BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **20**, No 2
Sofia • 2020 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2020-0019

The Implementation of Credit Risk Scorecard Using Ontology Design Patterns and BCBS 239

Jalil Elhassouni[1], *Abderrahim El qadi*[2], *Yasser El madani El alami*[3], *Mohamed El haziti*[1]

[1]*LRIT-CNRST* (*URAC'29*)*, Faculty of Sciences, Rabat IT Center, Mohammed V University of Rabat, Morocco*
[2]*High School of Technology in Sale, LCS, Faculty of Science, Mohammed V University of Rabat, Morocco*
[3]*ENSIAS, Mohammed V University of Rabat, Morocco*
E-mails: jalil.elhassouni@gmail.com abderrahim.elqadi@um5.ac.ma yasser.alami@um5.ac.ma melhaziti@yahoo.fr

Abstract**:** *Nowadays information and communication technologies are playing a decisive role in helping the financial institutions to deal with the management of credit risk. There have been significant advances in scorecard model for credit risk management. Practitioners and policy makers have invested in implementing and exploring a variety of new models individually. Coordinating and sharing information groups, however, achieved less progress. One of several causes of the 2008 financial crisis was in data architecture and information technology infrastructure. To remedy this problem the Basel Committee on Banking Supervision* (*BCBS*) *outlined a set of principles called BCBS 239. Using Ontology Design Patterns* (*ODPs*) *and BCBS 239, credit risk scorecard and applicant ontologies are proposed to improve the decision making process in credit loan. Both ontologies were validated, distributed in Ontology Web Language* (*OWL*) *files and checked in the test cases using SPARQL. Thus, making their* (*re*)*usability and expandability easier in financial institutions. These ontologies will also make sharing data more effective and less costly.*

Keywords**:** *Ontology design patterns, OWL, credit risk scorecard, decision support, BCBS 239.*

1. Introduction

In aftermath of the 2008 financial crisis, the Basel Committee on Banking Supervision (BCBS) [1] created by the central bank governors of the G10 and Switzerland reports that the weakest links in banking compliance data are due to problems in data architecture and Information Technology (IT), i.e., inadequacy or non-existence of common vocabulary, thereby, common semantics are messing. As a consequence, for many banking assertions or the decisions that depend on the data cannot be reliable. In order to enhance risk management and decision-making processes at financial institutions, the BCBS outlined a set of 14 principles (BCBS 239) [2]. L. Prorokowski and H. Prorokowski [3] found that BCBS 239 is a unique opportunity to modernize IT platforms and enhance risk data management. It proposes the best practice solutions (Fig. 1) for banks and other financial services to be able to implement it. As presented in Fig. 1, the end-user solutions are in general, based on artificial intelligence in order to assess potential future scenarios, predict behaviour in unprecedented ways and to extract meaningful information from the data. However, Artificial Intelligence (AI) is not enough to reach the desired results since a good content theory of the domain is not implemented. In fact, as shown in, Chandrasekaran, Josephson and Benjamins [4] state that once a good content theory is available, many different AI models might be used equally well to implement effective systems, all using essentially the same content. Thus, in order to deal with content theory and domain representation, a great interest was shown in ontologies. As Gruber [5] states, an ontology is a formal, explicit specialization of shared conceptualization. Therefore, they provide potential terms for describing well the knowledge domain to benefit many different mechanisms such as rule systems, fuzzy logic, machine learning, deep learning, neural networks or natural language processing to be used and implemented using and sharing the same content.

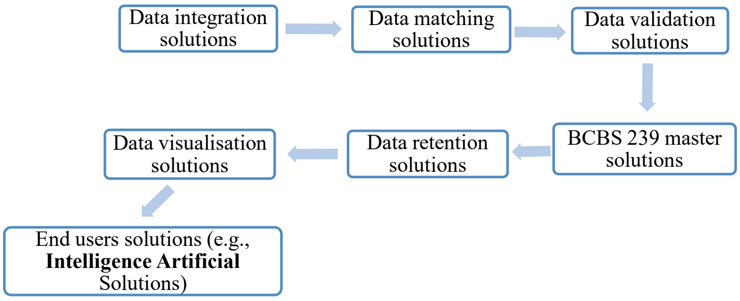

Data integration solutions
Data matching solutions
Data validation solutions
BCBS 239 master solutions
Data retention solutions
Data visualisation solutions
End users solutions (e.g., **Intelligence Artificial** Solutions)

Fig. 1. Solutions for the implementation of BSBS 239

In addition, ontologies permit to facilitate the data integration and matching as proved by Cruz, Xiao and Advis Lab [6] and Gagnon [7]; Subirats, Gil and García [8] and al. used them to personalize the visualization of the data. Ontologies constitute one of the most suitable solutions to implement BCBS 239. In this paper, we demonstrate the implementation of credit risk scorecard and applicant ontologies based on ODPs as a stepping stone towards a more complex and sophisticated framework for credit risk management and decision making processes at financial institutions. The credit risk scorecard and applicant ontologies are extended, enriched using logical axioms, validated, distributed in Ontology Web Language (OWL) files, and checked in the test cases using SPARQL.

The remainder of this paper is organized as follows: in Section 2, we review the relevant literature on the topic, followed by a detailed account of our approach in Section 3. Section 4, deals with limitations of this work. Finally, our paper end with some concluding remarks and future work.

2. Related work

In the last decade, the use of ontologies in the academia and industry fields has been exponentially expanded. Dwivedi and Kumar [9] used them to manage and process the files/documents of various departments and sections of higher educational institutions in paperless environment. Forbes et al. [10] applied ontologies in the healthcare field to provide navigable interactive healthcare guidance to assist ethnic minorities and aboriginal patients disadvantaged by languages, with limited use of the written text, The Gene Ontology Consortium [11] used them to represent and update continuously information concerning gene products and their functions and Yan, Zhang and Liu [12] used them to develop the ontology of the traditional Chinese medicine. They are also used by Kontopoulos et al. [13] in renewable energy to help no technical user to choose the domestic solar hot water system according to their needs, containing up-to-date information on its components and interrelationships, installation costs, etc. They have played a key role in the data management process from the knowledge representation as demonstrated by The Gene Ontology Consortium [11] passing through the integration of heterogeneous data sources as illustrated by Cruz, Xiao and Advis Lab [6] and Monnin et al. [14], data cleaning by Brüggemann [15] and the data publishing and access by Buranarach et al. [16], ending with big data management by Eine, Jurisch and Quint [17] and access control for cloud data by Michael, Kothandaraman and Kaliyan [18].

The ontology was also used in the financial industry; the unique and the reference one in the field is the Financial Industry Business Ontology (FIBO) [19]. It is a family of ontologies developed to standardize the terms used in the financial services industry. However, FIBO ontologies have some drawbacks such as:

1. The methodology used to build and maintain FIBO based on Agile Development Software principles and practices is in accordance with Winosky and the FIBO Leadership Team [20]. Initially, it was designed and adapted to support software development not artifacts such as ontologies. However, Abdelghany, Darwish and Hefni [21] make clear that the works done in this field are not mature enough to be largely approved by organizations. In addition, it was identified that ontology refinement and validation phase can be developed iteratively, whereas Axelsson et al. [22] confirm that other development phases involved larger challenges for adopting agile methods.

2. The absence of clear documentation and design conventions in the FIBO standards such the documentation made by the Gene Ontology (GO) in [23] and by Enterprise Ontology (TOVE) in [24] give the intention that certain classes and properties may be considered to be polysemic, which makes their re-usability in different financial industry applications by end-user developers a risky, complex task, and time consuming as Bennett [25] shows; such policy goes beyond the goals of open source communities.

3. Bennett [25] makes clear that the absence of a normalized and referenced method, leaving it to the individual modeller to find the best ways to represent things in problem domain.

4. The implementation of ontologies requires a lot of manpower and time to manually integrate them as described by Zaino [26].

To address some of these limitations we extend the work of El Hassouni et al. [27] based on ontology design patterns. It will permit our work to be flexible, expandable and re-usable as proved by Gangemi and Presutti [28], as well as reduce the cost (According to the 2015 McKinsey report, it is estimated that the implementation of BCBS 239 principles would cost approximately 230 million USD per G-SIB and 75 million USD in each D-SIB), improve the profitability and manage the risks. Our proposal, thus, is the keystone of credit risk management platform.

3. Credit risk scorecard pattern

In this paper we will develop an expandable and re-usable credit risk scorecard and applicant ontologies, which will be a keystone of ontology-based credit risk management platform through the implementation of the BCBS principles. This solution will permit to make good content theory and domain representation of the credit risk scorecard. It will help the financial institutions to develop a platform which conforms to the principles of BCBS 239 as well as provide user-ends solutions made available by AI mechanisms (e.g., machine learning and deep learning) which will be able to asses potential future scenarios, predict behavior in unprecedented ways, and extract meaningful information from the data. To realize this, we will extend and enrich the work initiated by El Hassouni et al. [27]. The latter has developed a credit risk scorecard and applicant ontologies based on ODPs. This work was the result of a group work by various experts: domain expert, specialists in database addressed by modeling, and ontology engineer working on ODP-based modeling process. The reconciliation of the differing perceptions of domain experts to their topic, credit risk scorecard and decision support tool – as data providers – and the Competency Questions (CQs) have led to a flexible, expandable and re-usable credit score card and applicant ontologies (Fig. 2).

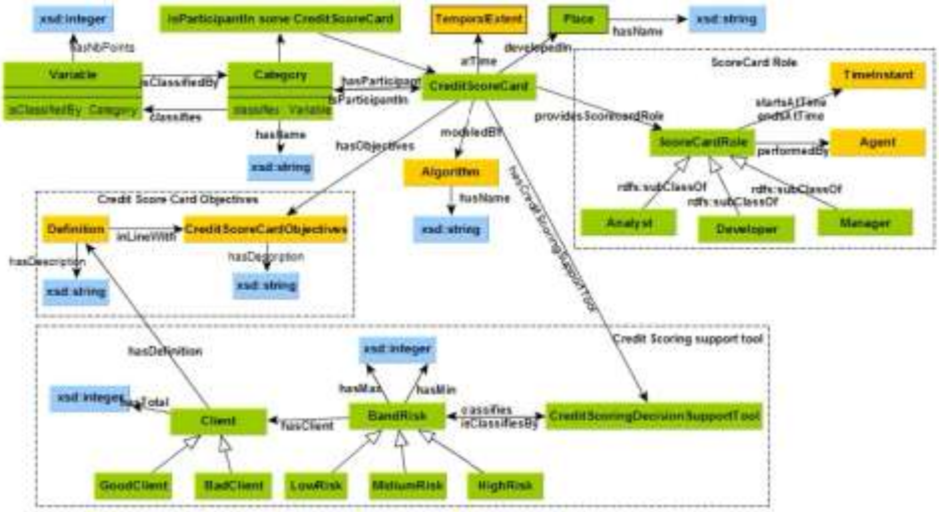


Fig. 2. The credit scorecard model [27]

Since the work on the credit risk scorecard and applicant ontologies is the same, we will outline here the credit risk ontology.

3.1. Axiomatization

The model depicted in Fig. 2 is not the scorecard ontology, but it gives the representation of credit scorecard classes. Representation alone is not sufficient to sustain all the weight of the philosophy of science's building found in Ferrario [29]. Suppes [30] declares that the best way to define this class of structure (e.g., credit risk scorecard) is via axiomatization. The axiomatic method constitutes a common framework for the discussion of scientific problems for people coming from different backgrounds, and even for people working in different but strongly related branches of the same discipline (e.g., credit risk scorecard analyst, manager and developer). One of the multiples advantages of axiomatization is standardization. It can create standard terminology and standard methods of conceptual analysis in various branches of science (one of the main uses of ontology in computer science is that of making explicit the intended meaning of a terminology; so, the effect towards finding a standard in the scientific practices that aim at explicitness present in the studies on ontology have both the goal of enhancing communication through the use of a formal ontology). A further advantage is that it has the value of the heuristic method. It can even be applied to disciplines in the process of formulation.

For this purpose, we use axioms in description logics as summarized by Krötzsch, Simancik and Horrocks [31] to describe the pattern and translate them to OWL. The axioms in ontology do not provide complete information; for that, the description logics have been designed [31]. Now, we know from the previous work of El Hassouni et al. [27] that the credit risk scorecard pattern consists of the following components:

1. The (re) use of the Agent Role pattern which reoccurs when modeling manager, analyst and developer of credit risk scorecard.
2. The (re) use of the Event pattern to model credit risk scorecard.
3. The (re) use of the Participation pattern to model credit risk scorecard variables.
4. The (re) use of the Classification Pattern to model categories of variables and credit scoring decision support tool.

To reuse Agent role, Event, Participation and Classifications patterns in practice, there is one way, importing the OWL serialization of four patterns into the OWL serialization of the credit risk scorecard. This means that all axioms and ontological commitments imposed by the patterns will be employed by credit scorecard pattern. Furthermore, the reuse of patterns may necessitate some adjustments and modifications following the use cases and modeling requirements. Due to limited space we cannot describe the full patterns, we can only outline Agent role and Event patterns.

3.1.1. Agent role pattern

Axioms for the Agent Role pattern are depicted in Fig. 3. The Axioms (1)-(5) capture the domain, range and the scope domain and range of each property in the pattern:

providesAgentRole, performedBy, startsAtTime and endsAtTime (e.g., the AgentRole and Agent entities provides the domain constraint and range respectively for the performedBy property). The Axiom (6) asserts that every pair of classes amongst AgentRole, Agent and TimeInstant are pair wise disjoint. Hence, nothing can be simultaneous: an agent and an agent role, or an agent and time instant, or agent role and time instant.

$$\top \subseteq \forall\ \mathbf{providesAgentRole.AgentRole} \tag{1}$$

$$\exists\ \mathbf{performedBy.Agent} \subseteq \mathbf{AgentRole} \tag{2}$$

$$\mathbf{AgentRole} \subseteq \forall\ \mathbf{performedBy.Agent} \tag{3}$$

$$\mathbf{AgentRole} \subseteq \forall\ \mathbf{starts.AtTime.TimeInstant} \sqcap \forall\ \mathbf{ends.AtTime.TimeInstant} \tag{4}$$

$$\top \subseteq \forall\ \mathbf{providesAgentRole.AgentRole} \tag{5}$$

$$\mathbf{DisjointClasses(AgentRole,\ Agent,\ TimeInstant)} \tag{6}$$

Fig. 3. Axiomatization for Agent role pattern

3.1.2. Event pattern

Axioms for the Event pattern depicted in Fig. 4, note that the event pattern imports the Agent Role pattern. Axiom (7) affirms that an event occurs at some temporal extent and some place. The axioms (8)-(10) express the domain, range and scoped domain and range for the properties: atTime and atPlace, while the axiom (10) asserts that every pair of classes amongst Event, TemporalExtent, Place, AgentRole and Agent are pair wise disjoint.

$$\mathbf{import:\ (1),\ (2),\ (3),\ (4),\ (5),\ (6)}$$

$$\mathbf{Event} \subseteq \exists\ \mathbf{atPlace.Place} \sqcap \exists\ \mathbf{atTime.TemporalExtent} \tag{7}$$

$$\exists\ \mathbf{atTime.TemporalExtent} \sqcup \exists\ \mathbf{atPlace.Place} \tag{8}$$

$$\mathbf{Event} \subseteq \forall \mathbf{atTime.TemporalExtent} \sqcup \forall \mathbf{atPlace.Place} \tag{9}$$

$$\mathbf{DisjointClasse(Event,\ TemporalExtent,\ Place,\ AgentRole,\ Agent)} \tag{10}$$

Fig. 4. Axiomatization for Event pattern

3.1.3. Axioms implementation

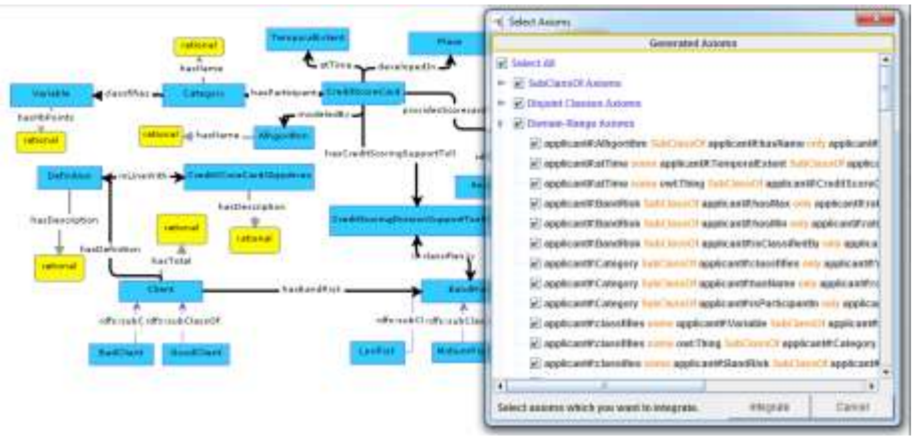


Fig. 5. The implementation of credit scorecard ontology axioms using Sarker, Krisnadhi and Hitzler's OWLAx plugin [32]

As our team comprises multi-domain experts, we choose to use a visual tool of Sarker, Krisnadhi and Hitzler [32] to implement axioms. The latter permit to generate most of axioms and add them automatically to the ontology (Fig. 5).

3.2. Validation of credit risk scorecard ontology

Validation of credit risk scorecard ontology is triply. Firstly, we make the diagnosis and repair of the credit risk scorecard ontology for quality validation. Secondly, we reason the logical consistency of the ontology. Thirdly, we make queries answering for usability validation of the ontology.

3.2.1. Quality validation

The validation of the quality of an ontology is an important part of ontology development. It is especially important when an ontology is expendable and re-usable.

We make the diagnosis and repair of credit risk scorecard ontology in order to check the technical quality of the ontology. However, manual diagnosis is always a tedious and time-consuming task. Therefore, we opt for using Poveda Villalón's [33] automatic online evaluation framework for OWL ontologies for many reasons: the online framework was accepted from the Semantic Web community, used by a high number of users worldwide, integrated by many systems, and it, further, enlarges the list of errors detected by most recent and available tools that Poveda Villalón [34] developed and implemented.

The credit risk scorecard ontology modeling does not contain any anomalies in which the quality of the ontology is checked through OOPS.

3.2.2. Consistency validation

We mean by consistency ontology validation task the validation of the ontology through reasoner over three standard inference services fixed by the unique international completion in the field, OWL Reasoner Evaluation (ORE) [35]:

1. Consistency checking.
2. Classification.
3. Realization.

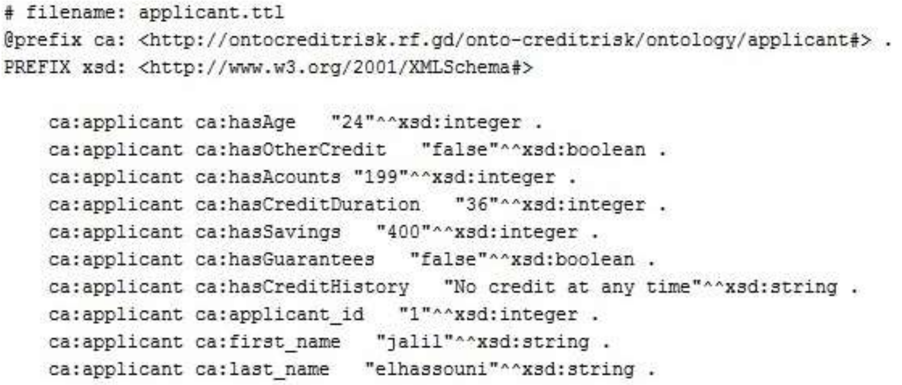
The credit risk scorecard ontology does not contain any contradictory facts in which logical consistency of the ontology is checked through HermiT tool provided by Glimm et al. [36] as one of the best reasoners detailed in the competition OWL Reasoner Evaluation (ORE) report [35] using its OWL API version [37].

3.2.3. Usability validation

We also use query responses to validate the credit risk scorecard ontology usability. To achieve this objective, we will use CQs as cited below at the assertion level by relying on SPARQL (Standard Protocol And RDF Query Language) as described by Blomqvist, Seil Sepour and Presutti [38].

What is the risk of an applicant who is under 25 applying for credit for the first time at the institution with no other credit, no non-payments, with an account having slightly positive balance (but less than €200), with a small amount of savings (less than €500), and without a guarantor applying for credit for 36 months?

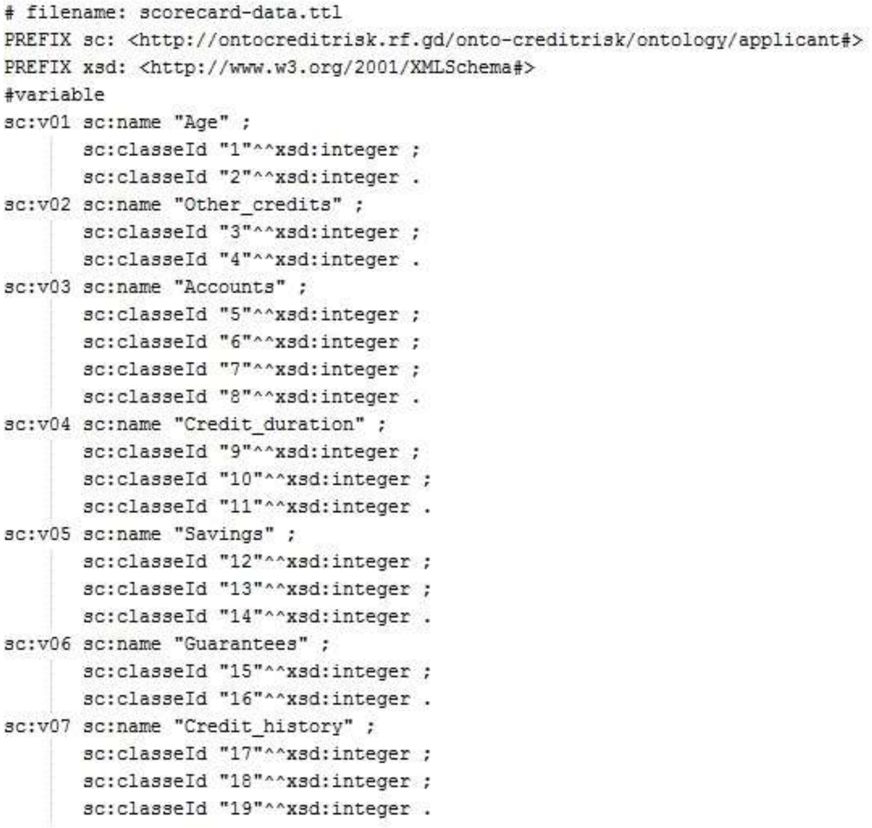
A variety of tools are available as open source on which SPARQL can be executed and tested. For our case, we choose Jana framework with ARQ developed by Apache Software Foundation [39]. Fig. 6 shows the representation of applicant data in turtle format and Fig. 7 shows the representation of data credit scorecard model in turtle format.



```
# filename: applicant.ttl
@prefix ca: <http://ontocreditrisk.rf.gd/onto-creditrisk/ontology/applicant#> .
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

   ca:applicant ca:hasAge   "24"^^xsd:integer .
   ca:applicant ca:hasOtherCredit   "false"^^xsd:boolean .
   ca:applicant ca:hasAcounts "199"^^xsd:integer .
   ca:applicant ca:hasCreditDuration   "36"^^xsd:integer .
   ca:applicant ca:hasSavings   "400"^^xsd:integer .
   ca:applicant ca:hasGuarantees   "false"^^xsd:boolean .
   ca:applicant ca:hasCreditHistory   "No credit at any time"^^xsd:string .
   ca:applicant ca:applicant_id   "1"^^xsd:integer .
   ca:applicant ca:first_name   "jalil"^^xsd:string .
   ca:applicant ca:last_name   "elhassouni"^^xsd:string .
```

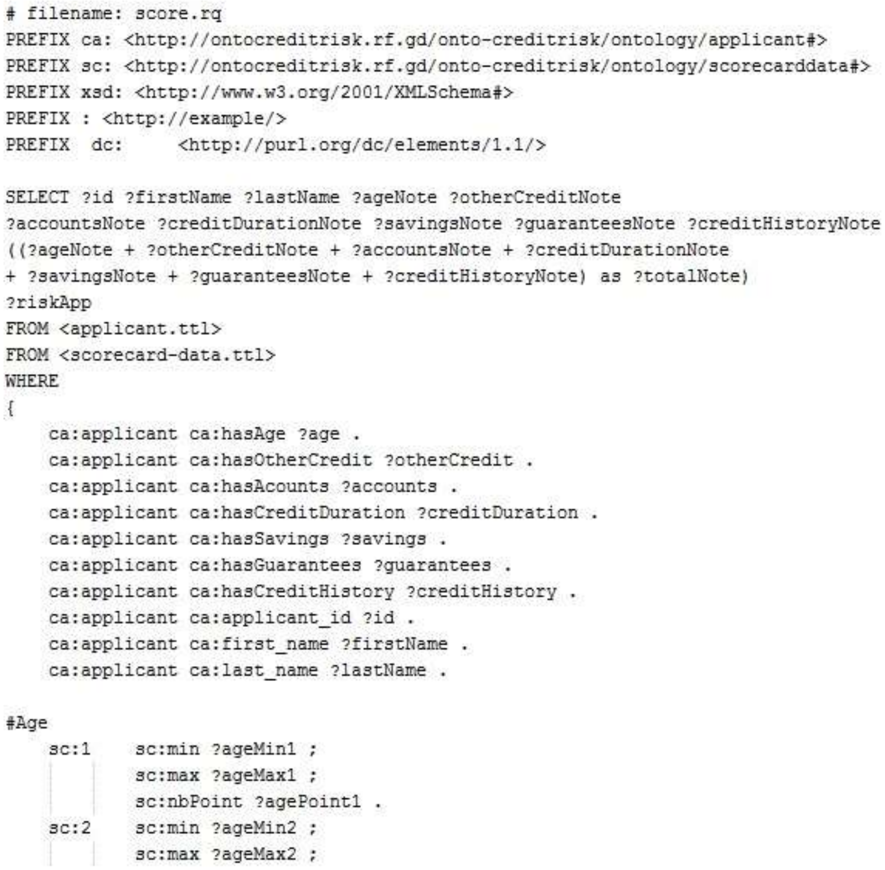
Fig. 6. The representation of applicant in turtle format – partial view



```
# filename: scorecard-data.ttl
PREFIX sc: <http://ontocreditrisk.rf.gd/onto-creditrisk/ontology/applicant#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
#variable
sc:v01 sc:name "Age" ;
       sc:classeId "1"^^xsd:integer ;
       sc:classeId "2"^^xsd:integer .
sc:v02 sc:name "Other_credits" ;
       sc:classeId "3"^^xsd:integer ;
       sc:classeId "4"^^xsd:integer .
sc:v03 sc:name "Accounts" ;
       sc:classeId "5"^^xsd:integer ;
       sc:classeId "6"^^xsd:integer ;
       sc:classeId "7"^^xsd:integer ;
       sc:classeId "8"^^xsd:integer .
sc:v04 sc:name "Credit_duration" ;
       sc:classeId "9"^^xsd:integer ;
       sc:classeId "10"^^xsd:integer ;
       sc:classeId "11"^^xsd:integer .
sc:v05 sc:name "Savings" ;
       sc:classeId "12"^^xsd:integer ;
       sc:classeId "13"^^xsd:integer ;
       sc:classeId "14"^^xsd:integer .
sc:v06 sc:name "Guarantees" ;
       sc:classeId "15"^^xsd:integer ;
       sc:classeId "16"^^xsd:integer .
sc:v07 sc:name "Credit_history" ;
       sc:classeId "17"^^xsd:integer ;
       sc:classeId "18"^^xsd:integer ;
       sc:classeId "19"^^xsd:integer .
```

Fig. 7. The representation of credit scorecard in turtle format – partial view

In the query below, Fig. 8 calculates the total score of the applicant and associate his credit risk.



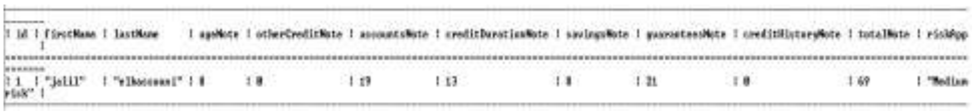
```
# filename: score.rq
PREFIX ca: <http://ontocreditrisk.rf.gd/onto-creditrisk/ontology/applicant#>
PREFIX sc: <http://ontocreditrisk.rf.gd/onto-creditrisk/ontology/scorecarddata#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX : <http://example/>
PREFIX  dc:     <http://purl.org/dc/elements/1.1/>

SELECT ?id ?firstName ?lastName ?ageNote ?otherCreditNote
?accountsNote ?creditDurationNote ?savingsNote ?guaranteesNote ?creditHistoryNote
((?ageNote + ?otherCreditNote + ?accountsNote + ?creditDurationNote
+ ?savingsNote + ?guaranteesNote + ?creditHistoryNote) as ?totalNote)
?riskApp
FROM <applicant.ttl>
FROM <scorecard-data.ttl>
WHERE
{
    ca:applicant ca:hasAge ?age .
    ca:applicant ca:hasOtherCredit ?otherCredit .
    ca:applicant ca:hasAcounts ?accounts .
    ca:applicant ca:hasCreditDuration ?creditDuration .
    ca:applicant ca:hasSavings ?savings .
    ca:applicant ca:hasGuarantees ?guarantees .
    ca:applicant ca:hasCreditHistory ?creditHistory .
    ca:applicant ca:applicant_id ?id .
    ca:applicant ca:first_name ?firstName .
    ca:applicant ca:last_name ?lastName .

#Age
    sc:1    sc:min ?ageMin1 ;
            sc:max ?ageMax1 ;
            sc:nbPoint ?agePoint1 .
    sc:2    sc:min ?ageMin2 ;
            sc:max ?ageMax2 ;
```

Fig. 8. SPARQL query executed in Jana framework with ARQ – partial view

The following output is obtained after executing the above code (Fig. 9).



```
-----------------------------------------------------------------------------------------------------------------------------------------------------------------
-------
| id | firstName | lastName     | ageNote | otherCreditNote | accountsNote | creditDurationNote | savingsNote | guaranteesNote | creditHistoryNote | totalNote | riskApp
 |
=================================================================================================================================================================
=======
| 1  | "Jalil"   | "elhassani" | 8       | 0               | 19           | 13                 | 8           | 21             | 0                 | 69        | "Medium
risk" |
-----------------------------------------------------------------------------------------------------------------------------------------------------------------
```

Fig. 9. SPARQL query executed in Jana framework with ARQ

From the answer query Fig. 9 over credit risk scorecard and applicant ontologies, we infer that our ontologies are valid for utilization.

4. Discussion

This work identifies and formalizes classes and relations in credit risk scorecard and applicant patterns. It demonstrates that these patterns are suitable for describing the data contents of credit risk scorecard and applicants.

Encoding credit risk scorecard and applicants using ontology design patterns is the basis of credit risk platform based on ontology design patterns. It will help financial institutions to develop a platform, which conforms to the principles of BCBS 239. It will equally provide user-ends solutions made available by AI mechanisms (e.g., machine learning and deep learning) which will be able to assess potential future scenarios, predict behavior in unprecedented ways, and extract meaningful information from the data.

5. Conclusions

In this paper, we have laid the foundation for ontology-based credit risk management platform, in order to enhance credit risk management and decision-making process at financial institutions. We develop a rich, expandable and re-usable credit risk scorecard and applicant ontologies which are based on ontology design patterns (ODPs) to improve the decision-making process in credit loan. This will lead financial institutions to develop a credit risk management platform which conforms to the principles of BCBS 239. Both ontologies were validated, distributed in Ontology Web Language (OWL) files and checked in the test cases using SPARQL. We extended the ontologies which have been developed by El Hassouni et al. [27] and enriched them using logical axioms. They were triply validated; the authors:

1. Made diagnosis and repair of credit risk scorecard ontology in order to check the technical quality of the ontology.
2. Reason logical consistency of the ontology (consistency checking, classification and realization).
3. Made queries answering for usability validation of the ontology.

To facilitate the re-uses and the extension of these ontologies, they have been distributed in Ontology Web Language (OWL) files.

Later on, our focus will be on ontology-based credit risk scorecard development. Thereafter, we can use Artificial Intelligence (machine learning and deep learning) to predict behavior of borrowers and new applicants in unprecedented ways and in real time in order to minimize the risks of financial institutions and enhance their competitivity in the financial market.

References

1. BCBS (n.d.). The Basel Committee – Overview. Retrieved September 2019. **https://www.bis.org/bcbs/index.htm**
2. BCBS (n.d.). Principles for Effective Risk Data Aggregation and Risk Reporting. Retrieved September 2019. **https://www.bis.org/publ/bcbs239.pdf**
3. Prorokowski, L., H. Prorokowski. Solutions for Risk Data Compliance under BCBS 239. – Journal of Investment Compliance, 2 November 2015, pp. 66-77.
4. Chandrasekaran, B., J. Josephson, R. Benjamins. What Are Ontologies, and Why Do We Need Them? – Intelligent Systems and Their Applications, IEEE, Vol. **14**, 1999, pp. 20-26.
5. Gruber, T. A Translation Approach to Portable Ontology Specifications. – Knowledge Acquisition, 1993, pp. 199-220. DOI:10.1006/knac.1993.1008.

6. Cruz, I., H. Xiao, Advis Lab. The Role of Ontologies in Data Integration. – Jounal of Engineering Intelligent Systems, Vol. **13**, 2005, No 4, pp. 1-18.
7. Gagnon, M. Ontology-Based Integration of Data Sources. – In: Proc. of 10th International Conference on Information Fusion, Quebec, Canada, 2007, pp. 1-8.
8. Subirats, L., R. Gil, R. García. Personalization of Ontologies Visualization: Use Case of Diabetes. – In: G. Alor-Hernández, J. Sánchez-Cervantes, A. Rodríguez-González, R. Valencia-García, Eds. Current Trends in Semantic Web Technologies: Theory and Practice, Vol. **815**, 2019, pp. 3-24.
9. Dwivedi, S., A. Kumar. Development of University Ontology for aSPOCMS. – Journal of Emerging Technologies in Web Intelligence, Vol. **5**, August 2013, No 3, pp. 213-221.
10. Forbes, D., P. Wongthongtham, J. Singh, S. Thompson. Ontology Supported Assistive Communications in Healthcare. – Communications of the Association for Information Systems, Vol. **34**, 2014, pp. 297-322.
11. The Gene Ontology Consortium. The Gene Ontology Resource: 20 Years and Still GOing Strong. – In: Nucleic Acids Res., 2018. DOI:10.1093/nar/gky1055.
12. Yan, H., F. Zhang, B. Liu. Granular Computing Based Ontology Learning Model and Its Applications. – Cybernetics and Information Technologies, Vol. **16**, 2016, No 4, pp. 29-44.
13. Kontopoulos, E., G. Martinopoulos, D. Lazarou, N. Bassiliades. An Ontology-Based Decision Support Tool for Optimizing Domestic Solar Hot Water System Selection. – Journal of Cleaner Production, Vol. **112**, 2016, No 5, pp. 4636-4646.
14. Monnin, P., C. Jonquet, J. Legrand, A. Napoli. PGxO: A Very Lite Ontology to Reconcile Pharmacogenomic Knowledge Units. – PeerJ Preprints 5:e3140v1., 2017. **https://doi.org/10.7287/peerj.preprints.3140v1**
15. Brüggemann, S. Rule Mining for Automatic Ontology Based Data Cleaning. – In: Y. Zhang, G. Yu, E. Bertino, U. Xu, Eds. APWeb 2008. 4976, Berlin, Heidelberg, Springer, 2008, pp. 522-527. **https://doi.org/10.1007/978-3-540-78849-2_52**
16. Buranarach, M., T. Supnithi, Y. Thein, T. Ruangrajitpakorn, T. Rattanasawad, K. Wongpatikaseree, A. O. Lim, Y. Tan, A. Assawamakin. OAM: An Ontology Application Management Framework for Simplifying Ontology-Based Semantic Web Application Development. – International Journal of Software Engineering and Knowledge Engineering, Vol. **26**, 2016, No 1, pp. 115-145.
17. Eine, B., M. Jurisch, W. Quint. Ontology-Based Big Data Management. – Systems, Vol. **5**, 2017, No 45.
18. Michael, A., R. Kothandaraman, K. Kaliyan. Providing Ontology-Based Access Control for Cloud Data by Exploiting. – International Journal of Intelligent Engineering and Systems, Vol. **12**, 2019, No 3, pp. 280-291. DOI:10.22266/ijies2019.0630.27.
19. EDM Council. The Financial Industry Business Ontology (FIBO). 2008. Retrieved Sptember 2019. **https://spec.edmcouncil.org/fibo/index.html**
20. Wisnosky, D. The FIBO Leadership Team. FIBO™ Build, Test, Deploy and Maintain Methodology. 30 September 2017. Retrieved 30 September 2019, from EDM Council. **https://spec.edmcouncil.org/fibo/doc/20170930_FIBO_BTDM.pdf**
21. Abdelghany, A. S., N. R. Darwish, H. A. Hefni. An Agile Methodology for Ontology Development. – International Journal of Intelligent Engineering and Systems, Vol. **12**, 2019, No 2, pp. 170-181. DOI:10.22266/ijies2019.0430.17.
22. Axelsson, J., E. Papatheocharous, J. Nyfjord, M. Törngren. Notes on Agile and Safety-Critical Development. – ASM SIGSOFT Software Engeneering, Vol. **41**, March 2016, No 2, pp. 23-26.
23. The Gene Ontology Consortium (n.d.). Gene Ontology Resource. Retrieved September 2019. **http://geneontology.org/**
24. Fox, M., J. Chionglo, F. Fadel (n.d.). TOVE Project. Retrieved September 2019. **http://www.eil.utoronto.ca/projects/tove-project/**
25. Bennett, M. Providing Conceptual Disambiguation for Terms in Reusable Ontologies: A Case Study from FIBO. – In: 6th International Workshop on Ontologies and Conceptual Modeling (Onto.Com), 2205. Cape Town, South Africa, 2018.

26. Zaino, J. Banking on FIBO: Financial Institutions Turn to Semantic Standard. 2016. Retrieved September 2019.
https://www.dataversity.net/banking-fibo-financial-institutions-turn-standard-value-compliance/
27. El Hassouni, J., A. El Qadi, M. Bazzi, M. El Haziti. Modeling with Ontologies Design Patterns: Credit Scorecard as a Case Study. – Indonesian Journal of Electrical Engineering and Computer Science, Vol. **17**, January 2020, No 1, pp. 429-439. DOI:10.11591/ijeecs.v17.i1.
28. Gangemi, A., V. Presutti. Ontology Design Patterns. – In: S. Staab, R. Studer, Eds. Handbook on Ontologies. International Handbooks on Information Systems. Berlin, Heidelberg, Springer, 2009, pp. 221-243.
29. Ferrario, R. Who Cares about Axiomatization? Representation, Invariance, and Formal Ontologies, Epistemologia. – Epistemologia, Vol. **2**, 2006, pp. 323-342.
http://www.cnr.it/prodotto/i/69465
30. Suppose, P. Heuristics and the Axiomatic Method. – Models and Methods in the Philosophy of Science. – Selected Essays, Vol. **226**, pp. 55-65. Preprint. – In: Synthese Library. Studies in Epistemology, Logic, Methodology, and Philosophy of Science, 1993.
https://doi.org/10.1007/978-94-017-2300-8_5
31. Krötzsch, M., F. Simancik, I. Horrocks. A Description Logic Primer. ArxIV; Preprint: 2012. arXiv CoRR abs/1201.4089.
32. Sarker, M., A. Krisnadhi, P. Hitzler. OWLAx: A Protégé Plugin to Support Ontology Axiomatization through Diagramming. – In: T. Kawamura, H. Paulheim, Eds. Proceedings of the ISWC 2016 Posters & Demonstrations Track Co-Located with 15th International Semantic Web Conference (ISWC'16), Kobe, Japan, 2016.
33. Poveda Villalón, M. (n.d.). OOPS! (OntOlogy Pitfall Scanner!). Retrieved September 2019.
http://oops.linkeddata.es
34. Poveda Villalón, M. Ontology Evaluation: A Pitfall-Based Approach to Ontology Diagnosis. Phd Thesis, Universidad Politécnica de Madrid, Madrid, 2016.
35. Parsia, B., N. Matentzoglu, R. Gonçalves, B. Glimm, A. Steigmiller. The OWL Reasoner Evaluation (ORE) 2015 Competition. – Automated Reasoning, 2015. DOI:10.1007/s10817-017-9406-8.
36. Glimm, B., I. Horrocks, B. Motik, G. Stoilos, Z. Wang. HermiT: An OWL 2 Reasoner. – Automated Reasoning, 2014, pp. 1-25. DOI:10.1007/s10817-014-9305-1.
37. Horridge, M., S. Bechhofer. The OWL API: A Java API for OWL. – Semantic Web Journal, Vol. **2**, 2011, No 1, pp. 11-21.
38. Blomqvist, E., A. Seil Sepour, V. Presutti. Ontology Testing – Methodology and Tool. – In: K. E. Management, Ed. Proc. of 18th International Conference EKAW 2012, Galway, Irland. Berlin, Heidelberg, Springer-Verlag, October 2012, pp. 216-226.
39. The Apache Software Foundation. Apache Jena – ARQ – A SPARQL Processor for Jena. 2017, Retrieved September 2019.
http://jena.apache.org/documentation/query

Received: 10.02.2020; Second Version: 11.03.2020; Accepted: 20.03.2020 (fast track)