Modeling Payments and Linked Obligation Settlements

Glenda Amaral[1], Tiago Prince Sales[1], and Giancarlo Guizzardi[1,2]

[1] CORE/KRDB, Free University of Bozen-Bolzano, Bolzano, Italy
{gmouraamaral,tiago.princesales,giancarlo.guizzardi}@unibz.it
[2] Services & Cybersecurity, University of Twente, The Netherlands

Abstract. Recently, digital innovation has revolutionized the world of payments and settlement services. Innovative technologies, such as the tokenization of assets, as well as new forms of digital payments, have challenged both current business models and the existing models of regulation. In this scenario, semantic transparency is fundamental not only to adapt regulation frameworks, but also to support information integration and semantic interoperability. In this paper, we deal with these issues by proposing an ontology-based approach for the modeling of payments and linked obligation settlements, that reuses reference ontologies to create ontology-based modeling patterns that are applied to model the domain-related concepts.

Key words: Economic Exchanges, Delivery versus Payment, OntoUML, gUFO

1 Introduction

Over the past decades, the financial industry has been disrupted by innovations that have shaken up the world of money, payments, and economic exchanges. These innovations, which include cryptocurrencies, blockchains and distributed ledger technologies, smart contracts [33], programmable money [6], and stablecoins [6], have challenged regulatory frameworks and business models in the financial industry. They have fostered the creation of financial products and services on top of decentralized technologies, giving rise to the concept of Decentralized Finance (DeFi) [37, 33]—the decentralized provision of financial products and services.

This disruption, alongside the entry of *big techs* into payments and financial services, pushed central banks to investigate new forms of digital money and prepare the grounds for central bank digital currencies (CBDCs). A CDBC is a form of digital money, denominated in the national unit of account, which is a direct liability of the central bank, such as physical cash and central bank settlement accounts [4]. The development of CBDCs as neutral means of payment and settlement assets enables the improvement of DeFi services and contributes to the open finance initiative by serving as a common platform around new payments ecosystems.

Open finance aims at empowering customers to have control over their data, so they can leverage it to have access to a wider range of financial products and services in a more open and competitive market [8]. It relies on standards, data sharing principles, and on a common understanding of key financial concepts to provide interoperability at different levels of DeFi ecosystems. Another important aspect of interoperability is related to the role played by central banks and regulatory authorities, which need to integrate a plethora of information from dynamic and complex decentralized environments to perform advanced analytics. This integrated view allows supervisors and regulatory entities to figure out what is going on, so they can ensure financial stability, manage financial risks, support anti-money laundering, combat the financing of terrorism, etc. Again, data needs to be clearly conceptualized and understood in order to be properly integrated and analyzed.

In this context, interoperability relies upon three interacting "layers" [29, p. 3]:

(i) a *social layer*, in which social actors interact to determine business models, regulatory frameworks and governance models;
(ii) an *information layer*, which supplies data stored both on-chain and off-chain; and
(iii) a *technical layer*, in which social actors interact to create, store, and obtain information via applications, networks, and consensus mechanisms.

In this paper, we deal with issues related to conceptual clarification (interoperability in the social layer) and semantic interoperability (interoperability in the information layer) in DeFi. We focus on the integration of DeFi off-chain data [37] and traditional finance data [37] that could be used in advanced analytics for regulation and supervisory purposes. It is important to note that DeFi ecosystems, as proposed by many countries, are ongoing systems under design, and the outcome is not clear yet. The initiatives around the world are at the stage of experimentation, proof of concept or pilot arrangements. Therefore, research on the integration of DeFi data and traditional finance is still in its infancy. To the best of our knowledge, the semantic-level integration of (DeFi) data and traditional finance data has not yet been addressed in the literature and is still an open issue.

We address this problem by proposing an ontology-driven conceptual modeling approach in which we (i) extract fragments of knowledge from reference ontologies to (ii) create ontology-based modeling patterns, which are (iii) systematically applied to represent concepts in the realm of money, payments, and economic exchanges. We first specify the models grounded on the Unified Foundational Ontology (UFO) [22], via the OntoUML [22] language, thus contributing to improving communication, problem-solving, and meaning negotiation among people. Then, we codify the models in gUFO [2], an implementation of UFO suitable for linked data applications, which contributes to dealing with semantic interoperability issues in heterogeneous scenarios.

We illustrate our approach by modelling payments and linked obligation settlements. In this scenario, innovative decentralized technologies allow financial and real tradeable assets to be digitally represented by what is known as digital tokens [11, 28]. An important aspect in this scenario is the utilization of settlement mechanisms that can prevent the risk that one counterparty irrevocably transfers the ownership of an asset, but does not receive the corresponding payment. A common way to mitigate this risk is to link the delivery and the payment legs, so that the asset moves if and only if the corresponding funds transfer occurs [9]. This settlement mechanism (a.k.a. *Delivery versus Payment*) is an example of a linked obligation settlement, a type of exchange transaction that must be performed atomically.

Although payments and linked obligation settlements have been modeled for decades [10, 13, 15, 19, 31], making sense of data in these new decentralized ecosystems is still a challenge. How to properly do analytics on financial data from different sources in decentralized heterogeneous ecosystems? The position defended here is that, in order to properly integrate data, it is necessary to make explicit their underlying ontological commitments. Let us take the example of two different ecosystem participants A and B that record information about payments. A and B may conceptualize the notion of payment in different ways. We cannot assume that just because the same term (e.g., payment) is used in both structures that they mean the same thing. For instance, Payment-A can refer to events, while Payment-B may refer to reified relationships [21]. In this case, the relation between Payment-A and Payment-B is one of manifestation, that is, instances of payment in one case (A) are manifestations of properties of payments as a bundle of relational aspects (B). For this reason, we advocate the use of ontology-based models, so that the nature of real world entities can be properly understood and represented.

This paper is organized as follows. In Section 2 we provide an overview of our research baseline, including payments and linked obligation settlements, UFO, the Core Ontology for Economic Exchanges, and the Reference Ontology of Money and Virtual Currencies, the latter two serving as conceptual foundation for our proposal. Then, in Section 3, we present our approach and use it to model payments and linked obligation settlements. In Section 4, we demonstrate our approach by modelling an application example. We present some related work on section 5 and conclude in Section 6 with some final considerations.

2 Background

2.1 Linked Obligation Settlements

In general, transactions involving the acquisition of goods, financial assets, or services have two settlement components: (i) the delivery of the good or service; and (ii) the transfer of funds [16]. According to the European Central Bank (ECB) [16], a payment is "a transfer of funds which discharges an obligation on the part of a payer vis-à-vis a payee". In this case, the payer is the party to a payment transaction which issues the payment order or agrees to the transfer of funds to the payee, while the payee is the final recipient of funds. When a payment is successfully made, the obligation between the payer and the payee is discharged. In the context of payments, settlement is an act that discharges obligations between two or more agents. For a payment instruction in a payment system, settlement occurs when funds are transferred from the payer's bank to the payee's bank. According to the ECB [16] "settlement discharges the obligation of the payer's bank vis-à-vis the payee's bank in respect of the transfer".

As explained in [7], a financial transaction involving two linked obligations may be settled by different mechanisms:

- **Payment versus Payment (PvP).** A settlement mechanism that ensures that the payment in one currency occurs if and only if the counterpart payment in another currency occurs as well [7]. This mechanism is typically used to mitigate settlement risk in foreign exchanges, which is the risk of delivering the currency sold without receiving the currency purchased (or vice versa).

- **Delivery versus Payment (DvP).** A settlement mechanism that links a securities transfer and a funds transfer in such a way as to ensure that delivery occurs if and only if the corresponding payment occurs [7].
- **Delivery versus Delivery (DvD).** A settlement mechanism that links two securities transfers in such a way as to ensure that the delivery of one security occurs if and only if the security in the other transfer is also delivered [7].

Recently, the European Central Bank and the Bank of Japan conducted a proof-of-concept, in the context of Project Stella [17], to explore how the settlement of two linked obligations, such as DvP, could be conceptually designed and operated in an environment based on the distributed ledger technology (DLT). In fact, such DLT-based Delivery versus Payment settlement can be applied not only in the context of financial assets but rather for all DLT use cases where assets such as immovable property, goods, or services are bought with money or exchanged with other assets.

2.2 The Unified Foundational Ontology (UFO)

The Unified Foundational Ontology (UFO) is an axiomatic domain-independent formal theory built on top of theories from formal ontology, philosophical logic, philosophy of language, linguistics, and cognitive psychology. It is organized in three main components: UFO-A, an ontology of endurants (objects) [22], UFO-B, an ontology of perdurants (events) [1], and UFO-C, an ontology of social entities [23].

UFO is formally connected to a conceptual modeling language (OntoUML). OntoUML was designed such that its modeling primitives reflect the ontological distinctions put forth by UFO, and its grammatical constraints follow UFO axiomatization. In fact, OntoUML is formally a pattern-language whose modeling primitives are *ontological design patterns*, representing UFO's constituting (micro)theories. Over the past decade, a number of ontology patterns have been derived from UFO, using OntoUML as a pattern language. An ontology pattern (OP) describes a particular recurring modeling problem that arises in specific ontology development contexts and presents a well-proven solution for the problem [18]. OPs are reused by analogy, i.e., by establishing a structural correspondence (or structural transfer) between the structure of the pattern and the one of the problem at hand. In this article, we focus on the use of domain-related ontology patterns, which are modeling fragments extracted of core/domain reference ontologies, containing pieces of knowledge that can be reused.

Furthermore, the "OntoUML Toolkit" contains several ontology engineering tools, such as ontological design patterns and anti-patterns, visual model simulation, and transformations for codification technologies. UFO has a partial translation to OWL termed gUFO [2], which is suitable for knowledge graph applications.

2.3 The Core Ontology for Economic Exchanges (COEX)

The Core Ontology for Economic Exchanges[1] (COEX) [32] is a well-founded reference ontology, specified in OntoUML, that formally characterizes the concept of economic

[1] The complete version of COEX in OntoUML and its implementation in OWL are available at `http://purl.org/krdb-core/economic-exchanges-ontology`.

exchanges based on the Action Theory of Economic Exchanges [30]. In this theory, an economic exchange is based on an agreement in which agents commit to performing certain reciprocal actions. This allows it to elegantly accommodate exchanges involving both products and services.

In COEX, when an `Offeree` accepts an `Economic Offering` proposed by an `Offeror`, the event of the offering founds a new relator of `Economic Agreement` between the two agents. This new relator has as parts the unconditional commitments of the agents (`Offeror Unconditional Agreement` and `Offeree Unconditional Agreement`) to fulfill the promised courses of actions. These agreements refer to types of actions, namely `Offered Contribution Type` and `Counterpart Contribution Type`. The actual event of `Economic Exchange` is required then to have as parts the event (action) of fulfillment of the offerer commitments as well as the event (action) of the fulfillment of the requested counterparts. Those events are of the right type, i.e. the `Offered Contribution` and the `Counterpart Contribution` match the type in `Offered Contribution Type` and `Counterpart Contribution Type` (respectively), cf. the relation `instantiation`. The relation `participation` models the fact that the offeror participates in the `Offered Contribution` event and the offeree participates in the `Counterpart Contribution` event. Figure 1[2] depicts a COEX diagram in OntoUML, which captures the aforementioned ontological notions.

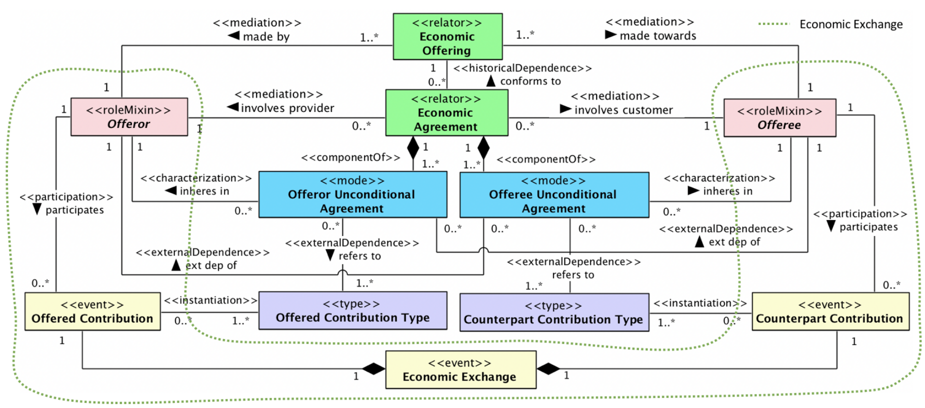

Fig. 1: A fragment of COEX [32] depicting economic exchanges.

2.4 The Reference Ontology of Money and Virtual Currencies (ROME)

The Reference Ontology of Money and Virtual Currencies[3] (ROME) [5] is a reference model, grounded on the UFO [22], that formalizes the characterization of money, currency, and virtual currencies (VC). The ontological distinctions between money and

[2] We adopt the following color coding in the OntoUML diagrams: types are represented in purple, objects in pink, qualities and modes in blue, relators in green, events in yellow, and datatypes in white.

[3] The complete version of ROME in OntoUML and its implementation in OWL are available at `http://purl.org/krdb-core/money-ontology`.

virtual currencies provided by ROME are important here because different rules and controls may apply to official money and VCs in activities such as issuance, risk assessment, risk mitigation, tax calculation, the elaboration of regulatory responses, among others. For this reason, we focus here on the representation of both currencies and VCs.

Nowadays, the status of money is supported by law, which specifies both the currency and the objects that are considered money in a particular country or region. It also defines a structure for the currency value domain. An example of structure is the one-dimensional structure of numbers with two decimal places defined for euros [25]. In Figure 2, we present a fragment of ROME that depicts the concept of `Money Status Function Description`, which defines a `Currency` and the `Monetary Object Types` that have the status of money. For example, the "Treaty on the Functioning of the European Union" [35] is an example of `Money Status Function Description`, which gives to euro banknotes and coins the status of money in the countries of the euro area. In this case, "euro" is the Currency, while "euro banknote" and "euro coin" are `Monetary Object Types`. The `Money Status Function Description` also defines a `Currency Quality Space Structure` for the `Currency Quality Space`. The former corresponds to a `Social Object` that prescribes a structure for the domain of values (e.g. number with two decimal places), while the latter corresponds to the value domain itself (see [22] for quality spaces).

Virtual currencies are similar to money within their user community. They also have their value grounded on a status function, which is defined in their underlying virtual currency scheme. Figure 2 presents a fragment of ROME that depicts the concept of `Virtual Currency Scheme Description`. An instance of such a description defines a `Virtual Currency`, a `Virtual Currency Token Type`, a `Virtual Currency Quality Space Structure`, and a `Virtual Currency Quality Space`. Examples of `Virtual Currency` include frequent flyer program points and privately-issued cryptocurrencies such as ETH (of the Ethereum blockchain platform). For an extensive discussion on money, currency and virtual currencies, please refer to [5].

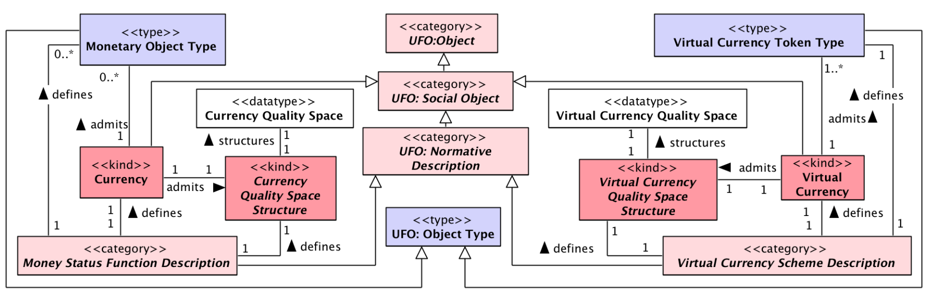

Fig. 2: A fragment of ROME [5] depicting currency and virtual currency.

3 Ontology-based Modeling of Payments and Linked Obligation Settlements

3.1 The Ontology-based Modeling Approach

In this section we present our ontology-based modeling approach and apply it to represent payments and linked obligation settlements, aiming at providing conceptual clarification and supporting semantic interoperability in the integration of DeFi off-chain data and traditional finance data. The three activities that compose our approach are described below.

1. **Extract knowledge fragments from reference ontologies**: In this step, we identify and extract fragments of core/domain reference ontologies, containing pieces of knowledge that describe the portion of reality that is intended to be represented, and that constitutes a well-proven modeling solution for the problem.
2. **Create ontology-based modeling patterns**: According to Buschmann et al. [14], "a pattern describes a particular recurring design problem that arises in specific design contexts and presents a well-proven solution for the problem". In this step, we reuse the model fragments extracted in step 1 to create ontology-based modeling patterns to represent recurrent structures in the domain.
3. **Apply the ontology-based modeling patterns to represent specific concepts in a particular domain**: In this step, we effectively apply the ontology-based modeling patterns identified in step 2 to model the problem at hand.

Our approach was inspired in the NeOn Methodology Framework [34]. NeOn provides guidance for the main activities in ontology engineering, making available detailed processes, guidelines and different scenarios for collaboratively building ontologies. In particular, we applied some of Neon methodological guidelines regarding reusing and reengineering ontological resources, which in our case are reference ontologies. One of the benefits of this approach is that pieces of knowledge from reference ontologies can be reused as needed: whole or extracts of it.

Figure 3 presents the customized version of the NeOn [3] methodology, suited to our particular context and needs. We defined two flexible scenarios, in which we applied some methodological directions of Neon for reusing and reenginnering ontological resources, namely reference ontologies.

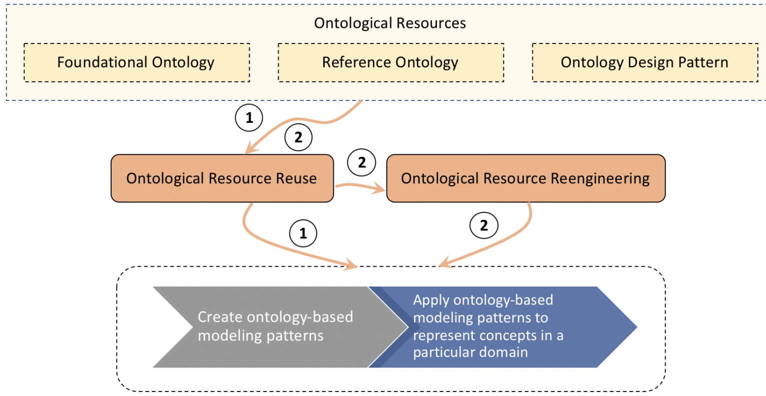

Fig. 3: Overview of the ontology-based modeling approach inspired in the NeOn methodology (adapted from [34])

3.2 Modeling Payments and Linked Obligation Settlements

For the modeling of payments and linked obligation settlements, we reuse concepts and relations defined in COEX [32] and ROME [5, 4] (step 1). As a result, we propose two ontology patterns (step 2) and apply them to model payments and different types of linked obligation settlements [9] (step 3), namely, Delivery versus Payment, Delivery versus Delivery and Payment versus Payment. We extend the notion of Delivery versus Payment and Delivery versus Delivery to consider not only financial assets (securities), but also any digital representation of assets (such as a share in a company, ownership of a piece of real estate, ownership of a car, or participation in an investment fund), which we name here *digital asset*. An example of digital asset is a token created on top of a blockchain network to represent the ownership of a real tradeable asset.

We start by defining a *digital transfer* as an event (action), in which the ownership of a digital object is transferred from one agent to another agent. By a *digital object* we mean a monetary amount or a digital asset [27]. For example, a digital payment is a digital transfer, in which a monetary amount is transferred from one agent to another. Similarly, a digital asset transfer is a digital transfer, in which a digital asset is transferred from a sender to a receiver. We define *digital exchanges* as events that have as parts two or more digital transfer events of: transferring a digital object to fulfill a commitment and transferring back another digital object to fulfill the requested counterpart. Confronting this view with COEX, we can see that this aspect is captured by the occurrence of an `Economic Exchange event` (Figure 1), which is composed of two events that represent the fulfillment of an economic agreement, namely the `Offered Contribution` and `Counterpart Contribution` (Figure 1). Moreover, what is brought about in digital transfers is the transferring of ownership of a digital object, which is in fact an event, see [6]. This is indeed a specific type of action, which can be straightforwardly accounted by either an `Offered Contribution` or a `Counterpart Contribution` in COEX. Therefore, digital exchanges can be seen as a specific type of economic exchange, in which the `Offered Contribution` and `Counterpart Contribution` are digital transfers between agents. This fragment of knowledge can be retrieved by isolating a part of the OntoUML [22] model that represent economic exchanges in COEX (area circled in green dotted lines, in Figure 1).

Based on these considerations, in Figure 4, we propose the Digital Transfer Pattern, represented in OntoUML [22]. In this pattern, a `Digital Transfer` is modeled as an event, which represents the action of transferring the ownership of a digital object (`Exchanged Digital Object`) from a `Sender` to a `Receiver`. As in COEX (Figure 1), both the `Sender` and the `Receiver` are UFO *agents* [22]. According to UFO, agent can be categorized into *human* (i.e. a person), *artificial* (i.e. artificial systems, such as information systems, cyber-physical systems, etc.) and *institutional* (i.e. organization). For reasons of space, we do not include a figure showing this agent categorization, but we refer the reader to [24](chap.3), for details. In the Digital Transfer Pattern, `Sender` and `Receiver` are modelled as *rolemixins* because they represent roles played by entities of different kinds (e.g., information systems and organizations). The same goes for `Exchanged Digital Objects`, which represent roles that can be played either by monetary amounts or by different kinds of digital assets. In Figure 5, we use the Digital Transfer Pattern to construct the Digital Exchange Pattern, which represents digital exchanges. It consists of a `Digital Exchange` event, composed of two or more `Digital Transfer` events.

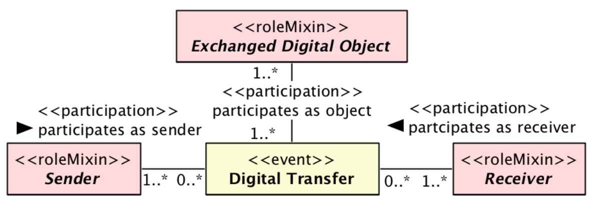

Fig. 4: The Digital Transfer Pattern.

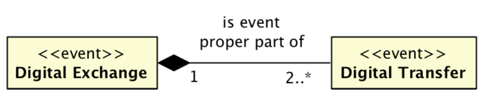

Fig. 5: The Digital Exchange Pattern.

In the sequel, we systematically apply and reuse both the Digital Transfer Pattern and the Digital Exchange Pattern to build a set of models that characterize the concepts and relations involved in the representation of payments and linked obligation settlements. Firstly, we specify the models using the OntoUML language [22]. Then, we generate their representation in gUFO.

Digital Asset Transfer. Digital asset transfer concerns the execution of actions aiming at transferring some sort of ownership rights to an asset from an agent to another agent. In Figure 6 we use the Digital Transfer Pattern (Figure 4) to model a `Digital`

`Asset Transfer` as an action (an UFO event [1]), in which a `Sender` agent transfers the ownership of a `Digital Asset` to a `Receiver` agent. The Turtle[4] fragment in Listing 1 shows the representation of a digital asset transfer in gUFO. This representation reproduces, with the limitations imposed by the expressiveness of the OWL language, the concepts presented in the OntoUML model in Figure 6. Figure 7 presents an instantiation example of the OntoUML model.

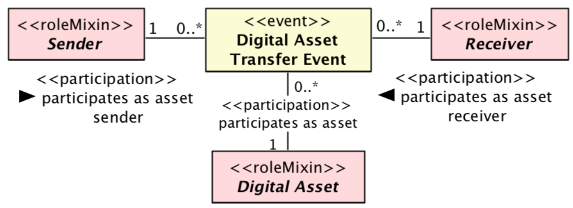

Fig. 6: Digital asset transfer in OntoUML.

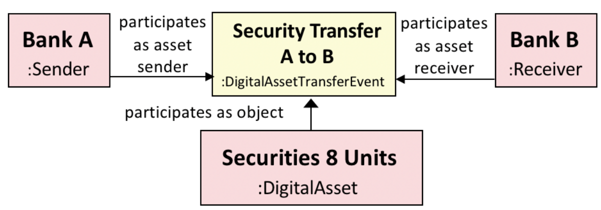

Fig. 7: Instantiation example of digital asset transfer.

Listing 1: Digital asset transfer represented in gUFO.

```
:DigitalAsset                 rdf:type gufo:RoleMixin ;
                              rdfs:subClassOf gufo:FunctionalComplex .

:DigitalAssetTransferEvent    rdf:type gufo:EventType,
                              rdfs:subClassOf gufo:Event .

:Receiver                     rdf:type gufo:RoleMixin ;
                              rdfs:subClassOf gufo:FunctionalComplex .

:Sender                       rdf:type gufo:RoleMixin ;
                              rdfs:subClassOf gufo:FunctionalComplex .

:participatesAsAssetSender    rdfs:subPropertyOf gufo:participatedIn ;
                              rdfs:domain :Sender ;
                              rdfs:range  :DigitalAssetTransferEvent .

:participatesAsAssetReceiver rdfs:subPropertyOf gufo:participatedIn ;
                              rdfs:domain :Receiver ;
                              rdfs:range  :DigitalAssetTransferEvent .

:participatesAsObject         rdfs:subPropertyOf gufo:participatedIn ;
                              rdfs:domain :DigitalAsset ;
                              rdfs:range  :DigitalAssetTransferEvent .
```

4 `https://www.w3.org/TR/turtle/`.

Payment. Payment concerns the transfer of a monetary amount in one currency from an agent to another. As explained in the modeling of Monetary Amount (Figure 10), we are considering here payments made both in official currencies and in virtual currencies.

In Figure 8 we use the Digital Transfer Pattern (Figure 4) to model a `Payment` as an action (an UFO event [1]), in which a `Sender` agent transfers a `Monetary Amount` to a `Receiver` agent. Figure 9 presents an instantiation example of this model. The Turtle fragment in Listing 2 shows the representation of a payment in gUFO.

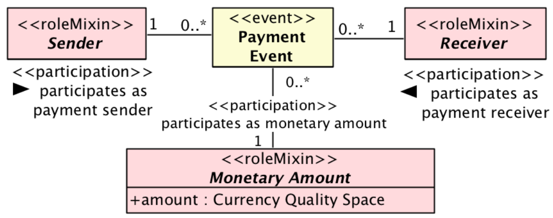

Fig. 8: Payment model in OntoUML.

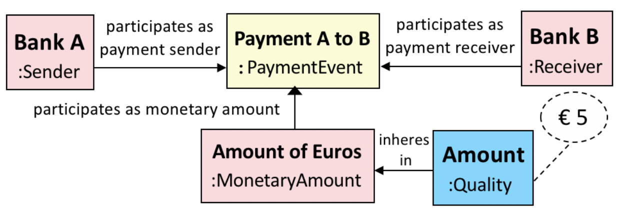

Fig. 9: Instantiation example of payment.

Listing 2: Payment represented in gUFO.

```
:MonetaryAmount               rdf:type gufo:RoleMixin ;
                              rdfs:subClassOf gufo:FunctionalComplex .

:Receiver                     rdf:type gufo:RoleMixin ;
                              rdfs:subClassOf gufo:FunctionalComplex .

:PaymentEvent                 rdf:type gufo:EventType ;
                              rdfs:subClassOf gufo:Event .

:Sender                       rdf:type gufo:RoleMixin ;
                              rdfs:subClassOf gufo:FunctionalComplex .

:participatedsAsPaymentSender rdfs:subPropertyOf gufo:participatedIn ;
                              rdfs:domain :Sender ;
                              rdfs:range :PaymentEvent .

:participatesAsPaymentReceiver rdfs:subPropertyOf gufo:participatedIn ;
                               rdfs:domain :Receiver ;
                               rdfs:range :PaymentEvent .

:participatedsAsMonetaryAmount rdfs:subPropertyOf gufo:participatedIn ;
                               rdfs:domain :DigitalAsset ;
                               rdfs:range :PaymentEvent .
```

Monetary Amount. To cope with the wide range of public and private payment means that emerged in recent times, our proposal considers not only payments made with real money and thus denominated in an official currency (e.g. Euro), but also payments using virtual currencies (eg. privately-issued cryptocurrencies like ETH). We rely on the notions of money, currencies, and virtual currencies defined in ROME [5] (Section 2.4) and on the concept of monetary amount defined in the Financial Industry Business Ontology (FIBO) [15]. According to FIBO, monetary amount corresponds to an amount of money specified in a currency. We extend FIBO's definition of monetary amount to consider also amounts specified in virtual currencies. Figure 10 shows an OntoUML diagram depicting the main concepts and relations involved in the representation of monetary amount.

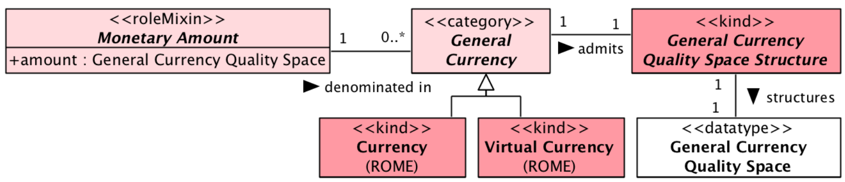

Fig. 10: Monetary amount in OntoUML.

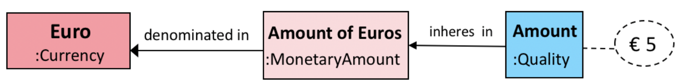

Fig. 11: Instantiation example of monetary amount.

In `Monetary Amount`, the property `amount` represents the quantity, which has a value in a `Currency Quality Space` [5] (cf. Section 2.4). For example, euro has a measurable value in one-dimensional structure of numbers with two decimal places. A Monetary Amount is denominated in a General Currency, which is specialized into Currency [5] and Virtual Currency [5]. For an extensive discussion on money, currency and virtual currencies, please refer to [5]. Figure 11 presents an instantiation example of the OntoUML model. The Turtle fragment in Listing 3 shows the representation of monetary amount in gUFO.

Listing 3: Monetary amount represented in gUFO.

```
:MonetaryAmount       rdf:type gufo:RoleMixin ;
                      rdfs:subClassOf gufo:FunctionalComplex.

:GeneralCurrency      rdf:type gufo:Category ;
                      rdfs:subClassOf gufo:FunctionalComplex.

rome:Currency         rdf:type gufo:Kind ;
                      rdfs:subClassOf :GeneralCurrency .

rome:VirtualCurrency rdf:type gufo:Kind ;
                      rdfs:subClassOf :GeneralCurrency .
```

```
:amount              rdfs:domain :MonetaryAmount ;
                     rdf:type owl:DatatypeProperty ;
                     rdfs:subPropertyOf gufo:hasQualityValue .

:denominatedIn       rdf:type owl:ObjectProperty ;
                     rdfs:domain :MonetaryAmount ;
                     rdfs:range :GeneralCurrency .
```

Delivery versus Payment. DvP can be seen as a specific type of digital exchange, in which the linked obligations are one or more digital asset transfers and the corresponding payment (or payments).

In Figure 12 we use the Digital Exchange Pattern (Figure 5) to model the `Delivery versus Payment` as an action (an UFO event [1]), composed of one or more `Digital Asset Transfers` (Figure 6) and one or more `Payments` (Figure 8). Figure 13 presents an instantiation example of this model. The Turtle fragment in Listing 4 shows the representation of the Delivery versus Payment in gUFO.

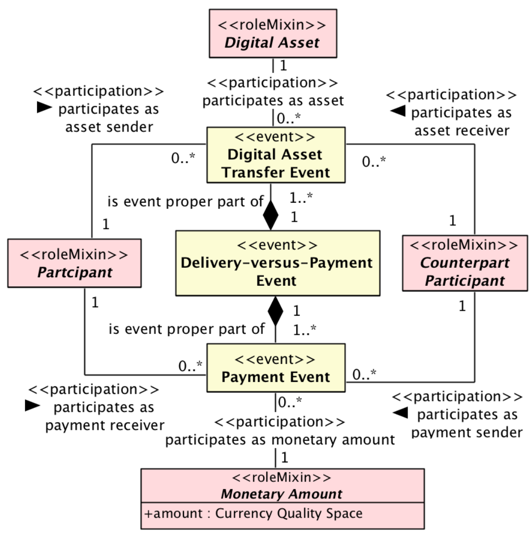

Fig. 12: Delivery versus Payment in OntoUML.

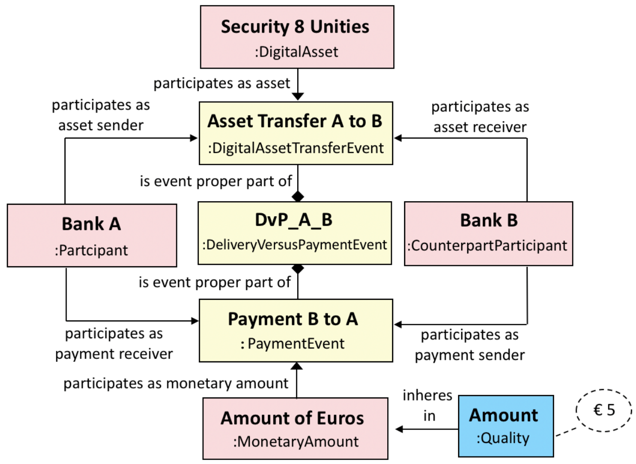

Fig. 13: Instantiation example of Delivery versus Payment.

Listing 4: Delivery versus Payment represented in gUFO.

```
:DeliveryVersusPaymentEvent     rdfs:subClassOf gufo:Event .

:DigitalAssetTransferEvent      rdfs:subClassOf gufo:Event ;
                         gufo:isEventProperPartOf :DeliveryVersusPaymentEvent .

:PaymentEvent                   rdfs:subClassOf gufo:Event ;
                         gufo:isEventProperPartOf :DeliveryVersusPaymentEvent .

:DigitalAsset                   rdf:type gufo:RoleMixin ;
                                rdfs:subClassOf gufo:FunctionalComplex .

:MonetaryAmount                 rdf:type gufo:RoleMixin ;
                                rdfs:subClassOf gufo:FunctionalComplex .

:Partcipant                     rdf:type gufo:RoleMixin ;
                                rdfs:subClassOf gufo:FunctionalComplex .

:CounterpartParticipant         rdf:type gufo:RoleMixin ;
                                rdfs:subClassOf gufo:FunctionalComplex .

:participatesAsAssetSender      rdfs:subPropertyOf gufo:participatedIn ;
                                rdfs:domain :Partcipant ;
                                rdfs:range :DigitalAssetTransferEvent .

:participatesAssetReceiver      rdfs:subPropertyOf gufo:participatedIn ;
                                rdfs:domain :CounterpartParticipant ;
                                rdfs:range :DigitalAssetTransferEvent .

:participatesAsAsset            rdfs:subPropertyOf gufo:participatedIn ;
                                rdfs:domain :DigitalAsset ;
                                rdfs:range :DigitalAssetTransferEvent .

:participatesAsPaymentSender    rdfs:subPropertyOf gufo:participatedIn ;
                                rdfs:domain :CounterpartParticipant ;
                                rdfs:range :PaymentEvent .

:participatesAsPaymentReceiver  rdfs:subPropertyOf gufo:participatedIn ;
                                rdfs:domain :Participant ;
                                rdfs:range :PaymentEvent .
```

```
:participatesAsMonetaryAmount   rdfs:subPropertyOf gufo:participatedIn ;
                                rdfs:domain :DigitalAsset ;
                                rdfs:range :PaymentEvent .
```

Payment versus Payment. PvP can be seen as specific type of digital exchanges, in which the linked obligations are two payments. In Figure 14 we use the Digital Exchange Pattern (Figure 5) to model the `Payment versus Payment` as an action composed of two `Payments` (Figure 8). Figure 15 presents an instantiation example of this model. The representation of both PvP in gUFO is analogous to the representation of DvP (Listing 4).

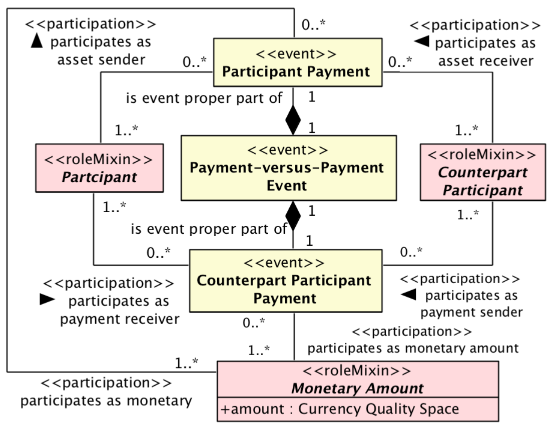

Fig. 14: Payment versus Payment in OntoUML.

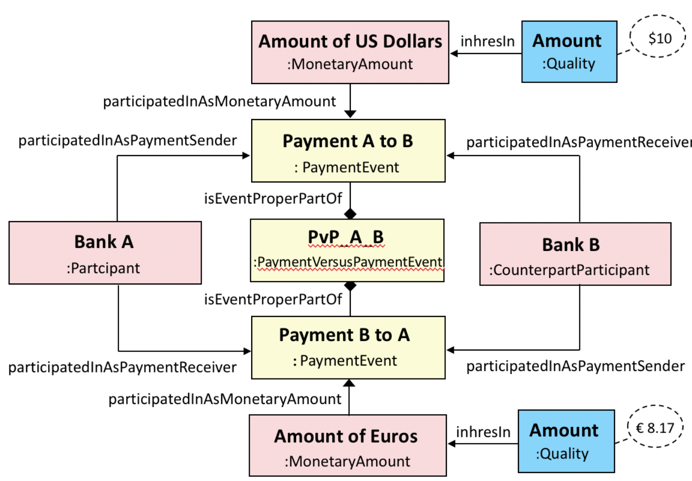

Fig. 15: Instantiation example of Payment versus Payment.

Delivery versus Delivery. DvD can be seen as specific type of digital exchanges, in which the linked obligations are two digital asset transfers. In Figure 14 we use the Digital Exchange Pattern (Figure 5) to model the `Delivery versus Delivery` as an action composed of two `Digital Asset Transfers` (Figure 6). Figure 17 presents an instantiation example of this model. The representation of DvD in gUFO is analogous to the representation of DvP (Listing 4).

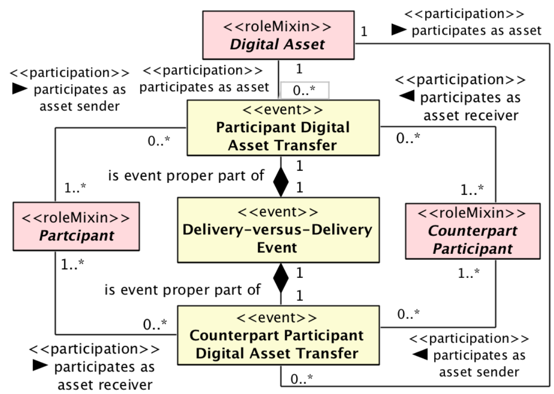

Fig. 16: Delivery versus Delivery in OntoUML.

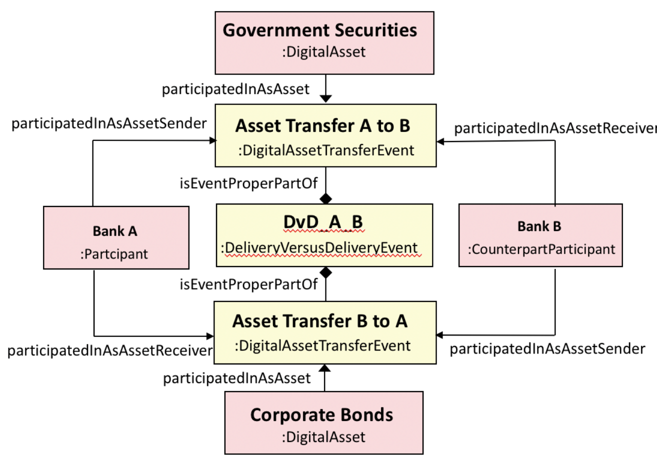

Fig. 17: Instantiation example of Delivery versus Delivery.

4 Application Example

A major concern of regulators is the use of digital money for illegitimate activities, like money laundering, terrorist financing, and tax evasion. Digital solutions for anti-money laundering and counter financing of terrorism based on artificial intelligence and data analytics can potentially help to identify risks and respond to, communicate, and monitor suspicious activity. Semantic interoperability is a fundamental aspect for applications in this context, as information from multiple and heterogeneous sources must be analyzed to detect unusual patterns, such as large amounts of cash flow at certain periods by particular groups of agents. Let us take as an example the assessment of a company named “Orange Corporate” regarding suspicious transactions. In order to detect unusual patterns, it is important to analyze all payment transfers performed by Orange Corporate. As the company operates on multiple ecosystems, it may be necessary to integrate DeFi on-chain and/or off-chain data with traditional finance data. Furthermore, from the perspective of the prevention of tax evasion, it is also important to be able to distinct payment transfers denominated in official currencies from the ones denominated in virtual currencies as different controls and rules may apply in each case. Figure 18 illustrates the application of the Payment and the Monetary Amount models (Listing 2 and 3) to support information integration in data analytics, for the example just described.

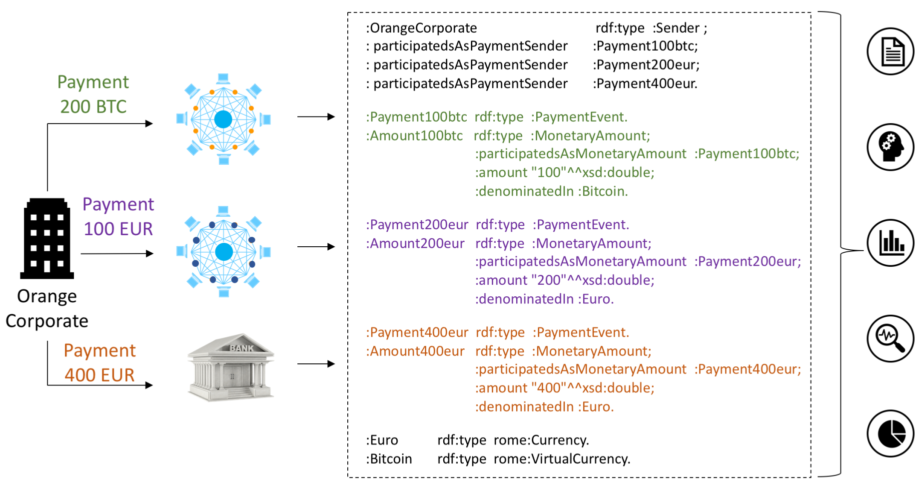

Fig. 18: Regulatory data analytics example.

5 Related Work

The notions of payments and settlement services have been addressed by financial standards such as the Financial Industry Business Ontology (FIBO) [15], which includes the modeling of payments and monetary amounts. However, these standards usually represent only payments made in official currencies. Payments made in virtual currencies such as privately-issued cryptocurrencies are not considered in their models.

Economic exchanges are a central notion in the Resource-Event Action (REA) ISO Standard [31]. In fact, as shown in [32], REA can be seen as subsumed by the Common Ontology for Economic Exchanges (COEX), on which our proposal is based. However, REA does not address the particularities of linked obligation settlement mechanisms (DvP, DvD and PvP) nor does it provide an ontological account of payments.

Another approach related to the notion of economic exchange is e3-value [20], an ontology-based methodology, commonly used for the modeling value exchanges. It adopts the economic value perspective by representing what is exchanged and by whom. The e3value ontology is based on the principle of reciprocity, denoting that every actor offers something of value, such as money, goods, services, etc., and gets a value in return. However, e3-value focuses on the exchanged value among actors in a generic way, leaving out the the particularities of linked obligation settlement mechanisms as well as the ontological distinctions between assets and payments.

Fischer-Pauzenberger and Schwaiger [19] proposed the OntoREA Accounting and Finance Model, which constitutes an ontology-based conceptualization of the accounting and finance domain, grounded on UFO. Similarly, Blums and Weigand [13] proposed a Reference Ontology of Complex Economic Exchanges for Accounting Information Systems, grounded on UFO, which is a commitment-based economic exchange ontology, whose conceptualization is based on the establishment and fulfillment of commitments and claims between exchange participants (enterprise and counterparty) along the exchange life-cycle. In [36], the same authors proposed a comprehensive approach for implementing economic exchanges in DLT. These three approaches are similar to ours in the sense that it uses a well-founded language to represent concepts in the realm of economic exchanges. However, they differ from our work as they do not consider the ontological distinctions between money and virtual currencies in the modeling of payments.

Finally, the Project Ellipse, launched by the Bank for International Settlements Innovation Hub [10] proposes the creation of an integrated regulatory data and analytics platform to support regulatory oversight. Although they rely on "common data models" to provide a common understanding and properly integrate information, their models do not consider the ontological distinctions between the concepts.

6 Final Remarks

In this paper, we proposed and ontology-based approach for the modeling of payments and linked obligation settlements, aiming at providing conceptual clarification and supporting semantic interoperability. Firstly, we created two domain-related ontology patterns by reusing pieces of knowledge extracted from reference ontologies. Then, we applied these patterns to model payments and linked obligations settlements in OntoUML. Finally, we exported the models to OWL using gUFO. These gUFO/OWL concrete artifacts can contribute to semantic web related initiatives in finances [12], as well as to the goal of transparency of financial data exchange according to FAIR principles [26].

This work is part of a broader research initiative in the domains of finance and economics. We have been working with concepts like money, value, risk, trust and economic exchanges, creating a federation of reference conceptual models which are now becoming a kind of network of models in economics and finance. This paper, in particular, is an application example of two of these ontologies: the ontology of money and the ontology of economic exchanges. One of our objectives is to raise awareness of the importance of making explicit the ontological commitments of financial data, so that data from different sources in decentralized heterogeneous ecosystems can be properly and safely integrated.

As future work, we plan to validate our models in practice, to support information integration between multiple DLT/blockchain networks and traditional finance datasets. We also plan to expand our analysis to different types of settlement agreements, including those involving more than two participants.

References

1. Almeida, J.P.A., Falbo, R.A., Guizzardi, G.: Events as entities in ontology-driven conceptual modeling. In: 38th Intl.Conference on Conceptual Modeling. pp. 469–483. Springer (2019)
2. Almeida, J., Guizzardi, G., Sales, T.P., Falbo, R.: gUFO: A Lightweight Implementation of the Unified Foundational Ontology (UFO). Tech. rep., Ontology & Conceptual Modeling Research Group (NEMO) - Federal University of Espirito Santo (2020)
3. Almeida Falbo, R.: Sabio: Systematic approach for building ontologies. In: ONTO. COM/ODISE@ FOIS (2014)
4. Amaral, G., Sales, T.P., Guizzardi, G.: Towards Ontological Foundations for Central Bank Digital Currencies. In: 15th Value Modelling and Business Ontologies Workshop (2021)
5. Amaral, G., Sales, T.P., Guizzardi, G., Porello, D.: A Reference Ontology of Money and Virtual Currencies. In: IFIP Working Conference on The Practice of Enterprise Modeling. pp. 228–243. Springer (2020)
6. Arner, D.W., Auer, R., Frost, J.: Stablecoins: risks, potential and regulation. Financial Stability Review. Nº 39 (Autumm 2020), p. 95-123 (2020)
7. Bank for International Settlements: A glossary of terms used in payments and settlement systems (2016)
8. Bank for International Settlements: Enabling open finance through APIs (2020)
9. Bank for International Settlements / International Organization of Securities Commissions : Principles for financial market infrastructures (2012)
10. Bank of International Settlements: Ellipse: regulatory reporting and data analytics platform. Available at `https://www.bis.org/about/bisih/topics/suptech_regtech/ellipse.htm` (2021/12/16) (2021)
11. Bech, M.L., Hancock, J., Rice, T., Wadsworth, A.: On the future of securities settlement. BIS Quarterly Review (2020)
12. Bennett, M.: The financial industry business ontology: Best practice for big data. Journal of Banking Regulation **14**(3), 255–268 (2013)
13. Blums, I., Weigand, H.: Financial reporting by a shared ledger. In: JOWO (2017)
14. Buschmann, F., Henney, K., Schmidt, D.C.: Pattern-oriented software architecture, on patterns and pattern languages, vol. 5. John Wiley & Sons (2007)
15. Enterprise Data Management Council: Financial Industry Business Ontology (2015)
16. European Central Bank: The payment system – payments, securities and derivatives, and the role of the Eurosystem. European Central Bank (2010)
17. European Central Bank, Bank of Japan: Securities settlement systems: delivery-versus-payment in a distributed ledger environment – Stella project report phase 2. ECB (2018)
18. Falbo, R.d.A., Guizzardi, G., Gangemi, A., Presutti, V.: Ontology patterns: clarifying concepts and terminology. In: 4th Workshop on Ontology and Semantic Web Patterns (2013)
19. Fischer-Pauzenberger, C., Schwaiger, W.S.: The OntoREA© Accounting and Finance model: ontological conceptualization of the accounting and finance domain. In: International Conference on Conceptual Modeling. pp. 506–519. Springer (2017)

20. Gordijn, J., Akkermans, H.: Designing and evaluating e-business models. IEEE intelligent Systems **16**(04), 11–17 (2001)
21. Guarino, N., Guizzardi, G.: Relationships and events: towards a general theory of reification and truthmaking. In: Conference of the Italian Association for Artificial Intelligence. pp. 237–249. Springer (2016)
22. Guizzardi, G.: Ontological foundations for structural conceptual models. Telematica Instituut Fundamental Research Series, No. 15, ISBN 90-75176-81-3 (2005)
23. Guizzardi, G., Falbo, R.A., Guizzardi, R.S.S.: Grounding software domain ontologies in the Unified Foundational Ontology (UFO). In: 11th Ibero-American Conference on Software Engineering (CIbSE). pp. 127–140 (2008)
24. Guizzardi, R.: Agent-oriented Constructivist Knowledge Management. Ph.D. thesis, University of Twente, Netherlands (2006)
25. ISO: Codes for the representation of currencies - ISO 4217:2015 (2015)
26. Jacobsen, A., et al.: FAIR Principles: Interpretations and Implementation Considerations. Data Intelligence **2**(1-2), 10–29 (2020)
27. Kud, A.: Substantiation of the Term "Digital Asset": Economic and Legal Aspects . International Journal of Education and Science **2**(1), 41–52 (2019)
28. Laurent, P., Chollet, T., Burke, M., Seers, T.: The tokenization of assets is disrupting the financial industry. Are you ready? Inside Magazine **19**, 62–67 (2018)
29. Lemieux, V.L., Feng, C.: Building Decentralized Trust. Springer (2021)
30. Massin, O., Tieffenbach, E.: The metaphysics of economic exchanges. Journal of Social Ontology **3**(2), 167–205 (2016)
31. McCarthy, W.: ISO 15944-4 - REA Ontology. ISO (June 2007)
32. Porello, D., Guizzardi, G., Sales, T.P., Amaral, G.: A core ontology for economic exchanges. In: 39th International Conference on Conceptual Modeling. pp. 364–374. Springer (2020)
33. Schär, F.: Decentralized finance: On blockchain-and smart contract-based financial markets. FRB of St. Louis Review (2021)
34. Suárez-Figueroa, M.C., Gómez-Pérez, A., Motta, E., Gangemi, A. (eds.): Ontology Engineering in a Networked World. Springer Berlin Heidelberg (2012)
35. Treaty on the functioning of the European Union. Official Journal of the European Union **C326**, 47–390 (2012)
36. Weigand, H., Blums, I., de Kruijff, J.: Shared ledger accounting—implementing the economic exchange pattern. Information Systems **90**, 101437 (2020)
37. Zetzsche, D.A., Arner, D.W., Buckley, R.P.: Decentralized finance (DeFi). IIEL Issue Brief **2** (2020)