

# Sharing Platform Ontology Development: Proof-of-Concept

Thomas Derave <sup>1,\*</sup> , Tiago Princes Sales <sup>2</sup> , Frederik Gailly <sup>1,3</sup>  and Geert Poels <sup>1,3</sup> 

<sup>1</sup> Department of Business Informatics and Operations Management, Ghent University, 9000 Ghent, Belgium; frederik.gailly@ugent.be (F.G.); Geert.Poels@UGent.be (G.P.)

<sup>2</sup> Faculty of Computer Science, Free University of Bozen-Bolzano, 39100 Bolzano, Italy; tiago.princesales@unibz.it

<sup>3</sup> Design & Optimisation Cluster, FlandersMake@UGent—Core Lab CVAMO, 9000 Ghent, Belgium

\* Correspondence: thomas.derave@ugent.be

**Abstract:** A sharing platform is a digital platform that facilitates access to underutilized goods by renting or lending them to others. Users become less dependent on ownership which improves efficiency, sustainability, and the sense of community. The Sharing Economy (SE) is considered a complex domain because value co-creation can occur between multiple types of platform users. Developing platform software that offers the right functionality for the intended digital platform is therefore challenging. To address this complexity, an ontology can be utilized, which is an explicit specification of a conceptualization that provides a controlled vocabulary and shared meaning regarding a domain. The use of a sharing platform ontology allows for more effective communication and knowledge sharing amongst stakeholders in platform development and eventually drives the platform software development process. However, currently, it is not known how to develop such an ontology. In previous research, we developed the Digital Platform Ontology (DPO) and an extension of the DPO for describing platform business model choices. The DPO describes the digital platform domain in terms of digital platform types, including sharing platforms. However, a method to use the DPO and its business model extension for developing an organization-specific ontology that describes the functionality and business model of a specific existing or intended sharing platform was lacking. In this paper, we develop such a method using the Design Science Research Methodology (DSRM) and demonstrate it using a proof-of-concept based on the BlaBlaCar and Couchsurfing platforms.

**Keywords:** organization-specific ontology; UFO; DPO; sharing platform; sharing economy; digital platform; conceptual model; OntoUML; DSRM



**Citation:** Derave, T.; Princes Sales, T.; Gailly, F.; Poels, G. Sharing Platform Ontology Development:

Proof-of-Concept. *Sustainability* **2022**, *14*, 2076. <https://doi.org/10.3390/su14042076>

Academic Editor: Colin Michael Hall

Received: 7 December 2021

Accepted: 31 January 2022

Published: 11 February 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The platform economy refers to activities in business, culture, and social interaction that are performed on or are intermediated by digital platforms [1]. For a number of years, specific attention has been devoted to sharing platforms, which are a specific type of digital platform operating in the Sharing Economy (SE) [2]. A sharing platform focuses on the sharing of underutilized assets in ways that improve efficiency, sustainability, and the sense of community [3].

The idea of sharing platforms and their economic advantages is not a recent phenomenon. The recently increased interest in developing sharing platforms is due to the emergence of new internet-based technologies, which allow users to easily communicate, establish an agreement, make a transaction, and reciprocate. The digitalization of the SE has enormously decreased transaction costs and free-riding behavior between unknown others [3,4]. Due to this technological evolution, sharing platforms including BlaBlaCar and Couchsurfing have recently emerged as a viable alternative to fulfilling a variety of consumer needs, ranging from car-sharing to tools renting to overnight accommodations that were previously provided by firms [5]. On the positive side, SE is promoted by practitioners, industry associations, policy-makers, and academics because of its purported sustainability

potential [6]. Sharing platforms have environmental impacts through a reduction in the total resources required, and they help reduce pollutants, emissions, and carbon footprints. For example, in the transportation sector, vehicle sharing behavior can have a positive environmental impact by decreasing the number of kilometers travelled. Such sharing activities can also stimulate long-lived changes in consumer behavior by shifting personal transportation choices from ownership to demand-fulfilment [3].

Technology alone is not sufficient to develop a successful sharing platform. It has been recognized that the Sharing Economy will only prove sustainable through mutual cooperation amongst all stakeholders, including public authorities, enterprises, and consumers [3]. This cooperation will not be easy, as the SE is mostly heterogeneous due to the interaction between different players from different domains and is therefore considered a complex domain. Many stakeholders in the SE struggle with knowledge sharing when involving inter-agency collaboration, resulting in issues with interoperability [7]. Due to this complexity, companies may use the ‘sharing economy’ as a marketing gimmick to disguise profit-motivation and exploitation under the pretense of making society a better place [3].

A solution to lowering this complexity is using a sharing platform ontology which is “an explicit specification of a conceptualization” [8] that “provides a controlled vocabulary, standard terms and consensus and a shared meaning among players to interoperate, communicate and share”. Without the availability of a domain ontology to provide a common language to facilitate communication, the SE domain cannot reach its maximum potential [7]. The standard, common vocabulary curated through an ontology can play an important role in the creation, understanding, and communication of new regulations. This improves platform governance, which is important as sharing will only prove sustainable through the mutual cooperation amongst public authorities, enterprises, and consumers [3]. Besides regulation, an ontology can also support the design of new sharing platforms that cater to the needs of both users and service providers [7]. An ontology promotes knowledge sharing, improves decision making, and supports knowledge reuse, and this again enables the rapid growth of the domain. New, niche platforms may emerge that tend to employ broader goals seeking to open up a pathway for the SE aligned with a transition to sustainability [9].

Currently, there is no clear framework to assist ontology development in the SE domain [7]. In addition, existing ontologies are not sufficient to replace the need for new ontology development [7]. How to develop an ontology that describes digital platforms that fall within the boundaries of the SE, and organization-specific ontologies that describe specific existing or intended sharing platforms is the research question that we investigate in this paper.

In previous research, we developed the Digital Platform Ontology (DPO) [10] and a business model extension to the DPO [11]. The DPO provides a description of the functionality of digital platforms in general and of specific functionality depending on the type of digital platform considered, including sharing platforms. The DPO business model extension allows describing platform functionality that is adapted to specific choices regarding the platform’s business model. In this paper, we demonstrate that the DPO and its business model extension can be used as a framework to develop sharing platform ontologies. To this end, we present a method for sharing platform development that is based on the DPO. This method is developed using the Design Science Research (DSR) approach of [12] and is based on a reflection of our experience in developing organization-specific ontologies for numerous existing digital platforms, in line with the ontology engineering approach of Ruy et al. [13].

The method developed has two phases and is demonstrated via a proof-of-concept consisting of two parts. The first part of our proof-of-concept shows how to use the DPO to develop a business model agnostic ontology for a sharing platform following the definition of Frenken and Schor [4]. The second part extends the proof-of-concept by developing organization-specific ontologies using the business model extension of the DPO. These

organization-specific ontologies are the result of reusing and combining classes, relationships, and constraints of the extended DPO to describe specific instances of the sharing platform type of digital platforms. The organization-specific ontologies are BlaBlacar and Couchsurfing, two existing sharing platforms operating different business models.

The modular design of the DPO simplifies the difference between types of digital platforms (including sharing platforms), and the extension of the DPO makes it easier for developers to analyze the influence of business model decisions on the creation of sharing platform software. Using the method we propose, DPO-based organization-specific ontologies of sharing platforms can be developed that offers specific advantages for platform software development. As mentioned by [14], an ontology can provide the requirements needed to integrate complex activities into an application and enables smooth systems integrations. In [7], it is concluded that a sharing platform ontology can “detect the structure of the business process”, “making it exceptionally easy to integrate different business model variations under the same environment”. An ontology can, therefore, “facilitate the design of the database, software, user interface, and components of applications” [7] in the SE.

The proposed method helps to increase the knowledge of software design in the SE, which triggers the conception of alternatives for monopolistic companies such as Airbnb and Uber, who are criticized for paying low wages, taking high commission fees, and avoiding taxes [15]. Consequently, the use of configurable sharing platform ontologies, as developed using our method, may facilitate the development of diverse, smaller, more alternative, and socially responsible platforms and thus contribute to the creation of a more socially responsible SE.

This paper is structured as follows; In Section 2, we give background on UFO, OntoUML, and the DPO. In Section 3, we discuss the methodology to develop our method. In Sections 4 and 5, we present each phase of our method. In Section 6, we discuss related work and our contributions. In Section 7, we discuss research limitations and outline future work. Finally, in Section 8, we present our conclusions.

## 2. Background: UFO, OntoUML and the Digital Platform Ontology (DPO)

The DPO [16] is based on the Unified Foundational Ontology (UFO), an ontology that defines basic concepts such as objects, events, social elements and their types, relations, and properties [17]. Therefore, the DPO is specified using OntoUML [18], a conceptual modelling language that is capable of representing the objects, events, and social entities of UFO. The models discussed in this paper are created using the OntoUML plugin for Visual Paradigm. This plugin supports UFO-A (i.e., an ontology of objects), UFO-B (i.e., an ontology of events), and UFO-MLT (i.e., an ontology for categorization using multi-level theory) [17,19,20]. Other ontologies in the SE domain mainly focus on the objects and entities (also called endurants) in our models reused from UFO-A, but the ability to capture events and user actions (also called perdurants), reusing UFO-B patterns, is even more important as user actions are considered the backbone of the SE domain [7]. The plugin uses colors to indicate the origin of the stereotypes: UFO-A stereotypes that have their origin in kinds are represented in red while the ones that originated in relators and have the ability to connect two or more kinds are represented in green; UFO-B stereotypes are represented in yellow and UFO-MLT types in purple.

In previous research, a digital platform is used alternately as the algorithm, the company owning the algorithm, a business model, or the offered service [21]. This conceptual confusion makes academic decision making difficult and makes decisions difficult to communicate [22]. As we wish to cover a wide range of digital platform types, we related the platform economy to the broader concept of the service economy, and therefore defined a digital platform as “a service offering by a digital platform company to the users that may be bound to an agreement”. The primary service offered is interactions between users; these interactions are enabled by a software [16]. This definition is very broad, as the intended interactions can consist solely of information transfer (e.g., WhatsApp, Tinder)

but can also include offerings of products (e.g., eBay), services (e.g., Airbnb), or investments (e.g., Kickstarter) [16].

The general DPO, shown in Figure 1, describes the functionality of any digital platform independent of its type or business model. The core ontology UFO-S [23] provides a clear account of service-related concepts, and therefore we can reuse UFO-S patterns to help define the differences between the software, the company owning the software, and the offered service by that company, independent of the chosen business model. The general ontology for digital platforms in Figure 1 can be interpreted as follows: A digital platform is a service offering as defined in UFO-S, offered by a platform company towards a target platform user community. This community is a collection of target platform users and can be persons or organizations. When a targeted user visits the platforms, he can initiate a user affiliation action binding him to an agreement (in this case, a user affiliation agreement) with the platform company. He becomes a platform user and from then on has a specific user role. The digital platform is enabled by the platform software that supports different kinds of platform supported actions. These platform supported actions are divided into user actions (i.e., platform actions performed by users), platform company actions (i.e., platform actions performed by an employee), and platform software actions (i.e., autonomous actions). The most basic user actions are the user affiliation actions, after which digital content creation (e.g., sending a message) and digital content consumption (e.g., receiving a message) can occur. When both creation and consumption take place, we talk about a communication action. The platform software must allow platform user interaction, which requires mutual communication between users (e.g., sending, receiving, replying, receiving).

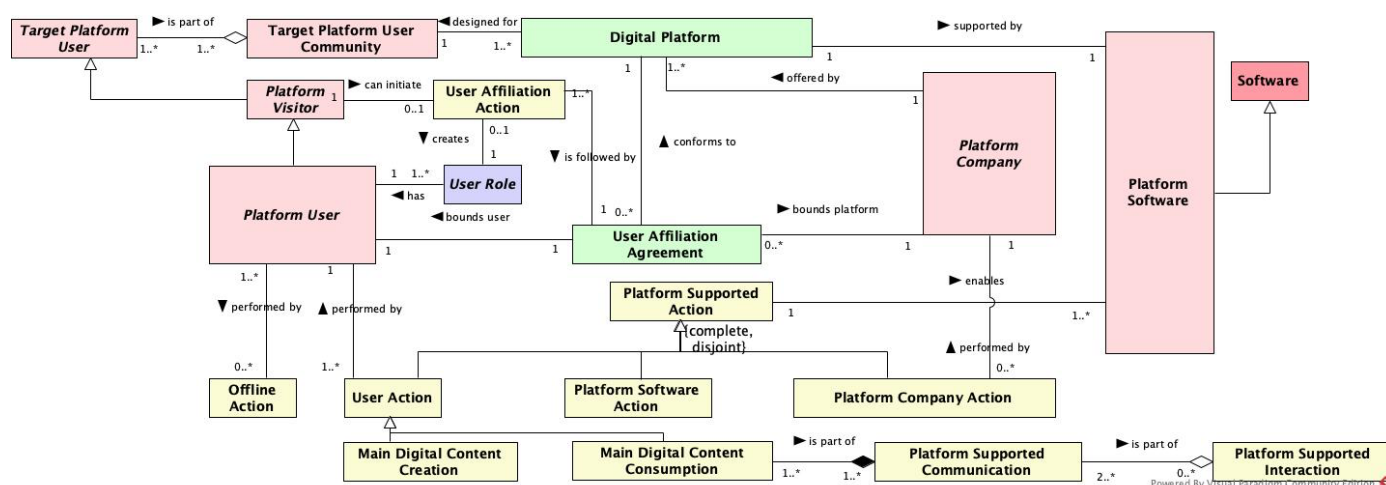


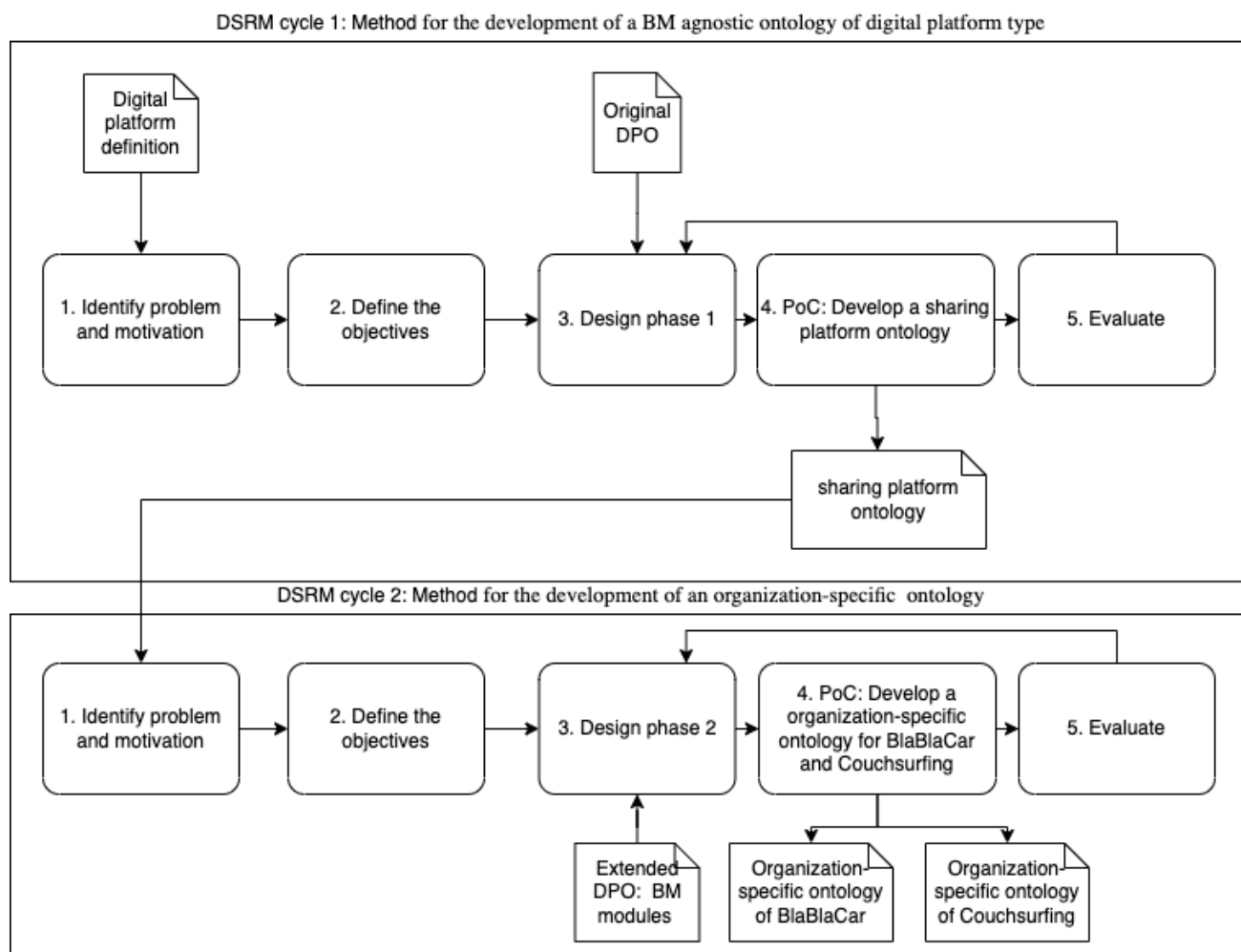
Figure 1. General digital platform ontology in OntoUML.

On top of the general digital platform ontology in Figure 1, the original DPO [24] also consists of different modules formalizing the differences between the digital platform types. The original DPO is developed using a literature review and validated with a sample of 44 existing digital platforms and 3 ontology experts using Competency Questions (CQs) [24]. CQs are a set of queries that the ontology must be capable of answering to be considered competent for conceptualizing the domain it was intended for [25].

A Business Model (BM) extension of the DPO, specifically for digital marketplaces, was developed by [11] and is based on a literature review of 31 papers and validated using a sample of 47 existing digital marketplaces. The digital platform type sharing platform is a subtype of a ‘digital marketplace’, which is again a subtype of a ‘digital platform’. Therefore, we can reuse both the original and extended DPO for our proof-of-concept to demonstrate the development of organization-specific sharing platform ontologies. The differences between these types will be further explained in Section 4.

### 3. Research Methodology

This paper proposes the development of a method with two phases, each phase developed on the basis of a separate Design Science Research Methodology (DSRM) cycle by Peffers et al. [12] and demonstrated with a ‘Proof-of-Concept’. An overview of the complete methodology is given in Figure 2.



**Figure 2.** Research Methodology: DSRM cycle 1 and 2.

#### 3.1. DSRM Cycle 1: Design of Method for the Development of a BM Agnostic Ontology of Digital Platform Type

The first phase of our method was designed during the first DSRM cycle (upper box in Figure 2) and had the objective to develop a BM agnostic domain ontology covering a specific digital platform type.

In the first step of the first DSRM cycle, we identify the problem and argue why developing the method as intended could offer a solution. As discussed in the previous section, certain digital platform types (e.g., sharing platform) are overloaded, without a controlled vocabulary, standard terms, and consensus among players to interoperate, communicate, and share. Therefore, an ontology development method that formalizes the platform type requirements can support a shared consensus. The definition of a digital platform, and therefore the scope of this research, is explained in the previous section.

Second, we defined the objectives of our method. Our method should be easy to use and not time-consuming. Besides that, it should be possible for a modeler to include all constraints of the digital platform type she wants to model. At last, the modeler should be able to validate if existing digital platforms are instances of the type it modelled.



In a third step, we design the first phase of our method. Here we based our methodology on the ontology engineering method called ‘reuse by analogy’ as proposed by Ruy et al. [13] to reuse ontology patterns within the ontology modules of the original DPO. To reuse ontology patterns ‘by analogy’, the modeler looks for ontology patterns that describe knowledge related to the type of situation she is facing. Once a pattern is selected, she has to identify which concepts in the domain (in this case, the domain of the digital platform type), correspond to the concepts in the pattern, and reproduce the structure of the pattern in the domain ontology being developed [26]. Due to the modular design of the DPO, we experienced that this reuse of patterns is relatively easy. Because we do not want to restrict the flexibility of the modeler, we chose a clear demonstration instead of a fixed set of rules. The method is explained in Section 4.1.

In step four, we demonstrated the first phase of our method with the first part of our proof-of-concept, described in Section 4.2, by developing the business model agnostic sharing platform ontology as illustrated in a series of videos.

Eventually, in the last step, we evaluate this first phase on the basis of the objectives as described in step 2. The evaluation of usability and modeling efficiency is, at this point, out of scope but will be conducted with case studies in future research, as described in Section 7. In this paper, we verify with the first part of our proof-of-concept, the ‘sharing platform ontology’, that all DPO modules for sharing platforms are used. We also verify that the existing sharing platforms BlaBlaCar and Couchsurfing are indeed instances of the sharing platform type based on the sharing platform ontology. This evaluation is discussed in Section 4.3.

### 3.2. DSRM Cycle 2: Design of Method for the Development of an Organization-Specific Ontology

The second phase of our method, which is intended for developing organization-specific ontologies starting from a BM-agnostic domain ontology for a specific digital platform type (i.e., the first phase result), was designed during a second DSRM cycle (lower box in Figure 2).

In the first step of this second DSRM cycle, we identify the problem and our motivation for developing the method. There is a large variety of BMs under which a digital platform of a certain platform type can operate [27]. It is important to understand the influence of these BM variations on the design of the platform software, especially the required functionality and user roles [11]. Furthermore, this BM-specific knowledge needs to be communicated amongst the stakeholders to guarantee a shared understanding of the intended platform and its BM choices [7].

Second, we define the objectives of our method. Again, our method should be easy to use and not time-consuming. On top of that, the organization-specific ontology developed by our method should be capable of capturing the required functionality, user roles, and at least the ‘happy path’ of how the software enables platform users to interact with each other. This happy path is a necessity to develop a prototype and Minimum Viable Product (MVP) [28], one of the main goals of our overall research project.

In a third step, we design the second phase of our method. This phase continues to work on the ontology of a specific platform type and extends it by reusing ontology patterns by analogy from the ontology modules of the BM extended DPO [11]. It is important to notice that multiple organization-specific ontologies of one digital platform type can be developed starting from the same digital platform type ontology without going through the first phase of our method repeatedly. The second phase of our method is explained in Section 5.1.

In step four, we demonstrated the second phase with the second part of our proof-of-concept in Section 5.2 by developing the organization-specific ontologies of BlaBlaCar and Couchsurfing, as illustrated in a series of videos.

Eventually, in the last step, we evaluate our method on the basis of the objectives defined in step 2. The ease-of-use and efficiency of the method are, at this point, out of scope but will be evaluated with case studies in future research, as described in Section 7.

Nevertheless, in this paper, we evaluate whether our method is able to produce ontologies that capture the functionality and cover the user interactions that are expected from a sharing platform. We do so by means of validation by an ontology expert of the deliverables of the method, referring to the second part of our proof-of-concept (i.e., the organization-specific ontologies of BlaBlaCar and Couchsurfing). We acknowledge that this is only an indirect evaluation of the method because we validate the end-product produced by our method.

For this validation, we use a known ontology validation technique, Competency Questions (CQs) [29]. To validate the usability of the organization-specific ontologies for software development, we formulate these CQs as user stories, which is a requirement documentation technique used in software engineering practice. A user story is a simple narrative expressing some software functionality that is expected by users [30]. User stories are articulated in the form of ‘As a [role], I want [goal], so that [benefit]’, with [role] specifying a type of user, [goal] describing the (inter)actions that the user wants the software to support, and [benefit] motivating the expected functionality from the user’s standpoint. By filling in the parts between the brackets, the ontology expert was asked to develop user stories based on the organization-specific ontology. After, these user stories were checked for their usefulness by an experienced software developer and frequent sharing platform user. User stories judged to be useful are user stories that provide sufficient information to start prototyping the platform software. If the ontology expert can derive useful user stories from the ontologies, then we have an indication of the quality of the ontologies that are developed using our method, so indirectly we contribute evidence of the quality of the method itself. The validation via CQs is discussed in Section 5.3.

#### 4. Development of the Business Model Agnostic Ontology for a Sharing Platform

An important problem for academia and practitioners alike was the conceptual confusion in different types of digital platforms, with important ramifications for their expected functionality. Platform types, including ‘sharing platform’, ‘digital marketplace’, ‘on-demand platform’, and ‘multi-sided platform’, are umbrella concepts or buzz words, and there was no consensus on what they comprise [6,31]. These digital platform types have a lot in common but also have substantial differences in business model and platform functionality. To add to this confusion, existing platforms try to be under the umbrella of certain ‘hyped’ platform types such as sharing platforms, because of the positive symbolic value of sharing [4]. These problems were tackled in the digital platform ontology modules [10] by conceptualizing the similarities and differences between the platform types in a way that all platform stakeholders can understand and discuss each type separately. An overview of the digital platform types as subclasses from each other is given in Figure 3, and the relevant modules related to each platform type are captured in the attributes of the classes. Important for this proof-of-concept is that a digital marketplace is a multi-sided digital platform that allows registration and transactions between its users. A sharing platform is a decentralized, user-oriented, C2C digital marketplace of under-utilized goods.

An overview of all original DPO modules is given in Figure 4. *Market sides* [32] is the number of different groups of platform users enabled to interact with each other. *User affiliation* [33] refers to different ways that users (per group) are connected to the platform. Transactions are only applicable on multi-sided platforms between users of different sides, while investments are always part of a transaction. *Centralization* [21,34] is the way the users can connect to each other. This can be via a decentralized search by the users of one side or via centralized, automated matching by the platform software. *Participation* [27,35] indicates if the market that is intermediated by the platform is Business-to-Business (B2B), Business-to-Consumer (B2C), Consumer-to-Consumer (C2C), or Peer-to-Peer (P2P). The latter case holds when platform participants are considered as ‘equals’, where C2C is a specialization of P2P when users of at least two sides are only allowed to be private persons. The *offering orientation* [36] differentiates between product selling, result-oriented services, or user-oriented offerings, including the leasing, renting, sharing, and pooling of a product.

Other studies typically refer to user-oriented offerings as the offering of a tangible good for temporary access [6]. Finally, *immediate access* and *under-utilized capacity* are only relevant for user-oriented offerings. A digital platform offers immediate access [37,38] if the offering can be delivered immediately after the moment of ordering. Under-utilized [4] indicates the use excess capacity of the offered product. Inclusive properties can have more than one module for the same property and are visually positioned under each other (e.g., user affiliation can be both registration and non-transaction, but not simultaneously non-transaction and investment). Exclusive properties cannot have more than one module and are always positioned next to each other (e.g., offering orientation is either product, result, or user). If a module for a property excludes a module for another property or makes another property irrelevant, then this is indicated in grey.

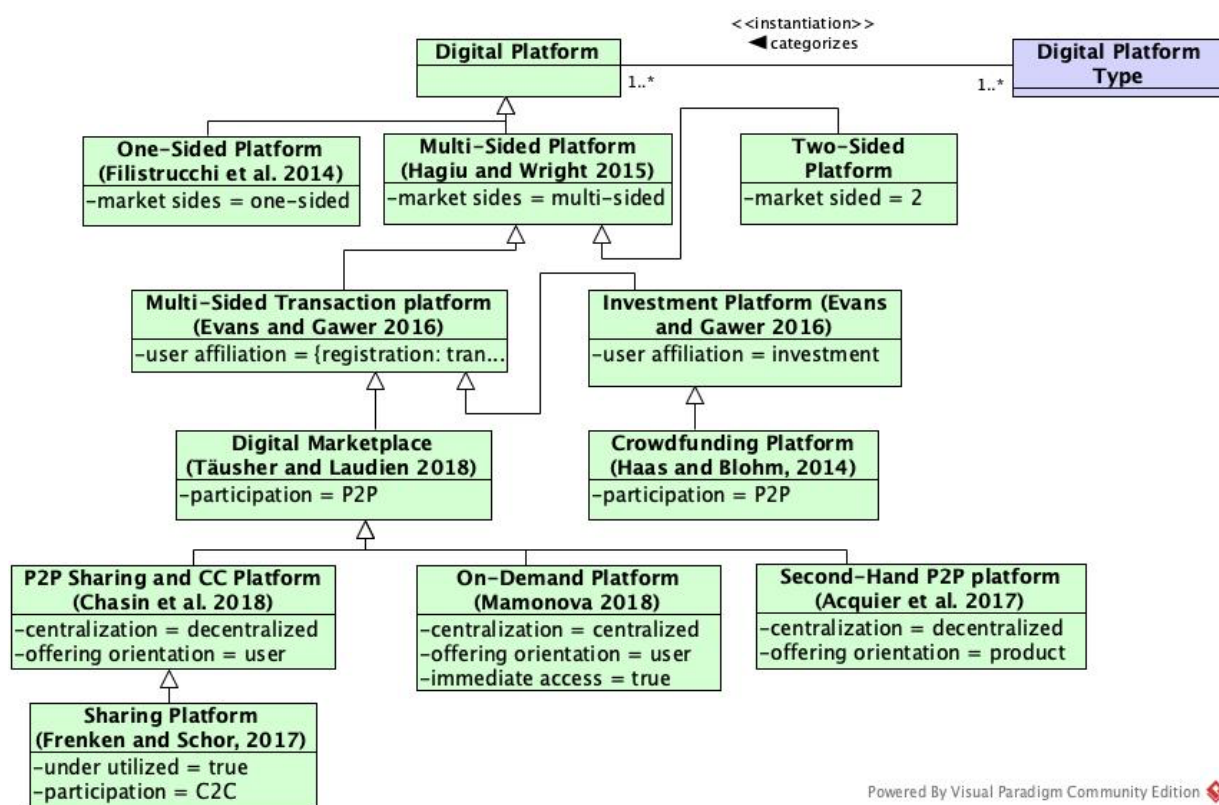


Figure 3. Digital platform types by [10].

To develop an ontology for sharing platforms as defined by Frenken and Schor [4] but initially independent of business model variations, we need to combine the ontology modules as displayed in the attributes of the sharing platform class (Figure 3). Know that this class also inherits the modules of all its parent classes. The sharing platforms defined by Frenken and Schor is a subset of the Peer-to-Peer (P2P) sharing and Collaborative Consumption (CC) platforms defined by Chasin et al. [39], which are again a subset of digital marketplaces as defined by Täuscher and Laudien [27] and so on. These relevant sharing platform modules are also highlighted in orange in Figure 4. In the remainder of this section, we will discuss one module, the multi-sided module, and provide a link with the other modules. After, we explain how to combine them into a business model agnostic sharing platform ontology.

Most businesses operate in a ‘traditional’ market and only support the direct interaction between a company and the customer. These businesses only favor direct network effects without enabling interactions between their customers [40]. A digital platform, on the other hand, enables interactions between the customers, who are referred to as ‘users’ of the platform. A market is multi-sided when the interactions are between the users of



different sides. In this case, users on one side are dependent on the number and quality of users of the other sides, and therefore multi-sided platforms also favor indirect network effects. In Figure 5, a multi-sided module is shown. It indicates that a multi-sided platform (offering) is always towards multiple sides of targeted users. Users of all sides are bound to the platform via a user affiliation agreement. This affiliation can be by registration, a transaction, and/or investment. The multi-sided platform allows for interactions between the users of at least two sides.

Property	Modules					
Market Sides	One-Sided	Multi-Sided				
User Affiliation	Registration					
	Non-transaction	Transaction				
		Investment				
Centralization		Decentralized				
		Centralized				
Participation		B2C				
		B2B				
		P2P			Non-P2P	
		C2C				
Offering orientation		Product	Result	User		
Immediate access					True	False
Under-utilized					True	False

Figure 4. Original DPO modules.

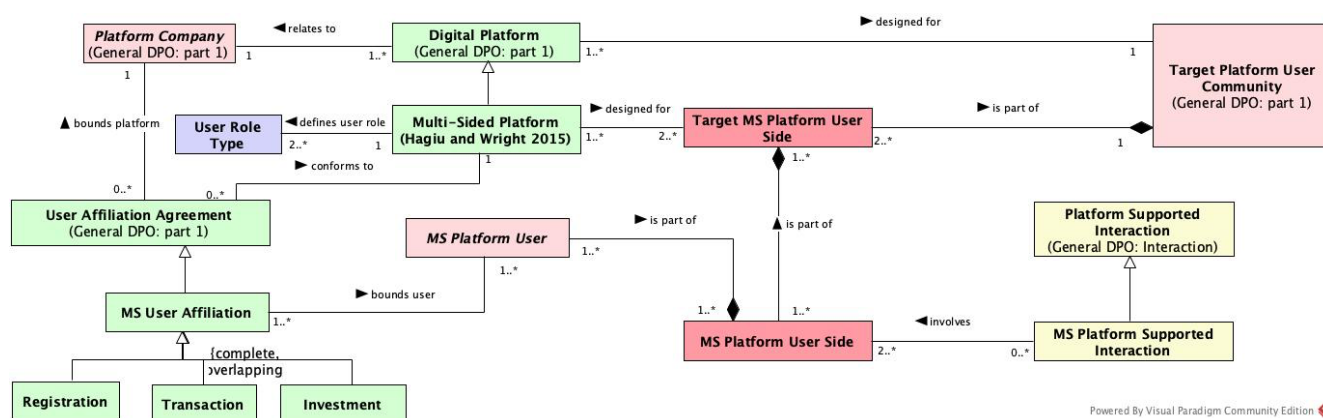


Figure 5. Multi-sided module in OntoUML.

#### 4.1. Method for the Development of a BM Agnostic Ontology of a Digital Platform Type

The first phase of our method on how to use the ontology modules, displayed in Figure 4, to develop a business model agnostic ontology for a specific digital platform type, is given below in a number of steps:

1. Download Visual Paradigm Community Edition, and install the OntoUML plugin (<https://github.com/OntoUML/ontouml-vp-plugin>, accessed on 1 December 2021)
2. Download the latest version of the extended DPO on <https://model-a-platform.com/ontology-versions/> (accessed on 17 November 2021) and open it with Visual Paradigm Community Edition (version 16.3, Visual paradigm, Hong Kong, China). All ontology modules (original and BM) can be visualized within the class diagrams.
3. Create a new class diagram for the platform type you want to model.
4. The classes of each module are gathered in a separate package visualized in the class repository of Visual Paradigm. These module packages are again grouped by property, as summarized in Figure 4. Go through each relevant module of the desired platform

type (e.g., the yellow modules in Figure 4 for sharing platform) from top to bottom, and drag and drop the classes into the newly created class diagram. Automatically, the colors indicating the UFO stereotype, relationships between the classes, and their multiplicities are visualized, with the module name of each class between brackets.

- Classes not relevant to the modeler's purpose can be deleted, and a subclass can be placed on top of the parent class in case the subclass further defines the relationships and constraints needed to model the envisioned shared platform, but the relationships of the parent class are still relevant for the modeler.

#### 4.2. Proof-of-Concept Part 1: Development of a Sharing Platform Ontology

In a first part of our proof-of-concept, we show and explain the development of the business model agnostic sharing platform ontology in Figure 6 in a series of videos (<https://model-a-platform.com/applying-the-ontology/>, accessed on 5 January 2022). The model shows that sharing platform visitors get to the website or mobile app and must perform a registration action before they can use the platform services. A registered user can create listings (for which this user becomes the offering creator). A platform visitor can perform listing searches (the user then becomes a target platform customer), after which a booking can be created between the users of both sides. The target platform customer that initiates the transaction (from now on referred to as 'booking', a transaction for services) then becomes a(n) (effective) platform customer, whereas the offering creator of the target listing becomes a platform provider. This booking can then be fulfilled by a service delivery to the platform customer. The delivery must involve a service and a physical good which is also used by the provider during or in between the service delivery(s) to comply with the under-utilization requirement. Both the provider and customer are private persons complying with the Customer-to-Customer (C2C) requirement.

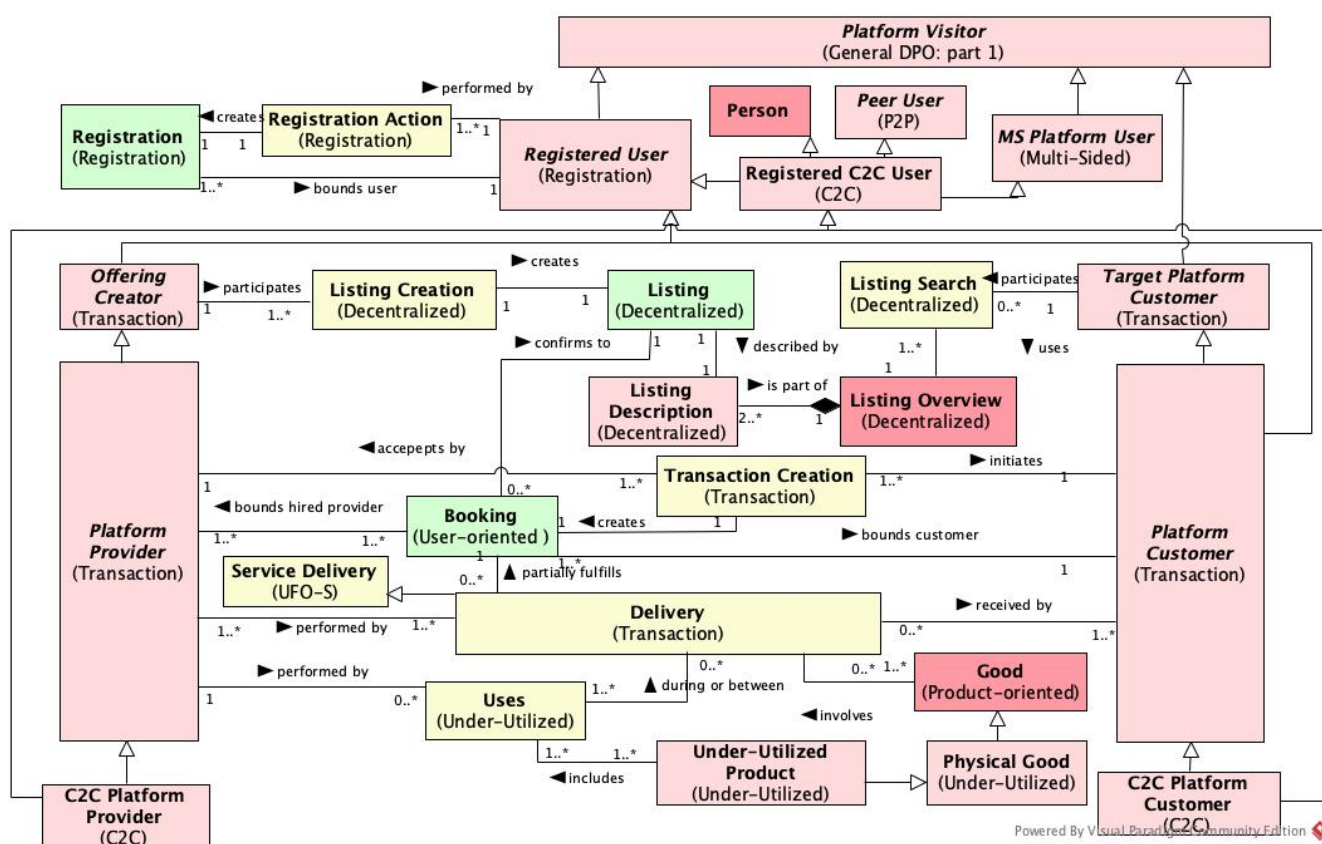


Figure 6. Business model agnostic sharing platform ontology based on DPO in OntoUML.

#### 4.3. Evaluation of the Sharing Platform Ontology

First, we validate if classes of all DPO modules for sharing platforms (colored orange in Figure 4) are used in Figure 6 (name of the module between brackets under the class name), which is a necessity to capture all constraints of the SE domain.

Second, we evaluate if both BlaBlaCar and Couchsurfing are instances of a sharing platform based on our model in Figure 6. BlaBlaCar is a ride-sharing application that intermediates between drivers and persons that want a ride. Couchsurfing, on the other hand, intermediates between homeowners and persons looking for a place to stay. For both platforms, users on both sides need to register. A customer can choose what ride or homestay he wants out of a list of offerings/listings. After the booking, there is a service delivery, including a physical good. In the case of BlaBlaCar, it is bringing the customer from A to B using a car. For Couchsurfing, it is renting out a room or bed for a number of nights. The model also explains why some existing platforms are sometimes wrongly defined within the SE, or at least following the definition of Frenken and Schor (2012). For example, for the ride-hailing app Uber the customer creates new capacity by ordering a car to drive the passenger from A to B. Without the order, the trip would not have happened in the first place, as the provider does not use the car during or between the service delivery. By contrast, in the case of BlaBlaCar, the customer occupies a seat that would otherwise not have been used as the driver had planned to go from A to B anyway. Another reason to place Uber outside of the SE is the centralized allocation of drivers to customers. It is not possible for a customer to search and choose the driver he or she wants from an overview of listings, as is the case for BlaBlaCar and other sharing platforms. Another example is Airbnb, which is partly operating in SE for their C2C rentals, but B2C rentals and accommodations that are not used by the owner during or between bookings are positioned outside the SE. Therefore, Airbnb cannot be fully described by the ontology shown in Figure 6.

#### 5. Development of Organisation-Specific Ontologies for BlaBlaCar and Couchsurfing

Besides the conceptual confusion in platform types, there is also a lack of knowledge concerning the user roles and the role-specific implications on the functionality of digital marketplaces (and its subtype sharing platforms) deploying different business models [21,41]. This confusion is partly due to the complex mechanism of value co-creation between users of different sides and the possible overlap between these roles. This issue is tackled by extending the Digital Platform Ontology (DPO), which now covers a wider variety of platform functionality as determined by specific business model choices captured in Business Model (BM) modules [11]. The proposed ontology modules in the extended DPO allow adjusting to a range of digital platform types and business model variations.

The BM modules are given in Figures 7 and 8, and to model a specific digital marketplace, at least one module for each property must be chosen (except the 'None' modules, which are empty). The modules in orange are mandatory for sharing platforms, as they are implied by the sharing platform definition (sharing platforms always offer offline services due to their user orientation, and require C2C transactions). In case the modules are not combinable within a single property, the module has an 'e' to indicate that an exclusive choice is required. All other modules are inclusive and allowed to be combined with modules of the same property. We also specify dependencies for modelling situations where the choice of modules is restrained by choice of values for other properties. Such dependencies are indicated by the thick boxes in Figures 7 and 8. For instance, only services can be specialized by their frequency (i.e., one-time or recurring). As another example, a listing with price discovery set by the provider or by the customer can only have a quantity-based or feature-based (or both combined) price calculation. In addition, revenue for the platform company in the form of a listing fee (cost for the registration of a new listing) always comes from the provider side. The 'other' revenue stream and revenue source groups all third parties, such as the income from advertising and service sales.

Property	Module				
Listing Type	Good Transfer			Service	
Listing Kind	Physical	Digital		Offline	Online
Listing Frequency	One-time <sup>e</sup>			Recurring <sup>e</sup>	
Price Discovery	None <sup>e</sup>	Set by Provider <sup>e</sup>	Set by Customer <sup>e</sup>		Set by Market <sup>e</sup>
Price Calculation	None <sup>e</sup>	By Quantity	By Feature		Auction <sup>e</sup> Quote <sup>e</sup>
Customer Type	Person			Organization	
Provider Type	Person			Organization	
Payment System	None <sup>e</sup>	Offline		In-house	External
Revenue Stream	Other	Subscription	Commission	Fixed Fee	Listing Fee
Revenue Source	Other	Customer		Provider	
Conversation System	Listing Conversation			Booking Conversation	
Review System	None <sup>e</sup>	By Customer		By Provider	

Figure 7. Extended DPO business model modules for BlaBlaCar (Note: <sup>e</sup> for exclusive modules).

Property	Module				
Listing Type	Good Transfer			Service	
Listing Kind	Physical	Digital		Offline	Online
Listing Frequency	One-time <sup>e</sup>			Recurring <sup>e</sup>	
Price Discovery	None <sup>e</sup>	Set by Provider <sup>e</sup>	Set by Customer <sup>e</sup>	Set by Market <sup>e</sup>	
Price Calculation	None <sup>e</sup>	By Quantity	By Feature	Auction <sup>e</sup>	Quote <sup>e</sup>
Customer Type	Person			Organization	
Provider Type	Person			Organization	
Payment System	None <sup>e</sup>	Offline	In-house	External	
Revenue Stream	Other	Subscription	Commission	Fixed Fee	Listing Fee
Revenue Source	Other	Customer		Provider	
Conversation System	Listing Conversation			Booking Conversation	
Review System	None <sup>e</sup>	By Customer		By Provider	

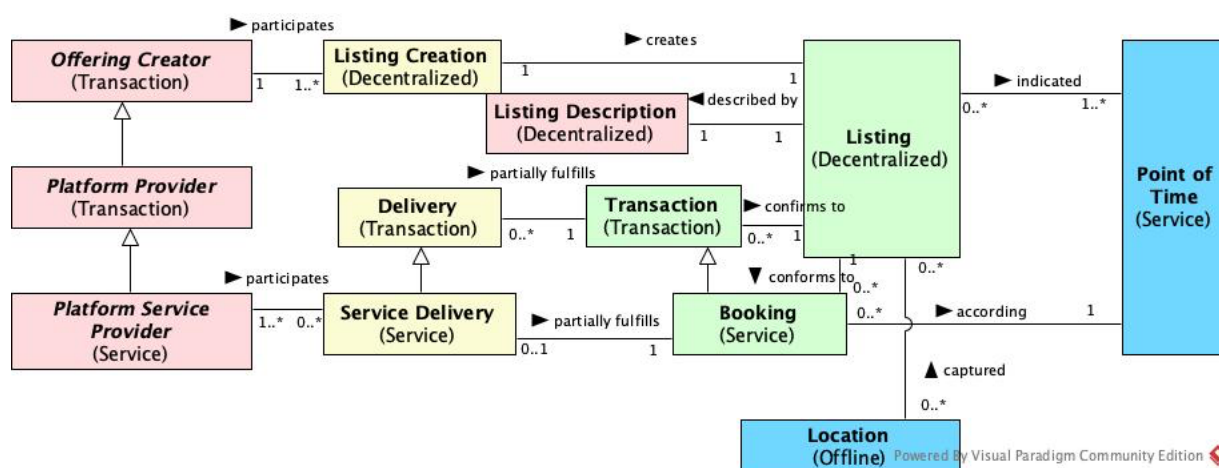
Figure 8. Extended DPO business model modules for Couchsurfing (Note: <sup>e</sup> for exclusive modules).

The ride-sharing application BlaBlaCar and the hospitality exchange service Couchsurfing are clearly operating in the SE, as discussed in the previous section, and due to their difference in BM, are ideal as a proof-of-concept to visualize the BM modules ontology application. These instances are mapped in blue in Figure 7 for BlaBlaCar and Figure 8 for Couchsurfing to indicate what modules are relevant to model an organization-specific ontology for each case.

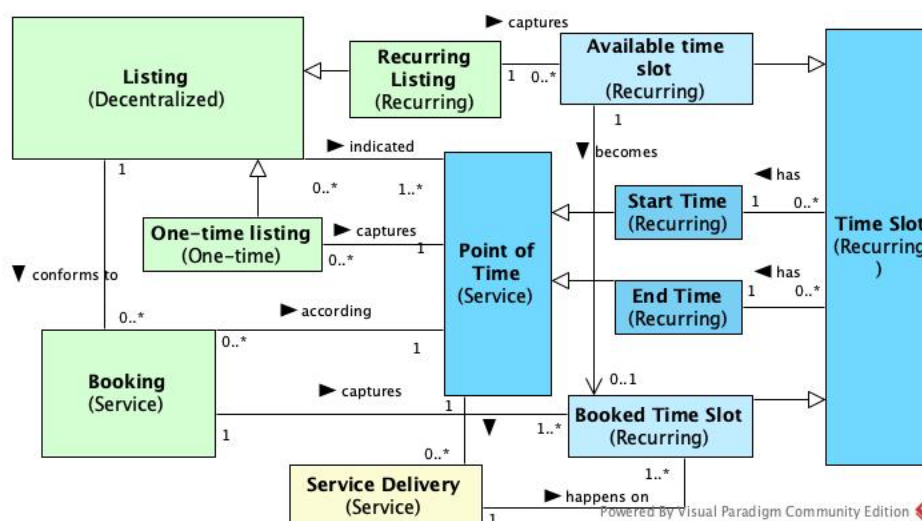
BlaBlaCar, on the one hand, intermediates between a driver (provider) and passengers (customers) for an offline service (ride-sharing), and the service only occurs once for each listing. The price is set by the provider based on quantity (number of persons). The payments are transferred by an external payment system, and the revenue stream is a commission of the booking price paid by the customer. The conversation system allows both conversations with messages before (listing conversation) and after (booking conversation) the booking. After the delivery of the service, both a review by the provider and by the customer towards each other are permitted.

Couchsurfing, on the other hand, intermediates between a homeowner (provider) and lodgers (customers) for an offline service (free stay), while the service can reoccur multiple times for one listing. There is no price setting, and the revenue is gathered using a subscription fee for all users by an external payment system. The conversation system allows both conversations with messages before (listing conversation) and after (booking conversation) the booking. After the delivery of the service, both a review by the provider and by the customer towards each other are permitted.

First, we will discuss four BM modules in more detail. Figure 9 show the modules for ‘Service’ and ‘Offline’. Figure 10 show the ‘One-time’ and ‘Recurring’ listing modules. The other BM modules can be visualized within the Visual Paradigm by opening the latest DPO version.



**Figure 9.** Offline service in OntoUML.



**Figure 10.** One-time and recurring listing in OntoUML.

When a customer creates a booking, a service needs to be delivered by the service provider, partially fulfilling this booking at a predefined point of time as indicated in the listing. When a service is offline, the location also needs to be captured in the listing and booking to specify where the service delivery will take place. Time and location are intrinsic qualities, structured values with multiple phases it can go through. The color coding of qualities is blue, following the usual conventions of OntoUML models [42].

A one-time listing can ultimately only result in one service delivery at a single, predefined point in time. This is the case for BlaBlaCar, where each listing can only result in one drive. A recurring listing, on the other hand, can have several available time slots, with



each time slot captured by the start time and end time. The period of booking is captured in a booked time slot and specifies the required time of the service delivery.

### 5.1. Method for Developing an Organization-Specific Ontology

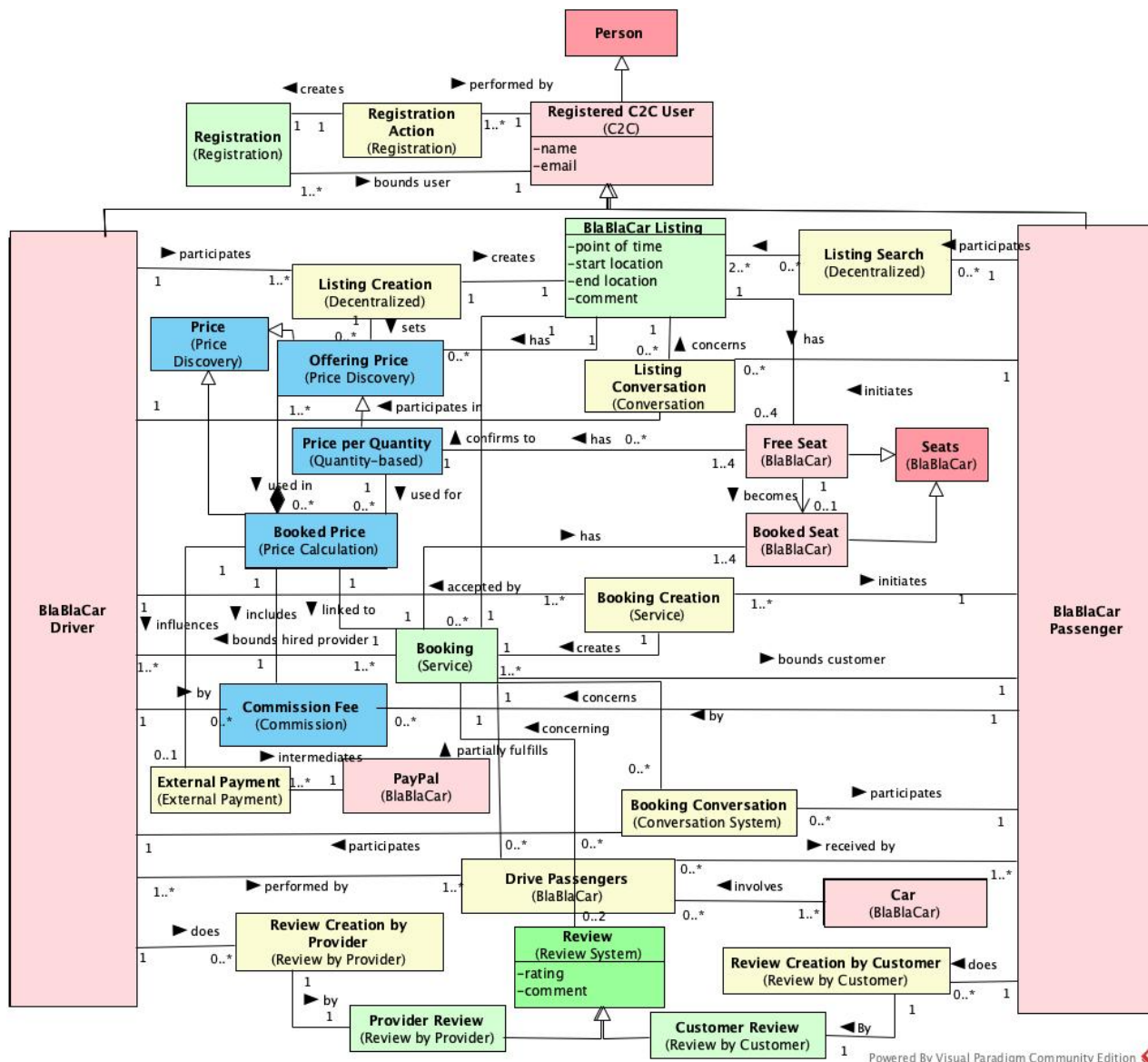
Phase two of our method on how to use the ontology modules, as displayed in Figures 7 and 8, to develop an organization-specific ontology for an envisioned or existing sharing platform, is given below in a number of steps:

1. Download Visual Paradigm Community Edition, and install the OntoUML plugin (<https://github.com/OntoUML/ontouml-vp-plugin>; accessed on 1 January 2021).
2. Download the latest version of the extended DPO on <https://model-a-platform.com/ontology-versions/> (accessed on 17 November 2021) and open it with Visual Paradigm Community Edition (version 16.3, Visual paradigm, Hong Kong, China). All ontology modules (original and BM) can be visualized within the class diagrams.
3. Create a new class diagram for your organization-specific ontology and copy and paste the BM agnostic sharing platform ontology (Figure 6) in this newly created class diagram.
4. The classes of each BM module are gathered in a separate package visualized in the class repository of Visual Paradigm. These BM module packages are again grouped within their property, as summarized in Figures 7 and 8. Go through each relevant module from top to bottom, and drag and drop the classes into the newly created class diagram. Automatically the color indicating the UFO stereotype, relationships between the classes, and their multiplicities are visualized, with the BM module name of each class between brackets.
5. Classes not relevant to the modeler's purpose can be deleted, and a subclass can be placed over a parent class when the subclass better defines the relationships and constraints needed to model the envisioned shared platform.
6. It is at the liberty of the modeler to create new classes and relationships to capture the platform-specific knowledge and required functionality of the platform he or she wants to model. The modeler has the freedom to tailor-make his model depending on its purpose, and after a newly created class is dropped into the organization-specific ontology, it is possible to select a suggested UFO stereotype to visualize the OntoUML color codes.
7. Attributes can be added to each class to visualize the platform-specific information it needs to capture. In addition, when a class is only connected to one other class with a 1 to 1 relationship, it is allowed to capture this class as an attribute instead to simplify the model.

### 5.2. Proof-of-Concept Part 2: An Organization-Specific Ontology for BlaBlaCar and Couchsurfing

In the second part of our proof-of-concept, we followed these steps using the relevant colored BM ontology modules in Figures 7 and 8 to develop two organization-specific ontologies, one for BlaBlaCar in Figure 11 and one for Couchsurfing in Figure 12. The full development of these models is again shown and explained in a series of videos (<https://model-a-platform.com/applying-the-business-model-ontology-modules/>, accessed on 2 January 2022). The organization-specific ontology of BlaBlaCar in Figure 11 can be read as follows: A person can perform a registration action, giving access to their name and email address to become a registered C2C user. This user can perform both a listing creation event (as a driver) and a listing search event (as a passenger). During the listing creation, the offering price per seat is set by the provider and captured in the listing. In addition, the number of free seats (maximum of 4), the point of time, the start location, end location, and comments are captured in the listing. During the listing search, a passenger can initiate a conversation concerning the listing with a potential driver. Subsequently, the passenger can create a booking for four available seats at most. The booked price is calculated based on the offering price and includes a commission. The other part is transferred via the external payment provider PayPal from the passenger to the driver. After the booking,

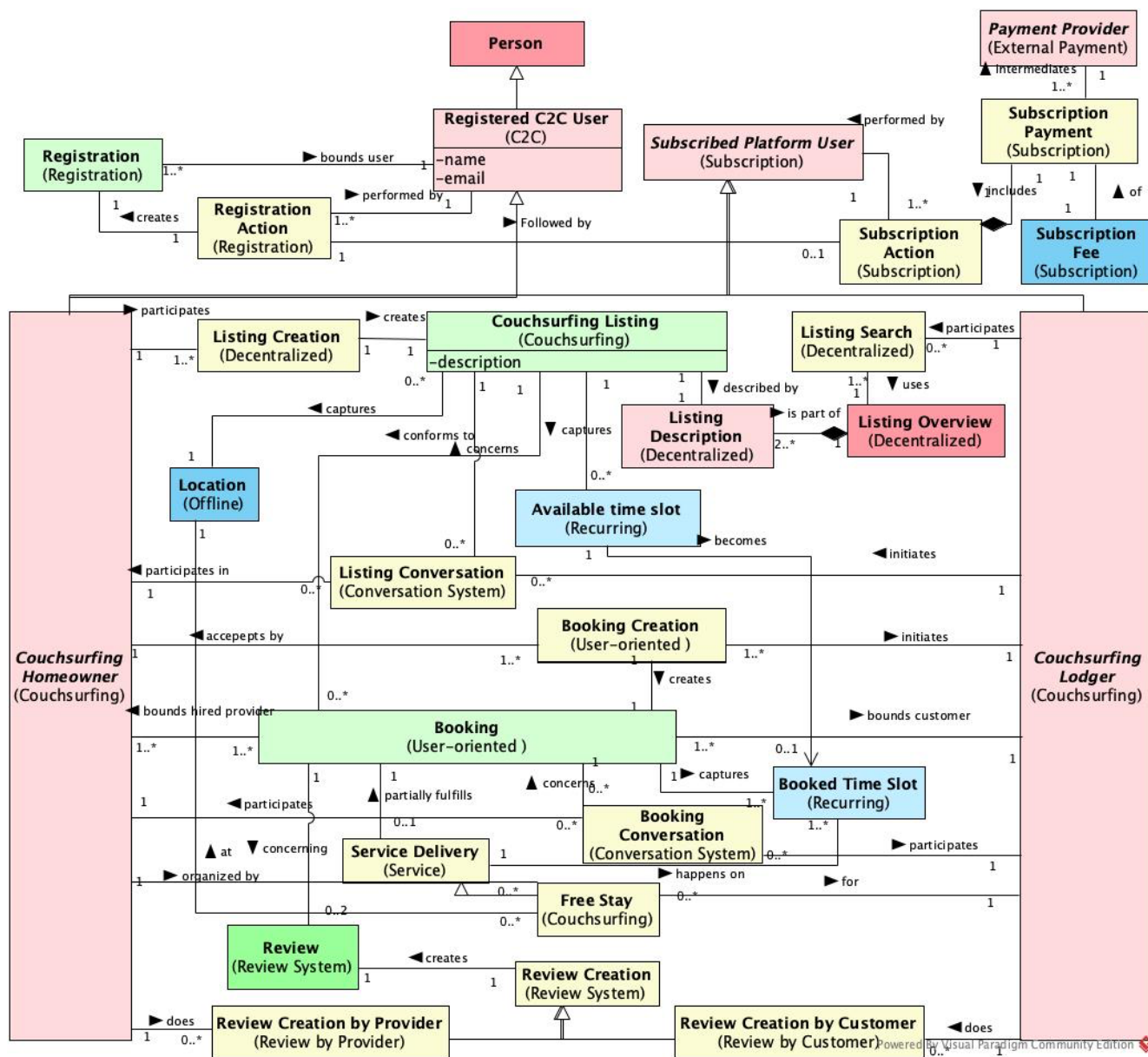
the passenger and driver can participate in a conversation concerning this booking and eventually, the driver drives the passengers in his car (partially), fulfilling the booking. After the drive, both users can create a review towards one another, including a rating and a comment.



**Figure 11.** Organization-specific ontology of BlaBlaCar in OntoUML.

The organization-specific ontology of Couchsurfing in Figure 12 can be read as follows: A person can perform a registration action, giving access to their name and email address followed by a subscription action to become a user. The subscription action includes the user paying a subscription fee intermediated by a payment provider. The user can perform both a listing creation event (as a homeowner) and a listing search event (as a lodger). During the listing creation, a description, location, and the available time slots are captured in the listing. During the listing search, the lodger can initiate a conversation concerning the listing with a potential homeowner. Subsequently, the lodger can initiate a booking that needs to be accepted by the homeowner, and the booked time slot is captured in the booking. After the booking, the lodger and homeowner can participate in a conversation concerning this booking and eventually, the lodger stays at the homeowner's house for free.

(partially), fulfilling the booking. After the stay, both users can create a review towards one another, including a rating and a comment.



**Figure 12.** Organization-specific ontology of Couchsurfing in OntoUML.

### 5.3. Evaluation of the Organization-Specific Ontologies

To validate the organization-specific ontologies and indirectly, the method by which these ontologies were developed, we make use of Competency Questions (CQs) in the form of user stories. The organization-specific ontologies of BlaBlaCar (Figure 11) and Couchsurfing (Figure 12) were given to the ontology expert, with more than 10 published UFO papers for which he created the user stories in the form of ‘As a [role], I want [goal], so that [benefit]’, given in Table 1. For instance, user story 9 of BlaBlaCar validates the need for pre-payment to simplify the payment and avoid disputes.

**Table 1.** User stories, derived from the organization-specific ontologies.

BlaBlaCar	
Nr	User Story
1	As a driver, I want to register so that I can create listings.
2	As a passenger, I want to register so that I can get a ride.
3	As a driver, I want to create a listing so that customers can find my listing.
4	As a passenger, I want to search for a listing so that I can find a ride.
5	As a passenger, I want to ask questions to the driver so that I have all the information I need for the trip.
6	As a driver, I want to talk to passengers so that I have more information and feel safer.
7	As a passenger, I want to book a seat so that I am certain I have a ride.
8	As a driver, I want to see who books a seat so that I know who I must pick up.
9	As a passenger, I want to pay the driver in an easy manner so that I do not need cash or have a dispute with the driver.
10	As a driver, I want to get paid so that I have some extra money.
11	As a passenger, I want to talk to the driver so that I can tell him I will be late or ask for the size of the boot.
12	As a driver, I want to talk to the passenger so that I can tell him about changes.
13	As a driver, I want to inform the passenger so that he/she knows the ride has started.
14	As a driver, I want to review the passengers so that it creates more trust between future users.
15	As a passenger, I want to review the driver so that it creates more trust in the community and ensures the driver drives slower next time.
16	As a passenger, I want to read the reviews so that I can make an informed decision.
1	As a homeowner, I want to register so that I can create listings and meet travelers.
2	As a user, I want to subscribe and make a payment of the subscription fee so that I can register and use the Couchsurfing services.
3	As a lodger, I want to search for listings so that I can book a couch/stay I can sleep on at a specific place at an available period.
4	As a homeowner, I want to create listings so that I eventually have guests in my house, meet new people, and create new friendships.
5	As a lodger, I want to talk to the homeowner so that I can tell him I will be late or ask for the size of the room.
6	As a homeowner, I want to talk to the lodger so that I can tell him about changes.
7	As a lodger, I want to book a stay for a certain period so that the homeowner is informed of my arrival.
8	As a homeowner, I want to accept bookings of an available period from lodgers I trust so that I do not have people in my house with bad intentions.
9	As a homeowner, I want to review the lodgers so that it creates more trust between future users.
10	As a lodger, I want to review the homeowner so that it creates more trust in the community and ensures the homeowner is eager to give me a warm welcome.
11	As a lodger, I want to read the reviews so that I can make an informed decision.

The usefulness of the user stories was evaluated by an experienced software developer with 6 years of software development experience at a Belgian software company. We verified that this software developer was familiar with the BlaBlaCar and Couchsurfing platforms and has experience with agile methods and user stories. All user stories created were considered useful, as they all require specific software platform functionality to be designed and programmed. Therefore, this evaluation indicates that our ontology development method supports developing organization-specific ontologies of sharing platforms from which requirements can be derived for creating platform software prototypes.

## 6. Discussion

The ultimate purpose of sharing platform organization-specific ontologies (such as the examples of BlaBlaCar and Couchsurfing presented in Figures 11 and 12) is to function as the conceptual design of the platform software. Using the DPO-based method illustrated in this paper, new ideas for sharing platforms and underlying business models can be explored, made explicit, and agreed upon by developing an organization-specific ontology, which can subsequently drive the development of the platform software. How to use organization-specific ontologies for the rapid development of platform prototypes is part of our future research, as will be explained in Section 7.



This purpose has motivated the design choice to focus our efforts first on the ‘happy path’ platform functionality, as captured by the DPO modules and business model ontology modules. Before the sharing platform can be put in operation, governance and regulatory mechanisms (e.g., for dispute resolution) need to be designed and integrated. Further extending the DPO will result in sharing platform ontologies that can capture additional knowledge of legal, social, and economic aspects. Some of this information is already included in the ontologies. For instance, these models already capture contracts and conditions by means of relators (green). Registration is an agreement between a user and the sharing platform company, a listing is a commitment of the provider towards its targeted customers, and a booking is a service agreement between two users of different sides (a provider and a customer). However, for the moment, governance and regulatory requirements are not (yet) included in the DPO (and therefore, these aspects are not present in the example ontologies shown in Figures 11 and 12). The modeler can add these elements during step 6 of the method. Further support for this in terms of DPO extensions is also part of our future research.

For positioning our contribution relative to related work on ontology development in the SE domain, we use the systematic literature review that was recently presented by Mohamad et al. [7]. Table 2 compares the organization-specific ontology developed by our method to alternative ontologies identified in this review.

First, our method was developed using a clear *research methodology*, the Design Science Research Methodology (DSRM) of Peffers et al. [12]; however, some of the ontologies in Table 2 were developed without a research method.

Second, many of the papers in Table 2 are not situated in the SE as defined by [4], or in the platform economy as defined by [16] because they do not discuss interactions between users (i.e., *inter-user interactions* in Table 2). This is an important condition, as reducing the complexity of the value co-creation mechanism between providers and consumers based on interactions, and the possible overlap between these roles is one of the main reasons why the use of a sharing platform ontology is important. Besides this sidenote, Mohamad et al. [7] encountered a number of issues for ontology development in the SE domain for which our method provides solutions.

An issue encountered by Mohamad et al. [7] is the complexity of the blooming IT-enabled *platforms* and the need for a uniform language that allows us to describe these emerging platforms. Our method partly solves this issue by enabling the modeler to develop a business model agnostic ontology of a specific platform type (in this paper, a sharing platform) that captures the terminology and scope of its domain. This is possible because the DPO captures the differences and similarities of platform types, making it possible to compare the research and knowledge of different types of digital platforms (including sharing platforms, on-demand platforms, digital marketplaces, multi-sided platforms, crowdfunding platforms, and other types of digital platforms). This business model agnostic ontology can then be used to validate if an existing or envisioned digital platform belongs to a certain type.

Another issue mentioned was the lack of focus on perdurants, better known as events in ontology design [7]. *Objects* are entities present wholly in each time of existence (i.e., endurants), while *events* are entities that only exist in parts at a time [17]. Objects can be either physical (e.g., car, people, house) or abstract (e.g., an organization, a user, a booking) and are visualized in OntoUML with red and green classes. Events, on the other hand, include processes and actions (e.g., a payment action, a registration action) and are visualized in OntoUML with yellow classes. “Previous ontologies are mainly focused on endurants, and very little attention is given to the perdurants to the point that it is simply neglected. Yet, ontology development for the SE domain needs to consider the importance of perdurants, as this can be considered the SE domain backbone” [7]. The DPO, and the method to develop an organization-specific ontology in this paper combine both objects and events, visualizing the types of users, what actions each user can perform, and with what result. This highly improves the quality and usability of our models in comparison to previous ones.



**Table 2.** Review result on ontology development in the SE domain by Mohamad et al. [7].

Type of Ontology	Paper	Language	Research Methodology	Inter-User Interactions	Platforms	Objects	Events	Framework	BM Specific
Domain ontology: Sharing economy description language	The Sharing Economy Meets the Semantic Web: An Ontology for the Matchmaking of Peers (Von Hoffen, 2017) [43]	RDF, OWL	DSRM	X	-	X	X	-	-
Application ontology: discover2share	Designing an Ontology-based Web Directory for the Discovery of Sharing and Collaborative Consumption Platforms (Von Hoffen et al., 2015) [44]	REF, OWL	-	X	-	X	-	-	-
Domain ontology: Digital marketplace	Domain Ontology for Digital Marketplaces (Derave et al., 2019) [45]	OntoUML	[46]	X	-	X	X	-	-
Application ontology: CrowdStrom	CrowdStrom: Analysis, Design, and Implementation of Processes for a Peer-to-Peer Service for Electric Vehicle Charging (Martin et al., 2018) [47]	-	DSRM	-	-	X	-	-	-
Application ontology: Smart bike-sharing system (SBSS)	Ontology-based multi-Agent smart bike-sharing system (SBSS) (Patel et al., 2018) [48]	RDF	-	-	-	X	-	-	-
Application ontology: CIK-Net	An Ontology-based Collaborative Inter-Organizational Knowledge Management Network (CIK-NET) (Leung and Lau, 2021) [49]	-	-	-	-	X	-	-	-
Application ontology: Geolocation analysis	A geolocation analytics-driven ontology for short-term leases: Inferring current sharing economy trends (Alexandridis et al., 2020) [50]	-	-	-	-	X	-	-	-
Organization-specific ontology	Sharing Platform Ontology Development: Proof-of-Concept (i.e., this paper)	OntoUML	DSRM	X	X	X	X	X	X

The last issue is the absence of a “clear *framework* to assist the ontology development in the SE domain. There is no existing ontology that can be reused to avoid developing SE ontology from scratch” [7]. The proof-of-concept in this paper demonstrates the method to develop an organization-specific ontology for sharing platforms. It can capture the versatility in the SE domain by reusing the classes of relevant modules, such as Lego bricks, to build up a tailor-made ontology depending on the needs of the modeler. Further, we give access to the extended DPO to enable the development of organization-specific ontologies of different platform types with a user-friendly method using Visual Paradigm. The extended DPO is aligned with UFO using the OntoUML plugin and represents the digital platform domain at one moment in time. Obviously, there is a continuous evolution and innovation in this domain, resulting in the appearance of new platform types. The modular structure of the extended DPO increases the flexibility of the modeler and allows for further extensions that can describe these new digital platform types and BMs (To support ontology versioning, we developed <https://model-a-platform.com/ontology-versions/>, accessed on 17 November 2021). In addition, an organization-specific ontology needs to be updated regularly during software development to make sure the model captures the intended design and functionality as good as possible. By modifying each design choice, the organization-specific ontology can also be used as a documentation artifact of what has been accomplished and in what direction the project is going.

Besides the issues of Mohamad et al. [7], our method also contributes to the SE domain by separating the development of a BM agnostic ontology covering a specific platform type and the development of a BM specific organization-specific ontology. This way, both BM independent and *BM specific* knowledge is captured, but the distinction is still clearly visible as each class can be linked to their module origin.

## 7. Limitations and Future Research

Besides our contributions, there are also a number of limitations. A first limitation is that an organization-specific ontology is dependent on the purpose of the modeler, and different organization-specific ontologies can exist for the same digital platform. An organization-specific ontology for regulatory purposes is different from an organization-specific ontology for software development purposes and will include other terminology, classes, and relationships. As discussed in Section 6, our models do not (yet) include unhappy paths or conflicts between users (e.g., car damage by a customer during a BlaBlaCar drive; a provider that does not comply with the expectations as defined in the listing for Couchsurfing). In future versions of the DPO, we plan to capture the possible influence of regulation, as a ‘one size fits all’ policy and regulatory approach is inappropriate [51]. Reusing ontology patterns of UFO-L, a core ontology that accounts for a comprehensive set of related legal notions, including rights and duties, no-rights and permissions, powers and liabilities, disabilities and immunities, as well as liberties [52], can be an aide in this matter. This way, stakeholders with an economic, legal, and IT background can easily communicate, which improves decision-making and hopefully drives the SE into a more sustainable direction. The complexity of SE platforms keeps increasing, triggering the need for well-developed ontologies.

It is important to notice that in addition to an organization-specific ontology, the modeler still needs to perform a market analysis and develop a business plan using frameworks such as the SWOT analysis and Porters five forces model. During this stage, an ontology can already help in the conceptualization of the idea, but only during the development of the software will the ontology reach its final formation.

In future research, we plan to develop a method for ontology-driven development of digital platform software. In an organization-specific ontology, classes and relators portray the required data structure for the sharing platform software, while events portray the required functionality. Pergl, Sales and Rybala [53] already describe the transformation of an ontological model into an implementation model. Other literature focuses on ontology-driven relational database development [53–56]. However, what it lacks is a method

for prototype deployment, which is a necessity to launch a Minimum Viable Product (MVP) fast and efficient [28]. An ontology is not only the collection of concepts, terms, constraints and relationships but also the formal, explicit, conceptual model of object ranges in a computational representation [57]. An organization-specific ontology can capture the required functionality for each user of a platform, and transformations between the organization-specific ontology and the code can be used to construct the final software [54].

Software development is known as a complex activity that is highly sensitive to human interaction and teamwork [57]. Therefore, developers adapted and started to use an agile approach with fast feedback, more client-focused, continuous improvements, and cross-functional teams. Besides the advantages of agile methods, there are also a considerable number of downsides. This includes a limited amount of documentation, fragmented output as teams work on different user stories and no clearly described finite end of the project [58]. Another downside of multiple teams working on different user stories is the widespread use of overlapping terminology and conflicting constraints for the components, user roles, and functionality of the intended software. An organization-specific ontology can document software development in an easy, structural, and flexible manner. It can be used to update the requirements throughout the development process using consistent terminology. Besides documentation, it is also possible to write user stories based on the ontology and include these users stories in a Kanban board to support agile thinking, visualize the progress of the project, and improve teamwork. We plan to investigate how an organization-specific ontology can support the development of a diverse set of digital platform prototypes operating different business models. A test case will be set up with aspiring entrepreneurs who plan to develop a prototype of their platform idea based on an organization-specific ontology they modeled using the method proposed in this paper. These organization-specific ontologies can then be evaluated using several criteria, including modeling time, change in perceived complexity, change in requirements quality, perceived flexibility of the ontology development, and change in the shared understanding of the terminology and functionality. On top of that, we will also evaluate the efficiency and perceived usefulness of the ontology-based prototype development process.

## 8. Conclusions

An ontology can be seen as a common language all stakeholders can use to improve communication and software development. This paper presents a method developed using the Design Science Research Methodology (DSRM) and demonstrated with a proof-of-concept on how to develop two types of ontologies that both have their use in the Sharing Economy (SE) domain. First, we propose a method for the development of an ontology of a digital platform type using the Digital Platform Ontology (DPO) of [10] in Section 4 and demonstrate it within the SE domain with the construction of a ‘sharing platform’ ontology independent of business model choices. The development of an ontology for different platform types can improve the communication of knowledge, assist in the understanding of the differences in platform types (e.g., sharing platform, digital marketplace, on-demand platform, crowdfunding platform), and help academia to set the scope of their research in the platform economy domain. Second, we also propose a method for the development of an organization-specific ontology of a digital platform instance using the Business Model (BM) ontology modules of [11] and demonstrate it by further adjusting the sharing platform ontology to develop the organization-specific ontology for two existing sharing platforms (‘BlaBlacar’ and ‘Couchsurfing’) operating different business models in Section 5. By dividing the ontology development method into two parts, we guarantee the separation of concerns of the platform type knowledge in the original DPO on the one hand and the BM knowledge on the other. Both the original DPO and the BM extension are based on UFO [17], a high-level ontology that provides us with basic concepts for objects, events, social elements, and relations, and are modelled in OntoUML [18], an ontology-driven conceptual modelling language capable of representing these concepts. All three ontologies are developed in Visual Paradigm by simply dragging and dropping classes of the relevant

modules, and the relationships and multiplicity constraints are automatically visualized. Within the organization-specific ontology, the modeler can create case-specific classes, redefine relationships, and limit constraints to tailor-make the model to his or her needs. An organization-specific ontology can be used for ontology-driven software development, describing the intended software by specifying various aspects of the system ranging from the requirements, functionality, and data of the system to the architecture, design, and deployment. The modular design makes it possible to analyze the influence of BM decisions on the organization-specific ontology, improving the flexibility, decision-making, and adaptability during the development of the intended software.

Sharing platforms still have the capacity to influence sustainable goals, as consumers can get cheap access to goods by renting or lending them from others, becoming less dependent on ownership [59]. BlaBlaCar, for example, facilitates carpooling, which decreases the need of users to buy a car and lowers the number of cars on the road. Sharing platforms such as Couchsurfing also have substantial social benefits, facilitating strangers to meet face-to-face, increasing meaningful contacts and social mixing [4]. Despite good prospects, the SE has not yet reached its full potential. Lowering the barrier of sharing platform development is therefore vital, as many digital platforms have the tendency to apply a ‘winner-takes-all’ strategy to create a monopoly. An essential element that creates incentives to enter and isolate the influence of competitors is increasing the differentiation of digital platforms. This way, network effects are mitigated, and divide-and-conquer strategies are less effective, which reduces the monopolization problem at the same time [33]. Due to the high complexity of the software design, in combination with the high costs and time needed to develop digital platform software [60], competitors with less diversification but a superior technology are still capable of monopolizing a market [33]. We believe that our ontology can accelerate the development of smaller, more alternative, and socially responsible sharing platforms and can thus contribute to the creation of a more socially responsible SE. Besides this, our ontology could also be used for regulatory purposes to retain monopolistic platform growth as it improves decision-making transparency.

**Author Contributions:** Conceptualization, T.D., F.G., T.P.S. and G.P.; methodology, T.D., F.G., T.P.S. and G.P.; software, T.D.; validation, T.D., F.G., T.P.S. and G.P.; formal analysis, T.D., F.G., T.P.S. and G.P.; investigation, T.D., F.G., T.P.S. and G.P.; resources, T.D., F.G., T.P.S. and G.P.; data curation, T.D., F.G., T.P.S. and G.P.; writing—original draft preparation, T.D., F.G. and G.P.; writing—review and editing, T.D., F.G. and G.P.; visualization, T.D., F.G., T.P.S. and G.P.; supervision, F.G., T.P.S. and G.P.; project administration, T.D., F.G., T.P.S. and G.P.; funding acquisition, F.G. and G.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Informed consent was obtained from all subjects involved in the study.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Kenney, M.; Zysman, J. The rise of the platform economy. *Issues Sci. Technol.* **2016**, *32*, 61–69.
2. Ranjbari, M.; Morales-Alonso, G.; Carrasco-Gallego, R.; Grijalvo, M. Business Model Innovation in Sharing Economy: A Benchmark Approach. In Proceedings of the 2nd International Conference on New Business Models—Exploring a Changing View on Organizing Value Creation: Developing New Business Models, Graz, Austria, 20–23 June 2017; Volume 2511, pp. 201–212.
3. Mi, Z.; Coffman, D.M. The sharing economy promotes sustainable societies. *Nat. Commun.* **2019**, *10*, 5–7. [CrossRef]
4. Frenken, K.; Schor, J. Putting the sharing economy into perspective. *Environ. Innov. Soc. Transit.* **2017**, *23*, 3–10. [CrossRef]
5. Zervas, G.; Proserpio, D.; Byers, J.W. The Rise of the Sharing Economy: Estimating the Impact of Airbnb on the Hotel Industry. *J. Mark. Res.* **2017**, *54*, 687–705. [CrossRef]
6. Curtis, S.K.; Lehner, M. Defining the sharing economy for sustainability. *Sustainability* **2019**, *11*, 567. [CrossRef]
7. Mohamad, U.H.; Ahmad, M.N.; Zakaria, A.M.U. Ontologies application in the sharing economy domain: A systematic review. *Online Inf. Rev.* 2021; ahead-of-print.
8. Gruber, T.R. A translation approach to portable ontology specifications. *Knowl. Acquis.* **1993**, *5*, 199–220. [CrossRef]

9. Martin, C.J. The sharing economy: A pathway to sustainability or a new nightmarish form of neoliberalism? *Ecol. Econ.* **2015**, *121*, 149–159. [\[CrossRef\]](#)
10. Derave, T.; Sales, P.T.; Gailly, F.; Poels, G. Comparing Digital Platform Types in the Platform Economy. In *Caise 2021: Advanced Information Systems Engineering*; La Rosa, M., Sadiq, S., Teniente, E., Eds.; Springer: Cham, Switzerland, 2021; Volume 12751, pp. 417–431.
11. Derave, T.; Sales, T.P.; Gailly, F.; Poels, G. Understanding Digital Marketplace Business Models: An Ontology Approach. In Proceedings of the 14th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modelling, Riga, Latvia, 24–26 November 2021; pp. 1–12.
12. Peffers, K.; Tuunanen, T.; Rotherberger, M.A.; Chatterjee, S. A Design Science Research Methodology for Information Systems Research. *J. Manag. Inf. Syst.* **2008**, *24*, 45–78. [\[CrossRef\]](#)
13. Ruy, F.B.; Reginato, C.C.; Santos, V.A.; Falbo, R.A.; Guizzardi, G. Ontology engineering by combining ontology patterns. In *Conceptual Modeling*; Johannesson, P., Lee, M., Liddle, S., Opdahl, A., Pastor Lopez, O., Eds.; Springer: Cham, Switzerland, 2015; Volume 9381, pp. 173–186.
14. Dermeval, D.; Vilela, J.; Bittencourt, I.I.; Castro, J.; Isotani, S.; Brito, P.; Silva, A. Applications of ontologies in requirements engineering: A systematic review of the literature. *Requir. Eng.* **2016**, *21*, 405–437. [\[CrossRef\]](#)
15. Kenney, M.; Zysman, J. Choosing a future in platform economy: The Implications and Consequences of Digital Platforms. *J. Chem. Inf. Model.* **2013**, *53*, 1689–1699.
16. Derave, T.; Sales, T.P.; Gailly, F.; Poels, G. Towards a Reference Ontology for Digital Platforms. In Proceedings of the 38th International Conference on Conceptual Modeling, Bahia, Brazil, 4–7 November 2019; pp. 1–14.
17. Guizzardi, G. Ontological Foundations for Structural Conceptual Models. Ph.D. Thesis, University of Twente, Enschede, The Netherlands, 27 October 2005.
18. Guizzardi, G.; Fonseca, C.M.; Benevides, A.B.; Almeida, J.P.A.; Porello, D.; Sales, T.P. Endurant types in ontology-driven conceptual modeling: Towards ontoUML 2.0. In *Conceptual Modeling*; Trujillo, J., Ed.; Springer: Cham, Switzerland, 2018; Volume 11157, pp. 136–150.
19. Carvalho, V.A.; Almeida, J.P.A. Toward a well-founded theory for multi-level conceptual modeling. *Softw. Syst. Model.* **2018**, *17*, 205–231. [\[CrossRef\]](#)
20. Benevides, A.B.; Bourguet, J.R.; Guizzardi, G.; Peñaloza, R.; Almeida, J.P.A. Representing a reference foundational ontology of events in SROIQ. *Appl. Ontol.* **2019**, *14*, 293–334. [\[CrossRef\]](#)
21. Sutherland, W.; Jarrahi, M.H. The sharing economy and digital platforms: A review and research agenda. *Int. J. Inf. Manag.* **2018**, *43*, 328–341. [\[CrossRef\]](#)
22. Wieringa, R. What Is a Platform. Available online: <https://www.thevalueengineers.nl/what-is-a-platform/> (accessed on 28 January 2022).
23. Nardi, J.C.; de Almeida Falbo, R.; Almeida, J.P.A.; Guizzardi, G.; Ferreira Pires, L.; van Sinderen, J.M.; Guarino, N.; Morais Fonseca, C. A commitment-based reference ontology for services. *Inf. Syst.* **2015**, *54*, 263–288. [\[CrossRef\]](#)
24. Derave, T.; Gailly, F.; Sales, T.P. A Taxonomy and Ontology for Digital Platforms. *Inf. Syst.* **2020**, 1–58.
25. Bezerra, C.; Freitas, F.; Santana, F. Evaluating ontologies with Competency Questions. In Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), Atlanta, GA, USA, 17–20 November 2013; Volume 3, pp. 284–285.
26. Falbo, R.A.; Guizzardi, G.; Gangemi, A.; Presutti, V. Ontology patterns: Clarifying concepts and terminology. In Proceedings of the 4th Workshop on Ontology and Semantic Web Patterns, Sydney, Australia, 21 October 2013; Volume 1188.
27. Täuscher, K.; Laudien, S.M. Understanding platform business models: A mixed methods study of marketplaces. *Eur. Manag. J.* **2018**, *36*, 319–329. [\[CrossRef\]](#)
28. Ries, E. *The Lean Startup*; Currency: Sydney, Australia, 2011.
29. De Falbo, R.A. SABiO: Systematic approach for building ontologies. In Proceedings of the 1st Joint Workshop ONTO.COM/ODISE on Ontologies in Conceptual Modeling and Information Systems Engineering, Rio de Janeiro, Brazil, 21 September 2014; Volume 1301.
30. Murtazina, M.; Avdeenko, T.V. The ontology-driven approach to support the requirements engineering process in scrum framework. In Proceedings of the International Conference Information Technology and Nanotechnology, Samara, Russia, 24–27 April 2018; Volume 2212, pp. 287–295.
31. Trabucchi, D.; Buganza, T. Fostering digital platform innovation: From two to multi-sided platforms. *Creat. Innov. Manag.* **2019**, *29*, 345–358. [\[CrossRef\]](#)
32. Hagi, A.; Wright, J. Multi-Sided Platforms. *Int. J. Ind. Organ.* **2015**, *43*, 162–174. [\[CrossRef\]](#)
33. Sanchez-Cartas, J.M.; Leon, G. Multi-sided Platforms and Markets: A Literature Review. *SSRN Electron. J.* **2019**, 1–62.
34. Acquier, A.; Carbone, V.; Massé, D. How to Create Value(s) in the Sharing Economy: Business Models, Scalability, and Sustainability. *Tehmol. Innov. Manag. Rev.* **2019**, *9*, 5–25. [\[CrossRef\]](#)
35. Ehikioya, S.A. A formal model of peer-to-peer digital product marketplace. *Int. J. Netw. Distrib. Comput.* **2018**, *6*, 143–154.
36. Ritter, M.; Schanz, H. The sharing economy: A comprehensive business model framework. *J. Clean. Prod.* **2019**, *213*, 320–331. [\[CrossRef\]](#)



37. Andersson, M.; Hjalmarsson, A.; Avital, M. Peer-to-peer service sharing platforms: Driving share and share alike on a mass-scale. In Proceedings of the 34th International Conference of Information Systems, Milan, Italy, 15–18 December 2013; Baskerville, R., Chau, M., Eds.; Association for Information Systems. AIS Electronic Library (AISeL): Atlanta, GA, USA; Volume 4, pp. 2964–2978.
38. Gobble, M.A.M. Defining the sharing economy. *Res. Technol. Manag.* **2017**, *60*, 59–61. [\[CrossRef\]](#)
39. Chasin, F.; von Hoffen, M.; Cramer, M.; Matzner, M. Peer-to-peer sharing and collaborative consumption platforms: A taxonomy and a reproducible analysis. *Inf. Syst. E-Bus. Manag.* **2018**, *16*, 293–325. [\[CrossRef\]](#)
40. Filistrucchi, L.; Geradin, D.; Van Damme, E.; Affeldt, P. Market definition in two-sided markets: Theory and practice. *J. Compet. Law Econ.* **2014**, *10*, 293–339. [\[CrossRef\]](#)
41. Hermes, S.; Maier, M.; Hein, A.; Böhm, M.; Krcmar, H. User Roles on Peer-to-Peer Sharing Platforms: A Critical Review of the Literature and Recommended Remedies. In Proceedings of the 53rd Hawaii International Conference on System Sciences, Honolulu, HI, USA, 7–10 January 2020; pp. 804–813.
42. Marek Suchánek: OntoUML Specification Documentation. 2020. Available online: [https://ontouml.readthedocs.io/\\_/downloads/en/latest/pdf/](https://ontouml.readthedocs.io/_/downloads/en/latest/pdf/) (accessed on 28 January 2022).
43. Von Hoffen, M. The Sharing Economy Meets the Semantic Web: An Ontology for the Matchmaking of Peers. In Proceedings of the 2017 IEEE 11th International Conference on Semantic Computing (ICSC), San Diego, CA, USA, 30 January–1 February 2017; pp. 213–219.
44. Von Hoffen, M.; Matzner, M.; Chasin, F. Designing an Ontology-Based Web Directory for the Discovery of Sharing and Collaborative Consumption Platforms. In Proceedings of the 2015 IEEE 17th Conference on Business Informatics, Lisbon, Portugal, 13–16 July 2015; Volume 1, pp. 108–115.
45. Derave, T.; Sales, T.P.; Verdonck, M.; Gailly, F.; Poels, G. Domain Ontology for Digital Marketplaces. In *Advances in Conceptual Modeling*; Guizzardi, G., Gailly, F., Suzana Pitangueira Maciel, R., Eds.; Springer: Cham, Switzerland, 2019; Volume 11787, pp. 191–200.
46. Ruy, F.B.; Guizzardi, G.; Falbo, R.A.; Reginato, C.C.; Santos, V.A. From reference ontologies to ontology patterns and back. *Data Knowl. Eng.* **2017**, *109*, 41–69. [\[CrossRef\]](#)
47. Martin, M.; Florian, P.; Jan, H.B.; Friedrich, C.; von Moritz, H.; Matthias, L.; Sarah, P.; Jorg, B. CrowdStrom: Analysis, Design, and Implementation of Processes for a Peer-to-Peer Service for Electric Vehicle Charging. In *Business Process Management Cases*; vom Brocke, J., Mendling, J., Eds.; Springer: Cham, Switzerland, 2018.
48. Patel, A.S.; Ojha, M.; Rani, M.; Khare, A.; Vyas, O.P.; Vyas, R. Ontology-based multi-Agent smart bike sharing system (SBSS). In Proceedings of the 2018 IEEE International Conference on Smart Computing (SMARTCOMP), Taormina, Italy, 18–20 June 2018; pp. 417–422.
49. Leung, N.K.Y.; Lau, S.K. An Ontology-based Collaborative Inter-Organizational Knowledge Management Network (CIK-NET). In Proceedings of the 23rd Australasian Conference on Information Systems (ACIS), Geelong, Australia, 3–5 December 2012.
50. Alexandridis, G.; Voutos, Y.; Mylonas, P.; Caridakis, G. A geolocation analytics-driven ontology for short-term leases: Inferring current sharing economy trends. *Algorithms* **2020**, *13*, 59. [\[CrossRef\]](#)
51. Codagnone, C.; Biagi, F.; Abadie, F. *The Passions and the Interests: Unpacking the “Sharing Economy”*; Publications Office of the European Union: Luxembourg, 2016.
52. Griffo, C.; Almeida, J.P.A.; Guizzardi, G.; Nardi, J.C. Service contract modeling in Enterprise Architecture: An ontology-based approach. *Inf. Syst.* **2019**, *101*, 101454. [\[CrossRef\]](#)
53. Pergl, R.; Sales, T.P.; Rybola, Z. Towards ontoUML for software engineering: From domain ontology to implementation model. In *Model and Data Engineering*; Cuzzocrea, A., Maabout, S., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; Volume 8216, pp. 249–263.
54. Rybola, Z.; Pergl, R. Towards OntoUML for software engineering: Transformation of Anti-rigid sortal types into relational databases. In Proceedings of the 6th International Conference (MEDI), Almería, Spain, 21–23 September 2016; pp. 1581–1591.
55. Rybola, Z.; Pergl, R. Towards OntoUML for software engineering: Optimizing kinds and subkinds transformed into relational databases. In *Enterprise and Organizational Modeling and Simulation*; Pergl, R., Babkin, E., Lock, R., Malyzhenkov, P., Merunka, V., Eds.; Springer: Cham, Switzerland, 2018; Volume 332, pp. 31–45.
56. Guidoni, G.L.; Almeida, J.P.A.; Guizzardi, G. Transformation of Ontology-Based Conceptual Models into Relational Schemas. In *Conceptual Modeling*; Dobbie, G., Frank, U., Kappel, G., Liddle, S.W., Mayr, H.C., Eds.; Springer: Cham, Switzerland, 2020; Volume 12400, pp. 315–330.
57. Clarke, P.; Mesquida, A.L.; Ekert, D.; Ekstrom, J.J.; Gornostaja, T.; Jovanovic, M.; Johansen, J.; Mas, A.; Messnarz, R.; Villar, B.N.; et al. An investigation of software development process terminology. *Commun. Comput. Inf. Sci.* **2016**, *609*, 351–361.
58. Lynn, R. Disadvantages of Agile. Available online: <https://www.planview.com/resources/articles/disadvantages-agile/> (accessed on 28 January 2022).
59. Botsman, R.; Rogers, R. *What’s Mine Is Yours: The Rise of Collaborative Consumption*; Harper Business: New York, NY, USA, 2010.
60. Handgraaf, S. Five Ways to Build an Online Marketplace Platform—And How to Choose Yours. Available online: <https://www.sharetribe.com/academy/ways-build-marketplace-platform/> (accessed on 28 January 2022).