Validation of a metadata application profile domain model

Mariana Curado Malta
CEOS.PP, Polythecnic of Oporto, Portugal
Algoritmi Center, UMinho, Portugal
mariana@iscap.ipp.pt

Helena Bermúdez-Sabel
LINHD - UNED, Spain
helena.bermudez@linhd.uned.es

Ana Alice Baptista
Algoritmi Center, UMinho, Portugal
analice@dsi.uminho.pt

Elena González-Blanco
Coverwallet, Spain
elena@coverwallet.com

Abstract

The development of Metadata Application Profiles is done in several phases. According to the Me4MAP method, one of these phases is the creation of the domain model. This paper reports the validation process of a domain model developed under the project POSTDATA - Poetry Standardization and Linked Open Data. The development of the domain model ran with two steps of construction and two of validation. The validation steps drew on the participation of specialists in European poetry and the use of real resources. On the first validation we used tables with information about resources related properties and for which the experts had to fill certain fields like, for examples, the values. The second validation used a XML framework to control the input of values in the model. The validation process allowed us to find and fix flaws in the domain model that would otherwise have been passed to the Description Set Profile and possibly would only be found after implementing the application profile in a real case.

Keywords: metadata; metadata application profile; Me4MAP

1. Introduction

The Semantic Web is an ecosystem of linked data, published, used and reused by agents related to communities of practice. The aim of these agents is to publish semantically interoperable data with data from other partners from the same community, and to profit from the open context that the ecosystem provides. In fact, the Semantic Web gives us this possibility of enriching data beyond borders and frontiers of communities since it is possible to start in a dataset "and then move through an unending set of databases which are connected not by wires but by being about the same thing" (Hawke, Herman, Archer, & Prud'hommeaux, 2013). Semantic interoperability is potentiated when data can be readily accessible with embedded information about its meaning, and it is possible through the use of common vocabularies and data models. In order to achieve maximum interoperability of its data, the development of semantic web applications requires obedience to *de jur*e and/or *de facto* standards. This implies careful and rigorous steps on the definition and design of its data and of its relationships with other data in the Web. One of the constructs that represents a semantic web data model is a Description Set Profile (DSP), which is, in turn, a component of a Metadata Application Profile (MAP). A MAP is a "generic construct for designing metadata records" (Coyle & Baker, 2009).

This paper is framed in a project funded by the European Research Grant (ERC), POSTDATA[1], which aims to provide means to make data about European poetry available as linked open data (LOD). Thus, POSTDATA is developing a MAP for the European poetry (MAP-EP). The

[1] http://postdata.linhd.es – accessed in July 31, 2018

POSTDATA work team is using the method for the development of metadata application profiles (Me4MAP) – see Curado Malta & Baptista (2013)– for its development. Me4MAP has been tested in several settings –see Curado Malta & Baptista (2017); Curado Malta, Baptista, & Parente (2015)– and this paper presents another one: European poetry provided by different institutions of the European poetry community of practice. The paper delineates how a domain model was developed in a context where non-interoperable structured data exists in 23 disperse databases that serve their own Websites, and also show in detail how this domain model was validated. The information herein presented is relevant both to the Metadata and the Digital Humanities communities. To the Metadata community because it provides a real-world example of a validation of a linked data domain model. To the Digital Humanities community because it gives information on how it is possible to create common models out of different contexts that will allow new studies across different repositories.

This paper is divided in four sections. The following section presents 1) Me4MAP and how it was used to develop the Domain Model, 2) the application domain where the MAP is being developed. Section 3 presents how the development of the Domain Model was done, presenting briefly the phases of construction of the Domain Model in the first sub-section and with more detail the phases of validation of the Domain Model in the second sub-section. The last section presents our conclusions and briefly explores future work.

2. Contextualisation

This section presents the context of this research project. The first sub-section presents the method for the development of metadata application profiles (Me4MAP) and why it is used in the development of the MAP-EP. The second sub-section introduces the European poetry community of practice as a context of the MAP-EP.

2.1 Me4MAP: a method for the development of metadata application profiles

The use of methods in any process of information systems development is important and the development of a MAP is no exception. In fact, a method introduces rigour in the process walking the developers through a path to follow and establishing which activities should be developed, when the activities may take place, how they interconnect and finally which milestones and deliverables they produce. The authors have been working in Me4MAP since 2012 and are using the process of developing MAP-EP as one more use-case to provide input for the improvement of Me4MAP.

Me4MAP presents a set of activities, organised in stages that are called the Singapore Stages. The name of the stages comes after the seminal document presented by Nilsson, Baker, & Johnston (2008). On stage S1 the Functional Requirements are defined, on stage S2 the Domain Model and on stage S3 the Description Set; these three stages are sequential and the deliverables of a previous stage feed the next stage.

As we will explain in the next paragraphs, we did not follow exactly Me4MAP for the Domain Model definition since the setting showed other possibilities.

On S1, Me4MAP defines a set of activities in order to obtain the Functional Requirements: S1.1 Definition of the Vision, S1.2 Development of the Work-Plan, S1.3 Definition of the Application Domain, S1.4 Elicitation of the high-level requirements and S1.5 Development of the Use-Case Model. The first three activities are general to all settings, the last two depend on the available resources of the setting that allow the work team to analyse the data needs of the community. And indeed, Me4MAP states that, depending on the resources available, it is possible to use other approaches to define the Functional Requirements.

Me4MAP says the Functional Requirements identified serve as input for the definition of the Domain Model. This is in fact nothing new since Me4MAP is inspired by the early stages of data

modelling used in the software development processes (Curado Malta & Baptista, 2013a) –e.g. Rationale Unified Process (Kruchten, 2004). But in our work we did not elicit functional requirements. In fact, since we had already structured data in the digital repertoires available on the Web, we decided to use the database structures of these repertoires as source to define the Domain Model[2].

2.2 Community of Practice: the European poetry

The willingness of an informal group of poetry scientists, that have been working together for some years, to publish data about poetry metrics in Linked Open Data provided the perfect opportunity to propose the development of a MAP for this specific community.

The MAP-EP is being developed in the scope of the POSTDATA project, a European Research Council Starting Grant – see Curado Malta, González-Blanco, Martinez, & Del Rio (2016) for more information about the project.

The research community of poetry works with digital repertoires of poetry. A repertoire is a catalogue that gives account of the metrical and rhythmical schemes of either a poetical tradition, a period or school, gathering a corpus of poems that are defined and classified by their main characteristics. These kind of repertoires may sometimes contain the text of the poem and information related to authors, manuscripts, editions, music, and other features, all of them related to the poems (Curado Malta et al., 2016)

These repertoires exist on the Web but are not interoperable (González-Blanco & Seláf, 2014). They have real data from research projects on poetry and this data has been structured by information modellers that have built these systems without concern with the possibility of interoperability. Since their interest laid in answering the particular research questions of their project, their goal is to just serve the specific needs of the local community. The poetry scientists want now to explore new possibilities; they want to cross or compare data from different traditions that is stored in different silos of information. Also, the possibility to link the data of those silos with other resources present in the LOD ecosystem is seen as a huge opportunity to enrich the data that already exists.

3. Developing the Domain Model

The development process of defining the Domain Model was made of two well-defined moments of construction and two well-defined moments of validation (see FIG.1). Nevertheless, there were certainly less distinct tasks of validation and construction since there were informal moments of discussion with poetry scientists during local presentations in the laboratory with visitors or in meetings with all the laboratory colleagues.

The process was iterative since we defined Version 0.1[3] (DM v0.1 in FIG.1) and validate it. Out of this first validation we issued Version 0.2[4] (DM v0.2 in FIG.1). Then, in a new period of construction, we defined Version 0.3[5] (DM v0.3 in FIG.1), finally this version was validated and we issued the first stable version of the Domain Model (DM v1.0 in FIG1 – version submitted to a scientific journal, waiting for editorial decision).

3.1 Building the Domain Model

The work team identified 23 important representatives of the community of practice. Seventeen provided the database structures of the digital repertoires. We used a reverse engineering process (Müller et al., 2000) to transform the logical data models of the databases into conceptual ones.

[2] See https://goo.gl/O0mqhI for the complete set of digital repertoires used in the whole process of the Domain Model definition – accessed in July 31, 2018
[3] Available at https://doi.org/10.5281/zenodo.832885 – accessed in July 31, 2018
[4] Available at https://doi.org/10.5281/zenodo.832906 – accessed in July 31, 2018
[5] Available at http://doi.org/10.5281/zenodo.1164193 – accessed in July 31, 2018

Curado Malta, Centenera, & González-Blanco (2017) and Bermúdez-Sabel, Curado Malta, & González-Blanco (2017) expound how the Domain Model was defined having as basis conceptual models of some databases.

Regarding the repertoires for which the delegates did not provide database structures, we decided to analyse the websites identifying their informational needs since they were openly available on the Web. By informational needs we mean the data the system needs to retrieve from the database to provide the information stated on the screen and the way it combines it. Firstly, we analysed the different pages or screens of the Website and how they were linked, and then for each screen we identified each dynamic field on the screen as data to be part of the Domain Model.

We have used the digital repertoire *MedDB – Base de Datos da Lírica Profana Galego-Portuguesa*[6] to conceptualise the framework of analysis. In fact, this database was used in the first moment of construction of the Domain Model (we had access to the database structure), but our idea was to test the results of the analysis against the structure of the database to verify whether the technique used was adequate and did not miss any important data.

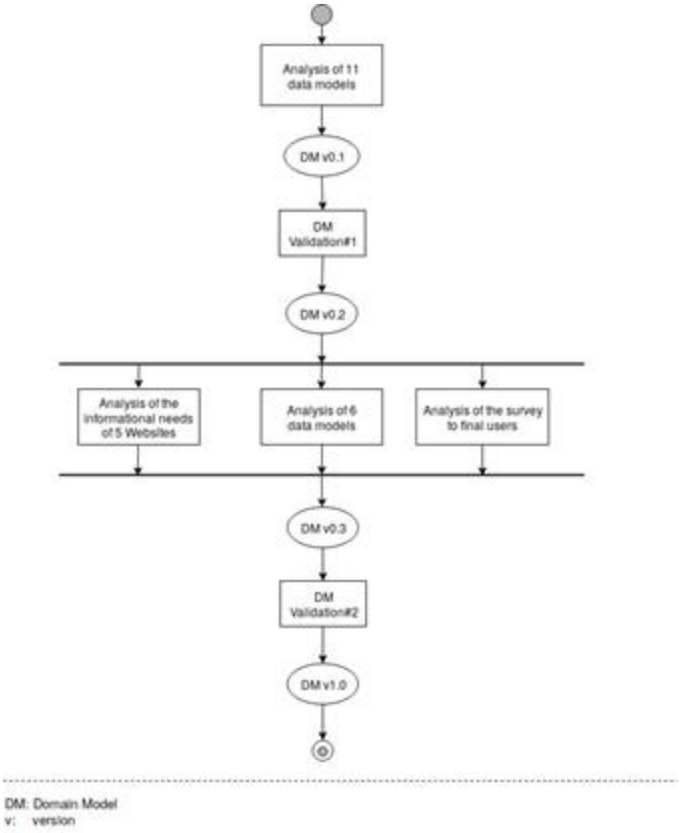


FIG. 1. The process of development of the Domain Model of the MAP-EP

The link http://doi.org/10.5281/zenodo.1117064[7] presents the report of this analysis showing the new data needs that were introduced in Version 0.3 of the Domain Model.

The work team also made available on the Web a survey to final users of the repertoires to understand the informational needs of these users, the link

[6] http://www.cirp.gal/meddb – accessed on July 31, 2018

[7] Accessed on July 31, 2018

https://doi.org/10.5281/zenodo.1117194[8] provides the results of the survey as well as the data needs that were introduced in Version 0.3 of the Domain Model.

The next sub-section presents the activities of validation (DM Validation#1 and DM Validation#2) that were developed.

3.2. Validating the Domain Model

We implemented two moments of validation: the first one validated Version 0.1 of the Domain Model –referred as "DM Validation#1" in FIG.1, and the second moment validated Version 0.3 of the Domain Model –referred as "DM Validation#2" in FIG.1.

The paradigm behind the class diagrams is object-oriented. The paradigm behind Linked Data is property centric and one of its benefits is that "it allows anyone to extend the description of existing resources, one of the architectural principles of the Web" (Brickley & Guha, 2004). The use of modelling techniques based on two distinct paradigms may pose some problems of expressiveness and coherence between the respective models. In our case, for clarity and ease of transposition to the property centric paradigm of the Resource Description Framework (RDF), we have mapped the relations between classes as properties that have those classes as their domain and/or range. For example, a `rel` relationship between class `A` and class `B` would be mapped as a `rel` property with domain `A` and range `B`.

Domain Model Validation#1

DM Validation#1 took place in March 2017 at UNED (Madrid), the university that hosts the POSTDATA project. We invited delegates of the digital repertoires that were firstly contacted during the definition of the state-of-the-art and thus were invited to participate as stakeholders. Delegates from ten different repertoires collaborated in the discussions of the Domain Model, nine of which participated in the validation test as well since their data models were analysed during the development of the Domain Model.

Delegates were all application experts (philologists). Each delegate received as work material:

- A paper sheet with the UML class diagram of the conceptual model of its own database: this diagram included the classes of the database, the relations between the classes and the attributes of each class. It is important to note that the names of the classes were the same as the ones appearing in the Domain Model.
- A spreadsheet file with a mapping between the logical model of the database of the delegate and the conceptual model (developed in the scope of POSTDATA) of the database.
- A paper sheet with the UML class diagram of the Domain Model: The diagram included the classes and the relations between the classes. It did not include the attributes of each class for reasons of readability[9].
- A spreadsheet with i) a list of the classes of the Domain Model and description of each class, ii) a list of the attributes of the Domain Model with description. The attributes were organised by classes, and iii) a list of the relations of the classes with domain and range information.

A testing sheet was used to execute the validation[10]. This testing sheet is organised as follows:

- Each sheet (see FIG.2) has the name of a class (e.g `Opus`), on the top of the sheet there is a cell that identifies the instance of the class (value of the cell "Instance label"). The sheet can be repeated as many times as the number of instances of the class that the resource

[8] Accessed on July 31, 2018
[9] Available at https://doi.org/10.5281/zenodo.437827 – accessed in July 31, 2018
[10] Available at https://doi.org/10.5281/zenodo.1226672 – accessed in 21 April, 2018

being tested has. Or, if needed and if there is space, other instances of the same class can be repeated in the same sheet.

- Each sheet has a list of the attributes (column "Property Label") of the class at hand. Each line represents an attribute and has the following columns: "range" (the type of the value of the attribute, e.g. int, text, boolean), "cardinality" (how many times the attribute can be repeated) and "value" (we can have more than one column named "value", depending on the cardinality of the attribute). The cell of the columns "value" should be filled in with the information of the resource related to that attribute.
- FIG.3 presents the last part of each sheet where there is a list of the relations (Column A - "Property label") between the class at hand and other classes (Column B - "Range"). The cells in the columns "Value" should be filled with the names of the instances of the class that are the range of the relations at hand. For example, the two instances of classes `Opus1` and `Redaction1` relate the following way: `Opus1—isRealisedThrough—Redaction1`. This is made explicit by filling in:
  - sheet "Opus" (see FIG.2), the cell "Instance label" with the value `Opus1` and,
  - the same sheet "Opus" (see FIG.3), the cell C26 with the value `Redaction1`.

Instance Label					
Property Label	**Range**	**Card.**	**Value**	**Value**	**Value**
altTitle		0-1			
betaReferenceID		0-1			
date		0-M			
dateNote		0-M			
domain		0-1			
duttonReferenceID		0-1			
genre		0-M			
IncipitariumCarminumLatinorumReferenceID		0-1			

FIG. 2. An excerpt of the test sheet: list of some attributes of the class Opus

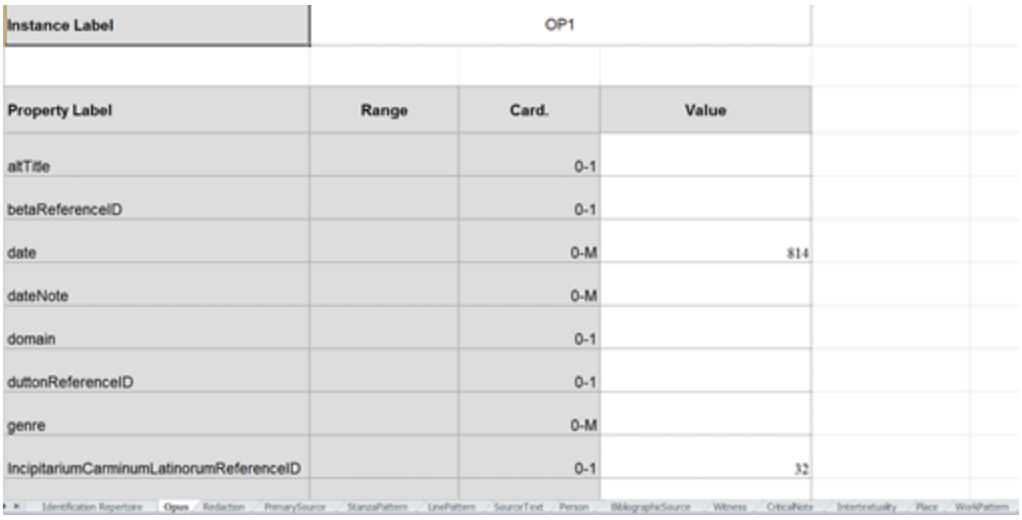
	A	B	C	D
24				
25	**Property Label**	**Range**	**Instance Label**	**Instance Label**
26	isRealisedThrough	Redaction		
27	isPreservedIn	AlternativeSource		
28	hasCreator	Person		
29	isPartOf	Ensemble		
30	isEditedIn	BibliographicSource		
31	isReferred	BibliographicSource		
32	comesFrom	Place		
33	isOrganisedIn	CatalogEntry		
34				
35				

ID Repertoire | Opus | Redaction | PrimarySource | Stanza | StanzaPattern | Line | Rhyme | LinePattern | Word | SourceText | Derived

FIG. 3. An excerpt of the test sheet: list of some relations of the class Opus

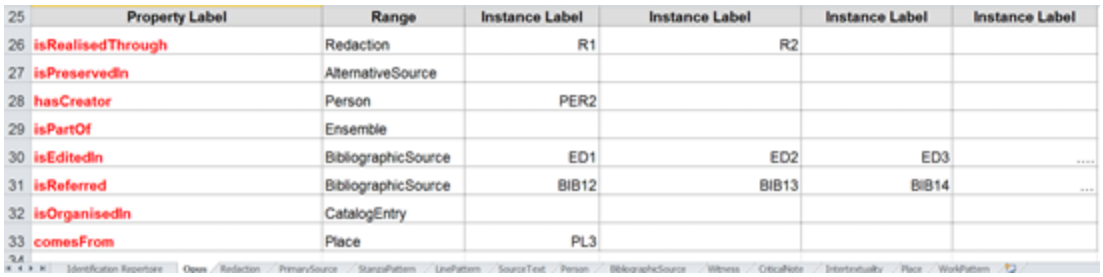
Before doing the hands-on session of validation, a testing sheet with an example of testing was given to the delegates and explained[11] for the delegates to understand the aim of the session. The example given used a resource sample from the repertoire *Corpus Rhythmorum Musicum*[12], one of the repertoires used to build the Domain Model. FIG.4 shows an excerpt of the validation example:

- There is an instance of the class `Opus` named `OP1`;
- Attributes of `OP1`, e.g. the date of creation (value: year 814) and the Reference ID of the catalogue Incipiarium Carminum Latinorum (value: 32);
- `OP1` relates to a certain number of other instances of classes (see FIG.5): `OP1 isRealisedThroug R1`, `OP1 isRealisedThrough R2`, `OP1 hasCreator PER2`, etc. All these instances of classes (`R1`, `R2`, `PER2`, etc.) have sheets where their attributes are defined;
- FIG.6 presents several instances of the class `Person` where `PER2` (the author of `OP1`) in the attribute `name` has the value "anonymous", meaning that `OP1` has an anonymous author.



Instance Label	OP1		
Property Label	**Range**	**Card.**	**Value**
altTitle		0-1	
betaReferenceID		0-1	
date		0-M	814
dateNote		0-M	
domain		0-1	
duttonReferenceID		0-1	
genre		0-M	
IncipitariumCarminumLatinorumReferenceID		0-1	32

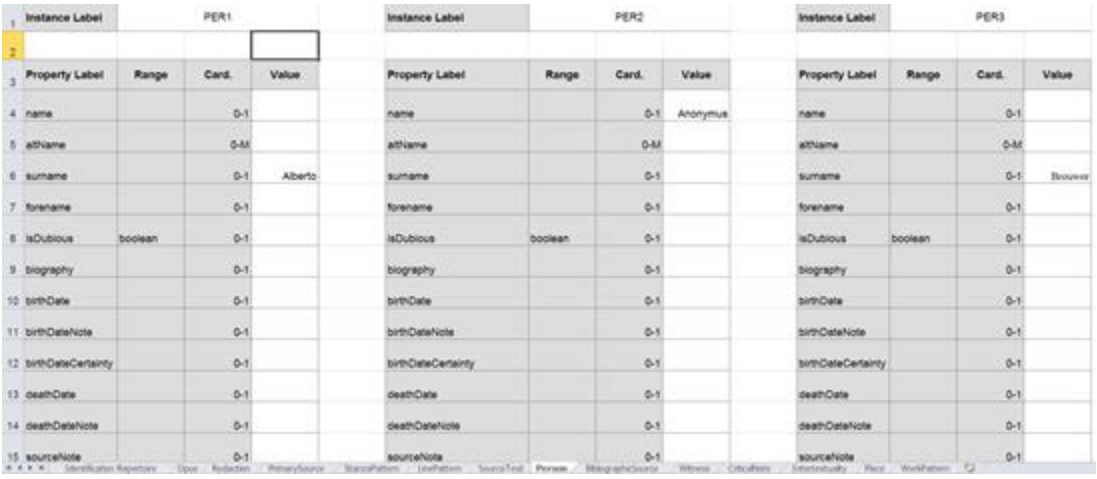
FIG. 4. An excerpt of the validation example: instance `OP1` of the class `Opus`



25	Property Label	Range	Instance Label	Instance Label	Instance Label	Instance Label
26	isRealisedThrough	Redaction	R1	R2		
27	isPreservedIn	AlternativeSource				
28	hasCreator	Person	PER2			
29	isPartOf	Ensemble				
30	isEditedIn	BibliographicSource	ED1	ED2	ED3	....
31	isReferred	BibliographicSource	BIB12	BIB13	BIB14	...
32	isOrganisedIn	CatalogEntry				
33	comesFrom	Place	PL3			

FIG. 5. An excerpt of the validation example: instance `OP1` of the class `Opus` and the relations with other instances of classes

[11] See https://doi.org/10.5281/zenodo.1226672 –accessed in July 31, 2018 – to download the file
[12] See http://www.corimu.unisi.it/ – accessed in July 31, 2018



Instance Label	PER1			Instance Label	PER2			Instance Label	PER3		
Property Label	Range	Card.	Value	Property Label	Range	Card.	Value	Property Label	Range	Card.	Value
name		0-1		name		0-1	Anonymus	name		0-1	
altName		0-M		altName		0-M		altName		0-M	
surname		0-1	Alberto	surname		0-1		surname		0-1	
forename		0-1		forename		0-1		forename		0-1	
isDubious	boolean	0-1		isDubious	boolean	0-1		isDubious	boolean	0-1	
biography		0-1		biography		0-1		biography		0-1	
birthDate		0-1		birthDate		0-1		birthDate		0-1	
birthDateNote		0-1		birthDateNote		0-1		birthDateNote		0-1	
birthDateCertainty		0-1		birthDateCertainty		0-1		birthDateCertainty		0-1	
deathDate		0-1		deathDate		0-1		deathDate		0-1	
deathDateNote		0-1		deathDateNote		0-1		deathDateNote		0-1	
sourceNote		0-1		sourceNote		0-1		sourceNote		0-1	

FIG. 6. An excerpt of the validation example: instances of the concept "Person" of the resource being described

We asked the delegates to choose some resources from their own digital repertoires and fill in the validation sheet with the correspondent values.

During the process of validation, we asked the delegates to register the issues that arose during the validation tests in the validation sheet. Also. at the end of the workshop we asked the delegates to upload the file(s) with the validation tests to a server in order to be analysed later by the work team. The delegates were also asked to fill in a form with the following questions:

- Could you describe all your data with the available elements? If not, please refer the difficulties.
- Did you have any difficulty in particular to describe your data? Were there any ambiguities?
- Is there anything else you want to add?

The work team used all the inputs given by the delegates to issue a Version 0.2 of the Domain Model.

Domain Model Validation#2

The DM Validation#2 was done on Version 0.3 of the Domain Model. In similarity to the previous process of analysis of the informational needs of the Websites, the digital repertoire *MedDB – Base de Datos da Lírica Profana Galego-Portuguesa* was used to conceptualise the framework of validation. After that, we have identified resources from digital repertoires that were not part of the 17 repertoires used as sources during the processes of construction, this way we could address at a certain point the general scope of the Domain Model. By "general scope" we mean that we expect this Domain Model to serve other contexts, in the same community of practice, then the ones used to create it.

This validation was done mainly by a master student of philology that did not participate in the processes of construction of the Domain Model. By using an external person to the team we wanted to give total freedom of interpretation of the model to see if again the Domain Model could respond to the needs of the community. This student was helped by members of the team, nevertheless we tried not to introduce any bias on the use-cases building.

The DM Validation#2 consisted in using real resources from the GUIs of the databases and, with that data, populate the Domain Model Version 0.3. For this work we created:

1. A description of the Domain Model in XML[13];
2. Schema files for the use-cases that validates their contents against the DM[14].

Besides the repertoire used as base example for each process, the aforementioned *MedDB*, we selected five different poetry projects and randomly chose, at least, one poetic resource from each one them. In total, we built nine use-cases.

The modelling of the use-cases consisted in describing the resource in XML using the classes, attributes and relations of the Domain Model. The schema file restricted both the classes and the different attributes and relations, so any elements that were not contained in the Domain Model could not be added. In addition, it also prevented the repetition of labels that identified the different instances of each class so to avoid ambiguities. This schema also controlled the relations between the different instances of class: except for the instance of class Opus, every instance of any other class had to be the range of at least one relation.

The construction of the use-cases affected the contents of the DM. Whenever a informational need not previously considered was detected, the elements required for enabling its modelling were added to the Domain Model so we had an updated version to validate against the use-cases. This means that the XML provided as representative of Version 0.3 represents a previous stage of the Domain Model than the use-cases.

With the information retrieved from the construction of the use cases, we created a report, organised by digital repertoire. This report is available on http://doi.org/10.5281/zenodo.1164854[15].

Out of Validation#2 we issued the first stable version of the Domain Model for European Poetry that is to be published in a scientific journal (waiting for editorial decision).

4. Conclusions and Future Work

A metadata Application profile (MAP) is a construct of the semantic web that enhances interoperability (Nilsson et al., 2008). When a community of practice publishes linked open data (LOD) in the semantic web using as reference the MAP of the community, all the data from its datasets will be ready to be used and combined automatically since they have exactly the same structure. Adding to this, if the developers of the MAP followed good practices while defining it, i. e., used standard vocabularies of the semantic web and referenced resources of other datasets inside borders of the same community or even outside, these data will be much enriched. An informal group of philologists, delegates of digital repertoires of European poetry, understood that they could profit from these possibilities. The POSTDATA project, financed by a European Research Council (ERC) Grant, started two years ago with the aim (among other goals) of providing means for this informal group –and later any organisation of the same community of practice– to publish LOD about European poetry. To achieve this goal, the POSTDATA work team decided to develop a MAP for the European poetry (MAP-EP) using Me4MAP, a method for the development of MAPs. This paper presents the work developed during the definition of the Domain Model of this MAP-EP, more specifically presents how the validation of the Domain Model was done. The process followed during the building of the Domain Model for European poetry reveals the importance of validation, hence the upgrade of version that each validation moment caused.

This validation included two steps: 1) The first moment had the aim to validate Domain Model version 0.1. This occurred in a workshop with the informal group just referred where they tested

[13] See https://github.com/postdataproject/Domain-Model-v.0.3/tree/master/domain-model – accessed in July 31, 2018 – for the XML file with the description of the Domain Model and the related schemas)

[14] See https://github.com/postdataproject/Domain-Model-v.0.3/tree/master/use-cases – accessed in July 31, 2018 – for the XML files of the use-cases and related schemas

[15] Accessed on July 31, 2018

the model using real resources from their own databases. This group was guided to populate a testing file with information from the chosen resources. The file was organised in such a way that it reproduced the structure of the Domain Model in worksheets of a spreadsheet; 2) the second moment had the aim to validate Domain Model version 0.3. This validation was an activity that consisted in using a set of use-cases, resources of other digital repertoires –other than the used in the building of the model– and feeding XML files with the information from the resources. The XML files were structured in a way that reproduced the Domain Model, and we used a XML framework to validate in real-time the values introduced to avoid any errors. This last validation activity resulted in version 1.0 of the Domain Model.

The first stable version of the Domain Model for the European poetry is a milestone of the whole process of developing the MAP-EP. The POSTDATA work team is now continuing the development of MAP-EP. The current work is focusing in aligning each concept of the DM (either class, attribute or relation) with the RDF vocabulary term that best describes it, as well as developing vocabulary encoding schemes to constrain certain properties.

This activity of developing a Domain Model in the framework of a MAP development was the opportunity for Me4MAP researchers to test the method in a new setting not tested before. Me4MAP was developed following a Design Science Research methodological approach –see Hevner (2007). During its development, the method was tested using an experimental situation with a worldwide group of the Social and Solidarity Economy (SSE) to collaboratively build a MAP for the Web Based Information Systems of the SSE community (Curado Malta, 2014; Curado Malta, Baptista, & Parente, 2015). Me4MAP researchers think that Me4MAP may be adequate in a context similar to the one used in the SSE community, but it needs validation in different settings. In fact the question of generalisability needs to be addressed as well as the limits of the Me4MAP applicability. This is why this new use-case of Me4MAP application is being monitored. The work described in this paper will be subject of reflection in order to give input for the improvement of Me4MAP.

Final Note: The authors are sorted in descending order according to their contribution to research and writing.

Acknowledgements

The authors would like to thank the delegates of the analysed repertoires for their availability and willingness to share information and to discuss issues related to their projects with the POSTDATA team.

The authors would like thank Patricia Garrido Teixeira for her work on the use-cases and Clara Martínez (LINHD-UNED) for her help on the metrical analysis of the digital repertoires.

The research presented in this paper was developed with the support of the Starting Grant research project: Poetry Standardization and Linked Open Data - POSTDATA (ERC-2015-STG-679528), funded by the European Research Council (ERC) under the European Union's Horizon2020 research and innovation programme.

Part of this work has been supported by COMPETE: POCI-01-0145-FEDER-007043 and FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2013.

References

Bermúdez-Sabel, H., Curado Malta, M., & González-Blanco, E. (2017). Towards Interoperability in the European Poetry Community: The Standardization of Philological Concepts. In Language, Data, and Knowledge (pp. 156–165). Springer, Cham. https://doi.org/10.1007/978-3-319-59888-8_14

Brickley, D., & Guha, R. V. (2004, February). RDF Vocabulary Description Language 1.0: RDF Schema. Retrieved 7 May 2018, from http://www.w3.org/TR/rdf-schema/

Coyle, K., & Baker, T. (2009). DCMI: Guidelines for Dublin Core Application Profiles (Working Draft). Retrieved 15 January 2018, from http://dublincore.org/documents/profile-guidelines/

Curado Malta, M. (2014, July 16). Contributo metodológico para o desenvolvimento de perfis de aplicação no contexto da Web Semântica. Universidade do Minho, Guimarães, Portugal. Retrieved from http://hdl.handle.net/1822/30262

Curado Malta, M., & Baptista, A. A. (2013). A method for the development of Dublin Core Application Profiles (Me4DCAP V0.2): detailed description. In Proceedings on International Conference on Dublin Core and Metadata Applications (pp. 90–103). Lisbon, POrtugal: Dublin Core Metadata Initiative. Retrieved from http://dcpapers.dublincore.org/pubs/article/view/3674

Curado Malta, M., & Baptista, A. A. (2017). The Development process of a Metadata Application Profile for the Social and Solidarity Economy: Computer Science & IT Book Chapter | IGI Global. In Developing Metadata Application Profiles (pp. 98–117). IGI Global. Retrieved from https://www.igi-global.com/chapter/the-development-process-of-a-metadata-application-profile-for-the-social-and-solidarity-economy/175868

Curado Malta, M., Baptista, A. A., & Parente, C. (2015). A DCAP for the Social and Solidarity Economy. In Proceeedings of the International Conference on Dublin Core and Metadata Applications (pp. 20–29). Lisbon, Portugal: Dublin Core Metadata Initiative. Retrieved from http://dcevents.dublincore.org/IntConf/dc-2015/paper/view/372

Curado Malta, M., Centenera, P., & González-Blanco, E. (2017). Using Reverse Engineering to Define a Domain Model: The Case of the Development of a Metadata Application Profile for European Poetry. In Developing Metadata Application Profiles (pp. 146–180). Retrieved from https://www.igi-global.com/chapter/using-reverse-engineering-to-define-a-domain-model/175870

Curado Malta, M., González-Blanco, E., Martínez, C., & Del Rio, G. (2016). Digital repertoires of poetry metrics: towards a Linked Open Data ecosystem. In Proceedings of the First Workshop on Digital Humanities and Digital Curation co-located with the 10th Conference on Metadata and Semantics Research (Vol. 1764, pp. 1–11). Gröningen, Germany: CEUR Workshop Proceedings.

González-Blanco, E., & Seláf, L. (2014). Megarep: A comprehensive research tool in medieval and renaissance poetic and metrical repertoires. Humanitats a La Xarxa: Món Medieval/Humanities on the Web: The Medieval World, 321–332.

Hawke, S., Herman, I., Archer, P., & Prud'hommeaux, E. (2013). W3C Semantic web activity. Retrieved 4 May 2018, from https://www.w3.org/2001/sw/

Hevner, A. R. (2007). The three cycle view of design science research. Scandinavian Journal of Information Systems, 19(2), 87–92.

Kruchten, P. (2004). The rational unified process: an introduction (3rd ed.). Boston, MA, USA: Addison-Wesley Professional.

Müller, H. A., Jahnke, J. H., Smith, D. B., Storey, M.-A., Tilley, S. R., & Wong, K. (2000). Reverse Engineering: A Roadmap. In Proceedings of the Conference on The Future of Software Engineering (pp. 47–60). New York, NY, USA: ACM. https://doi.org/10.1145/336512.336526

Nilsson, M., Baker, T., & Johnston, P. (2008). DCMI: Singapore Framework for Dublin Core Application Profiles. Retrieved 20 April 2018, from http://dublincore.org/documents/profile-guidelines/