DCPAPERS
Proc. Int'l Conf. on Dublin Core and Metadata Applications 2013

A Method for the Development of Dublin Core Application Profiles (Me4DCAP V0.2): Detailed Description

Mariana Curado Malta
Algoritmi Center, University of Minho and Polytechnic of Oporto, Portugal
mariana.malta@algoritmi.uminho.pt

Ana Alice Baptista
Algoritmi Center, University of Minho, Portugal
analice@dsi.uminho.pt

Abstract

Our research in progress project aims the design of a method for the development of Dublin Core Application Profiles (Me4DCAP). This paper describes Me4DCAP V0.2, an early version of a method for the development of DCAP. The development of Me4DCAP was based on a Design Science Research methodological approach and has as starting points the Singapore framework for DCAP and the Rational Unified Process; integrates knowledge from: (i) the practices of the metadata community concerning DCAP development, and (ii) software development processes and techniques, focusing on the early stages of the processes that deal with data modeling. Me4DCAP establishes a way for the development of a DCAP. It establishes which are the activities, when they should take place, how they interconnect, and which deliverables they will bring about; it also suggests which techniques should be used to build these deliverables.

Keywords: metadata; semantic web; application profile; Dublin Core application profile; DCAP; methods; development of DCAP; Me4DCAP; Web based information systems; design science research; interoperability; object role modeling; social and solidarity economy.

1. Introduction

The Semantic Web, or Web of Data, has technologies that ``enable people to create data stores on the Web, build vocabularies, and write rules for handling data. Linked data are empowered by technologies`` that started to emerge in 1999. It is about common formats for integration and combination of data from different sources (W3C, 2012). This data is mostly what is being called metadata, in the way that it is "data about data" (DCMI, 2011) and follows well-defined rules of metadata schemes. A metadata scheme is a set of ``metadata elements designed for a specific purpose, such as describing a particular type of information resource`` (Press, 2004, p. 4).

The Dublin Core Metadata Initiative (DCMI) is probably the most well-known and influential worldwide initiative in what concerns metadata. In order to provide "a foundation for the development of application-independent syntax specifications and constraint languages", DCMI developed the Dublin Core Abstract Model (DCAM) (Powell et al., 2007) that presents the components and constructs used in DCMI metadata. One of these constructs is the Dublin Core Application Profile (DCAP), - "a generic construct for designing metadata records" (Baker & Coyle, 2009). The definition of rules to build a DCAP is set in the "Singapore Framework for Dublin Core Application Profiles", a DCMI recommendation (c.f. Nilsson et al. (2008), from now on only referred as Singapore Framework.

A DCAP is a very important construct to implement interoperability; therefore it is essential to have a method to be able to develop such a construct in order to give DCAP developers a common ground of work. By method we mean a selection of techniques, the control of their usage and the integration of the obtained partial results (De Almeida & Pinto, 1995).

For the time being, the only guidelines available to develop a DCAP are stated in the Singapore Framework and the DCMI Guidelines for DCAP – c.f Baker & Coyle (2009), from

https://doi.org/10.23106/dcmi.952136137

now on referred as DCMI Guidelines - and they are very brief. In fact, a study that we have performed recently shows that there is no method to develop a DCAP (c.f Curado Malta & Baptista, 2012). The absence of guidelines showing life-cycle with standardised activities, as well as a set of well-defined design criteria, with defined techniques, leads a DCAP development into a non-systematic set of activities. Therefore, it is imperative to define a method for this purpose.

Me4DCAP is a method for the development of DCAP under construction in the scope of a PhD research project that uses the Design Science Research (DSR) methodology. We have already published two articles in conference reporting on the first version of this method (Me4DCAP V 0.1) and on the use of DSR to develop it. The feedback we had was used to tune Me4DCAP V0.2.

This paper reports research in progress, it describes in detail Me4DCAP V0.2 describing the sources used to justify its design. This description defines a way for the construction of each Singapore Framework component. It establishes a way through the DCAP development, when activities should take place, how they interconnect, and which deliverables they are expected bring about; it also suggests which techniques should be used to build these deliverables.

This document proceeds as follows. The following section presents the DSR methodological approach application. Next section presents Me4DCAP V 0.2 design description. Closing conclusions and future work are drawn in the final section.

2. Methodology

This work is based on a design science research (DSR) methodology. DSR aims at the development of innovative artifacts that solve real-world problems (Simon, 1996), thus "Design Science is inherently a problem solving process`` (Hevner et al, 2004, p. 82). An artifact is something that is artificial, constructed by humans (Hevner & Chatterjee, 2010); a project using a DSR methodology produces artifacts that can be either constructs, models, methods, or instantiations of one of these 3 artifacts (Hevner, 2007). The artifact developed in this DSR project is a method.

DSR methodology is well grounded in the literature; concerning taxonomy and theory we mention the writings of Gregor & Jones (2007) and of March & Smith (1995); concerning method and process, the writing of Peffers et al. (2008) and concerning frameworks and guidelines, the writing of Hevner (2007). Our work follows the framework of Hevner (2007) with its 3 cycles: the ”Relevance Cycle“ that works in the ”Environment”; the ”Design Cycle“ that works in the core activities of building the artifact, and the ”Rigor Cycle“ that works in the ”Knowledge Base” of scientific theories.

In the Design Cycle, in the construction moments, we have the Singapore Framework, the DCMI Guidelines and the Rational Unified Process (RUP) (c.f. Kruchten (2004)) as starting points. We then integrate other elements namely (c.f FIG. 1): (i) early stages of the software development from methods and techniques used by the Software Engineering Community; (ii) information from studies we performed (c.f Curado Malta & Baptista (2012)); and also (iii) the results of the analysis of semi-structured interviews conducted to 3 DCAP developers: DRYAD (Carrier, 2008), Scholarly Work Application Profile (SWAP) (Allinson & Powell, 2006) and Variazioni Musical Application Profile (VMAP) (Iglesias et al., 2009). In the evaluation moments of the Design Cycle, we are using an experimental situation with a group of the World community of Social and Solidarity Economy (SSE) (c.f Curado Malta & Baptista (2013b)) to build, together with this group, a DCAP for the Web Based Information Systems of the SSE community. We feed-back the construction moments of the Design cycle with the outputs of this experiment.

https://doi.org/10.23106/dcmi.952136137

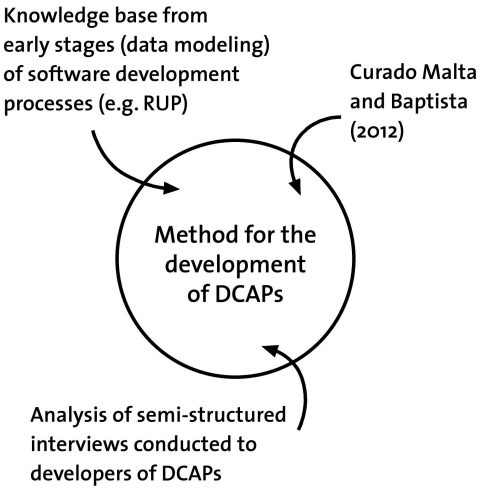

FIG. 1. The 3 inputs for the development of Me4DCAP in the construction moments of the DSR Design Cycle

3. Me4DCAP V0.2 Description

As stated earlier, Me4DCAP has as starting points the Singapore Framework, the DCMi Guidelines and RUP. According to the Singapore Framework, a DCAP is composed by:

- Functional Requirements (Singapore Stage 1)
- Domain Model (Singapore Stage 2)
- Description Set Profile (Singapore Stage 3)
- Usage guidelines (optional) (Singapore Stage 4)
- Syntax guidelines(optional) (Singapore Stage 5)

Me4DCAP defines a way for the construction of each component of the Singapore Framework. Using RUP as a basis, Me4DCAP establishes the way through the DCAP development: it establishes which are the activities, when they should take place, how they interconnect, and which deliverables they will bring about; it also suggests which techniques could be used to build these deliverables.

Me4DCAP has 4 phases (see FIG. 2): Scope Definition, Construction, Development and Validation. These phases are traversed along the project development as the Singapore Components are being developed.

Work planning begins in the phase "Scope Definition"; the goal of this phase is to define the DCAP application scope and to organize the work team. Part of the Functional Requirements Singapore Stage 1 is also developed in this phase; the rest overflows to the next phases, Construction and Development. In the Construction phase, the Domain Model Singapore Stage 2 development is initiated; it also overflows to the next phase, Development. The DSP Singapore Stage 3 is built in the Development phase. It is the climax of all construction done until this moment, since the DSP Singapore Stage 3 development work is based on the Domain Model Singapore Stage 2; and it is the Singapore Stage that defines the DCAP in its entirety. Finally in the Validation phase, the DCAP is validated. The Guidelines, Singapore Stage 4 and 5 are developed throughout the three last phases (Construction, Development and Validation).

https://doi.org/10.23106/dcmi.952136137

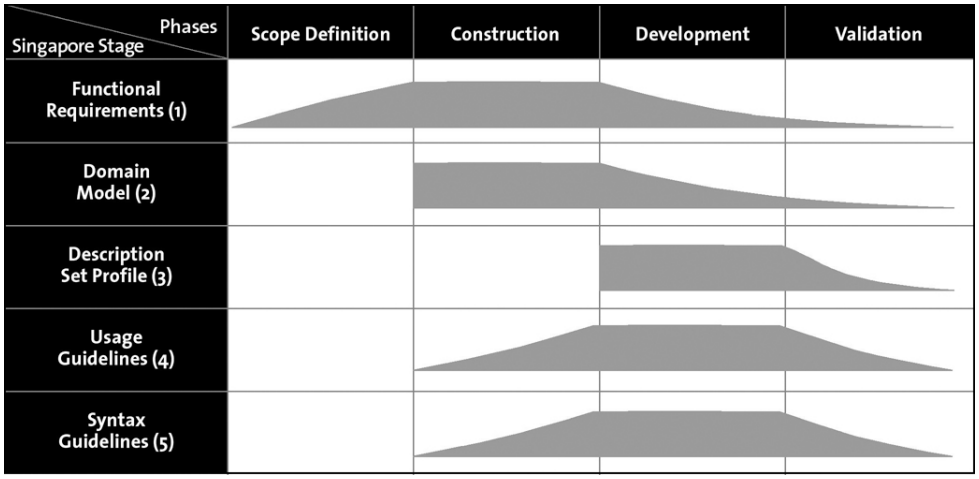

FIG. 2. The Me4DCAP V0.2 phases

3.1. The DCAP development work-team

Me4DCAP defines the following minimum profile roles types for the DCAP development work-team:

Project manager: the project manager "assumes a global overview of the project through a detailed interaction with the internal and external participants" (Borges et al., 2012, p. 63). It should be someone belonging to the organisations that are committed to the DCAP development. The person performing this role should have a scope vision of the organisations and of the goals they want to achieve with the DCAP.

System Analyst: the system analyst has two knowledge aspects: first, (s)he should have technical skills in requirements elicitation and in data modeling; second, (s)he should have "basic knowledge in management", should be able to understand in detail the application domain and should perceive "the real motivations and relevance of the requirements" (Borges et al., 2012, p. 62).

Integrator: the integrator should be a metadata designer/developer that understands the Semantic Web concepts, that has knowledge about the Description Set Profile (DSP) and the Resource Description Framework (RDF).

Final User: the final user is a user that will work with the Web system that will use the DCAP.

The work-team is then composed by several persons, with different roles, one or more per role, depending on the work load. It should be noted that the multi-disciplinary aspect of the team is very important and should be respected for the success of the DCAP.

In multi-disciplinary teams, however, it is important that the members of the work team speak all a common language, as it avoids misunderstandings and improves communication (Booch et al., 1999). For this reason, it is very important to build a **Glossary**. This should preferably be done from the beginning of the DCAP development process. A Glossary is a text document with the keywords (and their description) used in the DCAP. It should define important words commonly used by the work-team while constructing the DCAP.

A Glossary was also used by Methontology (Fernández-López et al., 1997) and by RUP (Kruchten, 2004).

3.2. The Me4DCAP V 0.2 life-cycle development model

Throughout a DCAP development, deliverables are being produced to help to reach the Singapore Framework Deliverables. FIG. 3 shows Me4DCAP life-cycle development model mentioning the deliverables that have to be produced and when they should be produced. The Me4DCAP life-cycle development model is iterative. The number of iterations depends on the

https://doi.org/10.23106/dcmi.952136137

dimension and complexity of the DCAP under development. Iterations will end when there is nothing new to "discover" or to add, depending on the results of the validation in laboratory and in production (see sections 3.5 and 3.7 to learn more about these validations)

FIG. 4 shows the dependence among the deliverables.

Next section will describe each Me4DCAP deliverable and the techniques that may be used to develop them. This description follows the Singapore Framework Stages order as they are at the center of all development. At the end of each deliverable description we show the sources of the "knowledge base" or of the "Environment" of the DSR approach that we use to justify the existence of that particular deliverable.

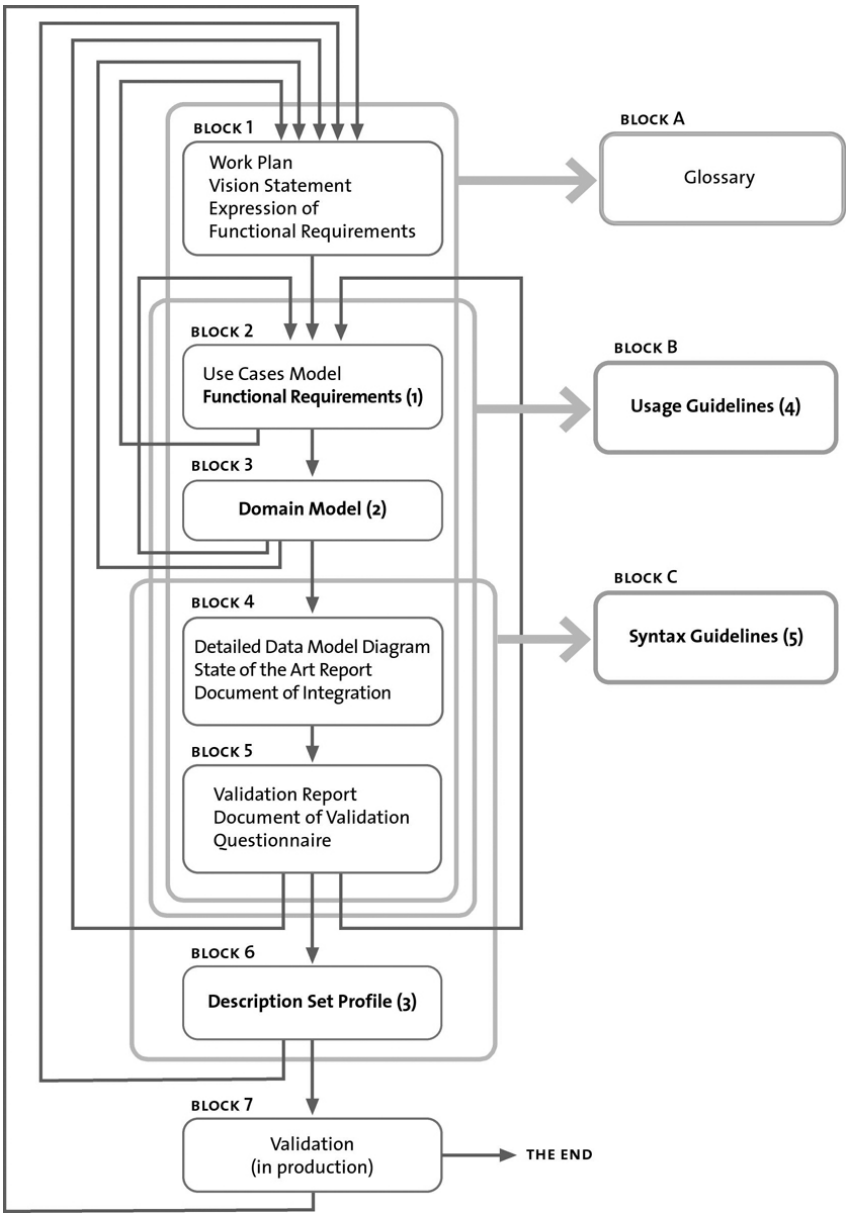

FIG. 3. Me4DCAP V0.2 life-cycle development model

3.3. Functional Requirements (Singapore Stage 1)

Me4DCAP defines the need to develop a set of 4 mandatory deliverables in order to build the mandatory Functional Requirements Singapore Stage: the Work Plan, the Vision Statement, the

https://doi.org/10.23106/dcmi.952136137

Expression of the Functional Requirements and the Use-Cases Model. The first deliverables to be built are the Vision Statement, the Work Plan and the Expression of the Functional Requirements, followed by the Use-Case Model.

The **Vision Statement** is a document that shows what developers want to reach with the DCAP development. It defines the scope of the DCAP; it is a simple plain text document with no more than 200 words, describing the boundaries for the DCAP usage. Me4DCAP recommends the brainstorming technique to develop the Vision Statement, where all members of the team should feel free to write ideas on a board (physical board or web tool), followed by discussion. In the end, the set of ideas chosen should be organized in simple sentences. The Project Manager leads the group on the development of this deliverable; all work-team members are expected to participate.

The TBM DCAP (c.f Calverley & Johnston (2009)), the WSDM method (c.f De Troyer & Leune, 1998)) and RUP refer the use of a vision Statement as the first step to build Functional Requirements.

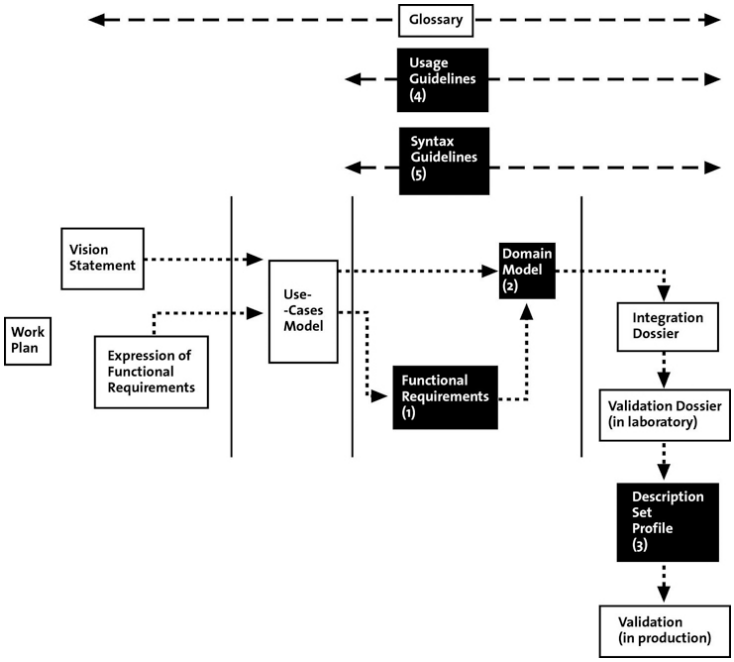

FIG. 4. The dependence among the deliverables in Me4DCAP V0.2

The **Work Plan** has as goal the time planning of the project activities; it is the follow-up project timing and serves as a guide for the work team of the DCAP development project. The Work Plan refers the timings of each phase, as well as the respective beginning and ending dates. It also defines the dates when each Component Stage should be ready. It will also be possible for the Work Plan to include information on the responsibilities of each element of the work-team in the phase or deliverable in question. The Work Plan may be a text document, a Gantt Chart or any other type of graph or scheme that the work-team finds more convenient. The Work Plan should be built by all members of the work-team, and negotiated among them in other to fit their time constraints; the Project Manager leads this work. It is acceptable that the Work Plan is modified as the project evolves.

The **Expression of Functional Requirements** is a list of the functional requirements expressed by the work-team, especially by the Final User and the Project Manager. This document should encompass very short description - with no more than 2 lines - per functional requirement. To develop this deliverable the work-team can use the same technique used in the Vision Statement deliverable. The Expression of Functional Requirements will be detailed further

https://doi.org/10.23106/dcmi.952136137

on, in the Use-Case Model deliverable. The System Analyst leads the group; all work-team members are expected to participate.

The **Use-Cases Model** is developed after the 3 deliverables previously described. Since Me4DCAP has an iterative life-cycle development model, we might have to revisit more than once the previous deliverables; there will be moments when the work-team will have to decide to release a draft version of the deliverables to follow the process, being aware that they will be working on those draft versions later on.

Use Cases "offer a systematic and intuitive means of capturing functional requirements" *(Booch et al., 1999*, p. 37). The Use-Cases will be used to develop the Functional Requirements and to understand the objects (and attributes) of the system to be studied. The Use-Cases Model is composed of:

- the UML Use-Case diagram with the actors that interact in the Use-Cases, describing the functionality of the system
- the set of all detailed Use-Cases

For information on how to build an UML Use-Case diagram see Booch et al. (1999) and Fowler (2004).

"Each use-case must include details about what has to be done to achieve its functionality" (*Schneider,* 2001, p.21) Every Use-Case should then be documented in detail. This documentation should set the sequence of actions - a specific sequence of events that happen in the system - that a system performs to bring added value to a specific actor. An actor is somebody or something (automata) outside the system that interacts with it (Kruchten, 2004). A Use-Case description is a flow of events description, and it may be developed using the template proposed by Schneider (2001). The System Analyst will need to know precisely the needs of the system in order to achieve certain objectives of functionality; (s)he should be able to identify the actions that will bring value to the system. So, the Use-Case description should be developed by the whole team with the leadership of the System Analyst and using the help of the template of the flow-of-events defined by Kruchten (2004) and a definition of Use-Case defined in the Glossary.

The Use Cases Model was also used by all the DCAP studied - TBM AP (Calverley & Johnston, 2009), SWAP (Allinson & Powell, 2006), VMAP (Iglesias et al., 2009), Images Application Profile (IAP) (Eadie, 2008) and DRYAD (Carrier, 2008). Also by the method ADWIS (Takahashi & Liang, 1997) and is referenced in the following literature: IMS Global Learning Consortium (2005), Friesen et al. (2002), DCMI (n.d.) and Buonazia et al. (2007

The **Functional Requirements** can be built after having developed the 4 described deliverables. This Singapore Component Stage 1 is mandatory. Functional requirements "guide the development of the application profile by providing goals and boundaries and are an essential component of a successful application profile development process. This development is often a broad community task and may involve managers of services, experts in the materials being used, application developers, and potential end users of the services". The Functional Requirements Singapore Stage 1 is a text document where general goals are mentioned, as well as specific tasks (Baker & Coyle, 2009). To develop the Functional Requirements the work-team should use the Expression of Functional Requirements and the Use-Cases Model in order to identify the functional requirements that the use-cases explicit. The work-team should use short sentences and write them on the board (on the physical board or on the working web tool). After that, the work-team should verify if there are repeated ideas of functional requirements on the board. Certain ideas speak more to some work-team members than to others, so each functional-requirement-idea should be distributed accordingly, in order to satisfy every member's specific requirements. Every member of the work-team should write some sentences describing more deeply the requirement-case that (s)he is responsible for. In the end of the process, all the requirements' cases should be put together on the board (on the physical board or web tool), and the whole

https://doi.org/10.23106/dcmi.952136137

group should discuss and review the final result. The System Analyst is expected to lead this work.

3.4. Domain Model (Singapore Stage 2)

The **Domain Model** is the mandatory Singapore Stage 2. It "captures the most important types of objects in the context of the system." (Booch et al., 1999, p. 119). According to Baker & Coyle (2009) "a domain model is a description of what things your metadata will describe, and the relationships between those things. The domain model is the basic blueprint for the construction of the application profile". The domain model is based on the Functional Requirements Singapore Stage 1 and on the Use-Cases Model deliverable described in Section 3.3. The Domain Model development can also use the help of other documents or information depending on the DCAP development context. In situations where access to documentation or the information systems databases is available, it is possible to resort to the Document Analysis technique to define it. Me4DCAP suggests the Domain Model to be developed using Object Role Modeling (ORM) (c.f. *Nijssen* & Halpin (1989) and Halpin (1996) for more information about ORM diagrams and Halpin (2009) for ORM modeling and the Semantic Web). The ORM diagram Domain Model identifies the classes of objects and the relationships among them. The attributes of the objects should be omitted since they are not needed at this stage and will be defined in the Detailed Diagram Data Model (see section 3.5). The Entity-Relationship diagram (ER) (c.f. Chen (1976) for more information about ER diagrams) showing the entities and the relationships among them can be an alternative to the ORM diagram (attributes should also be omitted).

The System Analyst is expected to do this work; the final version of the Domain Model should be discussed and validated by the whole work-team.

SWAP used an UML class diagram with details suppressed, as well as the work reported by Onyancha et al. (2001). RUP, being an UML based process, also uses UML. The WSDM (De Troyer & Leune, 1998) suggest that various techniques can be used: UML, the Object Modeling Technique (OMT) (c.f. Rumbaugh et al. (1990)), the ORM technique and the ER technique. Eadie (2008) also reports in its work the use of the ER technique.

3.5. Description Set Profile (Singapore Stage 3)

Me4DCAP defines the need to develop a set of 2 mandatory dossiers in order to develop the mandatory Singapore Stage 3 Description Set Profile (DSP):

- The Integration Dossier
- The Validation Dossier (in laboratory)

The **Integration dossier** comprises 3 deliverables: a Detailed Data Model Diagram, a State Of The Art Report and a Document of Integration. All these deliverables are mandatory.

The **Detailed Data Model Diagram** should present the Domain Model in detail, i.e. the definition of the object properties, whether they are mandatory or optional, repeatability, codification and multi-language possibility. Me4DCAP suggests the Detailed Data Model Diagram to be developed using the ORM technique. Other techniques can be used, as e.g ER. The System Analyst is expected to develop this deliverable; and the whole group should discuss and validate the results.

In this part of the DCAP development process, every class of object and attribute should also be described, in plain text, in the Usage Guidelines Singapore Stage 4 (see section 3.6 for more details about this deliverable).

After the definition of the Detailed Data Model Diagram, Me4DCAP defines as next step that the objects and attributes of this Data Model should be described, each and every one, by existent properties of the metadata schemes of the metadata community. According to Baker & Coyle (2009) this process is done in 2 steps:

- To build a **State Of The Art Report** with the existing metadata schemes - that are

https://doi.org/10.23106/dcmi.952136137

described in RDF - to find out from the existing schemes which properties can describe the identified attributes. This State of the Art Report should be done by the Integrator;

- To create new properties in case there are no properties on the metadata schemes of the State of the Art Report to describe some of the identified Detailed Data Model Diagram object attributes. The Integrator is expected to do this work.

All the analysed DCAP performed a state of the art; DCMI Guidelines (Baker & Coyle, 2009) also suggest such work, and all the following literature: Chen & Chen (2005), BSI (2005), Onyancha et al. (2001), Agostinho et al. (2004), Marzal García-Quismondo et al. (2006), Buonazia & Masci (2007) and Salokhe et al. (2008).

The existing information on the State Of The Art Report and on the Detailed Data Model Diagram will be used to build a **Document of Integration**. This Document shows, in a matrix, per line, every attribute and its constraints, described by the properties of the metadata schemes and encoding schemes chosen. This work should be done by the Integrator. A template of the Document of Integration can be downloaded from the repository of University of Minho, accessible through the URL http://hdl.handle.net/1822/24379.

Methontology (Fernández-López et al., 1997) and the literature CWA (2006); and also the DCMI Guidelines use a Document of Integration.

Me4DCAP defines as next step the execution of the validation of the work done until the present moment of the development process which is a validation in laboratory resulting in the "Validation Dossier". The **Validation Dossier** comprises 3 mandatory deliverables: a Validation Report, a Document of Validation and a Questionnaire.

A laboratory validation should take place, so as to check:

- its adequacy to what has been defined in the "Vision Statement" deliverable: a meeting of the work-team should take place to evaluate the answer to the defined vision (see Vision Statement in section 3.3). The work-team should make a report (text document) - the **Validation Report** - with the conclusions of the meeting and recommendations.
- DCAP adaptation to the resources that are going to be described. This validation work is done in 2 stages:
    a) Application of the DCAP to a resource sample. The work-team should identify a set of resources that constitutes a trustworthy sample of the application domain of the developing DCAP, and from there, the Final User, supported by the Integrator, are expected to complete the **Document of Validation** with data referring to each resource. The Document of Validation should be simple to fill in, where each element of the metadata is populated with the data that corresponds to the resource. This process should have the 2 Guidelines Singapore Stage 4 and Singapore Stage 5 as support material. A template of a Document of Validation can be downloaded from the repository of University of Minho, accessible through the URL http://hdl.handle.net/1822/24379
    b) **Questionnaire**. The persons that worked in the Document of Validation should answer to a set of questions to assess the difficulties of the validation process. The goal is to understand if there is data for which the DCAP has no description, or if there are DCAP elements, defined as compulsory, that could not be fulfilled with the information existing in a given resource, or any other type of difficulty or ambiguity. The questions to be asked to the DCAP validators could be like:

https://doi.org/10.23106/dcmi.952136137

i. Could you describe all your data with the available elements? If not, please refer the difficulties.

ii. Were there any DCAP metadata elements left that you could not fulfill? Which? Did this happen for lack of data or because you did not know how to do it?

iii. Did you have any difficulty in particular to describe your data? Were there any ambiguities?

iv. Is there anything else you want to add?

According to the results of the questionnaire, the process iterates or follows to the DSP development (c.f FIG. 2).

As shown in FIG. 5, SWAP and VMAP have also used a Document of Validation to perform laboratory validations. Also Agostinho et al. (2004) have used such a technique.

Once the Integration Dossier is finalised it is possible to pass on to the development of the Description Set Profile which is mandatory. The Integrator should detail the DCAP using the DSP framework defined by Nilsson (2008) having the Integration Dossier as support document. Further information, including implementation examples, can be found in Baker & Coyle (2009).

3.6. Guidelines (Singapore Stage 4 and Singapore Stage 5)

These guidelines are not mandatory in the Singapore Framework. Me4DCAP does not make them mandatory but recommends that they are developed, since it helps the final users of the DCAP application to apply correctly the properties and constraints.

The **Usage Guidelines** development is expected to start at the same time as Block 2, and this deliverable development flows with the rest of the DCAP development work (c.f FIG. 3). The DCMI Guidelines explain: "Description Set Profile defines the "what" of the application profile; usage guidelines provide the "how" and "why". Usage guidelines offer instructions to those who will create the metadata records. Ideally, they explain each property and anticipate the decisions that must be made in the course of creating a metadata record".

The Usage Guidelines should be developed mainly by the System Analyst. The description of the attributes and classes of objects is a type of information that can only be supplied by the application domain experts so the Project Manager and the Final User should also participate.

The development of the **Syntax Guidelines** should start at the same time as Block 4, and this deliverable development flows with the rest of the DCAP development work (c.f FIG. 3). The Syntax Guidelines describe "any application profile-specific syntaxes and/or syntax guidelines, if any" (Baker & Coyle, 2009). It is recommended the Syntax Guidelines are developed by the Integrator since it is a very technical document.

3.7. Finishing the DCAP development

A **Validation in production** of the DCAP should be performed. This process of validation can be done using a log registration technique or observing final-users working with the system that has implemented the DCAP developed. The Integrator is expected to do this observation. The suggested changes should be collected by the Integrator and reported back to the work-team from time to time in order to review and access the DCAP definitions. If there is new information to introduce in the process, the whole DCAP development process should start from Block 1 (c.f. FIG. 2), and every deliverable should be checked against this new information in order to develop a new DCAP version.

Having a Semantic Web application many interactions and many implications on other systems, DCAP changes implementation is a delicate task to manage: DCAP users (machines) are not standalone machines in such an environment, the whole system will feel the impact of the DCAP changes. The implementation of the changes that come out of the validation in production

https://doi.org/10.23106/dcmi.952136137

process is something that should be planned in advance and well discussed with the DCAP community in order to impose a minimum impact on the DCAP applications.

Regarding the sources used to justify the use of this deliverable, only RUP integrates such a deliverable (c.f FIG. 5).

4. Conclusions and future work

This paper is framed in a research in progress PhD project based on a Design Science Research (DSR) approach. The goal of this PhD project is to develop a method for the development of Dublin Core Application Profiles – (Me4DCAP); recent studies (c.f. Curado Malta & Baptista (2012)) have shown that there is none.

A Me4DCAP V0.2 was developed integrating the outputs of a validation of Me4DCAP V0.1. This paper describes in detail Me4DCAP V0.2 showing the sources used to justify its design. Me4DCAP has as starting points the Singapore framework for Dublin Core Application Profiles (DCAP), the DCMI Guidelines for DCAP and the Rational Unified Process; it also integrates knowledge from: (i) other software development processes and techniques, focusing on the early stages of the processes that deal with data modeling; and (ii) the practices of the metadata community concerning DCAP development. Me4DCAP detailed description establishes the way through the DCAP development. It establishes when activities must take place, how they interconnect, and which deliverables they will bring about; it also suggests which techniques could be used to build these deliverables. Me4DCAP defines a way for the construction of each Singapore Framework component. Me4DCAP description follows the order of these Singapore components.

The work presented is still in progress; Me4DCAP V 0.2 validation will be subject to two Confirmatory Focus Groups in order to build Me4DCAP V0.3 with this validation outputs.

We expect to find, in the future, partners that will be willing to use Me4DCAP V0.3 in their DCAP developments, for in the field validation of Me4DCAP. It is important to validate, among other things, Me4DCAP usefulness in order to understand if its use in a DCAP development will provide a better design.

Acknowledgements

This work is sponsored by FEDER funds through the Competitivity Factors Operational Programme - COMPETE and by National funds through FCT - Foundation for Science and Technology within the scope of the project: FCOMP-01-0124-FFEDER-022674.

We would like to thank the RIPESS group for accepting the DCAP-SEE development challenge as an experimental situation for our research project; without them the work described in this article was not possible.

We would like to thank Ida Cruz for the drawings of this paper.

References

Allinson, Julie, & Powell, Andy (2006). SWAP Application Profile.

Baker, Tom, & Coyle, Karen (2009). Guidelines for Dublin Core Application Profiles. Retrieved November 12, 2012, from http://dublincore.org/documents/profile-guidelines/

Booch, Grady, Jacobson, Ivar, & Rumbaugh, James (1999). The unified software development process (p. 463). Addison-Wesley.

Buonazia, Irene & Masci, Maria Emilia. (2007). Il PICO Application Profile. Un Dublin Core Application Profile per il Portale della Cultura Italiana. E-LIS. E-prints in Library and Information Science

https://doi.org/10.23106/dcmi.952136137

Borges, Pedro, Monteiro, Paula & Machado, Ricardo J. (2012). Mapping RUP Roles to Small Software Development Teams. Software Quality. Process Automation in Software Development, 59-70. Springer.

BSI. (2005, February). BS 8419-1:2005 Interoperability between metadata systems used for learning, education and training. Code of practice for the development of application profiles.

Calverley, Gayle & Johnston, Pete (2009). Time-based Media Application Profile: Definition Phase Report.

Carrier, Sarah (2008). The Dryad repository application profile: Process, development, and refinement.

Chen, Peter (1976). The entity-relationship model—toward a unified view of data. ACM Transactions on Database Systems (TODS), 1(1), 9–36.

Chen, Ya-ning, & Chen, Shu-jiun (2005). Metadata lifecycle model and metadata interoperability. 5th International Conference on Conception of Library and Information Science.

Curado Malta, Mariana, & Baptista, Ana A. (2012). State of the Art on Methodologies for the Development of a Metadata Application Profile. Metadata and Semantics Research, 343(July), 61–73. doi:10.1007/978-3-642-35233-1_6

Curado Malta, Mariana & Baptista, Ana A. (2013a). Me4DCAP V0.1: A method to develop Dublin Core Application Profiles. In N. Lavesson, P. Linde, P. P., editor, Proceedings of the 17th International Conference on Electronic Publishing - Mining the Digital Information Networks, pages 33 - 44. IOS Press.

Curado Malta, Mariana & Baptista, Ana A. (2013b). Social and Solidarity Economy Web Information Systems: State of the Art. In T. Torres-Coronas & M.-A. Vidal-Blasco (Eds.), Social E-Enterprise: Value Creation through ICT (1st ed., pp. 1–16). IGI Global.

CWA. (2006). Guidelines and support for building application profiles in e-learning (CEN Workshop Agreement, CWA 15555:2006 E). ftp://ftp.cenorm.be/PUBLIC/CWAs/e-Europe/WS-LT/cwa15555-00-2006-Jun.pdf. Retrieved from ftp://ftp.cenorm.be/PUBLIC/CWAs/e-Europe/WS-LT/cwa15555-00-2006-Jun.pdf

DCMI (2011). DCMI Glossary. Retrieved July 7, 2013, from http://wiki.dublincore.org/index.php/Glossary

DCMI. (n.d.). Dublin Core Education Application Profile (Working Draft of v0.4).

De Almeida, João F., & Pinto, José M. (1995). A investigação nas ciências sociais (p. 163). Editorial Presença.

De Troyer, O. M. F., & Leune, Corneli J. (1998). WSDM: a user centered design method for Web sites. Computer Networks and ISDN Systems, 30(1), 85–94.

Eadie, Mick (2008). Towards an application profile for images. Ariadne, 55.

Fernández-López, Mariano, Gómez-Pérez, Asuncíon, & Juristo, NNatalia (1997). Methontology: from ontological art towards ontological engineering.

Fowler, Martin (2004). UML distilled. Addison-Wesley Professional.

Friesen, Norm, Mason, Jon, & Ward, Nigel (2002). Building educational metadata application profiles. Proceedings of the International Conference on Dublin Core and Metadata for e-Communities (Vol. 2002, pp. 63–69).

Marzal García-Quismondo, Miguel Á., Calzada Prado, Javier, & Cuevas Cerveró, Aurora (2006). Desarrollo de un esquema de metadatos para la descripción de recursos educativos: el perfil de aplicación MIMETA. Revista española de documentación científica, 29(4), 551–571.

Gregor, Shirley and Hevner, Alan (2013). Positioning and presenting design science research for maximum impact. MIS Quarterly, pages 337–356.

Halpin, Terence (1996). Business rules and object-role modeling. Database Programming and Design, 9, 66–72.

Halpin, Terence (2009) Ontological Modeling: Part 1', Business Rules Journal, Vol. 10, No. 9 Retrieved June 27, 2013, from http://www.orm.net/pdf/OntologicalModeling1.pdf.

Hevner, Alan, & Chatterjee, Samir (2010). Design Research in Information Systems - Theory and Practice. (R. Sharda & S. Voß, Eds.) (Integrated., p. 320). Springer.

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

Iglesias, Carlos A., Garijo, Mercedes, Molina, Daniel, & De Juan, Paloma (2009). VMAP. A Dublin Core Application Profile for Musical Resources. In N. M. Fabio Satori Miguel A. Sicilia (Ed.), Metadata and Semantic Research: Third International Conference, MTSR 2009 (pp. 1–12). Springer Verlag.

IMS Global Learning Consortium, I. (2005). IMS Application Profile Guidelines: Part 1 - Management Overview. Retrieved from http://www.imsglobal.org/ap/apv1p0/imsap_oviewv1p0.html

Kruchten, Philippe (2004). The rational unified process: an introduction (p. 298). Addison-Wesley Professional.

Nijssen, Gerardus M., & Halpin, Terence (1989). Conceptual Schema and Relational Database Design: a fact oriented approach. Prentice Hall.

Nilsson, Mikael (2008). Description Set Profiles: A constraint language for Dublin Core Application Profiles.

https://doi.org/10.23106/dcmi.952136137

Nilsson, Mikael, Baker, Thomas, & Johnston, Pete (2008). The Singapore Framework for Dublin Core Application Profiles. Retrieved March 13, 2012, from http://dublincore.org/documents/singapore-framework/

Onyancha, Irene, Keizer, Johannes, & Katz, Stephen (2001). A Dublin core application profile in the agricultural domain. International Conference on Dublin Core and Metadata Applications (p. pp–185).

Peffers, Ken, Tuunanen, Tuure, Rothenberger, Marcus A., and Chatterjee, Samir (2007). A design science research methodology for information systems research. Journal of management information systems, 24(3):45–77.

Powell, Andy., Nilsson, Mikael, Naeve, Ambjörn, Baker, Thomas, & Johnston, Pete (2007). DCMI Abstract Model. Retrieved June 26, 2011, from http://dublincore.org/documents/2007/06/04/abstract-model/

Press, NISO (2004). Understanding Metadata. National Information Standards.

Rumbaugh, James R., Blaha, Michael R., Lorensen, William, Eddy, Frederick, & Premerlani, William (1990). Object-oriented modeling and design.

Salokhe, Gauri, Pesce, Valeria, & Liesthout, Joost (2008). Organization Metadata Application Profile. Retrieved from ftp://ftp.fao.org/gi/gil/gilws/aims/metadata/docs/organizationap.pdf

Schneider, Geri, & Winters, Jason P. (2001). Applying use cases: a practical guide (p. 245). Addison-Wesley.

Takahashi, Kenji & Liang, Eugene (1997). Analysis and design of Web-based information systems. Computer Networks and ISDN Systems, 29(8), 1167–1180.

Tremblay, Monica C., Hevner, Alan, & Berndt, Donald J. (2010). The use of focus groups in design science research. Design Research in Information Systems, 121–143.

W3C. (2012). Semantic Web. Retrieved March 11, 2013, from http://www.w3.org/standards/semanticweb/

https://doi.org/10.23106/dcmi.952136137

Appendix

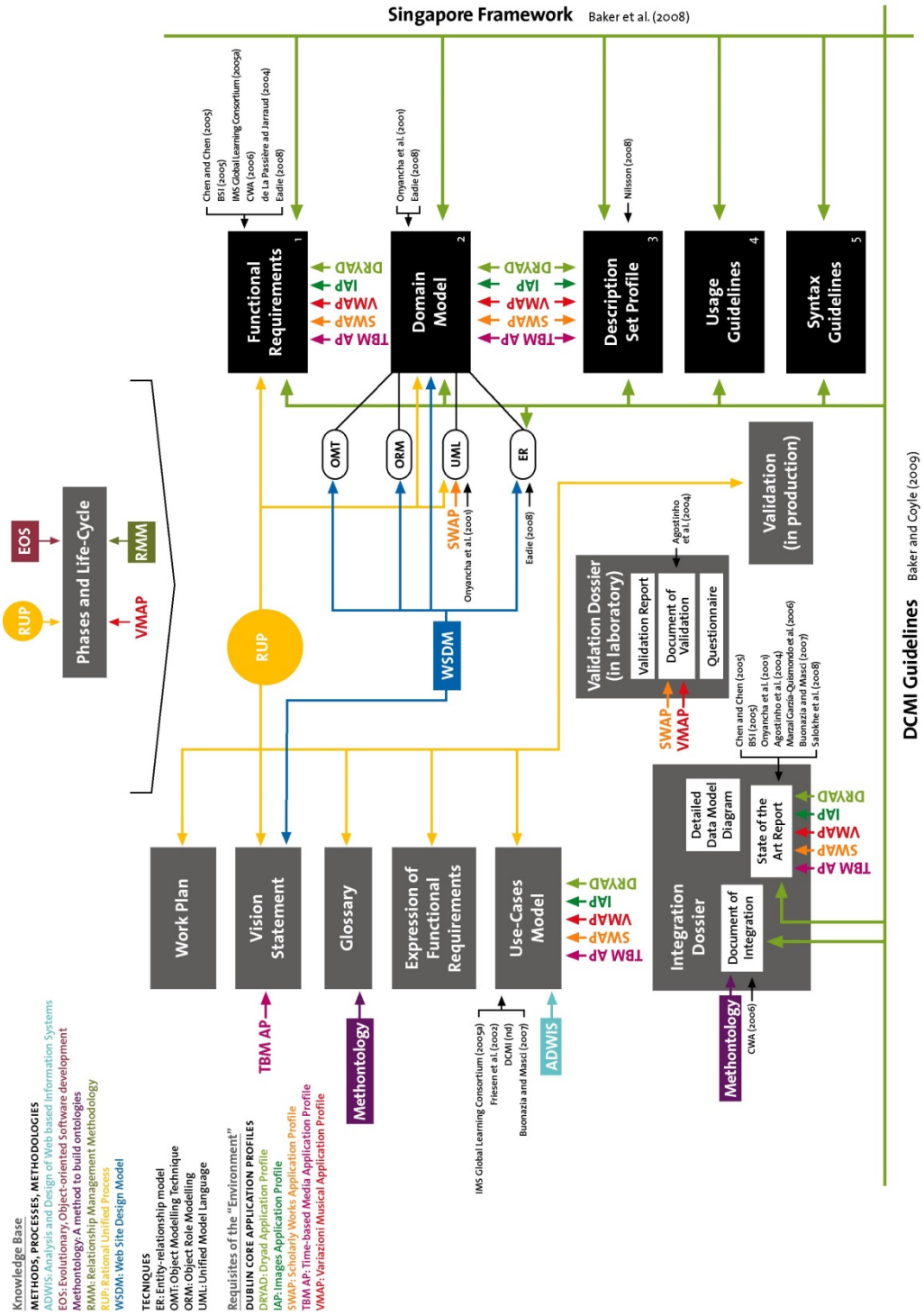

FIG. 5. MeDCAP development sources detailed description

https://doi.org/10.23106/dcmi.952136137