Green mobility data models and services for smart ecosystems

D2.3 A core vocabulary for shared mobility

Document Identification	
Contractual Delivery Date	31/08/2022
Actual Delivery Date	31/08//2022
Responsible Beneficiary	AIV
Contributing Beneficiaries	AIV, IMEC
Dissemination Level	PU
Version	1.0
Total Number of Pages:	60

Keywords
standardisation, co-creation, OSLO, application profile, vocabulary, mobility

This document is issued within the frame and for the purpose of the GreenMov project. This project has received funding from the European Union's Innovation and Networks Executive Agency – Connecting Europe Facility (CEF) under Grant AGREEMENT No INEA/CEF/ICT/A2020/2373380 Action No: 2020-EU-IA-0281. The opinions expressed and arguments employed herein do not necessarily reflect the official views of the European Commission.

(*) Dissemination level.-PU: Public, fully open, e.g. web; CO: Confidential, restricted under conditions set out in Model Grant Agreement; CI: Classified, Int = Internal Working Document, information as referred to in Commission Decision 2001/844/EC.

Document Information

Related Activity	Activity 2	Document Reference	D2.3
Related Deliverable(s)	D2.1, D2.2	Dissemination Level (*)	PU

List of Contributors	
Name	**Partner**
Thijs Geert	AIV
Scheldeman Arne	AIV
Driesen Alexis	AIV
Annelies De Craene	AIV
Callens Tom	AIV
Van de Vyvere Brecht	IMEC
Rojas Melendez Julian	IMEC
Filip Gosselé	IMEC

Document History			
Version	**Date**	**Change editors**	**Changes**
0.1	25/05/2022	Callens Tom (AIV)	Table of Contents
0.2	7/06/2022	Van de Vyvere Brecht (IMEC)	Improvements to Table of Contents
0.3	28/06/2022	Callens Tom (AIV)	Chapter 2 added
0.4	14/07/2022	Callens Tom, Arne Scheldeman (AIV)	Improvements to chapter 2 List of abbreviations added (chapter 2)

Document History			
Version	**Date**	**Change editors**	**Changes**
0.5	25/07/2022	Brecht Van de Vyvere (IMEC)	Chapters 3 and 4 added
0.6	27/07/2022	Filip Gosselé (IMEC)	Introduction, summary and conclusion added.
0.7	29/08/2022	Alberto Abella (FIWARE)	Review
0.8	30/08/2022	Tom Callens (AIV), Brecht Van de Vyvere (IMEC)	Revised executive summary, chapters, conclusion and formatting.
0.9	31/09/2022	María Guadalupe Rodriguez (ATOS)	Fix format and add information
1.0	31/08/2022	Clara Pezuela (ATOS)	FINAL VERSION TO BE SUBMITTED

Quality Control		
Role	**Who (Partner short name)**	**Approval Date**
Reviewer	Alberto Abella (FIWARE)	31/08/2022
Quality manager	Maria Guadalupe Rodriguez (ATOS)	31/08/2022
Project Coordinator	Clara Pezuela (ATOS)	31/08/2022

Table of Contents

List of Tables

List of Figures

List of Acronyms

Abbreviation / acronym	Description
AP	Application Profile
D2.1	Deliverable number 1 belonging to Activity 2 of the GreenMov project
D2.2	Deliverable number 2 belonging to Activity 2 of the GreenMov project
ETSI	European Telecommunications Standards Institute
INSPIRE	Infrastructure for Spatial Information in the European Community
INSPIRE	The directive aimed to create a European Union spatial data infrastructure in support of EU environmental policies and related policies or activities
ISA	Interoperability Solutions for European Public Administrations
JSON-LD	JavaScript Object Notation for Linked Data is a method of encoding linked data using JSON.
MobiVoc	Open Mobility Vocabulary
NeTEx	Network Timetable Exchange, a CEN Technical Standard for exchanging Public Transport schedules and related data.
NGSI	Next Generation Service Interfaces, Open licensed-API specification by ETSI
NGSI-LD	Next Generation Service Interfaces – Linked Data. Last version of NGSI protocol enabling the use of linked data principles.
ODALA	Collaborative, Secure, and Replicable Open Source Data Lakes for Smart Cities
OSLO	Open Standards for Linked Organisations
RDF	Resource Description Framework. A general method for description and exchange of graph data.
RDFS	Resource Description Framework Schema
SDM	Smart Data Models Program
URI	Uniform Resource Identifier is a unique sequence of characters that identifies a logical or physical resource used by web technologies
UML	Unified Modelling Language

Executive Summary

- **Motivation for the reader :**
  - In this document, you will discover how data models are created in a collaborative way and how different types of data models can co-exist and be mapped to provide precise semantics to Linked Data.
- **Main results and findings** :
  - The OSLO Passenger Transport Hubs, a new vocabulary, application profile and tooling in support of shared mobility use cases, were created according to the OSLO method. OSLO is a Flemish initiative to collaboratively describe semantics for linked data and builds upon existing (inter)national standards. The new OSLO data models are also shared via the perl.eu repository.
  - A passenger transport hub is a place with a diverse range of transport options (e.g. trains, trams, busses, taxis, shared cars, shared (e-)bikes, e-steps). These modes of transport are coordinated and supplemented with additional services (e.g. sandwich bars, coffee shops, and bike maintenance gear). The aim is to facilitate multimodality.
  - In the FIWARE NGSI-LD ecosystem, smart data models were created that are interoperable with OSLO.
  - By mapping OSLO and the FIWARE Smart Data Model, we have proven that both can co-exist and collaborate and contribute to the Linked Data strategy.
- **Short conclusions :**
  - The ultimate goal is to support Green mobility with open linked data. Interoperability has been proven by creating data models in OSLO and FIWARE and mapping them.

1 Introduction

In this deliverable we demonstrate that different types of semantic data models can co-exist. In particular, we show that interoperability between OSLO and NGSI-LD models is possible.

OSLO (Open Standards for Linked Organisations) is a Flemish initiative to collaboratively describe linked data semantics and build upon existing (inter)national standards. NGSI-LD (Next Generation Service Interfaces – Linked Data) is a method for defining linked data semantics, building upon the NGSI-protocol (Next Generation Service Interfaces). This method is standardised by ETSI (European Telecommunications Standards Institute). Data models using NGSI-LD is called a Smart Data Model (SDM). Both OSLO and NGSI-LD models are based on the Resource Description Framework (RDF). However, they use different RDF (Resource Description Framework) vocabularies to define relationships between classes.

To demonstrate the interoperability we first developed an OSLO vocabulary to model Passenger Transport Hubs. Secondly we created a application profile to support the mobility uses cases in GreenMov that use bike sharing data and train schedule information. Thirdly two new SDM's were created together with the mapping from the OSLO Passenger Transport Hubs Application Profile. The results of the mapping will be digested by a GreenMov service that forecasts the availability of shared bikes at a given bike-sharing station at a specific time.

1.1 Purpose of the document

This document describes the results of developing an OSLO core vocabulary for shared mobility, as indented in activity 2, task 2.3 of the Grant Agreement. Also, the original Unified Model Language (UML) and Resource Description Framework (RDF) equivalent sources of OSLO data models are translated into the NGSI standards to allow their implementation across the different pilots.

The outcome is delivered by month 12, as foreseen in the project roadmap.

1.2 Relation to other project work

The method of developing an OSLO vocabulary and application profile, which is used in chapter 2, is described in deliverable D2.1 Extended Smart Data Models v1.0 [1] and deliverable D2.2 Extended Smart Data Models v2.0 (which it is the update of D2.1.).

The OSLO Passenger Transport Hubs model is supporting the implementation of the green mobility use cases defined in deliverable D5.1 Requirements for the data sets and mobility services [2]

1.3 Structure of the document

This document is structured in 3 major chapters:

Chapter 2 presents the OSLO co-creation process and results of defining a common mobility vocabulary and application profile for the data exchange regarding places with a diverse range of transport options

Chapter 3 presents the creation of two Smart Data Models that are aligned with the OSLO Passenger Transport Hubs Application Profile.

Chapter 4 presents the building block for storing event data.

2 Defining the OSLO Passenger Transport Hubs application profile

2.1 Introduction

We developed a semantic data model for Passenger Transport Hubs to support the mobility use cases, aiming to boost the use of shared bicycles and public transportation by facilitating the data sharing between mobility service providers.

The OSLO Passenger Transport Hubs vocabulary and application profile was developed in co-creation with different stakeholders. During a business workshop in April 2022, the business requirements were gathered. During thematic workshops in May and June, the OSLO core vocabulary and application profile were defined. As a starting point, a semantic data model of passenger transport hubs, which is being developed in Flanders (in Dutch), was used. This OSLO Hoppin Points model was translated into English and extended according to the business needs of the stakeholders. The extensions will be translated into Dutch to enrich the OSLO Hoppin Points model, keeping both models in sync.

2.1.1 Open Standards for Linked Organisations (OSLO)

OSLO is an interoperability program in the region of Flanders, which brings together expertise from different business domains and governmental levels, independent of specific thematic use cases. The Flemish Government developed several domain models in line with international standards, including Interoperable Europe (ISA²) and INSPIRE, enriched by data extensions to comply with the local (European) context. The formal specification is published at data.vlaanderen.be [3] or purl.eu [4]. The thematic working groups, with over 500 authors from the public sector, private sector and academia, demonstrated that it is possible to raise the interoperability and foster the harmonisation of data coming from different use cases.

The OSLO method is described in deliverables [1] and D2.1 Extended Smart Data Models v2 (to be submitted by 31/8/2022).

2.1.2 Passenger transport hub

A passenger transport hub is a place with a diverse range of transport options. Depending on the location, you will find trains, trams, busses, taxis, shared cars, shared (e-)bikes, e-steps, a Park & Ride zone, bicycle parking spaces, etc. These modes of transport are coordinated and supplemented with additional services (e.g. sandwich bars, coffee shops, and bike maintenance gear). The aim is to facilitate multimodality.

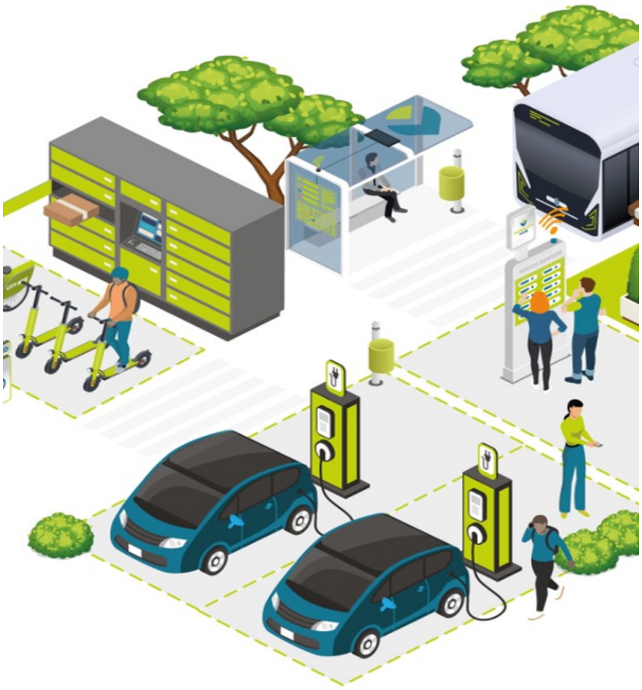


Figure 1. Outline of a passenger transport hub

2.1.3 OSLO Hoppin Points

In Flanders, places of modality cross-over are being branded as "Hoppin Points" To support the Hoppin Points, a new OSLO data model, Hoppin Points, is being developed (in Dutch) and is near completion (currently under public review).

The following (inter)national standards were taken into account:

- Inspire (transport networks) [6]
- Mobivoc [7]
- NeTEx (Stop Places) [8]
- OSLO Mobiliteit: Trips en aanbod [9] (OSLO Mobility: Trips and offers)
- OSLO Mobiliteit: Planning en dienstregeling [10] (OSLO Mobility: Planning and schedule)
- OSLO Openbaar domein [11] (OSLO Public domain)

The OSLO Hoppin Points [12] core vocabulary and application profile are published through the OSLO Standard register.

The OSLO Hoppin Points model will be used as a basis for the OSLO Passenger Transport Hubs track. The model will be translated in English and be extended to the European requirements. The extensions will be translated in Dutch to enrich the OSLO Hoppin Points model.

2.2 Co-creation process

Starting from existing international standards, vocabularies and datasets in the European Data Portal, the semantic agreements are reached in open thematic working groups, consisting of domain experts from the public sector, private sector and academia. These working groups follow the process and method within a formal governance framework (see deliverables D2.1 and D2.2).

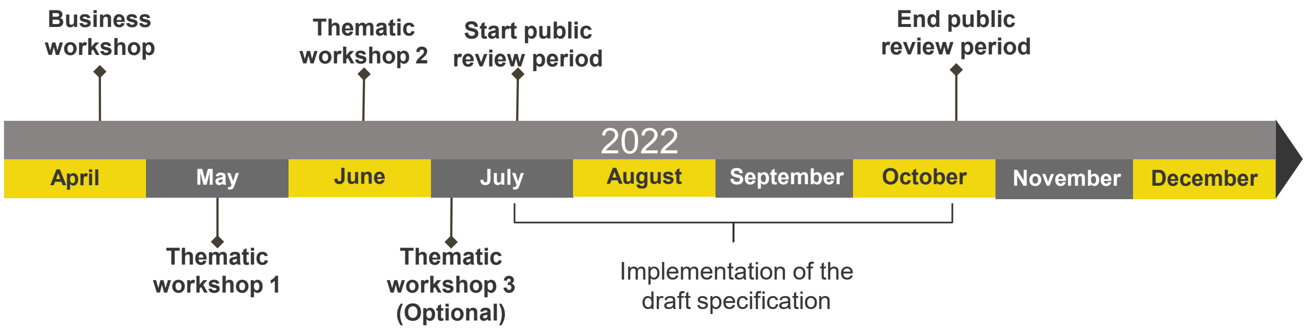


Figure 2. Roadmap for the creation of the OSLO Passenger Transport Hub

2.2.1 Stakeholders

The potential stakeholders of this process include:

Table 1. Potential stakeholders for creation and use of OSLO Passenger Transport Hub

Stakeholder type	Stakeholders name
Mobility providers	Dott, Bolt, Poppy, Velo, Villo!, Donkey Republic, Mobit, Cambio, Blue-Bike Company, Renault Mobility, Vélo Bleu, …
Mobility-as-a-Service providers	Olympus Mobility, Skipr, Whim, Optimile, …
Local authorities	Nice, Murcia, Molina de Segura, Ghent, …
Public Transport Companies	SNCF, RENFE, NMBS, DeLijn, TEC, MIVB/STIB, …
Knowledge institutions	Imec, VITO, UAntwerpen, Mobipunt vzw, …
Umbrella organisations	International Association of Public Transport, MaaS Alliance, ITS.be, Polis, …
GreenMov Partners	ATOS, FIWARE, IMREDD, IMEC, HOPU, MT3

Most of them were invited. The ones that participated are listed in the reports of the workshops [13].

2.2.2 Business workshop

The business workshop was held online on 26/04/2022.

After introducing the OSLO method, the GreenMov project, the context of passenger transport hubs and the OSLO Hoppin Points model, we held a brainstorming session to identify missing elements [14].

A primary discussion point was the accessibility part of the model. These were mainly misunderstandings of the model. For example, there was a remark that the model would only have the option of accessible or not, and 'grey' areas would be forgotten. Though, the model is able to divide accessibility into three parts:

1. Level of accessibility: e.g. Accessible with support, fully accessible
2. Component of accessibility: The component of the passenger hub to which the accessibility applies (e.g. infrastructure and mobility service)
3. Type of accessibility: e.g. information accessible, service accessible and physically accessible

There was also a requirement regarding the availability of train assistance for people with disabilities and reservations. However, reservations are not included in this model. Reservations are tackled within other standards, such as OSLO Mobility: Trips and offers. The OSLO: Passenger Transport Hub model, however, describes the transport hub in all its detail and the services you could use and its infrastructure.

There were several discussion points concerning the real-time capacity of the model to predict the availability of services. For example, if there are roadblocks or traffic jams, which result in a later arrival or arrival at another location. Or, if you want to know if bikes are available at a certain transport hub. Will the model be capable of handling these real-time adjustments?

2.2.3 Thematic workshops

Two thematic workshops were held (on 19/05/2022 [15] and 23/06/2022 [16]) to review the initial model's translation and discuss extensions based on new use cases introduced during the business workshop.

Accessibility

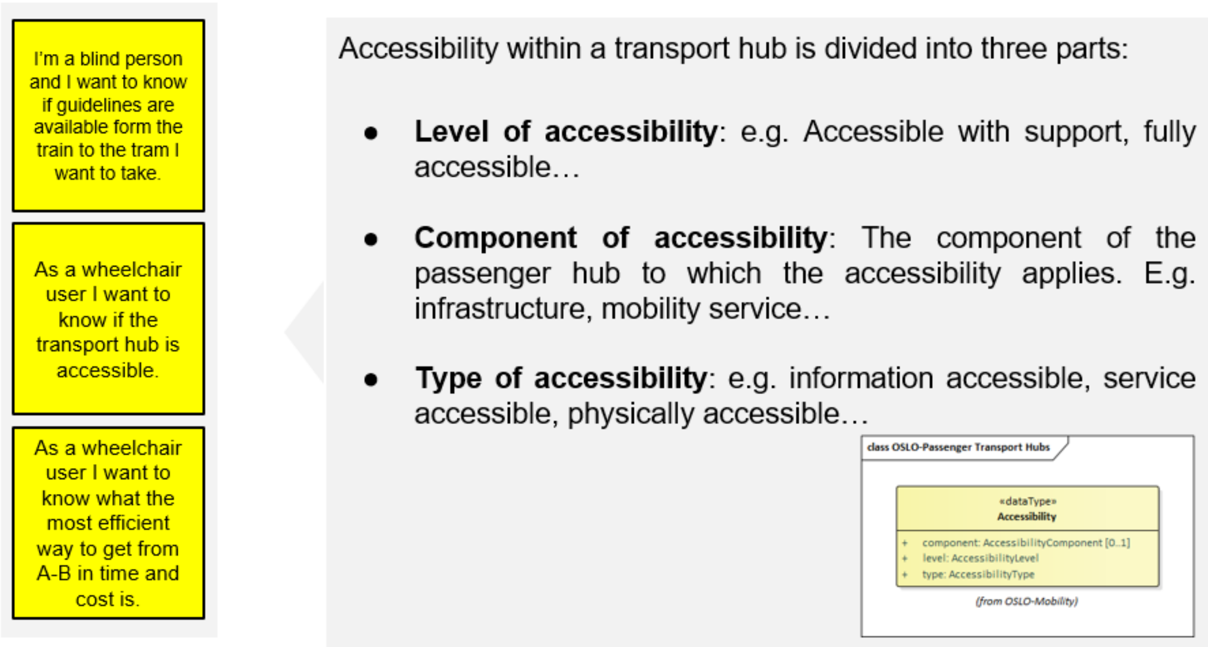


Figure 3. Accessibility requirements

To meet the many questions about accessibility (see Business workshop), it was explained that accessibility attributes are foreseen for all fundamental entities: 1) InfrastructureElements, 2) TransportNetworks, 3) MobilityServices, and 4) AdditionalServices.

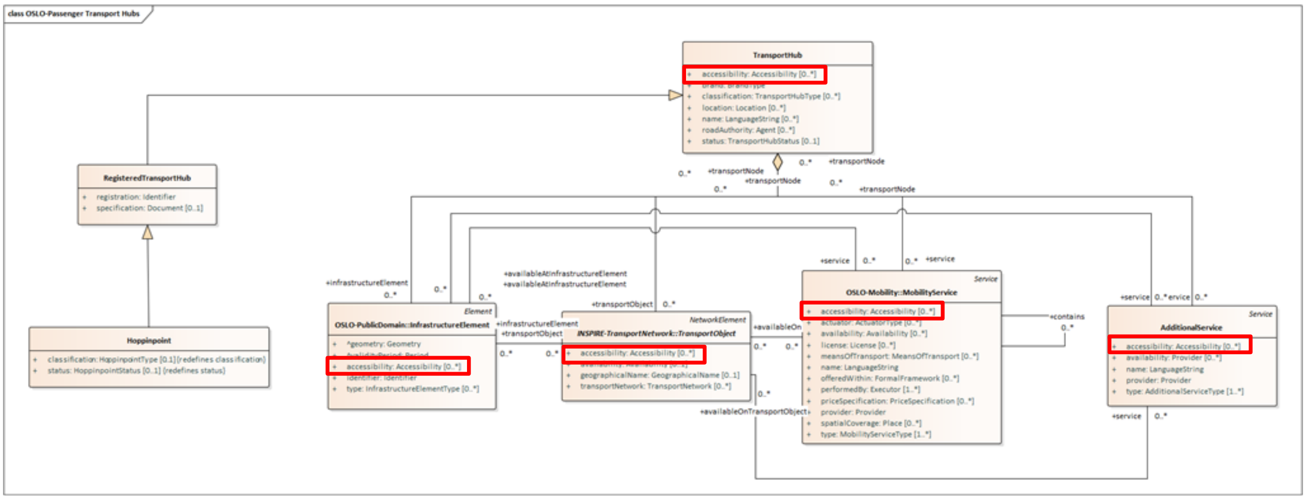


Figure 4. Accessibility in the fundamental entities

Real-time Capacity

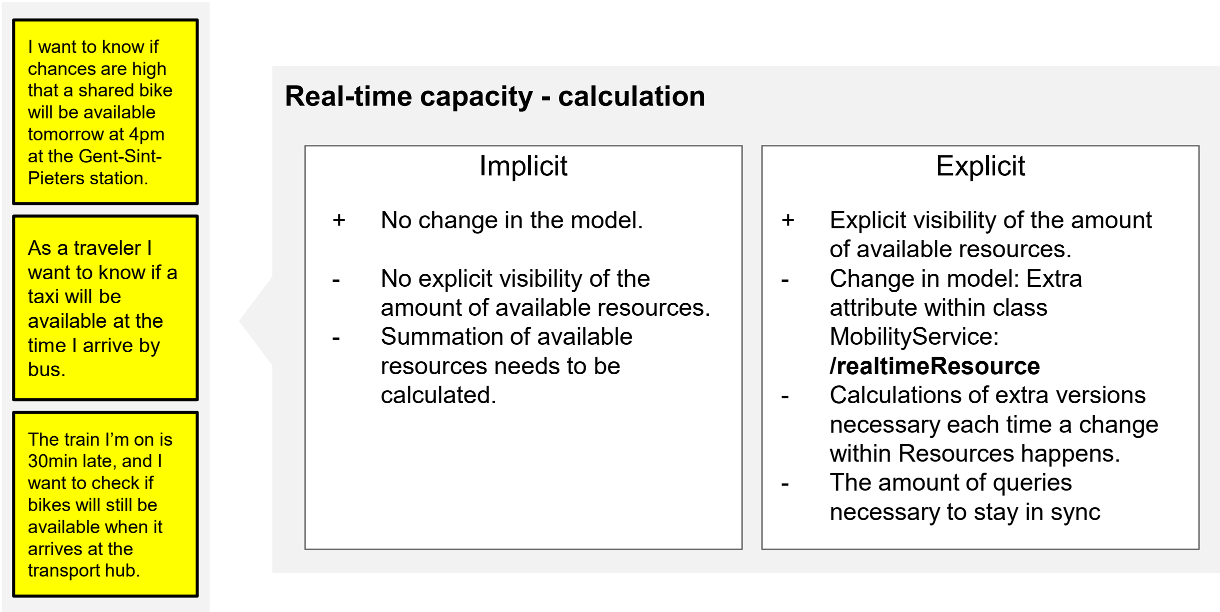


Figure 5. Real-time capacity requirements

One additional class has been added to the model, called ResourceReport. We added this class on demand because this positively affects the implementation of the model.

ResourceReport aims to generate a summary number based on different criteria. The decision to add this class is made based on the 'size' of possible use cases in real life. For example, if it is needed to generate a number for the number of rental bikes at a train station of a capital city or the number of available parking spots in a big city, this would otherwise generate a vast array of different resources. With the class ResourceReport, one number will satisfy the needs.

The cardinality of all the attributes, except the summary number, is zero to many. This cardinality is chosen because different filters/criteria could be added to generate a report. For example, the amount of available bikes with different actuator types (these can be electrical bikes, normal bikes, etc) at all possible locations near the train station.

The class consists of seven attributes. Firstly, actuator embraces the different propulsion types available like electrical, hybrid, non-electrical. The location is very broad. This can be any location where there is a service available, for example the station of Leuven. The means of transport consists of the different modes of transport that you can use like public transport, bicycles, steps, etc. The number returns the amount of resources available. For example, there are five bicycles available at the bike station at station Brussels-Central. The service represents the mobility service. The status, in general, is about whether a resource is available or unavailable. Lastly, the type returns the type of resource you are looking for. This can be, for example, a seating place, bicycles, bicycle pumps, etc.

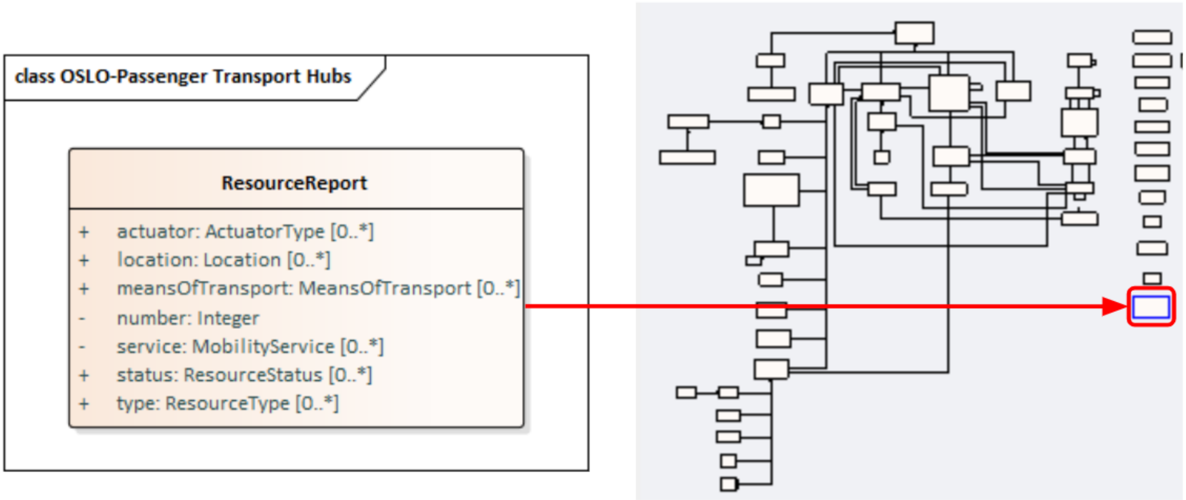


Figure 6. ResourceReport class with attributes

2.2.4 Publication of the draft standard

The model, together with the application profile and the vocabulary, was published on PURL.eu. This can be found by going to PURL.eu and selecting 'Passenger Transport Hubs' in either 'Application Profile' or 'Vocabulary'. Here the model and all classes with their attributes can be consulted at any time. For each class and attribute, you will find a definition and sometimes also a usage note. If you click on the object, you will be directed via a URI to the vocabulary, where this specific object is explained.

The publication of the OSLO-PassengerTransportHubs models can be found through following links:

- **Vocabulary**: https://purl.eu/ns/mobility/passenger-transport-hubs/
- **Application profile**: https://purl.eu/doc/applicationprofile/mobility/passenger-transport-hubs/

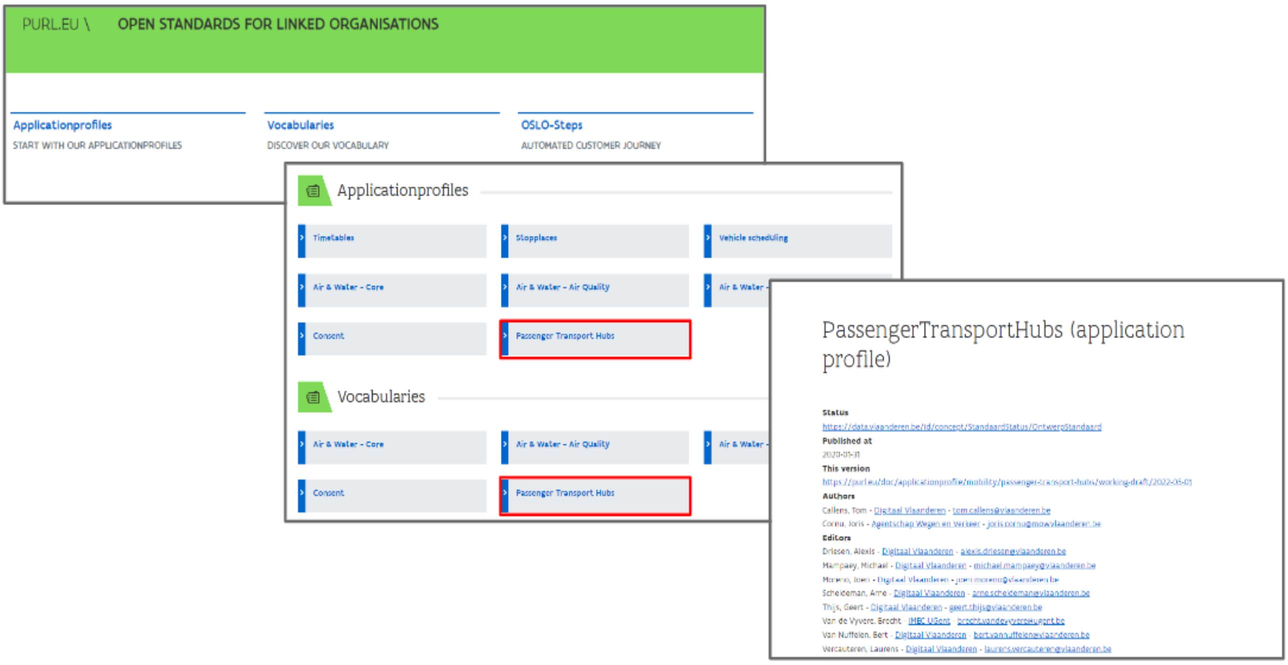


Figure 7. The published English OSLO vocabularia and application profiles

2.2.5 Public review

The public review started in mid-July and will end in mid-October. During this public review, anyone can raise issues on any aspects of the model (e.g. cardinalities, definitions, links, typo's ...) via GitHub [17].

README.md

OSLO Passenger Transport Hubs

Introduction

The goal of this track is to standardise the data structure for Passenger Transport Hubs in collaboration with all stakeholders. In order to semantically enrich data streams, to make transport hubs more findable and understandable, and to make this data more combinable with other data sources. We are creating the vocabulary based on international standards and the input from our stakeholders.

Reports and presentations

The reports and presentations of this track can be consulted via the map reports and presentations.

In the future the files will be available on purl.eu.

In this repository

In this map resources you can find the objectdiagrams created and explained during the workshops. If there is need for more info or documentation, you may send an email to arne.scheldeman@vlaanderen.be or create an issue via Github (see below).

Issues

Via the tab issues you can create comments, ask questions and give feedback for this specific track.

Figure 8: Publication and issue tracking

The end of the public review of OSLO Passenger Transport Hubs and OSLO Hoppin Points coincides. Any comments retained will be processed for OSLO trajectories. In other words, these two standards will be kept in sync.

2.2.6 Official standardisation

The official standardisation by the Flemish steering committee on ICT and information policy is foreseen at the beginning of 2023.

2.3 Fundamentals of the OSLO Passenger Transport Hubs vocabulary and application profile

The co-creation process resulted in the following candidates for standardisation:

- OSLO Passenger Transport Hubs vocabulary: https://purl.eu/ns/mobility/passenger-transport-hubs/
- OSLO Passenger Transport Hubs application profile: https://purl.eu/doc/applicationprofile/mobility/passenger-transport-hubs/

In this section, we give a description of the fundamentals of the developed vocabulary and application profile. The above links provide a formal and complete description of each class and property.

A vocabulary is a collection of reusable terms, more precisely classes and properties, of which their semantics is fixed by means of a label and a definition.

The application profile OSLO indicates how terms from the related vocabulary are used to exchange data on Transport Hubs for passengers. The full UML diagram is found in the annexe of this document. An overview is published online [18]. The model consists of two parts. The first part provides a general definition of a Transport Hub and describes its related elements, with a specific focus on passenger transport. The second part focuses on the traveller using Transport Hubs for his journey or route. Below we focus on part one. For more info on part two, we refer to the application profile OSLO Mobility: Trips and Offerings [19].

An OSLO Transport Hub consist of 1) InfrastructureElements, 2) TransportNetworks, 3) MobilityServices, and 4) AdditionalServices.

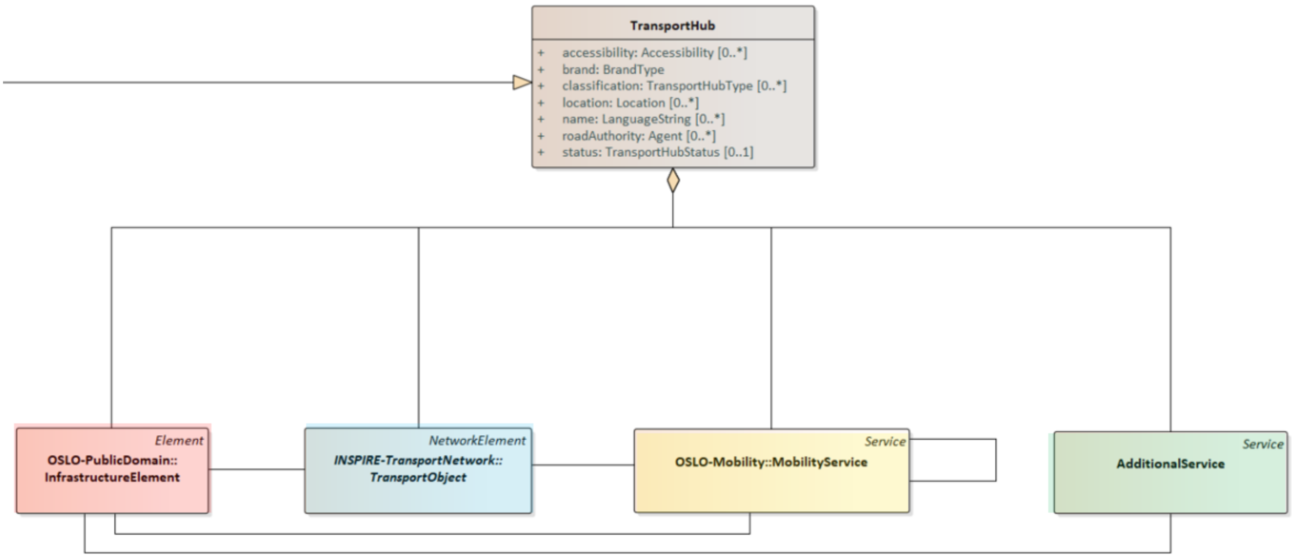


Figure 9. Major modelling parts of a Passenger Transport Hub

The class TransportHub has the following properties: accessibility, brand, classification, infrastructureElement, location, roadAuthority, service, service, status, title, transportObject. The class TransportHub is further specialised to Hoppinppoint, which is the Flemish version of a Transport Hub.

The **InfrastructureElements** represent the physical aspect of a Transport Hub; it concerns objects such as Information columns (like the Hoppin column), Stopplaces, Parkings, Bicycle racks, Chargers etc. which are necessary for the connected Transport networks and the services offered at the Transport Hubs. The InfrastructureElements and their specialisations are described within OSLO Public Domain – InfrastructureElements [20], OSLO Mobility - Timetables and Planning - Stopplaces [21] and out of Open Mobility Vocabulary [7].

The **TransportNetworks** available at a Transport Hub are accessible via TransportObjects such as TransportNodes (stops, stations...) and TransportConnections between those nodes. Also, VirtualNodes (points located on a link between two Transport Nodes, e.g. the position of an abandoned shared step within a city) are possible. The class TransportObject and its subclasses are reused from the vocabulary of OSLO Transport network [22].

Through the available InfrastructureElements and TransportObjects, **MobilityServices** are available at the Transport Hub. These can be scheduled services or shared steps, but just as well as parking spaces or bicycle pumps. A Mobility Service can include other Mobility Services, in the case of a Provider that describes its services at different granularity levels (at a high level, the service "shared mobility in Flanders" and at a lower level, the services "shared cars in Antwerp" and "shared bikes in Ghent"). The MobilityService and related classes come from the vocabulary OSLO-Mobility: Trips and Offerings [23].

At a TransportHub, we also usually find all kinds of **AdditionalServices**, including services that save the Traveler, additional trips such as mailboxes, bakeries, newspaper stores, etc.

Enumerations are basically described outside the application profile; hence the enumeration values shown in the diagram are illustrative.

2.4 Implementation case: shared bike mobility and public transportation

By providing the availability forecasting for shared bikes, we aim, within the European project of GreenMov, to improve the combined use of shared bike mobility and public transportation. We want less waiting time and to avoid the non-availability of shared resources at passenger transport hubs.

To enable such intelligent mobility services, all the different public and private providers should share data regarding the different transport modes in an interoperable way.

The objective is to develop a city service that forecasts the availability of shared bikes at a given bike-sharing station at a specific time. The focus of the use case is a passenger transport hub with a train station and "Blue-Bikes " [24]. Blue-Bikes are one specific bike sharing system provided by the Belgian Blue-Bike company. If the availability forecasting works for Blue-Bikes, it could also work for other bike sharing systems from other providers.

Figure 10. A passenger transport hub at a train station with Blue-Bikes

The requirements for this implementation case were presented in the OSLO business workshop and discussed in the OSLO thematic workshops.

3 Mapping OSLO core vocabularies to NGSI-LD

The OSLO and NGSI-LD models are both based on the Resource Description Framework (RDF). However, they use different RDF vocabularies to define relationships between classes. In this chapter, we discuss how OSLO core vocabularies can be implemented in the NGSI-LD ecosystem by creating NGSI-LD-compliant Smart Data Models (SDM). First, we will briefly discuss how OSLO and SDM differ in the semantic definition of properties. Then, we present two new SDMs for the OSLO Passenger Transport Hubs Application Profile.

3.1 Semantic foundations of OSLO and SDM

The OSLO core vocabularies, which are Application Profiles (AP) and Vocabularies, are aligned with the RDF Schema (RDFS) vocabulary to define classes and properties. RDFS uses the following properties to indicate the domain and range of a property:

- rdfs:domain is a property that states the class of possible subjects of a property
- rdfs:range is a property that states the class of possible objects of a property

This means that if you use this property, a consumer will expect that the domain and range of the property will probably be an instance of the classes mentioned by rdfs:domain and rdfs:range.

While RDFS suffices for knowledge representation in a certain domain, systems (e.g. NGSI-LD Context Brokers) may require the capability to add metadata to relationships. There exist a new extension of RDF (RDF*, pronounced RDF star) that provides this capability and is backwards-compatible. NGSI-LD also proposes an extension (the NGSI-LD metadata model) to solve this, but its normalised representation is not backwards-compatible with existing RDFS domain models. In the normalised NGSI-LD representation, properties (or relationships) can also be instantiated as rdfs:Class in order that metadata can be added. In the KeyValues representation, properties (and relationships) are used similar to RDFS. This implies that a Context Broker can return data semantically aligned with the OSLO core vocabularies using the KeyValues representation. However, a normalised representation of the OSLO models is required to ingest the data into the Context Broker. Also, this normalised representation must be defined per type of class to create a Smart Data Model. In the next section, we describe for two classes inside an OSLO model how these are reflected in SDM.

3.2 Mapping the OSLO Passenger Transport Hubs to NGSI-LD

This section will explore how the OSLO Passenger Transport Hubs Application Profile can be implemented as a SDM. We use the bike availability use case of GreenMov as reference to explain the mapping process. This use case requires two classes of the OSLO model:

- ResourceReport is a class to describe, among others, the number of available shared bikes
- BicycleParkingStation is a class to describe, among others, the location of the infrastructural building used for parking bicycles.

3.2.1 BicycleParkingStationPTH-AP

We created a SDM for the class BicycleParkingStation. Only the properties capacity and geometry that are required for the use case are currently added. In a later stage, other possible properties could be added, such as openingHours, entrance, exit, feature, validityPeriod, accessibility, identifier, type, accessibility, brand, classification, location, name, roadAuthority and status. In table 1, we can see the example normalised NGSI-LD payload where the station with URI "https://blue-bike.be/stations/141" has a total capacity of 20 places for bicycles.

Table 2. Example normalised NGSI-LD for BicycleParkingStation


```
{
  "id": "https://blue-bike.be/stations/141",
  "type": "BicycleParkingStation",
  "ParkingFacility.capacity": {
"type": "Relationship",
"object": {
    "type": "Capacity",
        "Capacity.total": {
            "type": "Property",
            "value": 20
        },
        "Capacity.vehicletype": {
            "type": "Relationship",
            "object": {
                "type": "VehicleType",
                "VehicleType.prefLabel": {
                    "@value": "bicycle",
                    "@language": "en"
                }
            }
        }
}
  },
  "InfrastructureElement.geometry": {
"type": "Relationship",
"object": {
"type": "Geometry",
"Geometry.wkt": {
"type": "Property",
"value": "POINT(3.313743000000 50.855703000000)"
```

```
}
}
 },
 "location": {
     "type": "Point",
     "coordinates": [
       3.313743000000,
       50.855703000000
     ]
 },
 "@context": [
   "https://brechtvdv.github.io/incubated/dataModel.OSLO.PassengerTransportHubs/context.jsonld",
   "https://uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context.jsonld"
 ]
}
```


Also, a JSON schema is created for this data model, which we can see in Table 3.

Table 3. JSON Schema of the BicycleParkingStation model


```
{
   "$schema": "http://json-schema.org/schema#",
   "$schemaVersion": "0.0.1",
   "modelTags": "",
   "$id": "https://github.com/smart-data-models/incubated/dataModel.OSLO.PassengerTransportHubs/raw/master/BicycleParkingStationPTH-AP/schema.json",
   "title": "Bicycle Parking Station PTH-AP schema",
   "description": "Bicycle Parking Station Schema meeting Passenger Transport Hubs AP Schema specification",
   "type": "object",
   "allOf": [
     {
       "$ref": "https://smart-data-models.github.io/data-models/common-schema.json#/definitions/GSMA-Commons"
     },
     {
```

```
"$ref": "https://smart-data-models.github.io/data-models/common-schema.json#/definitions/Location-Commons"
    },
    {
      "properties": {
        "type": {
          "type": "string",
          "enum": [
            "BicycleParkingStation"
          ],
          "description": "Property. NGSI Entity type. It has to be BicycleParkingStation"
        },
        "ParkingFacility.capacity": {
          "type": "object",
          "description": "Property. Model:'http://schema.mobivoc.org/#capacity'. ",
          "properties": {
            "type": {
              "type": "string",
              "enum": [
                "Capacity"
              ]
            },
          "Capacity.total": {
            "type": "number",
            "description": "Property. Model: 'http://schema.mobivoc.org/#totalCapacity'. Indicates the capacity of a resource."
              }
          },
          "required": [
            "type",
            "Capacity.total"
          ]
        },
        "InfrastructureElement.geometry": {
          "type": "object",
```

```
            "description": "Property. Model:'http://www.w3.org/ns/locn#geometry'. The geometry corresponding to the infrastructure element.",
            "properties": {
              "type": {
                "type": "string",
                "description": "Property. Model: 'http://www.w3.org/ns/locn#Geometry'",
                "enum": [
                  "Geometry"
                ]
              },
              "Geometry.wkt": {
                "type": "string",
                "description": "Property. Model:'http://www.opengis.net/ont/geosparql#asWKT'. Geometry expressed in wkt format."
              }
            },
            "required": [
              "type",
              "Geometry.wkt"
            ]
          }
        }
      }
  ],
  "required": [
    "id",
    "type",
  "ParkingFacility.capacity",
    "InfrastructureElement.geometry",
  "location"
  ]
}
```

3.2.2 ResourceReportPTH-AP

In table 4, we can see the example normalised NGSI-LD payload of the class ResourceReport with URI "http://example.org/resourcereport/1". Only 5 (non-electrical) bicycles are available for the above-mentioned station (cfr. table 1). Here, all possible properties (actuator, location, meansOfTransport, number, service, status and type) are incorporated.

Table 4. Example normalised NGSI-LD for ResourceReport


```
{
 "id": "http://example.org/resourcereport/1",
 "type": "ResourceReport",
 "ResourceReport.actuator": {
  "type": "Relationship",
  "object": {
    "type": "ActuatorType",
    "ActuatorType.preferredLabel": {
      "@value": "human",
      "@language": "en"
    }
  }
 },
 "ResourceReport.location": {
  "type": "Relationship",
  "object": "https://blue-bike.be/stations/141"
 },
 "ResourceReport.meansOfTransport": {
  "type": "Relationship",
  "object": {
    "type": "MeansOfTransport",
    "MeansOfTransport.preferredLabel": {
      "@value": "bicycle",
      "@language": "en"
    }
  }
 },
 "ResourceReport.number": {
```

```
  "type": "Property",
  "value": 5
},
"ResourceReport.reportTime": {
  "type": "Property",
  "value": "2022-08-29T12:20:47Z"
},
"ResourceReport.service": {
  "type": "Relationship",
  "object": "https://blue-bike.be/#me"
},
"ResourceReport.status": {
  "type": "Relationship",
  "object": {
    "type": "ResourceStatus",
    "ResourceStatus.preferredLabel": {
      "@value": "available",
      "@language": "en"
    }
  }
},
"ResourceReport.type": {
  "type": "Relationship",
  "object": {
    "type": "ResourceType",
    "ResourceType.preferredLabel": {
      "@value": "vehicle",
      "@language": "en"
    }
  }
},
"location": {
    "type": "Point",
    "coordinates": [
```

```
        3.313743000000,
        50.855703000000
      ]
  },
  "@context": [
    "https://brechtvdv.github.io/incubated/dataModel.OSLO.PassengerTransportHubs/context.jsonld",
"https://uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context.jsonld"
  ]
}
```

Also, a JSON schema is provided for ResourceReport in table 5 below. A benefit of the schema in comparison with the example payload is the explicit listing of possible enumerations.

Table 5. JSON Schema of ResourceReport

```
{
  "$schema": "http://json-schema.org/schema#",
  "$schemaVersion": "0.0.1",
  "modelTags": "",
  "$id": "https://github.com/smart-data-models/incubated/dataModel.OSLO.PassengerTransportHubs/raw/master/BicycleParkingStationPTH-AP/schema.json",
  "title": "Resource Report PTH-AP schema",
  "description": "Resource Report Schema meeting Passenger Transport Hubs AP Schema specification",
  "type": "object",
  "allOf": [
    {
      "$ref": "https://smart-data-models.github.io/data-models/common-schema.json#/definitions/GSMA-Commons"
    },
    {
      "properties": {
        "type": {
          "type": "string",
          "enum": [
            "ResourceReport"
          ],
```

"description": "Property. NGSI Entity type. It has to be ResourceReport. A summary of resources connected to mobility services based on defined filters by the person requesting the report."
},
"ResourceReport.actuator": {
"type": "object",
"description": "Property. Model:'https://purl.eu/ns/mobility/passenger-transport-hubs#actuator'. Engine of the means of transport.",
"properties": {
"type": {
"type": "string",
"enum": [
"ActuatorType"
]
},
"ActuatorType.preferredLabel": {
"type": "string",
"description": "Property. Model: 'http://www.w3.org/2004/02/skos/core#prefLabel'. Preferred label.",
"enum": [
"human",
"electricWithSupport",
"electric",
"combustionEngine"
]
}
},
"required": [
"type",
"ActuatorType.preferredLabel"
]
},
"ResourceReport.location": {
"type": "object",

"description": "Property. Model:'https://purl.eu/ns/mobility/passenger-transport-hubs#location'. Location of the Resource. This could be a bike parking station or the real-time location of the vehicle, e.g. in free-floating part transport."

},

"ResourceReport.meansOfTransport": {

"type": "object",

"description": "Property. Model:'https://purl.eu/ns/mobility/passenger-transport-hubs#Mobiliteitsdienst.vervoermiddel'. The type of means of transport of the Resource.",

"properties": {

"type": {

"type": "string",

"enum": [

"MeansOfTransport"

]

},

"MeansOfTransport.preferredLabel": {

"type": "string",

"description": "Property. Model: 'http://www.w3.org/2004/02/skos/core#prefLabel'. Preferred label.",

"enum": [

"car",

"bicycle",

"onFoot",

"airplane",

"boat",

"train",

"subway",

"tram",

"bus",

"step",

"motorcycle",

"truck",

"lift",

"escalator",

"treadmill",

```
        "pedelec",
        "scooter",
        "skateboard"
      ]
    }
  },
  "required": [
    "type",
    "MeansOfTransport.preferredLabel"
  ]
},
"ResourceReport.number": {
  "type": "integer",
  "description": "Property. Model:'https://purl.eu/ns/mobility/passenger-transport-hubs#number'. The number of resources."
},
"ResourceReport.reportTime": {
  "type": "date-time",
  "description": "Property. Model:'http://purl.org/dc/elements/1.1/date'. Point in time for which the report is valid."
},
"ResourceReport.service": {
  "type": "object",
  "description": "Property. Model:'https://purl.eu/ns/mobility/passenger-transport-hubs#service'. The MobilityService used within the ResourceReport."
},
"ResourceReport.status": {
  "type": "object",
  "description": "Property. Model:'https://purl.eu/ns/mobility/passenger-transport-hubs#status'. State of a Resource. E.g. reserved, inactive, available. Determines whether a resource can be used.",
  "properties": {
    "type": {
      "type": "string",
      "enum": [
        "ResourceStatus"
```

```
          ]
        },
        "ResourceStatus.preferredLabel": {
          "type": "string",
          "description": "Property. Model: 'http://www.w3.org/2004/02/skos/core#prefLabel'. Preferred label.",
          "enum": [
            "inactive",
            "deleted",
            "relocated",
            "available",
            "unavailable",
            "reserved",
            "inUse"
          ]
        }
      },
      "required": [
        "type",
        "ResourceStatus.preferredLabel"
      ]
    },
    "ResourceReport.type": {
      "type": "object",
      "description": "Property. Model:'http://purl.org/dc/terms/type'. Nature of the Resource.",
      "properties": {
        "type": {
          "type": "string",
          "enum": [
            "ResourceType"
          ]
        },
        "ResourceType.preferredLabel": {
          "type": "string",
```

```
        "description": "Property. Model: 'http://www.w3.org/2004/02/skos/core#prefLabel'. Preferred label.",
        "enum": [
          "seating",
          "vehicle",
          "parkingSpace",
          "room",
          "chargingStation"
        ]
      }
    },
    "required": [
      "type",
      "ResourceType.preferredLabel"
    ]
   }
  }
 }
],
"required": [
  "id",
  "type",
  "ResourceReport.number",
  "ResourceReport.reportTime"
]
}
```


The above-mentioned SDMs are still in draft and need to be integrated into the incubated repository before becoming official SDMs. They already give a good indication of how the OSLO classes could or should be implemented in SDM. Although it is time-consuming to prepare the contributions to SDM, SDMs should only be created for the classes that are supported by a use case.

3.3 Generic mapping

OSLO data can already be mapped to NGSI-LD before an official SDM is created. The transformation steps (splitting the OSLO object per type of entity and instantiating the properties and relations as classes) have been automated with a Javascript library [25], called rdfs2ngsi-ld, which can be reused in the pipeline between Context Producers and Context Brokers. In table 5, we can find the README of the tool.

Table 6. Instructions of the rdfs2ngsi-ld tool

rdfs2ngsild.js

JS library to convert RDFS objects (in JSON-LD format) to NGSI-LD compliant JSON-LD objects

What is the difference between an RDFS object and NGSI-LD object?

An example of an RDFS object:

```
{
"@context": "https://brechtvdv.github.io/demo-data/OSLO-airAndWater-Core-ap.jsonld",
 "@id": "https://lodi.ilabt.imec.be/odala/data/observations/16584343831",
 "@type": "Observation",
 "Observation.observedProperty": "http://www.wikidata.org/entity/Q48035511",
 "Observation.hasSimpleResult": "8.10 ug/m3"
}
```

Should be transformed to an NGSI-LD compliant object:

```
{
 "@context": [
  "https://uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context.jsonld",
  "https://brechtvdv.github.io/demo-data/OSLO-airAndWater-Core-ap.jsonld"
 ],
 "@id": "https://lodi.ilabt.imec.be/odala/data/observations/16584343831",
 "@type": "Observation",
 "Observation.observedProperty": {
  "@type": "Relationship",
  "object": "http://www.wikidata.org/entity/Q48035511"
 },
 "Observation.hasSimpleResult": {
  "@type": "Property",
```

```
   "value": "8.10 ug/m3"
  }
}
Use it
npm install @brechtvdv/rdfs2ngsi-ld.js
We expect an JSON-LD object at the input.
import { Ngsildify } from 'rdfs2ngsi-ld.js';
const ngsildify = new Ngsildify();
const input = {
   "@context": "https://brechtvdv.github.io/demo-data/OSLO-airAndWater-Core-ap.jsonld",
   "@id": "https://lodi.ilabt.imec.be/odala/data/observations/16584343831",
   "@type": "Observation",
   "Observation.observedProperty": "http://www.wikidata.org/entity/Q48035511",
   "Observation.hasSimpleResult": "8.10 ug/m3"
};
console.log(ngsildify.transform(input)); // Output will be an array of NGSI-LD compliant entities
[{"@context": [
  "https://uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context.jsonld",
  "https://brechtvdv.github.io/demo-data/OSLO-airAndWater-Core-ap.jsonld"],
      "@id": "https://lodi.ilabt.imec.be/odala/data/observations/16584343831",
      "@type": "Observation",
      "Observation.observedProperty": {
  "@type": "Relationship",
       "object": "http://www.wikidata.org/entity/Q48035511"},"Observation.hasSimpleResult": {
  "@type": "Property",
       "value": "8.10 ug/m3"}
```

}]

What it does

- Loop over the input object when it's an array
- Copy the JSON-LD context and add it with NGSI-LD's context
- loop over all the properties of the JSON object
- Based on the value of the property:
- if the value is a string and does not start with http or has a @value key, add as NGSI-LD Property
- else add as NGSI-LD relationship
- create an identifier if not provided, based on the subject id, relation and index

4 Building a building block for storing event data

A connector has been created to archive event data. These event data are harvested from NGSI-LD context brokers that expose an LDES through the NGSI-LDES component [26] or from other LDES-compatible systems. These event streams are then replicated and stored into a back-end system of choice (MongoDB, GraphQL, NGSI-LD) using connectors [27]. In GreenMov and ODALA, a connector [28] is developed to replicate an event stream into an NGSI-LD system, using the standardised HTTP bindings (Fig. 9 below). This can be an interesting approach for cities that want to store event data from other data publishers, such as vendors that maintain air quality sensors.

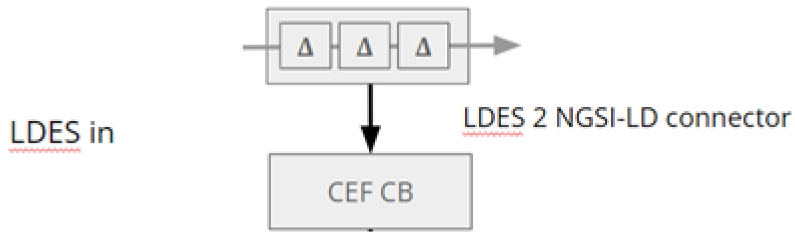


Figure 11. Data from an existing LDES can be ingested into an NGSI-LD Context Broker

5 Conclusions

The planned semantics or data models and vocabulary have been delivered.

- A vocabulary and application profile have been created with the OSLO framework. These semantics described the data for shared mobility at point where a user can switch the mode of transport.
- Two Smart Data Models for the same domain were created, using NGSI-LD.
- In D3.3. the project has provided the generic mapping between OSLO and Smart Data Models.

Hence the interoperability has been demonstrated. A developer who is used to work with JSON can evolve to Linked Data by working with JSON-LD. The necessary schema to expand JSON to JSON-LD has been provided.

6 References

[1] https://green-mov.eu/sites/GreenMov/files/public/content-files/2022/GreenMov%20-%20D2.1_Extended%20Smart%20Data%20Models_v1.0.pdf

[2] https://green-mov.eu/sites/greenmov/files/public/content-files/2022/GreenMov%20-%20D5.1_Requirements%20for%20the%20data%20sets%20and%20mobility%20services_v1.0.pdf

[3] http://data.vlaanderen.be/ns/

[4] https://purl.eu/

[5] https://green-mov.eu/sites/greenmov/files/public/content-files/2022/GreenMov - D2.1_Extended Smart Data Models_v1.0.pdf

[6] https://inspire.ec.europa.eu/theme/tn

[7] https://www.mobivoc.org/

[8] https://purl.eu/doc/applicationprofile/netex-belgium/stopplaces-ap/

[9] https://data.vlaanderen.be/standaarden/erkende-standaard/applicatieprofiel-mobiliteit-trips-en-aanbod.html

[10] https://data.vlaanderen.be/standaarden/erkende-standaard/applicatieprofiel-mobiliteit-dienstregeling-en-planning-stopplaatsen.html

[11] https://data.vlaanderen.be/standaarden/erkende-standaard/applicatieprofiel-openbaar-domein.html

[12] https://data.vlaanderen.be/standaarden/kandidaat-standaard/vocabularia-en-applicatieprofielen-hoppinpunten.html

[13] https://github.com/Informatievlaanderen/OSLOthema-passengerTransportHubs/tree/standaardenregister/reports

[14] https://github.com/Informatievlaanderen/OSLOthema-passengerTransportHubs/blob/standaardenregister/reports/26042022_OSLO%20Passenger%20Transport%20Hubs_Business%20Workshop_Report.pdf

[15] https://github.com/Informatievlaanderen/OSLOthema-passengerTransportHubs/blob/standaardenregister/reports/19052022_OSLO%20Passenger%20Transport%20Hubs_Thematic%20Workshop%201_Report.pdf

[16] https://github.com/Informatievlaanderen/OSLOthema-passengerTransportHubs/blob/standaardenregister/reports/23062022_OSLO%20Passenger%20Transport%20Hubs_Thematic%20Workshop%201_Report.pdf

[17] https://github.com/Informatievlaanderen/OSLOthema-passengerTransportHubs

[18] https://purl.eu/doc/applicationprofile/mobility/passenger-transport-hubs/working-draft/2022-05-01/html/overview.jpg

[19] https://data.vlaanderen.be/doc/applicatieprofiel/mobiliteit-trips-en-aanbod

[20] https://data.vlaanderen.be/doc/vocabularium/openbaardomein/infrastructuur

[21] https://data.vlaanderen.be/standaarden/erkende-standaard/vocabularium-mobiliteit-dienstregeling-en-planning.html

[22] https://data.vlaanderen.be/ns/transportnetwerk

[23] https://data.vlaanderen.be/ns/mobiliteit/trips-en-aanbod

[24] https://www.blue-bike.be/en

[25] https://github.com/brechtvdv/rdfs2ngsi-ld

[26] https://github.com/TREEcg/ngsi-ldes

[27] https://github.com/Informatievlaanderen/ldes2service

[28] https://github.com/TREEcg/connectors/tree/main/packages/connector-write-ngsi-ld

7 Annex

UML diagram of OSLO Passenger Transport Hubs

The UML-schema of the application profile is spread out over the next 18 pages.

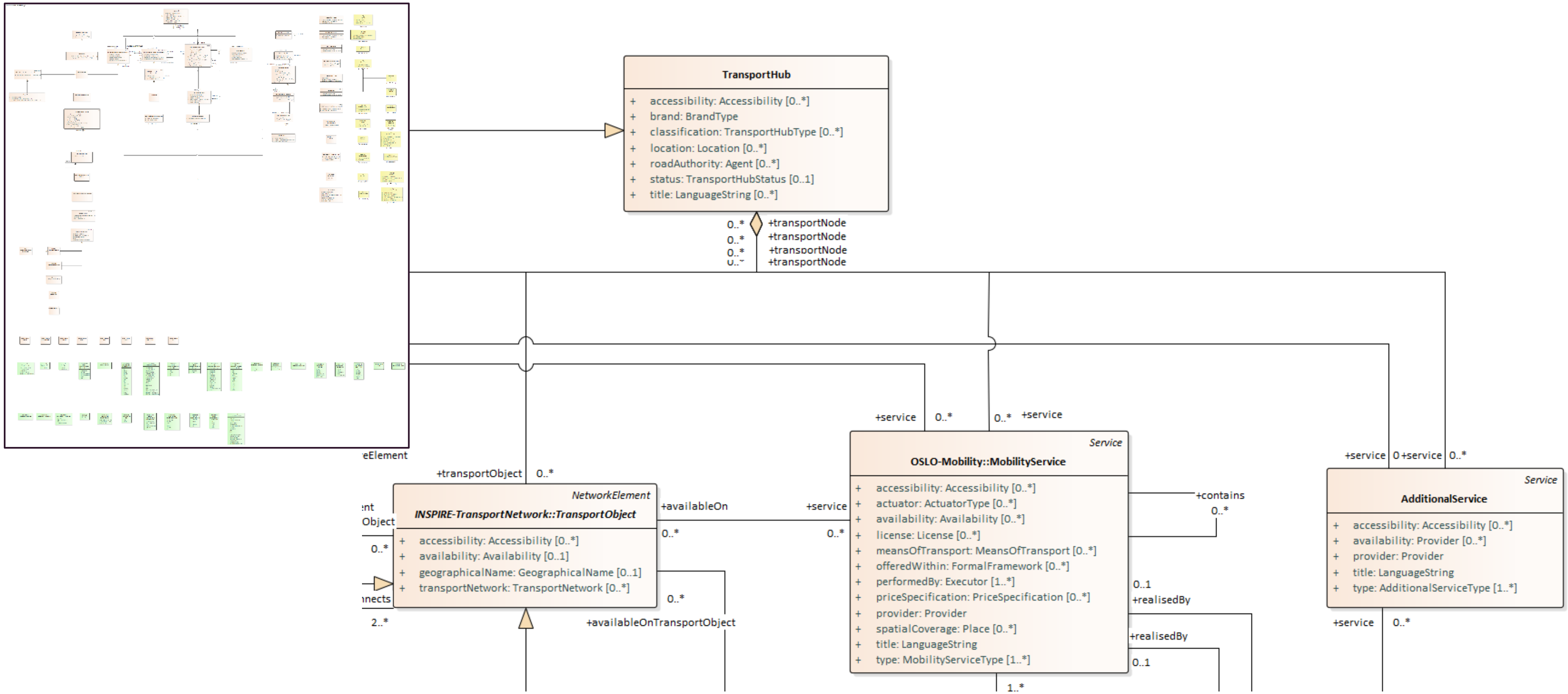


Figure 12. UML-schema of the application profile (1/18)

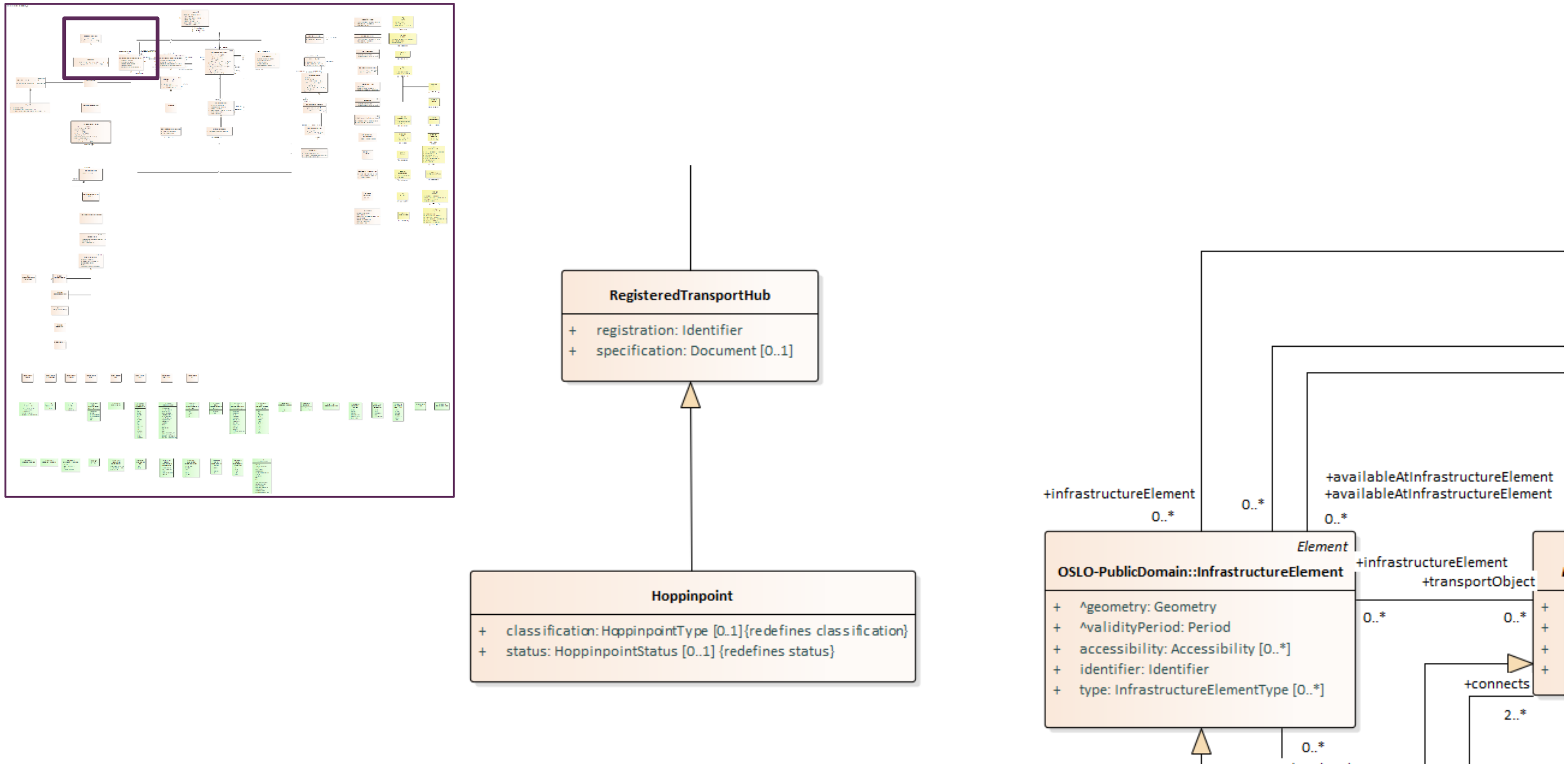


Figure 13. UML-schema of the application profile (2/18)

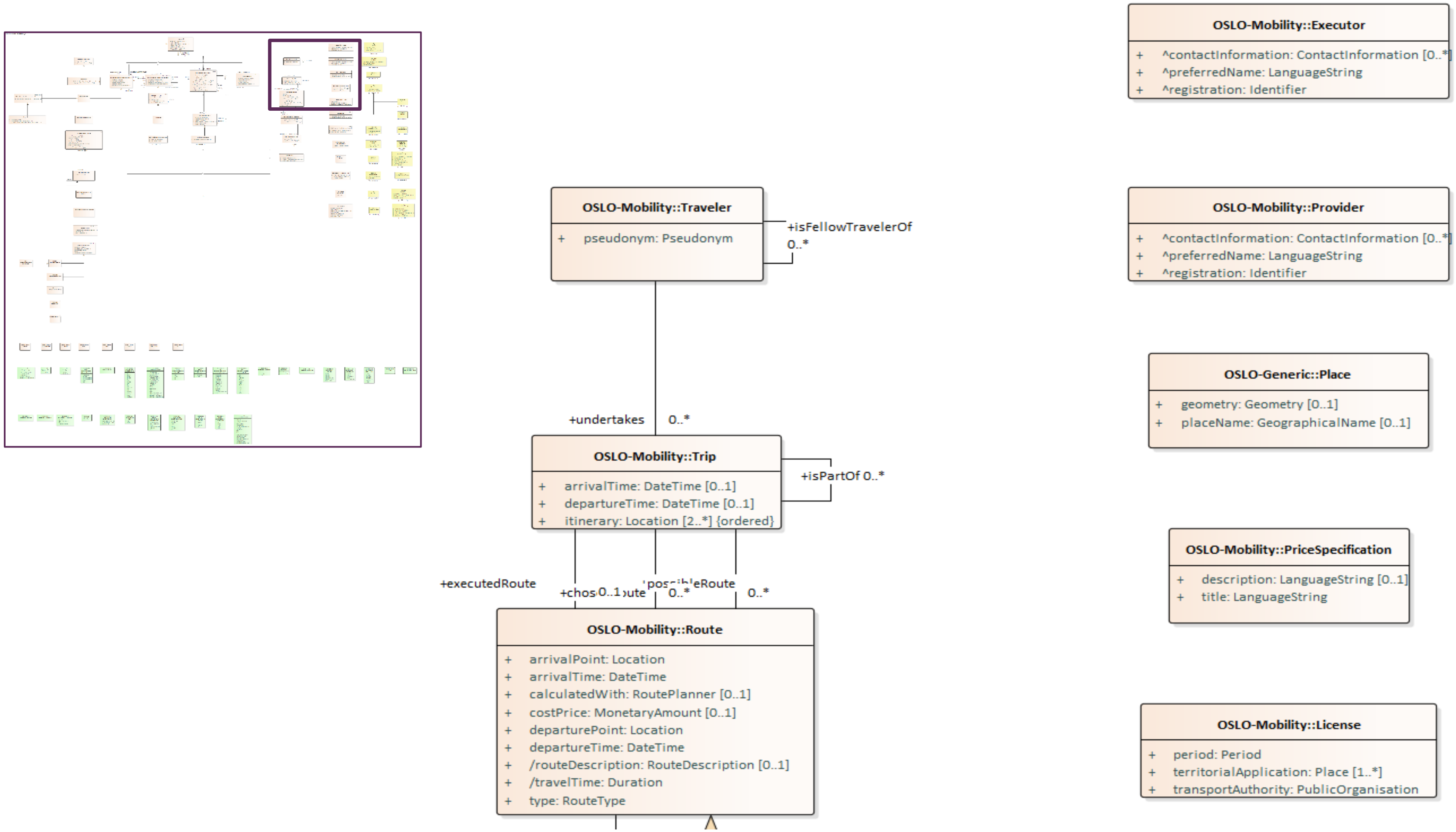


Figure 14. UML-schema of the application profile (3/18)

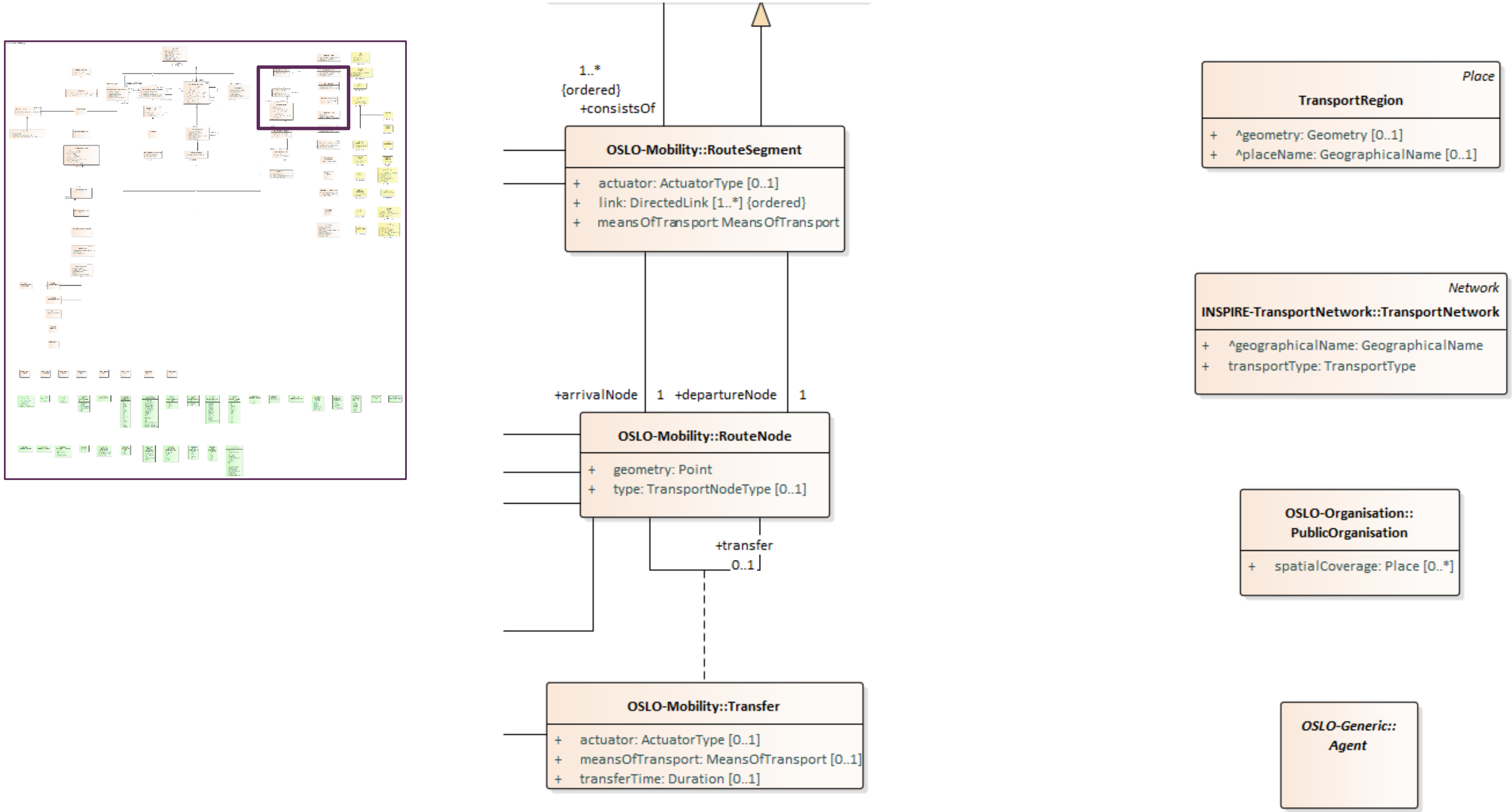


Figure 15. UML-schema of the application profile (4/18)

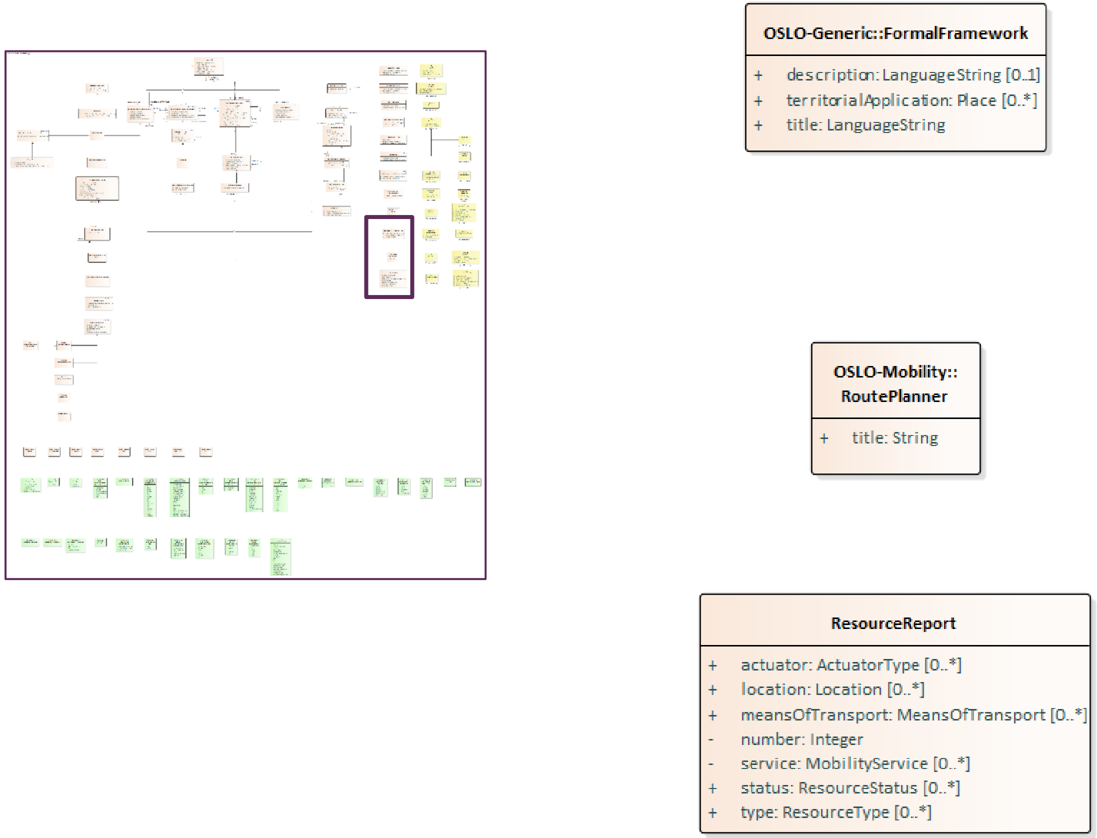


Figure 16. UML-schema of the application profile (5/18)

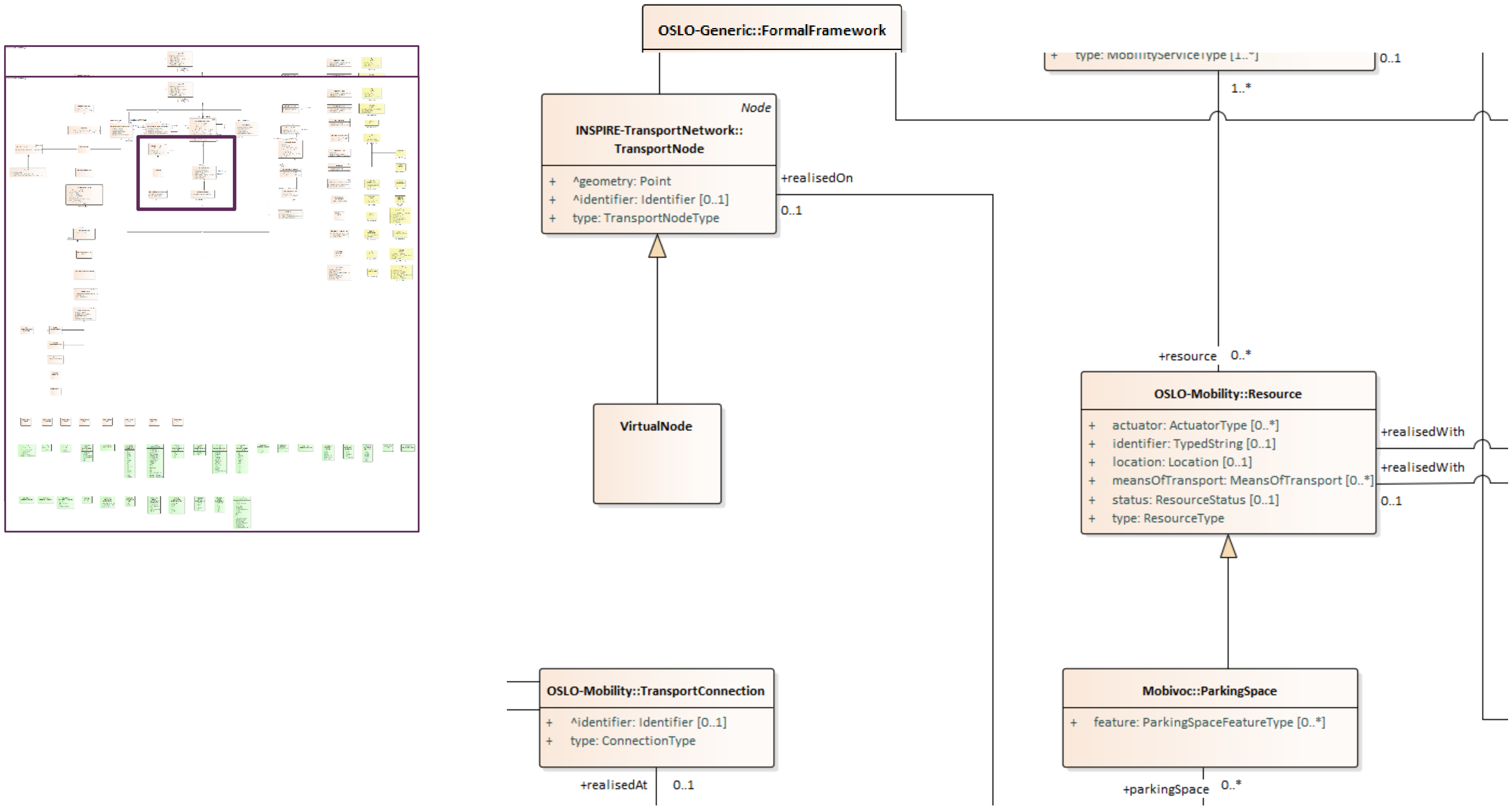


Figure 18. UML-schema of the application profile (6/18)

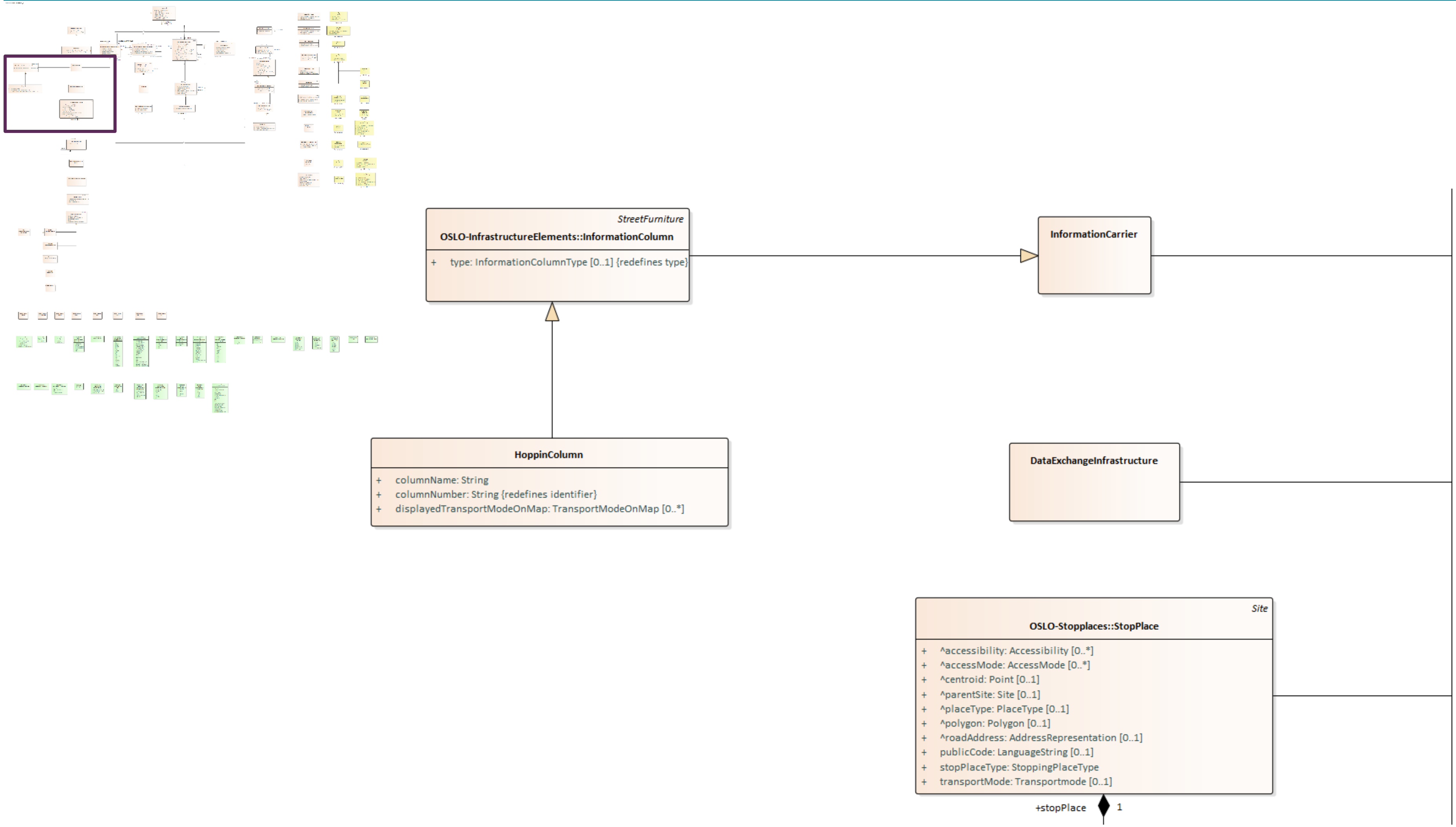


Figure 19. UML-schema of the application profile (7/18)

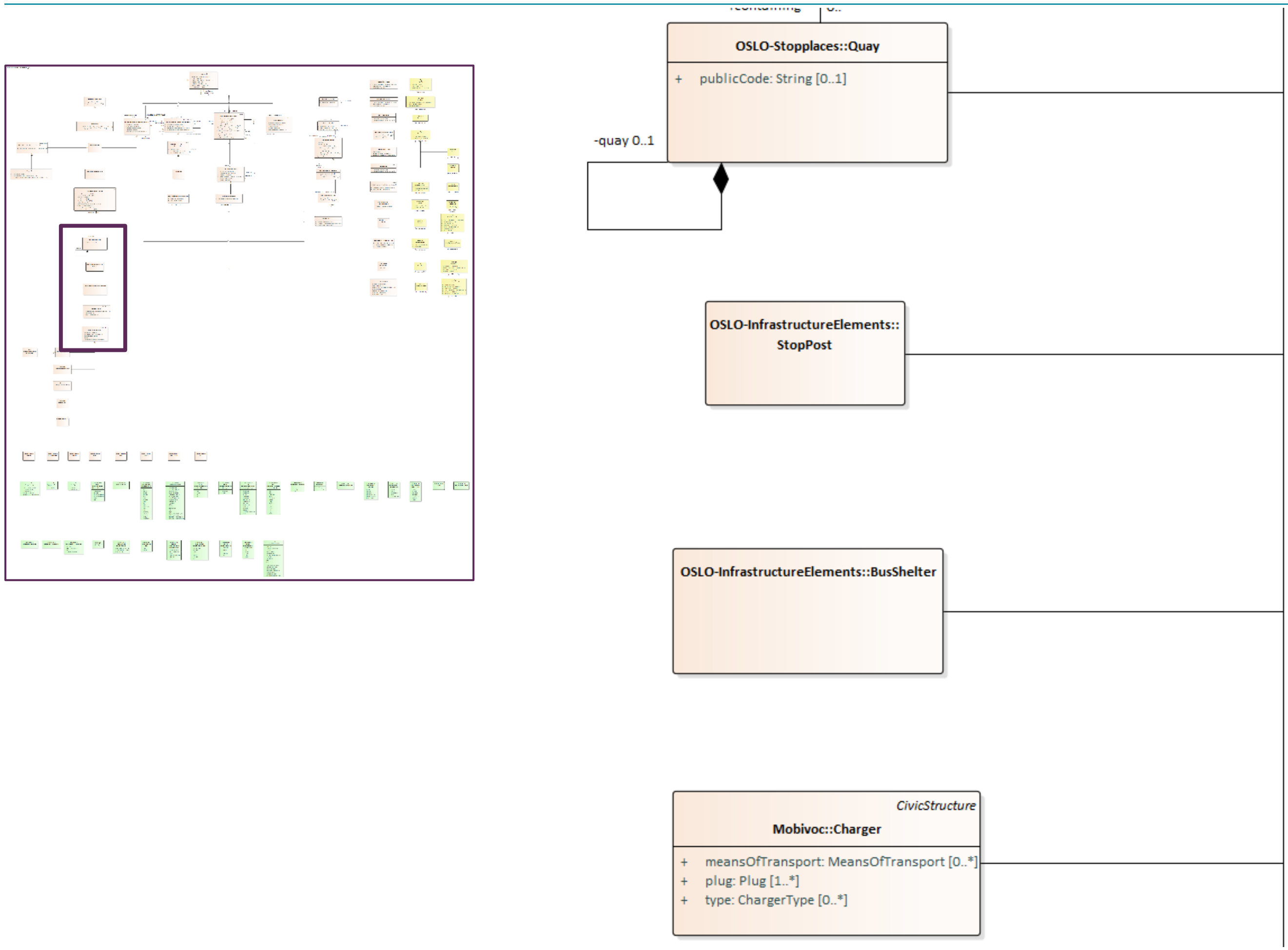


Figure 20. UML-schema of the application profile (8/18)

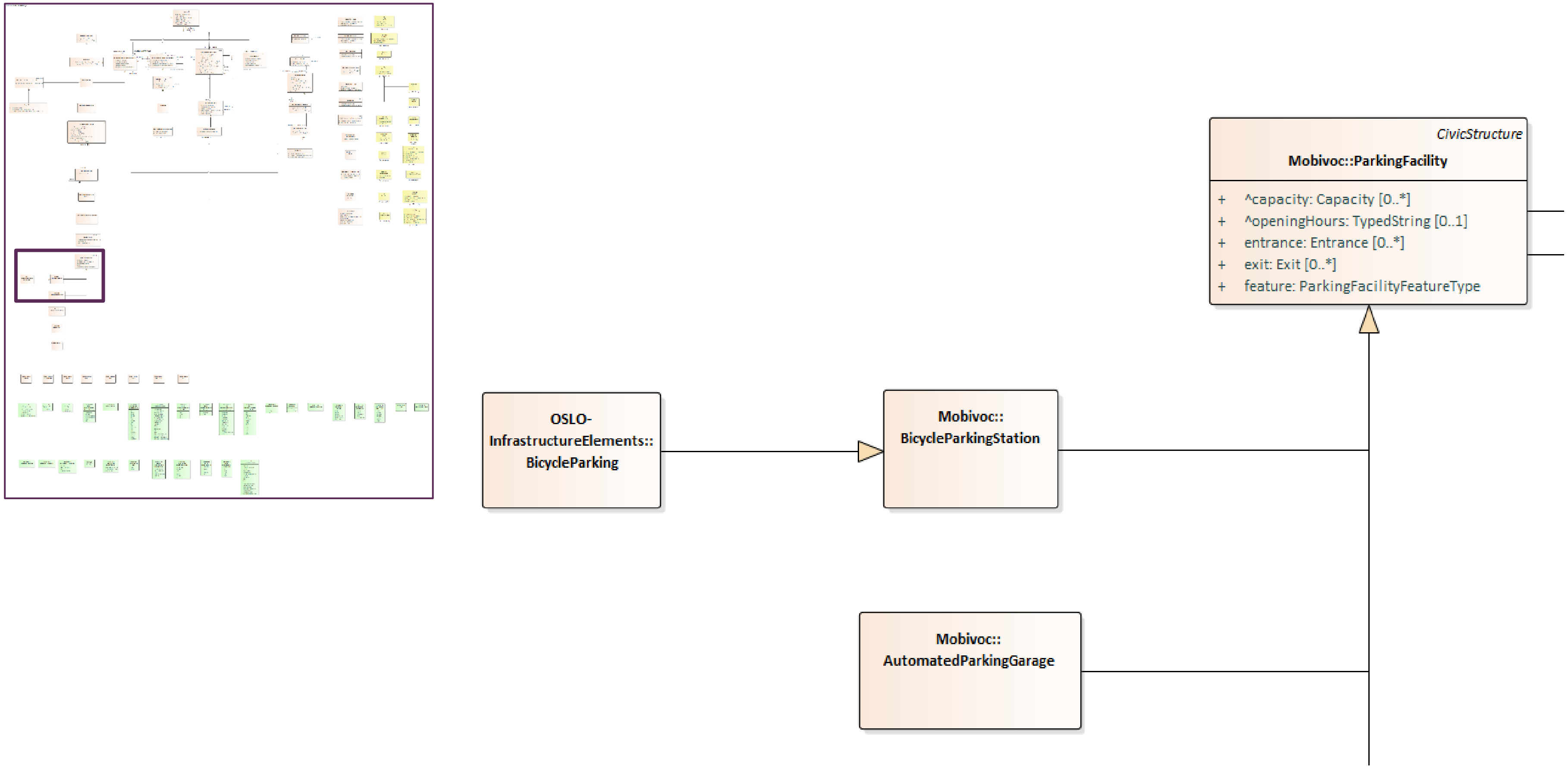
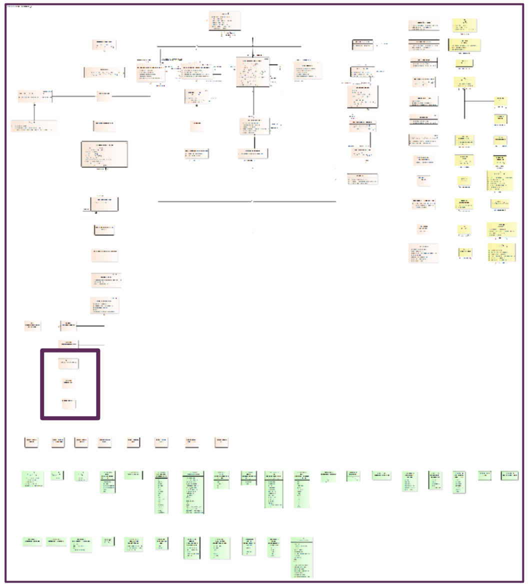


Figure 21. UML-schema of the application profile (9/18)



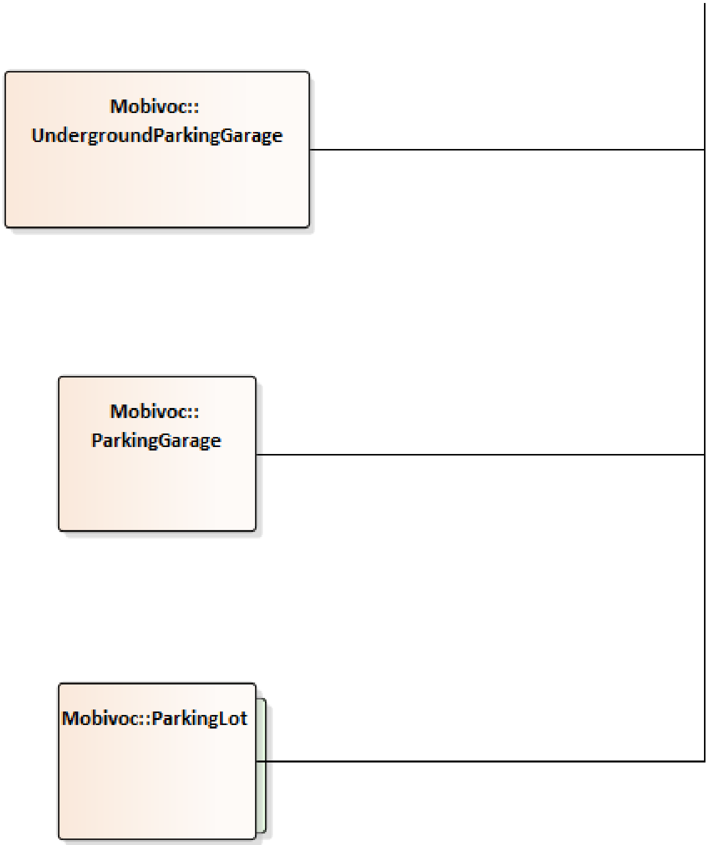


Figure 22. UML-schema of the application profile (10/18)

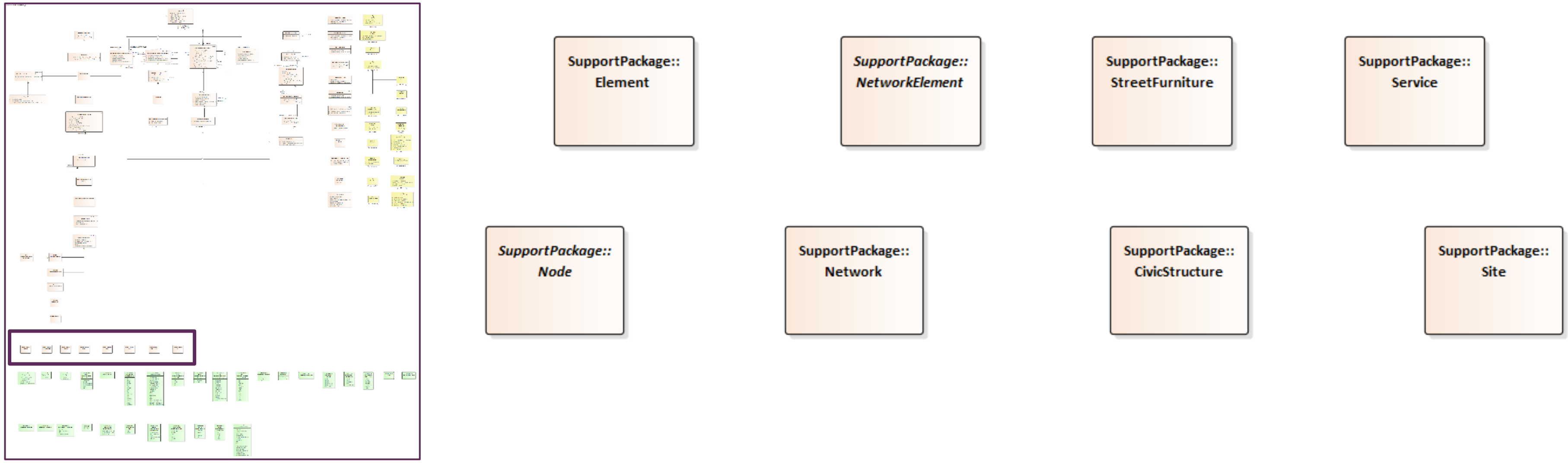


Figure 23. UML-schema of the application profile (11/18)

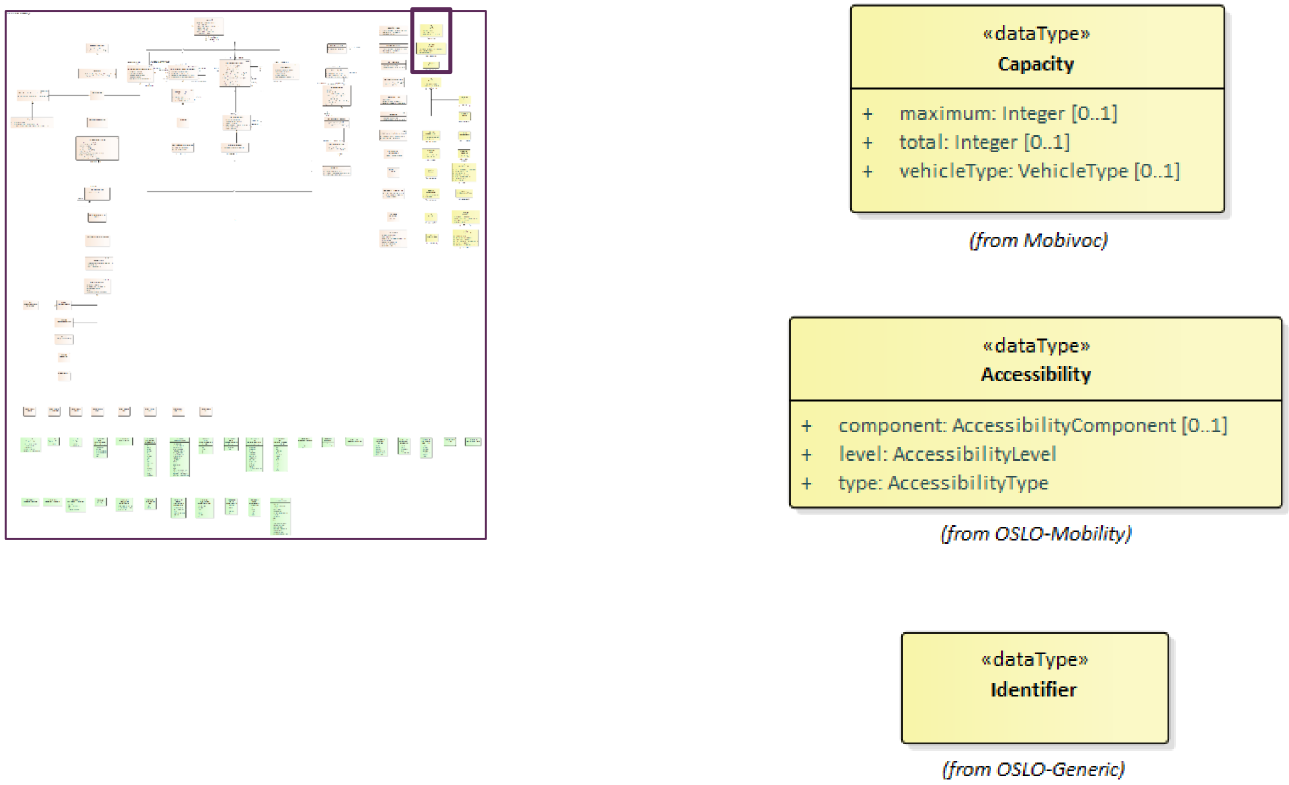


Figure 24. UML-schema of the application profile (12/18)

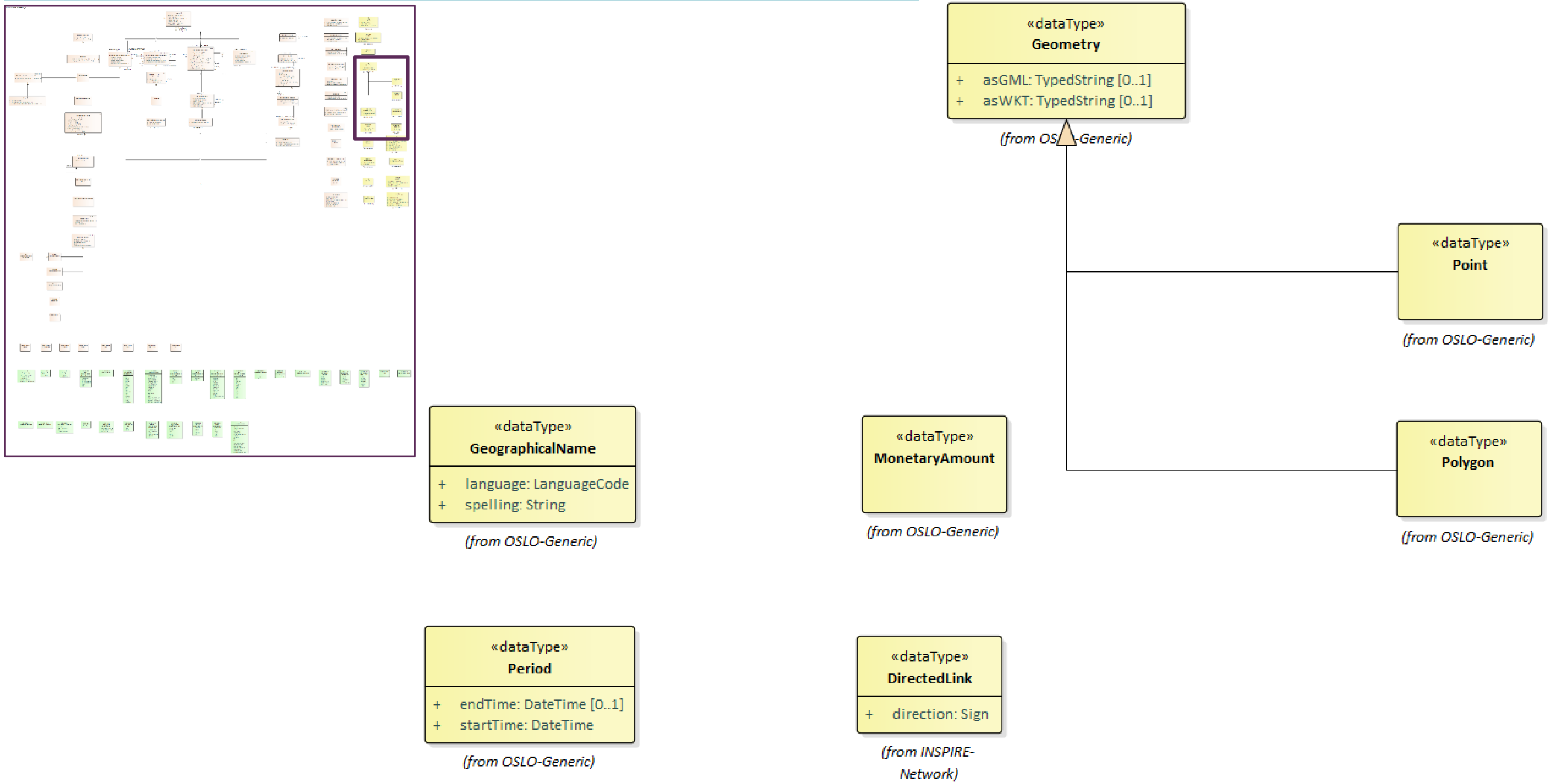


Figure 25. UML-schema of the application profile (13/18)

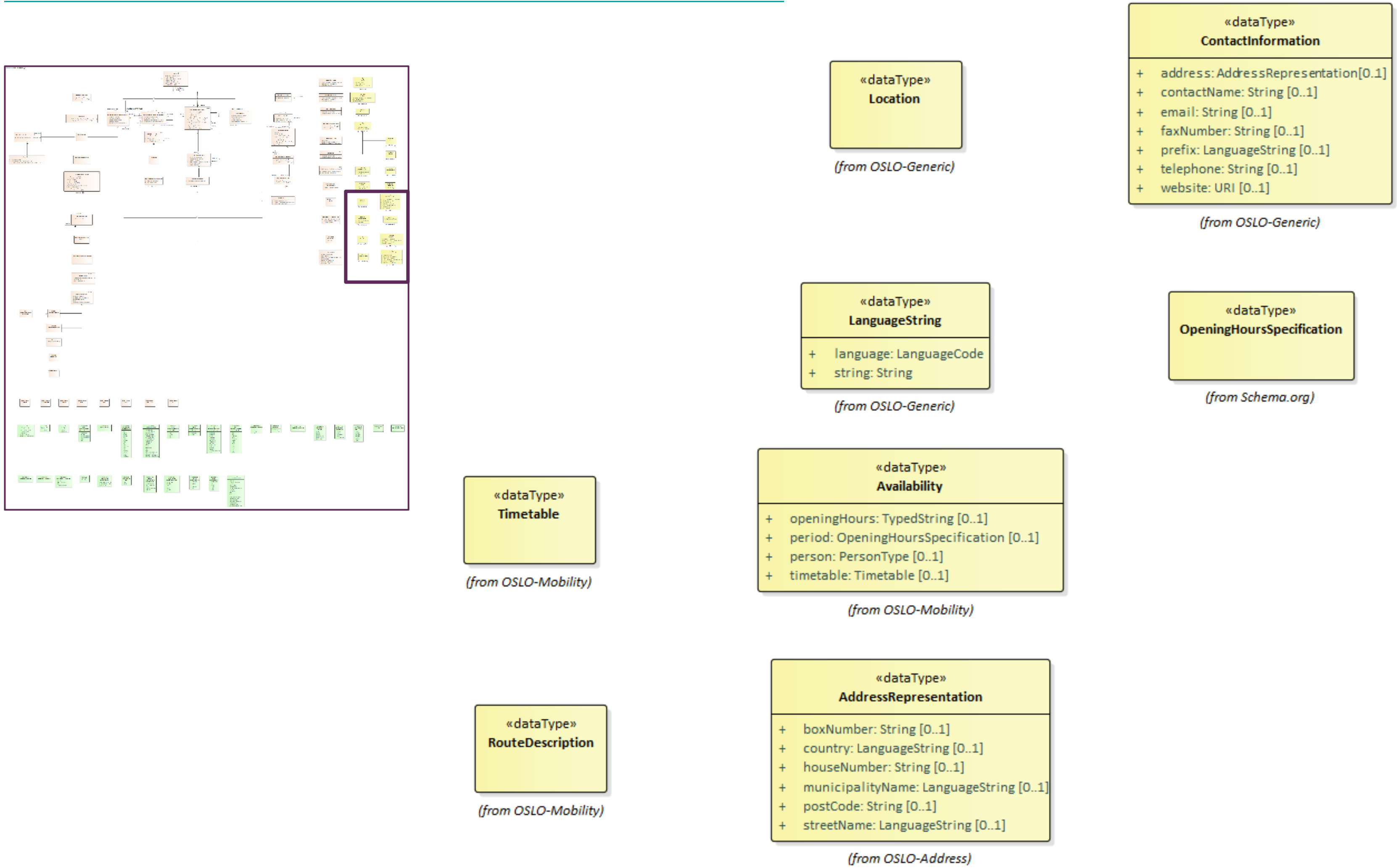


Figure 26. UML-schema of the application profile (14/18)

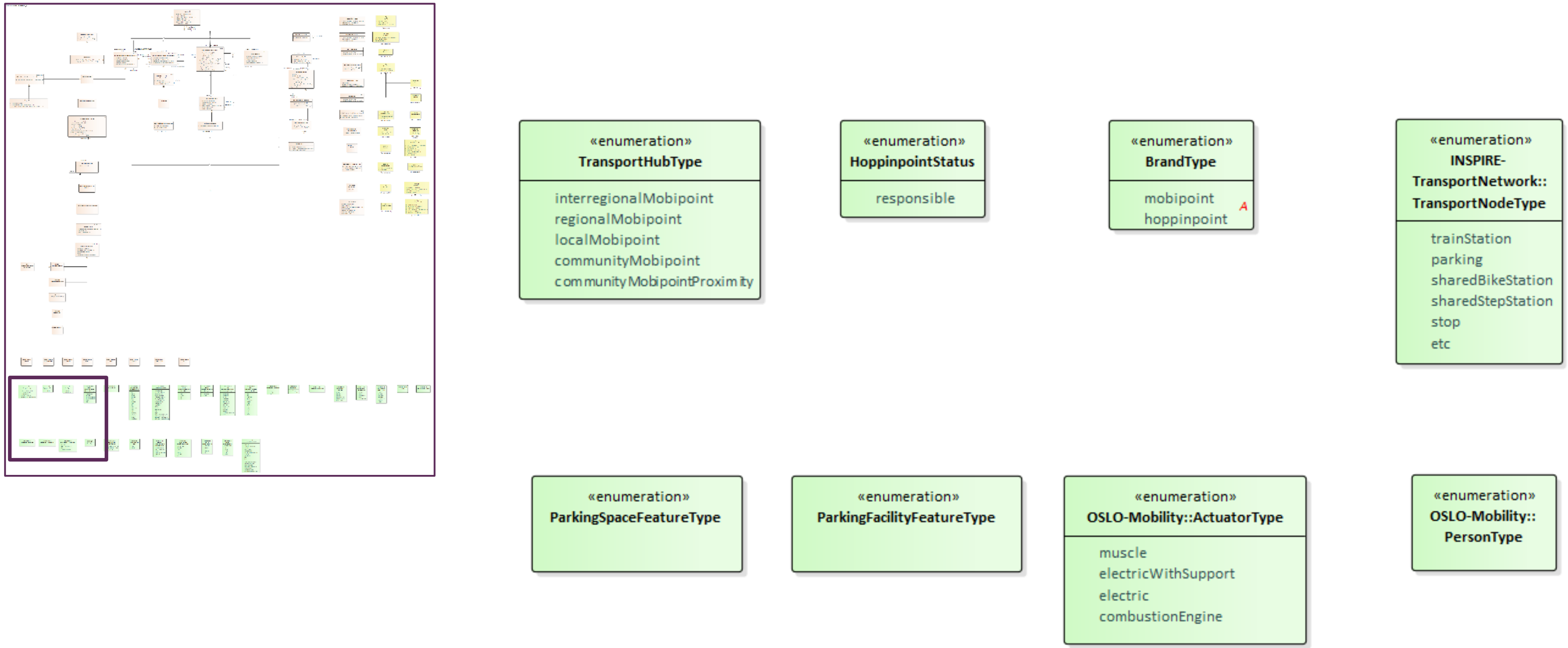


Figure 27. UML-schema of the application profile (15/18)

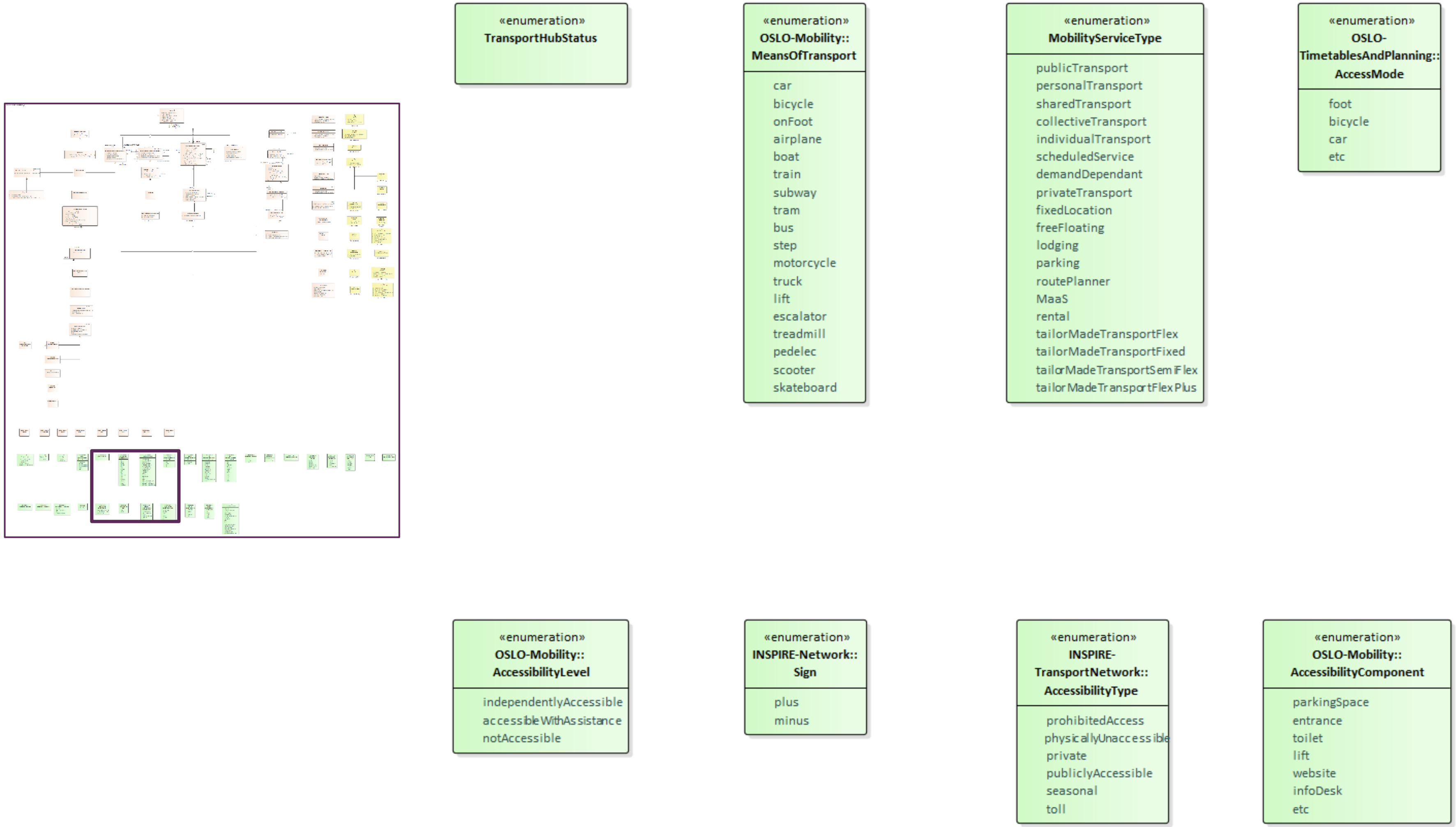


Figure 28. UML-schema of the application profile (16/18)

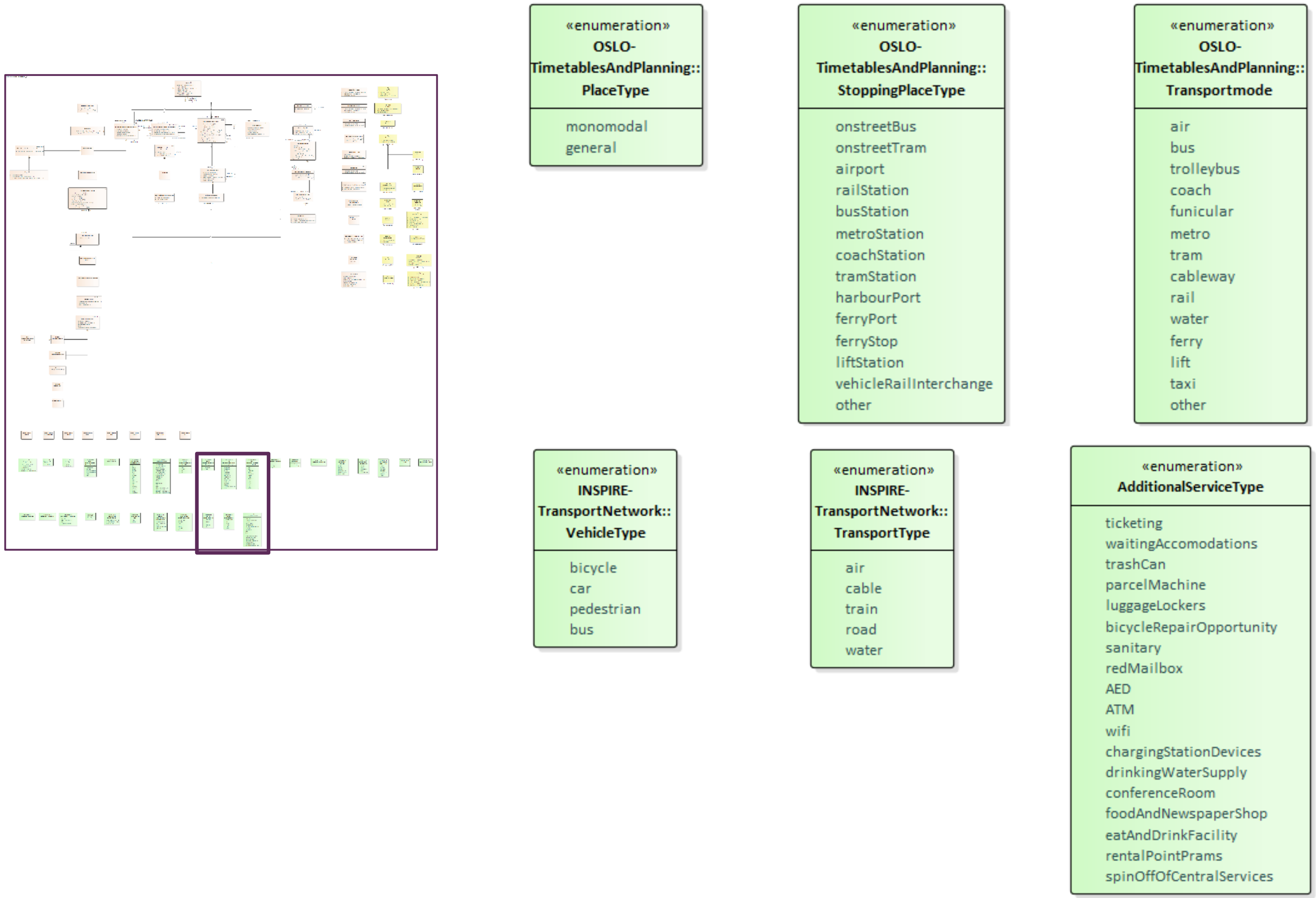


Figure 29. UML-schema of the application profile (17/18)

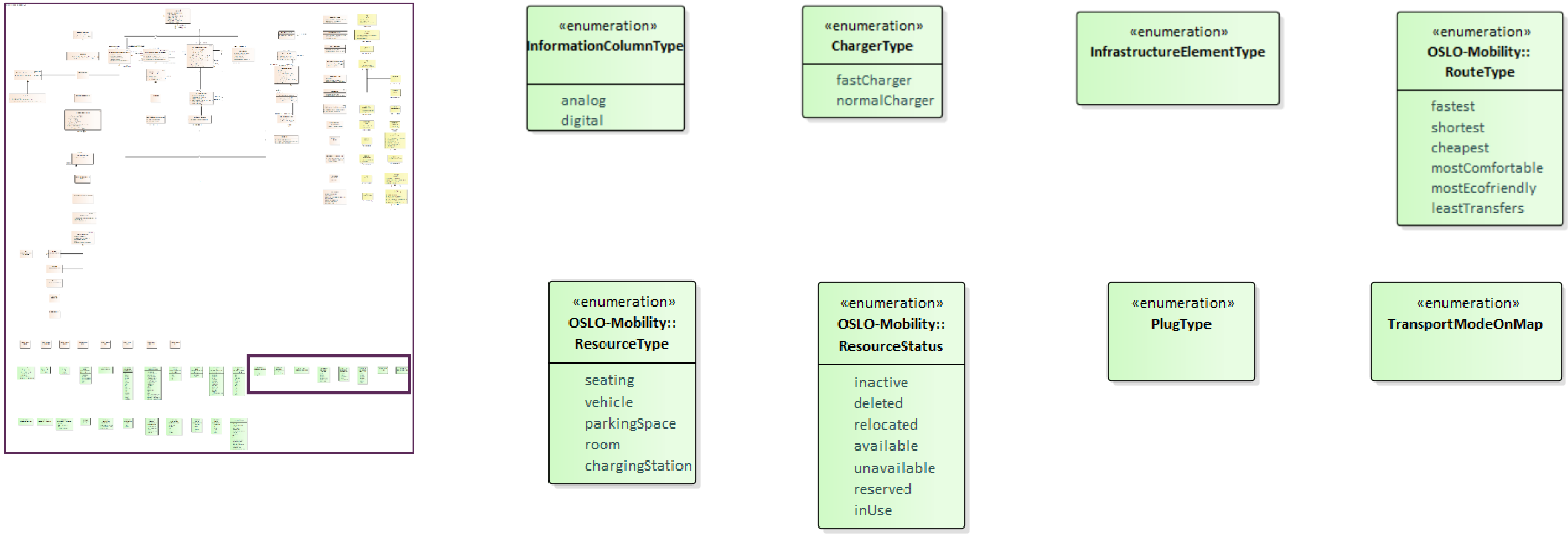


Figure 30. UML-schema of the application profile (18/18)