Integration of UML diagrams from the perspective of Enterprise Architecture

Luís Cavique [1][0000-0002-5590-1493]
Mariana Cavique [2][0000-0001-8489-8644]
Armando B. Mendes [3][0000-0003-3049-5852]

[1] Universidade Aberta, Lasige, Portugal
luis.cavique@uab.pt
[2] Universidade Europeia, Portugal
mariana.cavique@universidadeeuropeia.pt
[3] Universidade Açores, Algoritmi, Portugal
armando.b.mendes@uac.pt

Abstract. An integrated view of the information system has been an objective to deal with complexity. However, bibliography proposes many solutions with many synonyms depending on the layer, methodology, framework or tool used, that does not allow a broad view of the system. In this work we chose three basic elements of the information systems and we demonstrate how they are enough to integrate a set of essential UML diagrams. The proposed model firstly defines a set of UML diagrams for each layer of the Enterprise Architecture, and then heuristic rules are detailed in order to ensure vertical and horizontal alignment.

Keywords: UML, CRUD, enterprise architecture, organization alignment

1. Introduction

UML (Unified Modeling Language) [Fowler 2003] is a powerful tool that improves the quality of systems analysis and design. The use of UML iteratively in analysis and design, allows the fulfillment of the system requirements with object-oriented design, as well as with relational databases models.

The bibliography on UML tools is vast and it is presented at different levels and formats. However, it is usually presented in separate chapters, where each chapter refers in detail to use-case diagrams, class diagrams, activity diagrams, state diagrams, sequence diagrams and physical diagrams. UML unified a set of diagrams of different authors, where each UML diagram corresponds to a partial view of the system, keeping the holistic view poorly developed.

On the other hand, Enterprise Architecture [Lankhorst 2013] promises an integrated approach to deal with complexity, going beyond the symbolic models (such as the UML diagrams) and trying to achieve more coherent and meaningful tools, called semantic models.

In this work, our goal is to give an integrated view of the system, linking the UML tools to the CRUD matrix, achromic of <create, read, update, delete> [IBM 1978, Martin 1983]. An Enterprise Architecture perspective will be used, where we seek to find vertical and horizontal alignments, that ensures the consistency of the system.

Enterprise Architecture is usually defined by the three-layers model: business/process architecture, information system architecture and technological architecture, as shown in Figure 1.

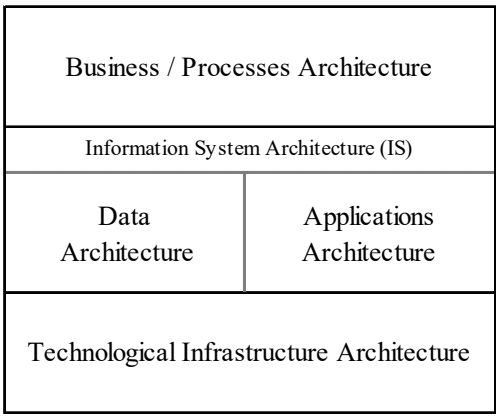

Figure 1. Three-layers model of Enterprise Architecture

In Enterprise Architecture there are many synonyms depending on the layer, methodology, framework or tool used, so it is important to find a synthesis with a reduced number of names. In this work, we chose only three basic elements: the actors, the activities and the data. Some of the synonyms are as follows: (i) actors is synonym of lines-of-responsibility, (ii) activities is synonym of applications, tasks, uses-cases or operational-processes, (iii) data is synonym of classes or informational-entities.

In the following examples we will use actors (α, β), activities (A, B, C) and data (X, Y, Z, W). This version of Enterprise Architecture with three layers, includes:

- Business / Processes Architecture: where processes are made up of activities (A, B, C) and managed by human actors (α, β);
- Information System Architecture: with two different software groups, the data (X, Y, Z, W) and the applications (A, B, C);
- Technological Infrastructure Architecture: which consists of hardware components and basic software (operating systems and database management systems).

The challenge of Enterprise Architecture is to create a vertical alignment that allows the communication among the business team, IS team and IT team, merging the three layers into a single architecture.

The paper is organized in four additional sections. In Section 2, related work is presented. Section 3 introduces an integrated view of UML tools. Section 4 details the proposed method. Finally, in Section 5 conclusions are drawn.

2. Related work

In this section we introduce the subjects of: CRUD matrix, Enterprise Architectures and topic about alignment in information systems.

CRUD Matrix

CRUD matrix was popularized by James Martin [1983] in his book Managing the Data-base Environment. CRUD matrix crosses information between applications and data classes.

This approach intends to obtain a compact view of the system, as well as the 'Design Structure Matrix' techniques [Eppinger, Browning 2012], in order to avoid coupled sub-systems. Both techniques try to find a matrix with the main diagonal filled and the smallest number of elements in the remaining matrix.

Similarly, 'Axiomatic Design' [Suh 2001] studies the transformation of the customer needs into functional requirements and related them to a set of design parameters using a design matrix. The customer needs correspond to the system requirements which are materialize in functionalities/applications. And, the data architecture corresponds to the design parameters and the CRUD matrix to the design matrix. Such as the axioms of axiomatic design, the goal of the CRUD matrix is to maintain the independence of the functional requirements in order to minimize the information content of the design.

applications vs classes	X	Y	Z	W
A	CRUD			
B	R	CRUD		
C	R		CRUD	
others				CRUD
CRUD counters	1311	1111	1111	1111

Figure 2. CRUD matrix with counters

In Figure 2, the CRUD matrix is shown with CRUD counters, that validate possible inconsistencies [Cavique 2020a]. The number 1311 indicates that there are 1 Create, 3 Reads, 1 Update and 1 Delete. There must be a single application that performs Create, Update and Delete, (CUD), and there may be multiple applications with Read operator. Preferably we will have CRUD counters with 1N11, i.e. a unique Create, Update and Delete and multiple Read operators.

Enterprise Architectures

Enterprise Architecture [Lankhorst 2013] reuses the term ‘architecture’ from building and construction, referring to a holistic view of the enterprise. ArchiMate is one of the most popular enterprise modelling language, created to be a meta-model of tools like UML or BPMN. However, the proposed ArchiMate language evolves into fields more and more specific, increasing the lexical complexity and losing the necessary simplicity of an architecture.

In the book of Desfray and Raymond [2014] proposed the EAP (Enterprise Architecture Profile) language which extends the UML concepts in order to represent all TOGAF objects. In the sub-title, the authors promised a practical guide using UML and BPMN, which we believe has not been fully achieved.

In Barros et. al [2000] the authors modeled business processes, business entities, business roles and business events using UML language. However, only the business layer was dealt with, lacking the information system and technological layers.

Silingas and Butleris [2009] proposed an approach to customizing UML tools for domain-specific modeling needs. The authors reused the generic Zachman framework for answering to the 6 Wh questions in Business, System, and Technology models. However, they do not provide a clear procedure to apply the approach.

Perez-Castillo et al. [2019] mapped EA, concluding that the process is costly and subject to errors, which may discourage enterprises from adopting EA.

In this work we will choose the UML diagrams, since they have been on the market for decades and there is a wide community of professionals already familiar.

Information system alignment

The authors [Pereira, Sousa 2005] and [Vasconcelos 2017] presented a set of three heuristic rules that guarantee the alignment of the information system. The heuristic rules, shown in Figure 3, can be summarized as follows:

#1 The data architecture must support the architecture of the business processes;

#2 Each process activity is automated by a single application;

#3 Each data set is managed (CUD) by a single application in the CRUD matrix;

Rules #1 and #2 guarantee vertical alignment and rule #3 guarantees horizontal alignment.

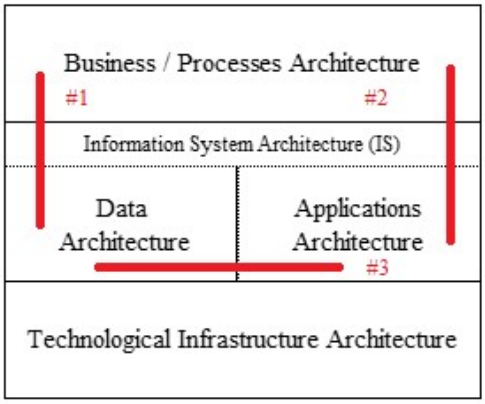

Figure 3. Heuristic rules for enterprise alignment

3. Integrated view of UML tools

BSP (Business Systems Planning) [IBM 1978] presents four basic elements for Information Systems Planning, coined as the Iron Cross: organization, applications, data and technological systems [Rocha, Freixo 2015].

In this approach we use three of the four elements. Our UML analysis considers the following entities: the actors, the use-cases/applications and the data classes [Cavique 2020a]. Next, we demonstrate that three elements are enough to guide the

set of UML diagrams. Although CRUD matrix is not a UML tool, it can complete a system view as shown in Figure 4.

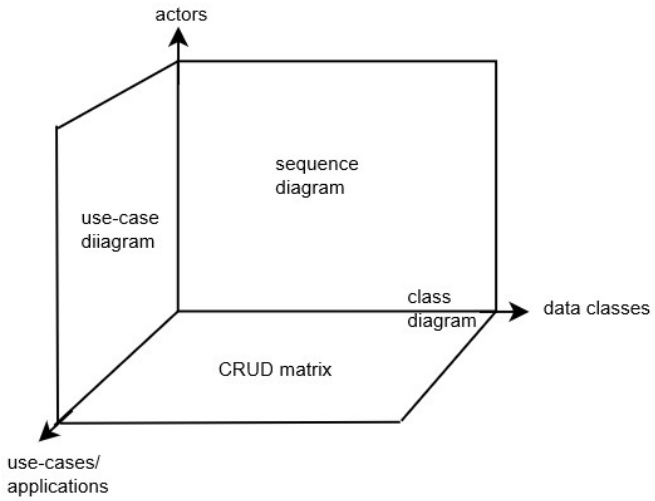

Figure 4. UML diagrams and CRUD matrix guided by the three essential elements

With the three vectors (actors, applications, classes) combinations of an element, of pair of elements and of a trio can be generated.

The only diagram with a single element is the class diagram, which is a consequence of the business narrative.

The pair (actors, applications/use-cases) is called a use-case diagram. A second pair (applications, classes) is represented by the CRUD matrix. A third pair (actors, classes) is represented by the sequence diagram.

Finally, the set of all sequence diagrams represents the trio of (applications, actors, classes).

4. Proposed Model

In the proposed model, firstly we define UML diagrams for each layer of the Enterprise Architecture, then we present the alignment heuristic rules for Enterprise Architecture and finally a procedure to check alignment is detailed.

4.1. UML diagrams for Enterprise Architecture

In this work we intend to detail the Enterprise Architecture using UML diagrams for modeling and systems analysis. Figure 5 shows the three layers of Enterprise Architecture associated with UML diagrams. In the 1st layer (above) it is shown the processes diagram with a sequence of activities. In the 2nd layer, the list of applications (in the center on the right), the data diagram (in the center on the left), the CRUD matrix merging the applications and data items (in the center on the

middle) are presented. Finally, in the 3rd layer, the infrastructure diagram is shown below.

To represent the process diagram, the BPMN (Business Process Model and Notation), an activity diagram with partitions ('swim-lanes') in UML or a simplified version in a UML use-case diagram can be used.

For the representation of the data, an Entity-Relationship diagram or a class diagram in UML can be used. Also, in the 2nd layer there is a list of applications and a CRUD matrix that unify applications and data.

Finally, the infrastructure diagram can be represented by an architecture diagram or UML implementation using components and nodes.

4.2 Heuristic rules to align the Enterprise Architecture

The set of diagrams must be aligned, in order to assure consistency in their articulation. Two alignments are generally considered, vertical and horizontal alignment.

The whole architecture begins with a description of the business area, that we call 'Business Area Narrative'. This narrative can be obtained through meetings, focus groups, interviews (structured or unstructured) or case studies. The narrative generally includes a survey of the 'as-is' system (past and present) and the intended or 'to-be' (future) system, associated with a set of functional requirements.

The analysis of the system considers two possible views: the view of the functional analyst who deals with the end users oriented to the process architecture (1st layer) and the view of the organic / systems analyst oriented to the IS/IT architecture (2nd and 3rd layer). For that reason, from the business narrative two heuristic rules are applied, rule #1 related to data architecture and the new rule #0 is connected to the process architecture.

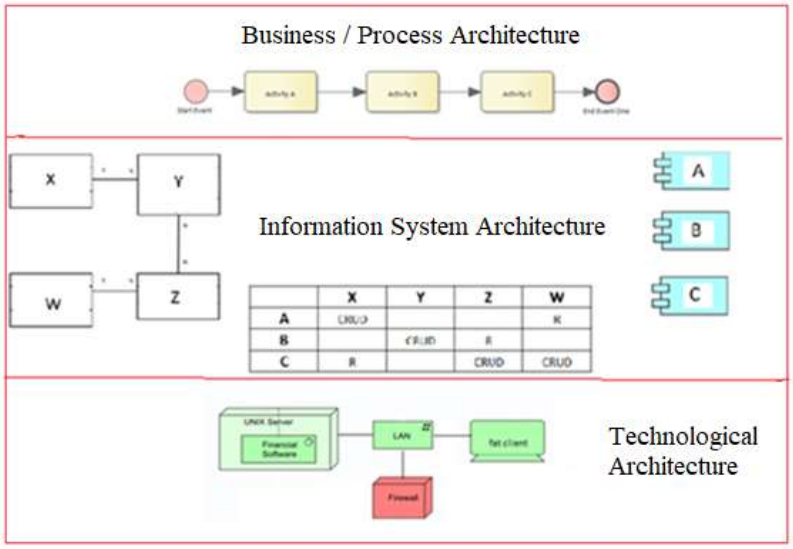

Figure 5. Three layers of the Enterprise Architecture associated with the diagrams

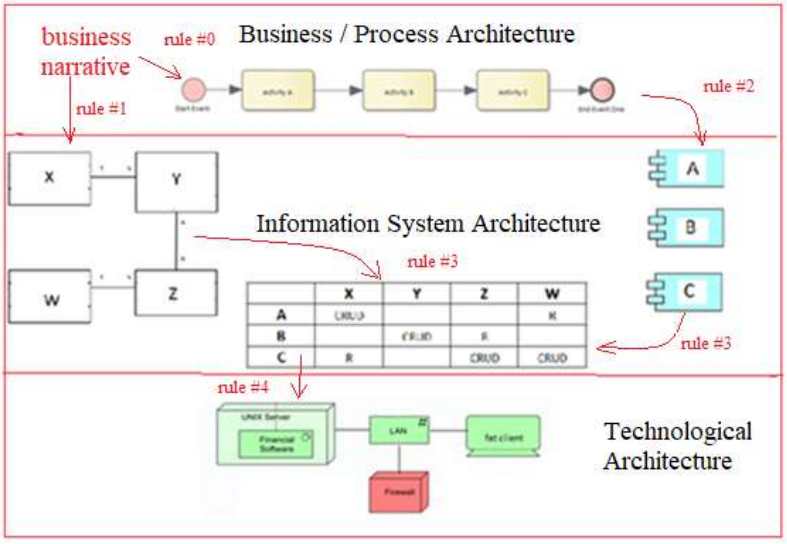

Figure 6. Heuristic rules to align the Enterprise Architecture

The activities that can be automated are called applications, following rule #2. And applying rule #3, the crossing of data with applications is represented in the CRUD matrix.

The authors (Pereira, Sousa 2005) and (Vasconcelos 2017) presented a set of three heuristic rules that guarantee the alignment of the information system. Based on these works, we add rule #0 and rule #4 to our model, which connect the referred diagrams.

Figure 6 illustrates a set of rules to achieve the Enterprise Architecture alignment. The heuristic rules are as follows:

#0 The process architecture must support the narrative of the business area;

#1 The data architecture must support the narrative of the business area;

#2 Each process activity is automated by a single application;

#3 Each data set is managed (CUD) by a single application;

#4 Each infrastructure is associated with one or more applications.

The development of the system begins with the Business Narrative. The application of rules #0 and #1 initiates the two possible views of the system. Then rule #2 is applied, which is responsible for the digital transformation of activities into applications. Rule #3 uses data and applications to create the CRUD matrix. Finally, rule #4, associates a technological infrastructure with one or more applications.

Note that rules #0, #1, #2 and #4 guarantee vertical alignment and rule #3 guarantees horizontal alignment.

Next, we will detail the procedure for modeling the Enterprise Architecture alignment in three layers.

4.3 Procedure to check alignment

Procedure 1 checks the alignment in the Enterprise Architecture and numbered in the same way as the layer number: (1) for the 1st layer of Business processes, (2) for the

2nd layer of information system, IS, and (3) for the 3rd layer, of information technology, IT.

Procedure 1:
(0) define the business narrative
(1) define architecture processes: list activities and actors, activity diagrams
(2) define information system architecture
(2.a) define data architecture: class diagram
(2.b) define CRUD matrix: data versus applications
(2.c) detail CRUD matrix with sequence diagrams
(3) define physical architecture: infrastructure diagram

(0) Define the business narrative. The narrative should refer in a logical way, the actors, activities and data, as well as the way they are articulated in the business.

(1) Define the process architecture (or sequence of activities). List the activities, list the actors and fill in the matrix activities versus actors. Develop a UML use-case diagram, or a UML activity diagram where lines of responsibility with the respective system actors are included. Apply rule # 0, where the process architecture must support the narrative of the business area.

(2) The Information System Architecture we divided into three subpoints:

a) define data architecture: class diagram

b) define CRUD matrix: data versus applications

c) detail CRUD matrix with sequence diagrams

(2.a) Define Data architecture. Create an Entity-Relationship diagram or a UML class diagram for the data architecture that represents the necessary data that correspond to the requirements referred to in the narrative. Apply rule # 1, where the data architecture must support the narrative of the business area.

(2.b) Apply rule # 2 where each activity, which is automated, must correspond to a single application. Produce the CRUD matrix representative of the relationship between applications and classes (data). Apply rule # 3, where each data set is managed (CUD) by a single application.

(2.c) Detail the CRUD matrix for each application, associating a sequence diagram, which details the message sequences between the classes.

(3) Define technological architecture. The UML infrastructure includes technological platforms, servers, client computers, databases, operating systems, etc. Apply rule # 4, associating one or more applications to each infrastructure.

Finally, the three lists of the essential elements of the system (actors, activities and data classes) should be reviewed and the heuristic rules for vertical and horizontal alignment checked.

5. Conclusions

Since organizations are getting more global, bigger, with more relationships and consequently more complex, an effort to obtain a holistic view is increasingly necessary.

Enterprise architecture, composed by three layers, promises companies to deal with digital transformation through a clear presentation and allowing horizontal and vertical alignment of business and IT in a holistic manner.

However, to answer these challenges scientific literature proposes solutions with many synonyms depending on the layer, methodology, framework or tool used, that do not allow a broad view of the system. Additionally, new languages evolve into fields more and more specific, increasing the lexical complexity and losing the necessary simplicity of an architecture. Perez-Castillo et al. [2019] concluded that the EA process is costly and subject to errors, which may discourage enterprises from adopting this view.

In this work we proposed an approach with the three layers of EA, the four elements of the iron cross with five rules for organization alignment, using UML diagrams, since they have been on the market for decades and there is a wide community of professionals already familiar.

BSP (Business Systems Planning) presents four basic elements for Information Systems Planning, coined as the Iron Cross. In this work we chose three basic elements of the information systems and we demonstrate how they are enough to integrate a set of essential UML diagrams. Although CRUD matrix is not a UML tool, it can complete a system view. The proposed model firstly defines a set of UML diagrams for each layer of the Enterprise Architecture, and then heuristic rules are detailed in order to ensure vertical and horizontal alignment.

The procedure to check the alignment in the Enterprise Architecture is based on the works of [Pereira, Sousa 2005] and [Vasconcelos 2017], which present a set of three heuristic rules that guarantee the alignment of the information system. In our model, we extended the set of rules in order to connect the UML diagrams and the CRUD matrix.

The proposed procedure to check the alignment in the Enterprise Architecture is divided into three steps, for an equal number of organization layers. In a previous step the business narrative is defined. Then business architecture, the information architecture and the technological architecture, apply the (i) use-case/activity diagrams, (ii) class diagrams, CRUD matrix with sequence diagrams to detail the applications, and (iii) infrastructure diagram, respectively. The steps between diagrams are supported by the referred heuristic rules, allowing the alignment of the system.

This work is an attempt to reduce and standardize the multiple synonyms in information systems. We also integrate UML diagrams with CRUD in an Enterprise Architecture point of view, allowing a holistic view of the organization.

The presented approach is based on several years of teaching this UML approach to undergraduate students [Cavique 2020a, Cavique 2020b]. In future work, we plan to establish key performance indicators, KPI, in order to measure the ability to adapt to real-world systems.

References

1. Barros A., Duddy K., Lawley M., Milosevic Z., Raymond K., Wood A. (2000), Processes, Roles, and Events: UML Concepts for Enterprise

Architecture, In: Evans A., Kent S., Selic B. (eds), UML 2000: The Unified Modeling Language, Lecture Notes in Computer Science, Springer, vol. 1939, pp. 62-77.
2. Cavique, L. (2020a), Modelação de Sistemas de Informação: Elementos essenciais na visão integrada das ferramentas do UML com a matriz CRUD, Recursos Educativos, Universidade Aberta, Portugal.
3. Cavique, L. (2020b), Arquitetura Empresarial: na perspetiva da integração de diagramas UMLRecursos Educativos, Universidade Aberta, Portugal.
4. Desfray P., G. Raymond (2014), Modeling Enterprise Architecture with TOGAF: A Practical Guide Using UML and BPMN, Elsevier Inc., ISBN 978-0-12-419984-2.
5. Eppinger, S.D., T.R. Browning (2012), Design Structure Matrix Methods and Applications, MIT Press, Cambridge.
6. Fowler, M. (2003), UML Distilled: A Brief Guide to the Standard Object Modeling Language, Addison-Wesley Professional, 3rd Edition, ISBN: 978-032-119-368-1.
7. IBM Corporation (1978), Business System Planning Information - System Planning Guide, International Business Machines Corporation, 2nd edition, New York.
8. Lankhorst, M. (2013), Enterprise Architecture at Work: Modelling, Communication and Analysis, 3rd Editions, Springer, ISBN: 978-364-229-650-5.
9. Martin, J. (1983), Managing the Database Environment, Prentice-Hall, Englewood Cliffs, New Jersey, ISBN: 013-550-582-8.
10. O'Neill, H., M. Nunes, P. Ramos (2010), Exercícios de UML, FCA, ISBN: 978-972-722-616-0.
11. Pereira, C.M., P. Sousa (2005), Enterprise architecture: business and IT alignment, In H. Haddad, L.M. Liebrock, A. Omicini, R.L. Wainwright, eds., SAC, 1344–1345, ACM.
12. Perez-Castillo R., F. Ruiz-Gonzalez, M. Genero, M.Piattini (2019), A systematic mapping study on enterprise architecture mining, Enterprise Information Systems, vol. 13:5, pp. 675-718.
13. Rocha, Á., J. Freixo (2015), Information Architecture for Quality Management Support in Hospitals, Journal of Medical Systems, 39, 125, Springer Link.
14. Silingas D., R. Butleris (2009), Towards customizing UML tools for enterprise architecture modeling, In: Nunes M.B., P. Isaías, P. Powell (eds), IADIS International Conference Information Systems, ISBN: 978-972-8924-79-9.
15. Silva, A., C. Videira (2008), UML - Metodologias e Ferramentas CASE. Edições Centro Atlântico, volume nº1, 2ª edição, ISBN: 978-989-615-009-9.
16. Suh, N.P. (2001), Axiomatic Design: Advances and Applications, Oxford University Press, ISBN 0-19-513466-4.
17. Vasconcelos, A. (2017), Slides Aulas Arquiteturas Tecnológicas Empresariais, DEI, Instituto Superior Técnico, Universidade Lisboa, Portugal.