dateTime` | *Subproperty of:* `time:inXSDDateTime`

`vumo:hasStartDateTIme`
A property used to express the start date and time of a resource, e.g. event.
Subproperty of: `vumo:hasDateTime`

`vumo:hasFinishDateTIme`
A property used to express the finish date and time of a resource, e.g. event.
Subproperty of: `vumo:hasDateTime`

`vumo:hasDescription`
A property used to express a textual description about something, e.g. line.

`vumo:hasID`
A property used to provide an identification which, ideally, should be unique within a given semantic context.
Subproperty of: `rdfs:label`

`vumo:hasFriendlyID`
A property used to provide an identification which is expected to be human-readable for a given purpose.

Subproperty of: `vumo:hasID, rdfs:label`

`vumo:hasInternalID`

A property used to provide an identification which is not necessarily expected to be human-readable for a given purpose.
Subproperty of: `vumo:hasID, rdfs:label`

`vumo:hasLicensePlate`

A property used to indicate the license plate registration of a vehicle.
Subproperty of: `vumo:hasID, rdfs:label` | *Domain:* `vumo:Vehicle`

`vumo:hasMeasure`

A property used to indicate a measure related to a resource, e.g. sensor reading.
Subproperty of: `vumo:hasMeasure`

`vumo:hasAverageSpeed`

A property used to indicate the average speed of a resource, e.g. vehicle.
Subproperty of: `vumo:hasMeasure`

`vumo:hasCount`

A property used to indicate the count of something, e.g. number of cars read by a `vumo:SensorReading`.
Subproperty of: `vumo:hasMeasure`

`vumo:hasDuration`

A property used to indicate the duration of a resource, e.g. event. VUMO is able to infer the duration, provided that the property `vumo:hasStartDateTime` and `vumo:hasFinishDateTime` are asserted for the same resource.
Subproperty of: `vumo:hasMeasure`

`vumo:hasEmissionRate_gmi`

A property used to express the emission rate of a given `vumo:EngineTechnology` in grams per miles (g/mi).
Subproperty of: `vumo:hasMeasure`

`vumo:hasFlow`

A property used to express the emission rate of a given `vumo:EngineTechnology` in grams per miles (g/mi).
Subproperty of: `vumo:hasMeasure`

`vumo:hasNumberOfAvailableBicycles`

A property used to express the number of bicycles available for hire in a `vumo:BicycleStation`.
Subproperty of: `vumo:hasMeasure`

`vumo:hasNumberOfDamagedBicycles`

A property used to express the number of bicycles unavailable for hire in a `vumo:BicycleStation` due to technical problems.
Subproperty of: `vumo:hasMeasure`

`vumo:hasNumberOfFatalities`

A property used to express the number of fatalities due to an event, e.g. `vumo:Accident`.
Subproperty of: `vumo:hasMeasure`

`vumo:hasNumberOfInjuredCyclists`
A property used to express the number of injured cyclists due to an event, e.g. `vumo:Accident`.
Subproperty of: `vumo:hasMeasure`

`vumo:hasNumberOfInjuredDrivers`
A property used to express the number of injured drivers due to an event, e.g. `vumo:Accident`.
Subproperty of: `vumo:hasMeasure`

`vumo:hasNumberOfInjuredPassengers`
A property used to express the number of injured passengers due to an event, e.g. `vumo:Accident`.
Subproperty of: `vumo:hasMeasure`

`vumo:hasRadius`
A property used to indicate the radius of something. For instance, the event class `vumo:Nearby-StopsRequest` may require the specification of a radius corresponding to the desired distance within someone is willing to search for a stop.
Subproperty of: `vumo:hasMeasure`

`vumo:hasName`
A property used to provide a name about something, e.g. `vumo:Operator`.
Subproperty of: `rdfs:label`

`vumo:hasRatingScore`
A property used to express the a quantitative rating about a `vumo:RatingComponent`, within the context of a `vumo:RatingStatement`.

`vumo:isRequired`
A property used to express if a `vumo:InputVariable` is required for the correct display of a `vumo:VisualizationTechnique`.

A.2.2 Object properties

`vumo:assignedBy`
A property used to indicate assignment of something, e.g. a fare zone can be assigned by a transportation authority.
`vumo:hasCompatibleValueType`
A property used to indicate the compatible value type of an `vumo:InputVariable`.
Domain: `vumo:InputVariable`

`vumo:hasInputVariable`
A property used to indicate an input variable of a `vumo:VisualizationTechnique`.

`vumo:hasInteractionTask`
A property used to indicate an `vumo:InteractionTechnique` that a `vumo:Technique` provides.
Domain: `vumo:Technique` | *Range:* `vumo:InteractionTask`

`vumo:hasOperator`
A property used to indicate an `vumo:Operator` of something, e.g. `vumo:TicketValidation`.

`vumo:hasOutputVariable`
A property used to indicate an output route of a `vumo:Transformation`. VUMO's built-in rules allow the automatic inference of the output variables of a transformation, thus using it manually requires extra care.

`vumo:hasPath`
A property used to express the path of something, e.g. `vumo:RoutePlan`, `vumo:Route`.

`vumo:hasRatingCategoricalValue`
A property used to express a categorical value for a rating.

`vumo:hasRatingComponent`
A property used to indicate what a rating is about.

`vumo:hasRatingStatement`
Definition
Domain: `vumo:Rating` | *Range:* `vumo:RatingStatement`

`vumo:hasReferenceFrame`
A property used to indicate the `vumo:ReferenceFrame` of a `vumo:Technique`.
Domain: `vumo:Technique` | *Range:* `vumo:ReferenceFrame`

`vumo:hasRoute`
A property used to indicate a route of a `vumo:Line`.
Domain: `vumo:Line` | *Range:* `vumo:Route`

`vumo:hasInboundRoute`
A property used to indicate an inbound route of a `vumo:Line`.
Subproperty of: `vumo:hasRoute` | *Domain:* `vumo:Line` | *Range:* `vumo:Route`

`vumo:hasOutboundRoute`
A property used to indicate an outbound route of a `vumo:Line`.
Subproperty of: `vumo:hasRoute` | *Domain:* `vumo:Line` | *Range:* `vumo:Route`

`vumo:hasSensorReading`
A property used to indicate a `vumo:SensorReading` made by a device, e.g. `vumo:Sensor`
Range: `vumo:SensorReading`

`vumo:hasSpatialDimensionality`
A property used to indicate the `vumo:SpatialDimensionality` of a `vumo:Technique`
Domain: `vumo:Technique` | *Range:* `vumo:SpatialDimensionality`

`vumo:hasSpatialDistribution`
A property used to indicate the `vumo:SpatialDistribution` of a `vumo:SpatialReferenceType`. This property is used by hard-coded `vumo:SpatialDistributionAxiom`.

`vumo:hasSpatialResourceReference`
A superproperty used to group all properties that provide a spatial reference to an entity, e.g. `vumo:Event`.

`vumo:hasDestination`
A property used to indicate the destination of something, e.g. `vumo:RoutePlan`.
Subproperty of: `vumo:hasSpatialResourceReference`

`vumo:hasFinalNode`
A property used to indicate the final `vumo:Node`, e.g. `vumo:RouteSegment`.
Subproperty of: `vumo:hasSpatialResourceReference`

`vumo:hasInitialNode`
A property used to indicate the initial `vumo:Node`, e.g. `vumo:RouteSegment`.
Subproperty of: `vumo:hasSpatialResourceReference`

`vumo:hasOrigin`
A property used to indicate the origin of something, e.g. `vumo:RoutePlan`.
Subproperty of: `vumo:hasSpatialResourceReference`

`vumo:hasZoneBoundaryPoint`
A property used to indicate each `vumo:Point` that forms the boundary of a `vumo:Zone`
Subproperty of: `vumo:hasSpatialResourceReference` | *Domain:* `vumo:Zone`

`vumo:isValidInsideZone`
A property used to indicate that something, e.g. `vumo:Ticket` is valid inside a `vumo:Zone`.
Subproperty of: `vumo:hasSpatialResourceReference` | *Range:* `vumo:Zone`

`vumo:locatedInZone`
A property used to indicate that something, e.g. `vumo:Node` is located in a `vumo:Zone`.
Subproperty of: `vumo:hasSpatialResourceReference` | *Range:* `vumo:Zone`
Inverse of: `hasZoneElement`

`vumo:mapsOntoInputVariable`
A property used to map an output variable of a `vumo:Transformation` onto an input variable of a `vumo:Technique`.

`vumo:occursAtCoordinates`
A property used to indicate that something, e.g. `vumo:Event` occurs at a given set of coordinates, usually expressed as a blank node.
Subproperty of: `vumo:hasSpatialResourceReference`

`vumo:occursAtNode`
A property used to indicate that something, e.g. `vumo:Event` occurs at a given node.
Subproperty of: `vumo:hasSpatialResourceReference`

`vumo:hasTemporalResourceReference`
A property used to indicate the temporal reference of something, e.g. `vumo:Event`, by using a resource instead of a literal. This property is useful for encoding well-defined intervals of time.

`vumo:hasTicketType`
A property used to indicate the `vumo:TicketType` of a `vumo:Ticket`.
Domain: `vumo:Ticket` | *Range:* `vumo:TicketType`

`vumo:hasUserProfile`
A property used to indicate the `vumo:DomainUserProfile` of a `vumo:DomainUser`.
Domain: `vumo:DomainUser` | *Range:* `vumo:DomainUserProfile`

`vumo:isAboutTransformation`
A property used to indicate the `vumo:Transformation` evaluated by a `vumo:CrossRating`.

`vumo:isAboutVisualizationTechnique`
A property used to indicate the `vumo:Technique` evaluated by a `vumo:CrossRating`.

`vumo:isCompatibleWith`
A property used to indicate that a `vumo:Transformation` is compatible with a `vumo:Technique`, and vice-versa.

`vumo:isNotCompatibleWith`
A property used to indicate that a `vumo:Transformation` is not compatible with a `vumo:Technique`, and vice-versa.

`vumo:isRelatedToLine`
A property used to indicate that a resource, e.g. `vumo:Reading` is related to a `vumo:Line`.

`vumo:isRelatedToPollutant`
A property used to indicate that an `vumo:EmissionsProfile` is related to a `vumo:Pollutant`.

`vumo:isRelatedToRoute`
A property used to indicate that a resource, e.g. `vumo:Reading` is related to a `vumo:Route`.

`vumo:isRelatedToRouteSegment`
A property used to indicate that a resource, e.g. `vumo:Reading` is related to a `vumo:RouteSegment`.

`vumo:madeBySensor`
A property used to indicate that a `vumo:Reading` was made by a `vumo:Sensor`.
Domain: `vumo:Reading` | *Range:* `vumo:Sensor`

`vumo:madeByDomainUser`
A property used to indicate that a `vumo:Rating` was made by a `vumo:DomainUser`.

`vumo:madeByVehicle`
A property used to indicate that something, e.g. `vumo:Trip` was made by a `vumo:Vehicle`.
Range: `vumo:Vehicle`

`vumo:occursInLine`
A property used to indicate that an `vumo:Event` occurs in a `vumo:Line`.

`vumo:occursInRoute`
A property used to indicate that an `vumo:Event` occurs in a `vumo:Route`.

`vumo:occursInTrip`
A property used to indicate that an `vumo:Event` occurs in a `vumo:Trip`.

`vumo:occursInVehicle`
A property used to indicate that an `vumo:Event` occurs in a `vumo:Vehicle`.

`vumo:operatedBy`
A property used to indicate that a resource, e.g. `vumo:Node` is operated by another resource, e.g. `vumo:Operator`
Inverse of: `vumo:operates`

`vumo:operates`
A property used to indicate that a resource, e.g. `vumo:Operator` operates another resource, e.g. `vumo:Node`.
Inverse of: `vumo:operatedBy`

`vumo:ownedBy`
A property used to indicate that a resource, e.g. `vumo:Vehicle` is owned by another resource, e.g. `vumo:Operator`.
Inverse of: `vumo:owns`

`vumo:owns`
A property used to indicate that a resource, e.g. `vumo:Operator` owns another resource, e.g. `vumo:Vehicle`.
Inverse of: `vumo:ownedBy`

`vumo:usesTicket`
A property used to indicate that a resource, e.g. `vumo:TicketValidation` uses a `vumo:Ticket`.

A.3 Instances

A.3.1 DomainUserProfile

`vumo:Operational`

`vumo:Strategic`

A.3.2 InteractionTask

`vumo:Filtering`

`vumo:Overview`

`vumo:DetailsOnDemand`

`vumo:DynamicQueries`

`vumo:SemanticZoom`

A.3.3 ReferenceFrame

`vumo:Geographic`

`vumo:Abstract`

A.3.4 SpatialDimensionality

`vumo:2D`

`vumo:3D`

A.3.5 TemporalArrangement

`vumo:Linear`

`vumo:Cyclic`

A.3.6 TemporalDimensionality

`vumo:Static`

`vumo:Dynamic`