

The OntoREA© Accounting and Finance Model: A Retroactive DSRM Demonstration Evaluation

Christian Fischer-Pauzenberger, Walter Schwaiger

► **To cite this version:**

Christian Fischer-Pauzenberger, Walter Schwaiger. The OntoREA© Accounting and Finance Model: A Retroactive DSRM Demonstration Evaluation. 10th IFIP Working Conference on The Practice of Enterprise Modeling (PoEM), Nov 2017, Leuven, Belgium. pp.81-95, 10.1007/978-3-319-70241-4_6. hal-01765260

HAL Id: hal-01765260

<https://hal.inria.fr/hal-01765260>

Submitted on 12 Apr 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



The OntoREA© Accounting and Finance Model: A Retroactive DSRM Demonstration and Evaluation

Christian Fischer-Pauzenberger¹[0000-0002-9071-9523], Walter S.A. Schwaiger¹[0000-0003-1868-7018]

¹ Technische Universität Wien, Institute of Management Science, Theresianumgasse 27, 1040
Wien, Austria
{christian.fischer-pauzenberger, walter.schwaiger}@tuwien.ac.at

Abstract. Derivative instruments have special characteristics that make them difficult to understand and to handle in financial instrument accounting. In the OntoREA© Accounting and Finance Model [1] such instruments are conceptualized and integrated into the REAC Business Ontology [2] as well as the OntoREA Accounting Model [3]. But the OntoREA© model is developed at a very abstract level so that no real cases are used for demonstration and evaluation. This shortcoming is addressed in this article by demonstrating and evaluating the *Collective* conceptualization of derivative instruments in OntoREA© from a retroactive design science methodological (DSRM) [4] perspective within the model driven software development (MDD) context. Along the model transformations in this context the OntoREA© model serves as platform independent (PIM) model. For demonstration purposes its direct translation into a platform specific (PSM) as well as an implementation specific (ISM) model are demonstrated for a real derivative instrument. The evaluation shows that the requirements of derivative instruments are met adequately.

Keywords: OntoREA© Accounting and Finance Model, OntoUML, Model Driven Development, Design Science Research Methodology

1 Introduction

A derivative is a financial instrument:

- *Whose value changes in response to the change in an underlying variable such as an interest rate, commodity or security price, or index;*
- *That requires no initial investment, or one that is smaller than would be required for a contract with similar response to changes in market factors; and*
- *That is settled at a future date.* [5]

This is the definition of derivative instruments in the International Financial Reporting Standards (IFRS). The three characteristics of derivative instruments say that their value is derived from an underlying variable (that's where their name comes from), the zero or low investment comes from the fact that these instruments are only contracted commitments that are fulfilled in the future where the underlying and the contracted prices are exchanged between the contracting agents. The zero investment means that a derivative instrument can have a value of zero. A good example for such

a contract is a forward which is defined as follows: *Contracts to purchase or sell a specific quantity of a financial instrument, a commodity, or a foreign currency at a specified price determined at the outset, with delivery or settlement at a specified future date. Settlement is at maturity by actual delivery of the item specified in the contract, or by a net cash settlement.* [6]

At the *contracting date* the buyer and the seller of the forward contract agree to exchange at the future *expiration date* the underlying good. In e.g. a *stock forward* the *underlying* of the contract is an explicitly defined *stock* (e.g. the XY stock). In the easiest case the forward *contract size* is one stock. In this instance the buyer of the forward receives in the future the stock from the seller. In reciprocity for this receipt the buyer pays the seller the *forward price*. Normally the forward price is set at the contracting date so that the initial value of the forward is equal to zero. This valueless property of the forward causes a problem in accounting as such contracts cannot be considered neither as assets nor as liabilities.

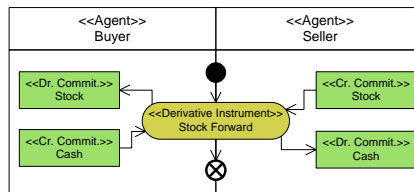


Figure 1: Derivative Instrument – UML-Activity Diagram

In Figure 1 the stock forward is conceptualized with an UML-activity diagram. The contracting between the two agents, i.e. the buyer and the seller, is the activity that initiates the forward contract. There are no exchanges at the contracting date. In the contract only the future related commitments are specified. This can be seen in the figure where the buyer has a debit commitment to the stock which is a credit commitment for the seller who has to deliver the stock. On the other hand the reciprocity requires that at the expiration date the buyer has to pay in cash the forward price to the seller.

The OntoREA© accounting and finance model [1] integrates the REAC Business Ontology [2] with the OntoREA Accounting Model [3]. Its name suggests that it is specified in the OntoUML language [7]. Derivative instruments are included according to the *no-arbitrage pricing theory* from the Nobel laureates Black/Scholes [8] and Merton [9]. Specifically the derivative instruments are conceptualized according to their corresponding *hedge portfolio* [10] – which underlies the no-arbitrage pricing theory – as *Collective* with a *MemberOf* relationship to two *Economic Resources*.

This abstract conceptualization of derivative instruments is hard to grasp. In the following this understanding problem is solved by demonstrating the precise meaning of the *Collective* conceptualization with a *stock forward* running example. Furthermore the usage of the OntoREA© model will be demonstrated in the model driven (software) development (MDD) context [11]. The UML activity diagram in Figure 1 corresponds to the computational independent *CIM model*, the OntoREA© model constitutes the platform independent *PIM model*. According to the *forward engineering* approach in MDD the PIM model is transformed into a platform specific *PSM model* and finally into an implementation specific *ISM model*. More precisely, the MySQL [12] database

model (PSM) and RStudio dataframe model (ISM) [13], [14] are derived in two transformation steps. For demonstrating the MDD process up to the software application a prototypical ISM application will be developed in RStudio Shiny.

A design science research methodology applied in accounting information systems [4] consists of six activities, i.e. problem identification and motivation, definition of the objectives of a solution, design and development, demonstration, evaluation and communication. In this light the OntoREAC Accounting and Finance model [1] addresses the first three activities quite well. This can be seen by the developed accounting and finance model which integrates the special temporal modal and identity-related peculiarities of financial instruments into the REA-based Asset-Liability-Equity (ALE) accounting model. Due to its publication the model also accomplishes the DSRM communication activity. With respect to the demonstration and evaluation activities there are shortcomings: there is no demonstration of the models' usage in e.g. an MDD context and its adequacy concerning the representation of the derivative instruments is not evaluated.

The primary research objective is twofold: Firstly, fostering the meaning of OntoREAC's *Collective* conceptualization of financial instruments. Secondly, the DSRM validation of this conceptualization in two steps

- by demonstrating its usage in the two model transformations within the MDD context, i.e. from the PIM to the PSM model as well as from the PSM to the ISM model, and
- by evaluating the adequate coverage of the financial instruments' special peculiarities.

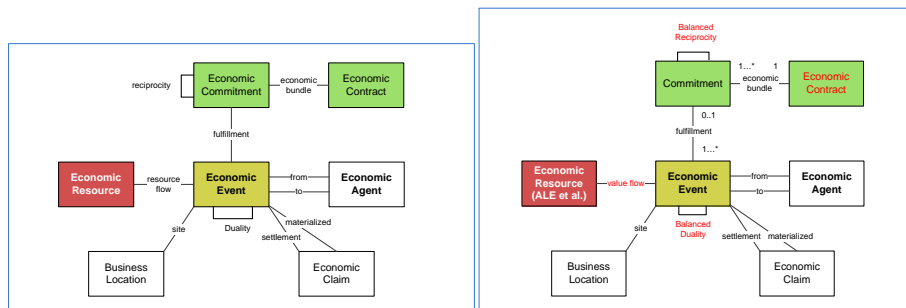


Figure 2: REAC Model (left) and REA©-based ALE Accounting Model (right)

The paper is organized as follows. In the subsequent section the OntoREAC accounting and finance model is presented. In the MDD context this model constitutes the abstract PIM model that is successively transformed into more specific models. In the next section the two transformations are demonstrated. Firstly, the MySQL database model (PSM) is derived. Of special importance is the specification of exclusivity constraint, which underlies the OntoUML *Collectives* constraints, for the MySQL database in the object constraint language (OCL). Secondly, the derived PSM model is transformed into an RStudio data model (ISM). In this transformation step the exclusivity constraint is via the switch function already expressed in the R programming language. This specification has the advantage that it can be directly used

in the prototypical ISM application in RStudio Shiny. The prototypical application is shown in the next section. In the final section the paper is concluded.

2 The OntoREA[©] Accounting and Finance Model: PIM Model

The OntoREA[©] accounting and finance model is based upon the *REAC Business Ontology* [2] as well as the *OntoREA Accounting Model* [3]. Both models can be seen in Figure 2. The REAC business ontology extends the original *REA Accounting Model* [15] by including among others a forward looking perspectives in form of commitments. The REA[©]-based ALE accounting model eliminates deficiencies in the REAC business ontology in order to make it appropriate for Asset/Liability/Equity (ALE) accounting purposes.

In Figure 3 the REA[©]-based ALE accounting model is slightly modified in order to enhance its expressiveness with respect to the OntoREA[©] accounting and finance model. The modification consists of the reification of the *Balanced Duality* relationship and the *Balanced Reciprocity* relationship. Now it is explicitly expressed that each business transaction is based upon a contract, i.e. a spot contract in a spot market and a future contract in a future market.

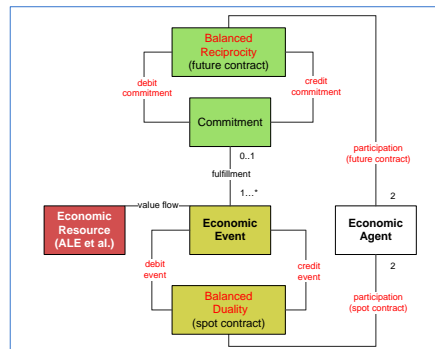


Figure 3: REA[©]-based ALE Accounting Model – Modified version

For understanding the meta-physical semantics of the OntoUML languages a short primer is given: “In conceptual modeling entity types and relationship types are the most fundamental constructs [16]. Taken from [3] the following entity types (subsequently marked in bolded letters), which are derived from the UFO typification tree (marked in italic letters) are of importance for the OntoREA accounting model:

- **Kinds** are *rigid substance sortals* and they provide their own identity principle (rather than just carrying it). *Kinds* are also considered as an OntoUML model’s backbone [17] and they are used to model resources, events and agents as endurants.
- **SubKinds** are *rigid sortals* and do not provide their own identity principle, they are merely inheriting the principle from another *Substance Sortal*.

- **Roles** are *anti-rigid sortals* and therefore can change their instantiation in a modal sense according to an extrinsic generalization condition. Furthermore, Roles are relational-dependent, they have to rely on at least one other universal. Roles get their identity principle through the generalization relation from the instance of its parent universal.
- **Phases** are *anti-rigid sortals* as well with a significant distinction to Roles. Phases are relational (i.e. external) independent. Due to the predetermined disjoint and complete generalization sets, instances of Phases, in contrast to instances of Roles, can change according to their intrinsic (and not extrinsic) generalization condition. As Roles, Phases also get their identity principle through the generalization relation.

Next to the above mentioned entity types the following relationship types will be important for the OntoREA accounting model to be developed:

- **Relator Universals**, are *moment universals* which represent the objectification of a relational property. Relator Universals are existentially dependent on a multitude of individuals, thus, mediating them. Relators are the foundation of the so-called **Material Relations** [18] and act as truthmakers of the relation.
- **Formal Relations** hold directly between entities without requiring any intervening individual.” [1].

Next to that, two additional *meronymic* meta-properties are used in the OntoREA© accounting and finance model in order to address the peculiar *parthood relationship* associated with derivative instruments:

- **Collectives** are – like Kinds – *rigid substance sortals* and that provide their own identity principle. The difference to *Kinds* lies in the scope of the universals. The *Kind* universal specifies individual universals whereas the *Collective* universal goes beyond individual universals by specifying collections.
- **MemberOf Parthood-Relationships** are relationships where both the *Collective* and its constituting *Kinds* have their corresponding identity principles. The *Beatles* are an example for such a collection. Seen as an instantiation of a *Collective* universal the Beatles as a group has its own identity. The members of the Beatles as persons are instantiations of a *Kind* universal and consequently have their personal identity.

The remaining entity types and relationship types of UFO are not needed for the OntoREA© accounting and finance model. The current and complete version of UFO is specified in the UFO reference [19].

Equipped with the vocabulary and syntax given in the OntoUML primer the OntoREA© accounting and finance model [1] can be presented. Figure 4 shows the model. The OntoUML meta-physical properties are marked with <<Guillemet>> parentheses. In contrast to the original version of the model one minor modification is included in order to avoid misunderstandings within the MDD contextual model transformations: The *Null* naming of the third derivative instrument phase is replaced by the word *Off Balance* due to the special importance of the *Not Null* restriction used in platform specific database PSM models.

For the exploration of the OntoREA© accounting and finance model in Figure 4 it’s advisable to start with the *Balanced Duality* and the *Balanced Reciprocity* classes. As is indicated in the REA©-based ALE accounting model in Figure 3, both classes are

Relator classes that reify the monetary balancing between the associated debit and credit entries. As such they possess a truthmaker that mediates between the debit and credit entries. In both cases the truthmaker is a contract, i.e. a *spot market contract* in the balanced duality case and a *future market contract* in the balanced reciprocity case. For simplicity this connection to the truthmaker is referenced explicitly only in the balanced reciprocity case by setting the *Economic Contract* in parentheses.

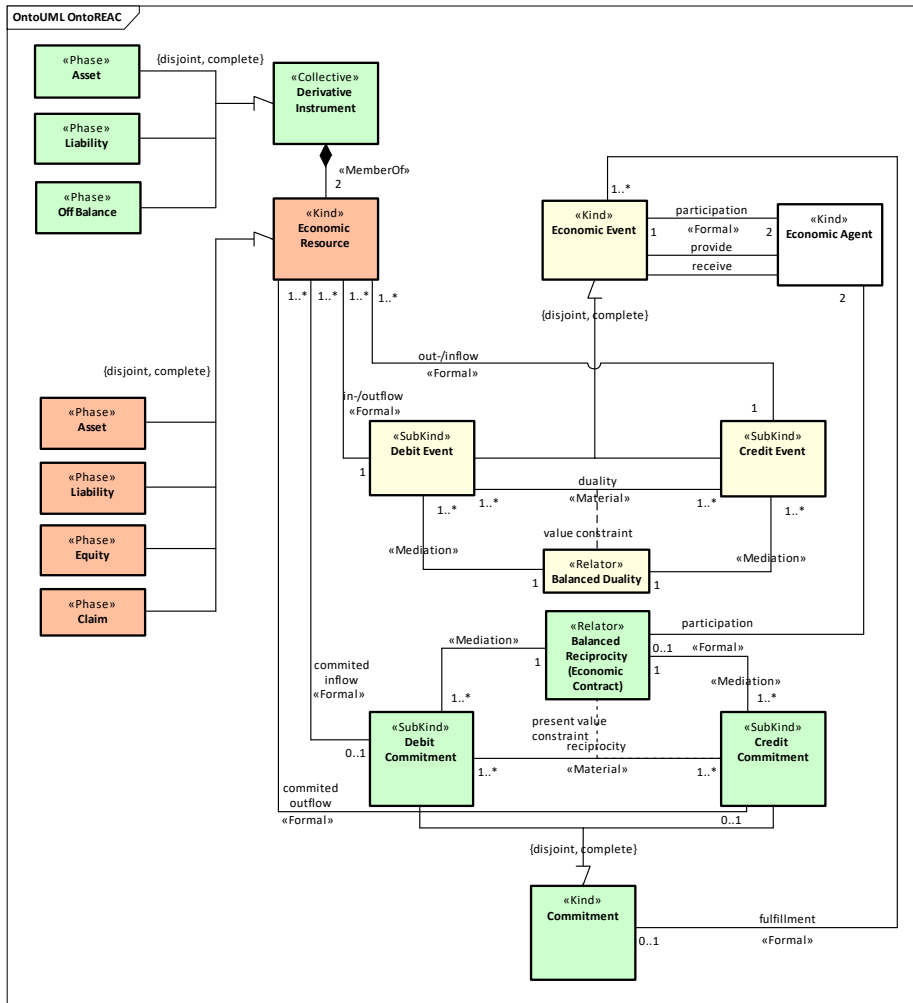


Figure 4: OntoREA© Accounting and Finance Model – Conceptual Model (PIM)

A derivative instrument, which is the primary focus in this article, is an *Economic Contract* that obeys the *Balanced Reciprocity relationship* between its debit and credit commitments. Furthermore there is a 0..1 cardinality from the *Economic Resource* class to the *Debit Commitment* class as well as to the *Credit Commitment* classes. If there are economic resources linked to the debit and credit commitments then these are the

resources in the hedge portfolio of the derivative instrument. As such the derivative instruments are economic contracts that are linked to their corresponding hedge portfolio. In the top-left corner of Figure 4 the *Collective* class Derivative Instrument is shown. “This class has a *MemberOf* relationship to the *Kind* class Economic Resource. The cardinality says that derivative instruments have two economic resource members, i.e. one asset and one liability. Furthermore the *Collective* class Derivative Instrument is typified via a *Phase* partition that consists of the *Phase* classes Asset, Liability and Off Balance. According to this the derivative instruments can change the phases over time without losing its identity by switching from the *Phase* Off Balance, i.e. an off-balance position, into an on-balance position either to the *Phase* Asset or the *Phase* Liability and so on.” [1].

The *Collective* conceptualization of derivative instruments in the OntoREA© is quite different to the ALE phases of economic resources. The asset, liability, equity and claim phase partition of economic resources relate to individual resources of a *Kind* class whereas the asset, liability and off balance phase partition of derivative instruments relate of a *Collective* class that consists of the 2 economic resources in its hedge portfolio. Accordingly the two phase partitions will be treated differently in the transformation from the PIM model into the PSM model.

3 The OntoREA© Model in the MDD-Context: Transformations from PIM via PSM into ISM

Let’s start with a concrete derivative instrument example, i.e. a long position in a stock forward contract. At the beginning of the year (*contracting date*) we are buying a stock forward where we receive one XY stock (*contract size*) at the end of the year (*expiration date*) and for which we have to pay the forward price. Currently the one-year interest rate, which corresponds to the contract’s *initial time to maturity* of 12 months, amounts to 4 percent ($4\% = 0.04$). By taking the interest rate as the substitute for the cost of carry, the forward price is equal to 104, which is the current stock price of 100 plus the 4 % interest rate. For this forward price the fair value of the stock forward is equal to zero. This means, the stock forward is a valueless position taken at the contracting date. Considering such a position from the accounting recognition perspective shows the problem that according to the valueless property in the initial measurement it cannot be assigned neither as asset nor as liability.

Over time the fair value of the contract, i.e. the stock forward value, depends on the actual stock prices and the remaining times to maturity. In the mid-year pricing, where the stock price is assumed to be 100, the forward value amounts to -1.98. In this case the forward contract appears due to its negative value as a liability position in the balance sheet. For the final measurement at the expiration date it is assumed that the stock price is 120. In this case the forward contract has a positive value of 16 so that it appears as an asset in the balance sheet.

Considering the evolution of forward value over its life cycle, i.e. from the contracting date up to the expiration date, shows the temporal modal nature of the balance sheet recognitions. Starting from an off balance position the stock forward changes to a liability and ends up as an asset position. This recognition structure mainly

depends on the assumed development of the stock prices which is taken for illustrative purposes. In general there are different stock price developments possible so that the stock forward recognitions are different as well. As stock prices are stochastic over time, the stock forward's balance sheet assignments also have a *stochastic temporal modal nature*. This specific nature of forward contracts can be seen in Table 1 in the boldfaced rows.

Contracting date:	01.01.2020
Expiration date:	31.12.2020
Initial stock price:	100
<u>Initial interest rate:</u>	<u>4%</u>
Stock Asset:	100
Loan Liability:	100
Forward value: = A - L	0 (Off B.)
Pricing date #1:	30.06.2020
Actual stock price:	100
<u>Actual time to maturity:</u>	<u>6 months</u>
Stock Asset:	100
Loan Liability:	101,98
Forward value: = A - L	-1,98 (L)
Pricing date #2:	31.12.2020
Actual stock price:	120
<u>Actual time to maturity:</u>	<u>0 months</u>
Stock Asset:	120
Loan Liability:	104
Forward value: = A - L	16 (A)

Table 1: Stock Forward (Running Example) – Contract Specification and Pricing

Furthermore, Table 1 shows the *hedge portfolio* at each pricing date in form of the *stock asset* and the *loan liability*. The net value of this portfolio gives the forward value. In parentheses the balance sheet assignments are given which depend on the sign of the net values, i.e. a zero value gives an off balance position, a positive value gives an asset position and a negative value gives a liability position.

The hedge portfolio forms the conceptual basis of the no-arbitrage pricing theory. In the case of forward contracts a *static hedge portfolio* is sufficient to exactly replicate (duplicate) the forward values over time. The static portfolio consists of buying one stock (asset), financing the purchase price with a loan (liability) and holding this asset/liability hedge portfolio until the forwards expiration date. In the absence of transaction costs the static buy-and-hold portfolio gives the same forward value development as the long position in the forward contract.

Equipped with this profound understanding of the initial and subsequent pricing of the stock forward contract the transformation of the OntoREA© PIM model into PSM models can be addressed. According to the chosen platform there are different possible PSM models. In this article the popular MySQL database platform [12] is taken which specifies the *MySQL database model* as *MySQL PSM model*.

As in this article the focus lies on the transformation of the *Collective* conceptualization of derivative instruments, the *Collective* transformation from the OntoREA© PIM model into the MySQL PSM model is addressed now. The result of this transformation can be seen in Figure 5. At the center is the *Derivative_Instrument* table. The relationships to its phases are the links to left side located tables for *Asset*, *Liability* and *Off_Balance*. To each table there is a 0..1 cardinality so that the derivative instrument can but must not be in each phase, and each table has the *Derivative_ID* as

its primary and foreign key (pFK). The three phases of the derivative instruments are transformed into separate tables due to the phases' anti-rigidity. All three phase tables have the Boolean *Mark_to_Model* attribute specified as TINYINT data type in the MySQL PSM model. This attribute allows the switching between the representation of derivative instruments either in form of their hedge portfolio, i.e. the mark-to-model attribute is true, or in form of the portfolio's net value if the mark-to-model attribute is false. Furthermore the *Off_Balance* table does not contain the *Fair_Value* attribute like the *Asset* and *Liability* tables as its valueless property is automatically given whenever the net value of the hedge portfolio is neither positive nor negative.

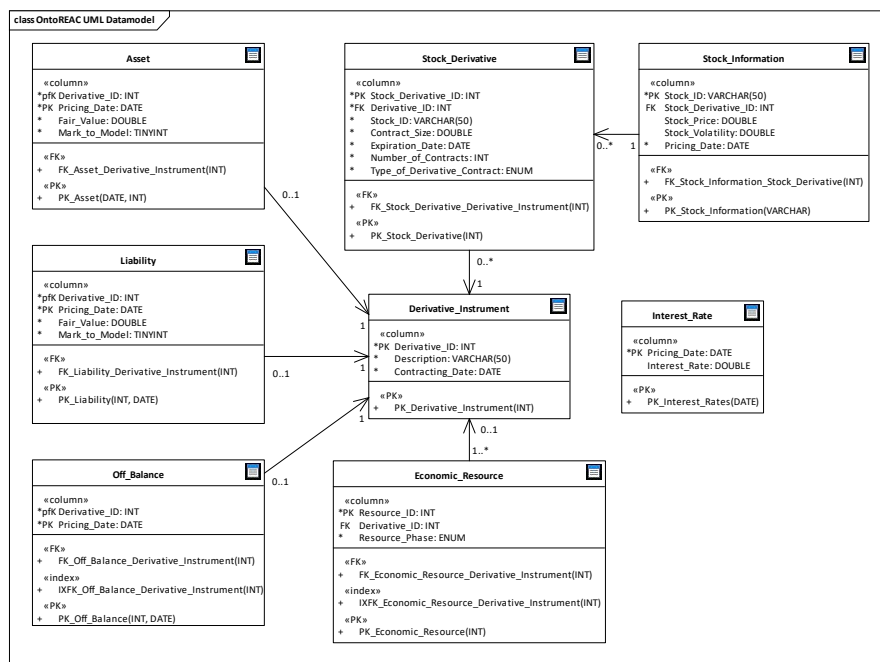


Figure 5: Relational Database Model – MySQL Database Model (PSM)

Next to the relationships to the balance sheet phases of the *Derivative_Instrument* class, Figure 5 also contains the relationship to the hedge portfolio. This relationship is set by the 0..1 cardinality to the *Economic_Resource* table so that one economic resource can but must not be associated with a derivative instrument. Furthermore this option is indicated by the missing * (NOT NULL CONSTRAINT) in the *Derivative_ID* foreign key (FK) attribute of the *Economic_Resource* table so that the relationship can be *Null*. On the other side of the relationship there are two economic resources assigned to the derivative instrument via the cardinality of 1..*. The two related economic resources are an asset as well as a liability position. To be precise, in the IFRS reporting standards not the hedge portfolio constituents but only the portfolio's net value is recognized in the balance sheet. Due to the rigid relationship of the hedge portfolio's constituting resources to their balance sheet phases, these resource phases are modeled via an enumeration with respect to asset, liability, equity and claim.

The final important point to mention in Figure 5 relates to the specification of the derivative instrument. The concrete specification of stock derivatives can be seen by the relationship of the *Derivative_Instrument* table to the *Stock_Derivative* table. The cardinality of 0..1 says that derivative instruments can but must not be a stock derivative. On the other side each stock derivative is a derivative instrument. The stock derivative's main defining attributes are the *Contract_Size*, the *Expiration_Date* and the *Type_of_Derivative_Contract*. The inclusion of the contract type information allows the differentiation between stock forwards and stock options, e.g. in form of stock calls and stock puts. The difference in the information base needed for pricing of forwards and options is the *Stock_Volatility* included in the *Stock_Information* table. This information is only relevant for stock options due to the optional right of the option buyer to execute the option at the expiration date in the case of European options [8] only in favorable constellations. As can be seen in Table 1 this information is not needed for pricing the stock forward contract. This difference between forwards and options is the reason why the forwards are members of the *unconditional derivative instrument type* and the options belong to the *conditional derivative instrument type*.

```

Algorithm. OCL invariant for the exclusivity of derivative instrument phases in PSM

context d: Derivative_Instrument inv exclusivePhaseReferences :
def : assetExists : Boolean =
  Asset.allInstances()->exist (a|a.Derivative_ID = d.Derivative_ID)
def : liabilityExists : Boolean =
  Liability.allInstances()->exist (l|l.Derivative_ID_d = d.Derivative_ID)
def : offbalanceExists : Boolean =
  Off_Balance.allInstances()->exist (o|o.Derivative_ID_d = d.Derivative_ID)
(assetExists xor liabilityExists) xor offbalanceExists

```

Table 2: OCL Exclusivity Constraint – MySQL Database Model (PSM)

Now a closer look onto the tables for the *Asset*, *Liability* and *Off_Balance* phases of the derivative instrument is taken. The 0..1 cardinality to each table specifies individual optional links. But what is missing in this specification is the exclusivity constraint, i.e. that the derivative instrument must be in one of the three tables at each point in time. Such a restriction can be specified [17] in the *Object Constraint Language* (OCL). In Table 2 the exclusivity constraint is defined as an OCL *invariant* for the *context* of the *Derivative Instrument* table. After defining the Boolean expressions for the existence of asset, liability and off balance, they are combined in an exclusive statement in order to express their exclusiveness. The combination of this OCL defined exclusivity constraint with the MySQL database PSM model in Figure 5 completely specifies the *Collective* conceptualization of derivative instruments in the MySQL PSM model.

After the OntoREA© PIM model is transformed into the MySQL PSM database model, the second transformation can be performed, i.e. the transformation from the MySQL PSM model into the RStudio Shiny dataframe ISM model. The dataframes in the ISM model are a composite data type in the R programming language that can contain several variables with different individual data types. Accordingly all tables in the MySQL PSM model can be represented as dataframes in the RStudio Shiny ISM model.

Figure 6 contains the RStudio Shiny dataframe ISM model which is visualized with the R package *datamodelr*. This model consists of 8 dataframes that have the same naming like the 8 tables in the MySQL PSM model. The dataframes of the three

derivative instrument phases, i.e. *Asset*, *Liability* and *Off_Balance* have the foreign key (annotated by the tilde ~) as their primary key (annotated by its underlining). This corresponds to the primary foreign key (pfK) specification in the MySQL PSM model.

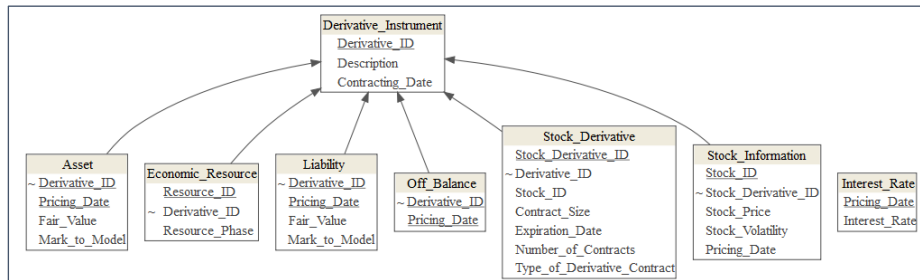


Figure 6: Data.frame Model – RStudio Shiny Data Model (ISM)

The exclusivity constraint with respect to the three derivative instrument phases derived from the corresponding OCL constraint is specified as *switch function*. The switch exclusivity constraint is given in Table 3. The exclusive assignment to one phase at each point in time is given by the hedge portfolio’s net value dependent specification of all phases as cases (case 1: positive net value -> asset; case 2: negative net value -> liability; case 3: else -> off balance) and the exclusiveness of the assignment within the switch function.

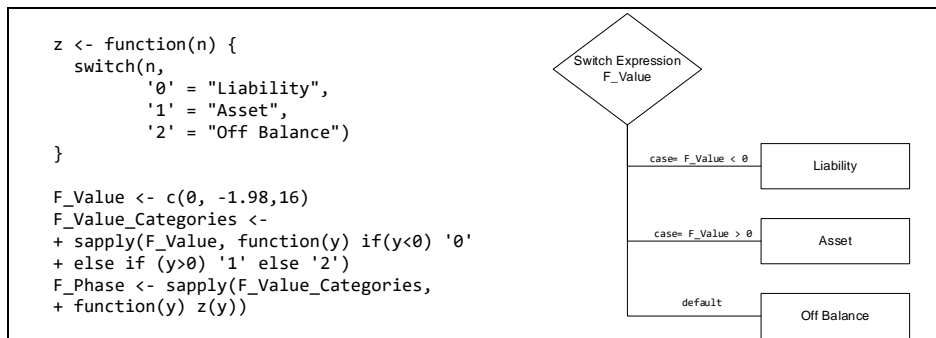


Table 3: Switch Exclusivity Constraint – RStudio Shiny Data Model (ISM)

4 The OntoREA© Model: RStudio Shiny ISM-based Application

The RStudio Shiny ISM dataframe model can easily be translated into an RStudio Shiny application (ISM-based application) as all dataframes in the RStudio Shiny ISM dataframe model (Figure 6) and the switch exclusivity constraint for the three derivative instrument phases (Table 3) are already specified in the R programming language.

Figure 7 shows in the top left corner the user interface for the initial pricing of the *prototypical RStudio Shiny application* which is based upon the RStudio Shiny ISM

dataframe model and which implements the switch exclusivity constraint. In the upper left panel the parameters for the initial pricing are specified, i.e. the initial interest rate of 4 %, the initial time to maturity of 12 months and the initial stock price of 100. The resulting no-arbitrage forward price of 104 is shown in the middle of the lower panel. Furthermore this panel also shows the stock asset value of 100, the loan liability value of -100 and the resulting net value of the hedge portfolio amounting to zero. According to the switch function this corresponds to an off balance phase assignment of the stock forward at its contracting date.

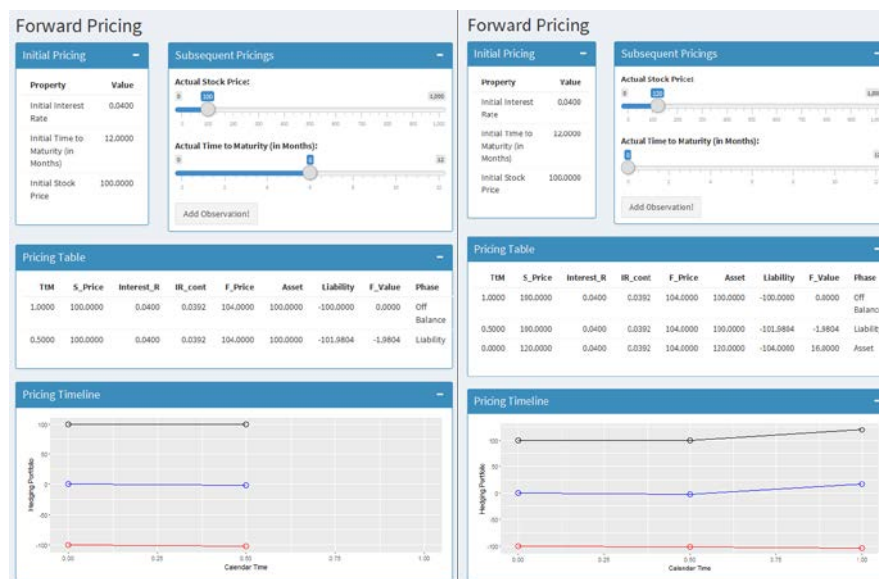


Figure 7: Prototypical ISM-based Application (Subsequent Pricings)

Figure 7 shows the user interface for the subsequent pricing of the stock forward for two different dates. In both dates it is assumed that the initial interest rate did not change compared to its original level of 4 %. So only the stock price and the time to maturity are changing over time. Their actual values are set with the sliders that are available in the subsequent pricing activity.

- The left hand side contains the input parameters and the results for the mid-term pricing at the end of June: The actual stock price is still 100 and the time to maturity reduced to 6 months. In this constellation the stock asset value is 100, the loan liability value is -101.98 and the resulting net value of the hedge portfolio amounts to -1.98. The negative portfolio value evaluated in the switch function gives a liability phase assignment of the stock forward.
- The right hand side contains the input parameters and the results for the final pricing at the end of December: The actual stock price rose to 120 and the time to maturity reduced to zero. In this constellation the stock asset value is 120, the loan liability value is -104 and the resulting net value of the hedge portfolio

amounts to +16. The positive portfolio value evaluated in the switch function gives an asset phase assignment of the stock forward.

This prototypical ISM-based application illustrates the *stochastic modal temporal behavior* of forward contract which is captured in OntoREA© via the *Collective* OntoUML meta-property of derivative instruments.

Although only a stock forward contract is used in the running example, it has to be mentioned that the *Collective* conceptualization also holds for derivative instruments in form of options. Options are defined as follows: *Contracts that give the purchaser the right, but not the obligation, to buy (call option) or sell (put option) a specified quantity of a particular financial instrument, commodity, or foreign currency, at a specified price (strike price), during or at a specified period of time. These can be individually written or exchange-traded. The purchaser of the option pays the seller (writer) of the option a fee (premium) to compensate the seller for the risk of payments under the option.*[5]

The no-arbitrage pricing theory also holds for options. For options a *dynamic hedge portfolio* exactly replicates (duplicates) the option value over time. In the dynamic hedging fractions of the stock are bought and they are partially financed by a loan liability. Over time the investment and financing has to be continuously adjusted according to the revealing stock prices. At the expiration date the hedge portfolio gives the same result as the initial buying an option contract.

5 Conclusions

The title of this article indicates a design science research methodological (DSRM) demonstration and evaluation [4] associated with the OntoREA© accounting and finance model. This investigation necessitates due to the missing real case usage and the missing adequacy evaluation in the original [1] OntoREA© artifact. By using a stock forward contract as running example the hedge portfolio meaning of the derivative instrument's *Collective* conceptualization with a *MemberOf* relationship to two *Economic Resources* was enlightened and illustrated with a real derivative instrument.

The *Collective* conceptualization served as starting point in the MDD's PIM model which was successively transformed in two steps into a MySQL PSM database model and an RStudio Shiny ISM dataframe model. The exclusivity constraint of the derivative instrument's asset, liability and off balance phases were specified in the OCL language for the MySQL PSM model and the switch function in the R programming language for the RStudio Shiny ISM model. The setup of the ISM model in the R programming language finally allowed a direct coding that resulted in a prototypical RStudio Shiny application.

Overall, the consistent usage of the *Collective* conceptualization in the MDD context and the real case demonstration of its appearance in the different model transformations demonstrate the conceptualization's usage required in the corresponding DSRM activity. The DSRM evaluation activity is performed by showing with the stock forward example that the derivative instruments are adequately represented with respect to their

identity-related peculiarity, their temporal modal peculiarities in the asset liability and off-balance phases as well as the exclusivity constraint defined thereupon.

Summing up. The two research objectives are met as follows: By using a rolling example the OntoREA©'s *Collective* conceptualization of derivative instruments is presented in a numerical and easy-to-grasp manner. With the demonstrated usage and adequacy evaluation of the *Collective* conceptualization the second research objective is met. By fulfilling this objective the DSRM required real case usage demonstration and adequacy evaluation are provided which are missing in the most innovative part of the original OntoREA© artifact. Having provided this kind of DSRM validity evidence the next step should be a formal validation of the OntoREA© model for detecting possible misspecifications by performing e.g. a model simulation in the logic-based Alloy language.

References

1. Fischer-Pauzenberger, C., Schwaiger, W.S.A.: The OntoREA© Accounting and Finance Model: Ontological Conceptualization of the Accounting and Finance Domain. In: The 36th International Conference on Conceptual Modeling (2017).
2. Geerts, G.L., McCarthy, W.E.: Policy Level Specifications in REA Enterprise Information Systems. *J. Inf. Syst.* 20, 37–63 (2006).
3. Fischer-Pauzenberger, C., Schwaiger, W.S.A.: The OntoREA Accounting Model: Ontology-based Modeling of the Accounting Domain. *Complex Syst. Informatics Model. Q.* (2017).
4. Geerts, G.L.: A design science research methodology and its application to accounting information systems research. *Int. J. Account. Inf. Syst.* 12, 142–151 (2011).
5. IFRS: International Accounting Standard 39 Financial Instruments: Recognition and Measurement, http://ec.europa.eu/internal_market/accounting/docs/consolidated/ias39_en.pdf.
6. IAS 39 — Financial Instruments: Recognition and Measurement, <https://www.iasplus.com/en/standards/ias/ias39>.
7. Guizzardi, G.: Ontological Foundations for Structural Conceptual Model, <http://doc.utwente.nl/50826>, (2005).
8. Black, F., Scholes, M.: The Pricing of Options and Corporate Liabilities. *J. Polit. Econ.* 81, 637 (1973).
9. Merton, R.C.: Theory of rational Theory option pricing. *Bell J. Econ.* 4, 141–183 (1973).
10. Cox, J.C., Ross, S.A., Rubinstein, M.: Option pricing: A simplified approach. *J. financ. econ.* 7, 229–263 (1979).
11. Pastor, O., España, S., Panach, J.I., Aquino, N.: Model-driven development. *Informatik-Spektrum.* 31, 394–407 (2008).
12. Sparks, G.: Database modelling in UML, www.sparxsystems.com/downloads/whitepapers/Database_Modeling_In_UM

- L.pdf.
13. An Introduction to R, <https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf>.
 14. Hillebrand, J., Nierhoff., M.H.: *Mastering RStudio—Develop, Communicate, and Collaborate with R*. Packt Publishing Ltd. (2015).
 15. McCarthy, W.E.: *The REA Accounting Model - A Generalized Framework for Accounting Systems in a Shared Data Environment*, (1982).
 16. Guizzardi, G., Wagner, G., Almeida, J.P.A., Guizzardi, R.S.S.: Towards ontological foundations for conceptual modeling: The unified foundational ontology (UFO) story. *Appl. Ontol.* 10, 259–271 (2015).
 17. Rybola, Z., Pergl, R.: Towards OntoUML for Software Engineering: Transformation of Anti-rigid Sortal Types into Relational Databases. In: *International Conference on Model and Data Engineering*. pp. 1–15. Springer International Publishing (2016).
 18. Sales, T., Barcelos, P., Guizzardi, G.: Identification of semantic anti-patterns in ontology-driven conceptual modeling via visual simulation. *4th Int. Work. Ontol. Inf. Syst. (ODISE 2012)*, Graz, Austria. (2012).
 19. Ontology Project: UFO-A Specification, <http://ontology.com.br/ufo-a/spec/>.