A pattern based methodology for analyzing enterprise architecture landscape

Jihane Lakhrouit[1], -Karim Baïna[1]

[1] **University Mohammed V, ENSIAS**
Alqualsadi research team on Enterprise Architecture
Rabat, Morocco

Abstract

Nowadays, enterprise architecture (EA) has garnered considerable attention from both practitioners and academics in the fields of information systems and business management. Enterprise architecture (EA) is an approach to managing the complexity of an organization's structures, information technology (IT) and business environment, and facilitating the integration of strategy, personnel, business and IT towards a common goal through the production and use of structural models providing a holistic view of the organization. In this paper, we present a complete pattern based methodology for evaluating the complexity of enterprise architecture. Our objective is to propose an evaluating methodology for guiding designers and architects in evaluating and improving the EA models. Furthermore, our enterprise architecture patterns system will be used for an automated support so as to manage the evaluation of enterprise architecture complexity.

Keywords: *Enterprise Architecture; EA patterns; Analysis of Enterprise Architecture; Complexity.*

1. Introduction

The Enterprise Architecture defines how information and technology will support the business operations and provide benefits for the business. EA is not only a kind of theory to support business and IT alignment, but also a useful and practical methodology. Combining EA and complexity analysis has the advantages of being able to analyze the IT systems in their enterprise wide context. In previous research, managing complexity has been identified as one major challenge in enterprise architecting [29], however IS literature has not mentioned a holistic, i.e. applicable to all dimensions and elements of an EA, approach to quantify complexity in EAs so far[30].

The goal of this paper is to (1) analyze the current state-of-the -art regarding dimensions of complexity and approaches for evaluating EA complexity, (2) identify the components of enterprise architecture based on the information pattern and (3), precise available metrics to calculate the complexity of enterprise architecture using the network theoretic model.

The paper is structured as follows: the second section describes the state of the art of our research, the third section details our proposed approach and presents some of our results, the fourth section presents the numeric results of the components complexity of the proposed method and finally, the last section is dedicated to conclude our paper.

The case study which is used in this paper is an abbreviated version of a study under development in a private university. This is done to give a more comprehensive presentation of how the method can be used and to demonstrate the efficacy of our approach.

2. State of the art

In the following section, we present our literature review related to our research. We define the complexity and its dimensions and the existing methods in the literature which discussed and developed a method of evaluating enterprise architecture.

2.1 Complexity – definition and dimensions

According to Davis and LeBlanc [5] the complexity of application architecture is "number of its components or elements, kind or type of elements and structure of the relationship between elements". On the infrastructure architecture level defined complexity as "The complexity can be defined here as the dramatic increase in the number and heterogeneity of included components, relations, and their dynamic and unexpected interactions in IT solutions"[32], another definition proposed by [31] covers all aspects of complexity "The complexity can be defined on the basis of the number and variety of components and interactions plus the rate of change of these". From the different definitions cited we can notice that the complexity is a fuzzy term, because different stakeholders have generally different views and conceptions of complexity term, as shows the figure 1 below.

From these definitions we will clarify the dimensions of complexity and proposed a global definition which is "The complexity of architecture is the description of its structure and quantification of the numbers and heterogeneity of components and relations between them over the time".

The figure 2 shows the four dimensions of the complexity of enterprise architecture, although the number of components and relationships can be determined by simply counting the respective elements, heterogeneity, calculating change rates and the architectural structure must be calculated using formulas and measures that we must clarify. In this paper, we will discuss only the dimension of enterprise architecture landscape

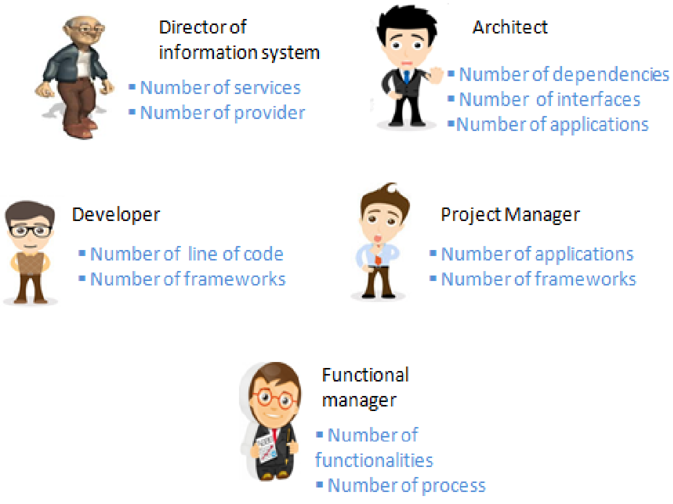

Fig. 1. The different definitions of complexity

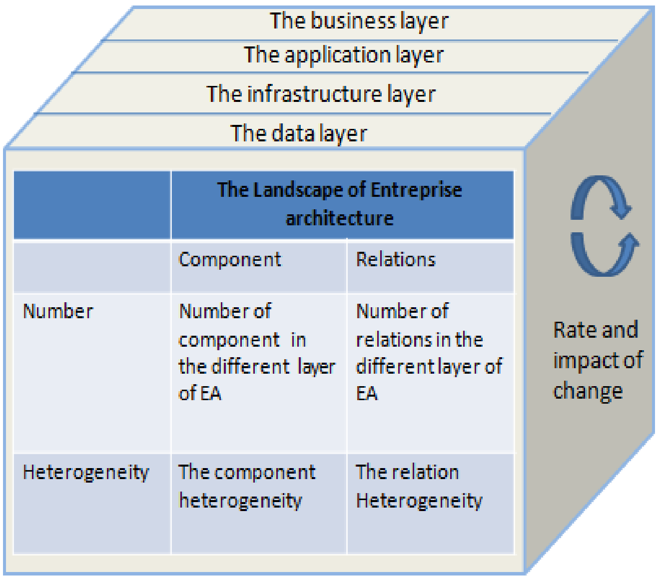

Fig. 2. The dimensions of enterprise architecture complexity

2.2 Methods for measuring complexity

This section present two types of approaches, the first present and calculate only the complexity of one layer of enterprise architecture EA and the second types are more holistic and propose a generic approach for one or more layers of enterprise architecture.

2.2.1 Analysis approaches specific to one layer for evaluating complexity of enterprise architecture

In the literature, the methods of measuring the complexity capture very different aspects of this concept. Two of the most interesting and most relevant aspects affecting the development effort are: The size which refers to the size of the program (for example lines of code) Or functional size (for example function points) and the complexity of the software structure interpreted through the complexity of the design or structure of the code structure [39][40].

Mocker [7] identified the complexity of the application layer as the age of applications and the number of functional requirements defined for each application. Based on the available literature, he identified four different measures to quantify the complexity: the interdependence, the diversity in technologies, the heterogeneity of standards and of technologies. The method proposed by Widjaja [8] revolves around two axes : the heterogeneity of a landscape as a statistical property that can be measured by statistical indices. The measurement to quantify the heterogeneity is the entropy which is proposed by Jacquemin and Berry [9] and a generic mathematical model to quantify the heterogeneity in computer landscape.

2.2.2 Holistic approaches for evaluating complexity of enterprise architecture

During the analysis of the identified contributions only few methods were presented to quantify complexity and the existing methods merely cover parts of an EA, not the EA as a whole. Often the application is so specific that it is not possible to transfer the method to other dimensions of an EA.

In the paper [8] it discussed the metrics for EAs and application landscapes are introduced as decision support techniques based on analysis of structural dependencies. The approach emphasizes on operational risk, failure propagation and availability, based on a practitioner survey. In order to explicate the structural dependencies analyzed in the paper, an information model with derived attributes is used, along with a Bayesian calculation formalism. An EA level application example is also given in the paper [8] with visual analysis of ex post information about failure propagation to compare different project proposals for the evolution of the application landscape. Thus, the project portfolio management process is supported.

Lagerström et al. [12] proposed to use an approach pervasive in the software architecture discipline— Design Structure Matrix—to visualize the hidden structure of an AL and thereby identify spots of increased complexity. Schuetz et al. [10] introduce a metric to quantify the structural complexity of an IT landscape, which is also applicable to application landscape. The proposed approach of Schutz [10] revolves around the conceptualization of the complexity of EA by adopting the concept of the system to the context of EA. This

IJCSI International Journal of Computer Science Issues, Volume 13, Issue 6, November 2016
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org https://doi.org/10.20943/01201606.1524

approach presented a holistic conceptualization of complexity but don't apply it in the different layers of EA.

After define and clarify the complexity of dimensions we present the approach to the management of enterprise architecture management pattern approach that we adopted for the analysis of the EA complexity.

3. Our proposal formalization for the representation of enterprise architecture management pattern

In our approach we propose to design and reuse enterprise architecture management EAM patterns to create the patterns analysis of complexity. Firstly, we clarify the benefits of using the EAM process and secondly we present the approach of EAM pattern and thirdly we will propose new representation formalism.

3.1 Problems standard approaches to enterprise architecture

This section details the EAM pattern approach, which has been introduced by [33] contrasting the variety of approaches from academia and practice, which exhibit at least one of the following problems: EA management is usually introduced from scratch, not considering related initiatives already present in or outside the organization. EA management frameworks, like Zachman [34], TOGAF [35], etc., are usually either too abstract and therefore not "implementable", or too extensive to be used in real world. Lacking an actual starting point for the EA management initiatives, companies tend to call for proposal to a wide number of potential EA stakeholders. Consolidating their demands and integrating their information needs an all-embracing EA management approach is likely to develop, which would demand a vast amount of data to be gathered, although only a part of it would be needed to address the pain points of the company.

If an approach has been implemented, it is mostly not documented, why certain decision have been taken, e.g. why a special entity has been introduced to the information model [34][35][41][42]. This leads to information models, which cannot be adapted or extended due to the fact that no one knows what aspects rely on which parts of the model. Approaches proposed e.g. by organizations or standardization groups are usually a "complete or nothing" approach, meaning that it is supposed to be introduced as one single piece instead of an incremental introduction. This results in an EA management approach that cannot evolve according to the maturity level of the company.

3.2 Presentation of enterprise architecture management pattern

The EAM pattern language developed by Buckl [33] distinguishes between four different types of patterns:

- M-Patterns. Methodologies define steps to be taken in order to address given objectives. These objectives are addressed by procedures defined by the methodology. Others refer to them as Process Patterns.

- V-Patterns. Viewpoints provide the languages used by methodologies. A viewpoint proposes a way to present data stored according to one or more information model patterns.

- I-Patterns. Information models represent underlying models for the data visualized in one or more viewpoints. An information model pattern conveys an information model fragment including the definitions and descriptions of the used information objects.

- The Objective, The EAM pattern language includes a list of typical objectives. They can be used as an entry point and help to select appropriate patterns within a given context.

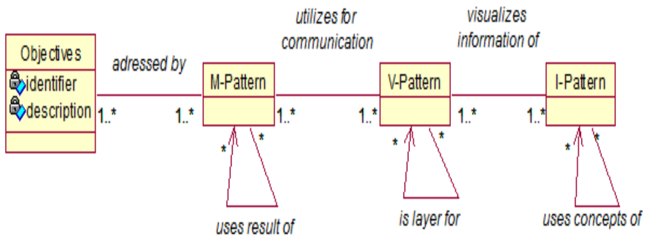

Fig. 3. The conceptual model underlying the EAM pattern language.

EAM pattern approach tries to solve the problems listed above, it is based on best practices, with precise and well documented instructions, such as information model which specifies exactly what data that must be maintained to obtain specific objective. In addition, it is an approach based on the goals, which is expandable because it is based on models and can include justifications for design decisions.

To improve readability, the comparability and structure of enterprise architecture management EAM pattern, all of the patterns follow a formalism proposed in the catalog [15]. This formalism is similar to the structure proposed by [21] it consists of the name of the model, the problem, the solution, the consequences of the solution and the version of the pattern. This formalism does not meet all the reporting needs, which we can mention:

- The lack of formalization of headings: the formalism is poorly suited to automated techniques of research patterns which are required by the designer. The different sections of the pattern and in particular those intended for the understanding and selection patterns are often described in natural language with unstructured text. Only the section expressing the

IJCSI International Journal of Computer Science Issues, Volume 13, Issue 6, November 2016
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org https://doi.org/10.20943/01201606.1524

solution was the formalization of object (via diagrams or algorithms).

- The lack of formalization of inter-relationship patterns: a formal description of the relationships between patterns facilitates the use and organization of its. Still, there already exist formalisms such as P-Sigma [36] and PROPEL [23] that define and formally represent the inter-patron relationship, but at the same time it cannot be applied directly to EAM patterns without adaptation.

To improve the formalism proposed in the catalogue and consider these constraints cited above, we propose in the next section the new formalization of enterprise architecture.

3.3 New Formalization of enterprise architecture management pattern

The formalism, proposed to specify our EAM patterns is an extension of P-Sigma formalism because on one hand it details the overview part and on the other hand it proposes the relations between patterns, but this is insufficient to model the relationship between EAM pattern because P-Sigma represents the relationships for the pattern which have the same type but in the approach of EAM pattern we will need to clarify the relationship of the same type and also the relationship between two different types(for example between the V-Pattern and I-Pattern).
The proposed formalism is grouped into four sections: Overview, Solution, relations and example.
The overview, is dedicated to the selection of patterns, it composed of the columns heading; classification to identify the pattern, the type of pattern (M-Pattern, V-pattern, I-Pattern), context to describe the precondition for the implementation of the pattern, the layer of enterprise architecture(Business, Application, Technology), Problem to define the goal addressed by the pattern and Force to clarify the strengths and weaknesses of the reuse of this pattern to guide the designer's choice.

The section solution, expresses the solution of a pattern using model or approach solution, it consists of model (a pattern of information or perspective that offers a solution in the form of a class diagram and / or view at a modeling language). In the second case for methodology pattern, the new formalism of enterprise architecture pattern, the solution with a class diagram or a point of view is replaced by a solution as an activity diagram of UML [The section Example presents one or more examples of adaptation of the proposed solution, to better understand how to reuse the pattern. The section Relationship allows organizing relations between patterns. There are two types of relationships: RelationSameTypePattern for relations between the same types of patterns or RelationDifferentTypePattern to present the relations between the three types of patterns. RelationSameTypePattern consists of Uses relationships (Represents the relationship: X pattern uses pattern Y in its solution) Requires (The X pattern mentions the pattern Y in context, and Y is applied before X. "), Raffine (a pattern X is a refinement of a pattern Y if X can solve the problems Y) and Alternative(a pattern X is an alternative of a pattern Y if X can replace the problems of Y). RelationDifferentTypePattern has three relationships "Address" (The M-pattern X addresses the objective Y) "uses" (The M-Pattern X utilize the V-Pattern Y)

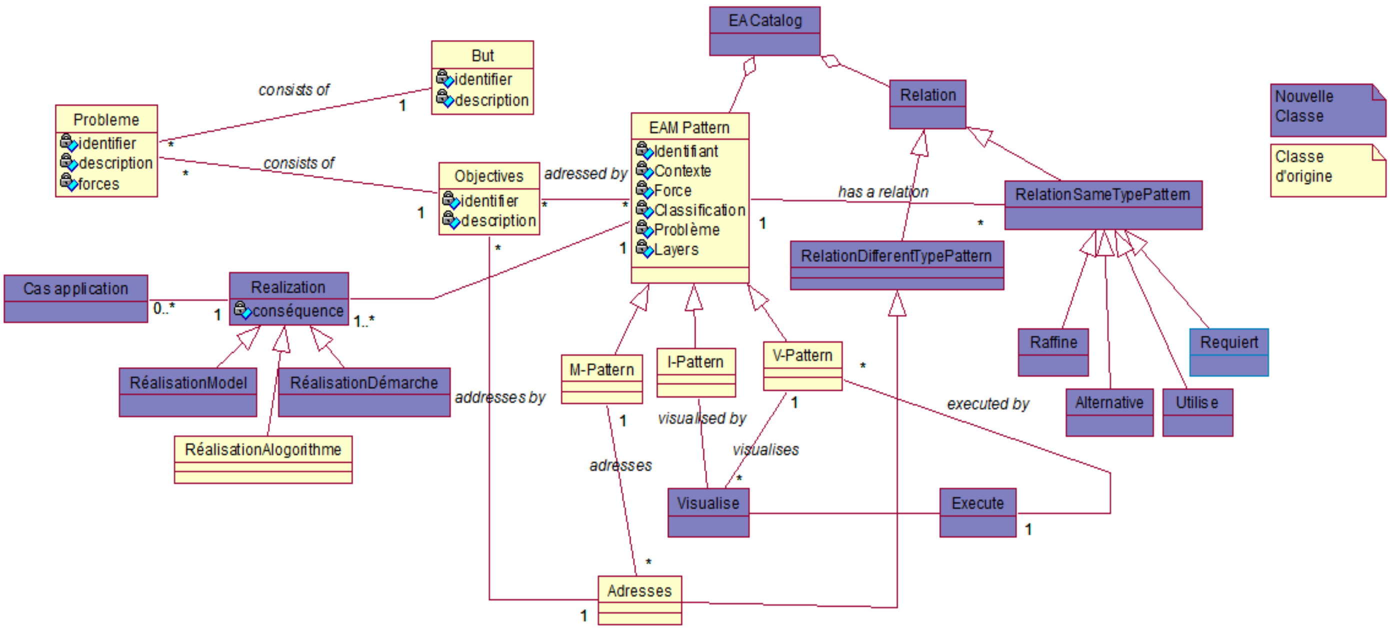

Fig. 4. The new formalism of enterprise architecture pattern

IJCSI International Journal of Computer Science Issues, Volume 13, Issue 6, November 2016
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org https://doi.org/10.20943/01201606.1524

and "visualizes" (the V-Pattern X visualizes the I- Pattern Y).

In this section we proposed a formalism to represent the EAM pattern, the figure below shows the class diagram of the proposed formalism. The following section details the objectives, the information patterns, the visualization patterns and proposed methodology pattern for analyzing enterprise architecture complexity.

4. Our proposal for evaluating enterprise architecture complexity

Based on the definition proposed, we remark that complexity consist of four dimensions, in this paper we will propose methodologies patterns to analyzing the dependences of enterprise architecture landscape. To describe each enterprise architecture pattern, we use the EA-formalism proposed in the previous section, but before we define the objectives of our methodology pattern.

4.1 Conceptual Foundation: Measures of Network Analysis

Before exploring the use of networks analysis NA in enterprise architecture landscape, we precise in this section the conceptual foundation of our work, introducing basic concepts of NA and clarifying their meaning in our context. Rooted in graph theory, NA conceptualizes and visualizes structures that emerge from any interaction or connection as networks and allows a quantitative analysis of the network nodes' relationships.
As indicated, the representation of the IT landscape as a network of nodes and edges is central to our approach. Nodes represent the EA components, which we will precise in the next section with the concept of I-Pattern; edges represent relationships and interdependencies between the components.
To analyzing enterprise architecture landscape, two visions are considered in network analysis: A "micro-view" which considers the individual structure of each node and a "macro-view" that provide complete visualization of the network and provides an assessment of the level of connectivity. The table below details the metrics considered in our approach:

TABLE I. THE NETWORK METRICS USED IN OUR APPROACH

Dimension	Metric	definition
Micro-view	Degree centrality	$DC(i) = \frac{\sum x_{ij}}{n-1}$
	Closeness centrality	$CP(i) = \frac{n-1}{d(i,j)}$
	Betweenness centrality	$CI(i) = \frac{\sum g_{ijk}}{\sum g_{jk}}$
	Eigenvector centrality	$Ce(vi) = \frac{1}{\lambda}\sum_{j=1}^{n} Aj,iCe(vj)$
	Average neighbor degree	$k_{nn,i} = \frac{1}{N(i)} \sum_{j \in N(i)} k_j$
Macro-view	Density	$Density = \frac{2L}{n(n-1)}$
	Modularity	$Modularity = \sum_{C \in P} \frac{Wc}{W} - \left(\frac{Dc}{2W}\right)^2$
	Clustering	Applying the partitioning algorithm using modularity for each iteration.

Degree centrality (C_D) represents the number of relations of a given node and thus indicates the degree of "activity" [30, 31, 32] of applications within the IT landscape. Formally, it can be defined as follows:

$$C_D(i) = \frac{\sum x_{ij}}{n-1}$$

Where x_{ij} equals 1 if there is a link between applications i and j, and $x_{ij} = 0$ otherwise.

Closeness centrality (C_C) measures the geodesic distance of a given node to all other nodes in the network. The node that can reach all other nodes in the fewest steps is most central. C_C can be formalized as

$$Cc(i) = \frac{n-1}{d(i,j)}$$

Where d_{ij} is the number of links in a shortest path from application i to j ($i \neq j$).

Betweenness centrality (C_B) represents the "number of shortest paths that pass through a given node" and therefore indicates whether an application plays some kind of a gatekeeper function, controlling data exchange in the overall network. In mathematical terms, it can be written as

$$C_B(i) = \frac{\sum g_{ijk}}{\sum g_{jk}}$$

Where g_{jk} denotes the number of shortest paths from component j to k ($j, k \neq i$), and g_{jik} is the number of shortest paths from component j to k passing through application i.

IJCSI
www.IJCSI.org

The eigenvector centrality (CE), quantifies the extent to which nodes are connected to other central nodes in the network. For computing this measure for a given node, the relationships to other nodes are thus weighed based on these nodes' centralities:

$$-\sum \mathrm{Aj}$$

Eventually, we also consider overall graph density (as the number of edges L divided by the maximum number of edges in a full graph)

$$\overline{}$$

Modularity is defined as the number of edges falling within groups minus the expected number in an equivalent network with edges placed at random.

$$\sum - \quad (-)$$

A network N having n nodes: 1, 2, ⋯ n. P a partition of the set of nodes in k (k≤ n) groups: C1, C2, C3, ... Ck. Wc the number of edges within the group C. Dc the sum of degrees of all nodes in the group C. W the number of the egdes in the network.

4.2 Objectives for analyzing enterprise architecture landscape

In this section, we define the objectives of our approach to measure the dimensions of the EA complexity. The objectives are measuring the heterogeneity and dependence between EA components. For this we specify three objectives to achieve:

C-102: Present Enterprise Architecture landscape and specify the dependence between the business, the application and the infrastructure layer. C-103: Measuring the heterogeneity of EA components. C-104: Measuring the dependencies between EA components.

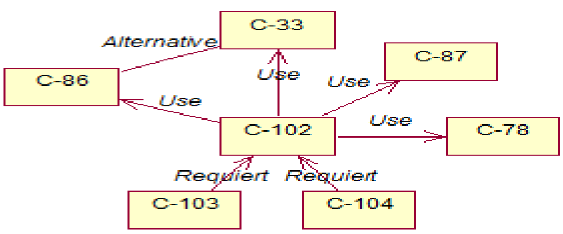

Fig. 5. The relations between our objectives and the objectives of EAM pattern catalogue

C-33: Which applications are used by which organizational units?

C-86: Which business applications are hosted by which organizational unit?
C-87: Which business processes are supported by which business application?
C-78: To which extent are the business processes supported by business applications? Which business processes are supported manually? Can the automated support be extended?
In this paper we will discuss the solution for the objectives c-102 et c-104.

4.3 I-patterns for analyzing enterprise architecture landscape

This section presents the information patterns for the analysis of the EA. We define two patterns of information, the first to model the structure of the EA and the second to transform the landscape on the graph.

- EA landscape I-pattern

Overview:

TABLE II. THE OVERVIEW OF THE EA LANDSCAPE I-PATTERN

Classification	EA landscape
Context	Use{I-25}, Use{I-18}, Use{I-56}, Use{I-12}
Layer	Business, Application, Technology
Problem	This pattern shows how to present the components which defined the enterprise architecture landscape
Force	This pattern presents the EA structure landscape

Realization:

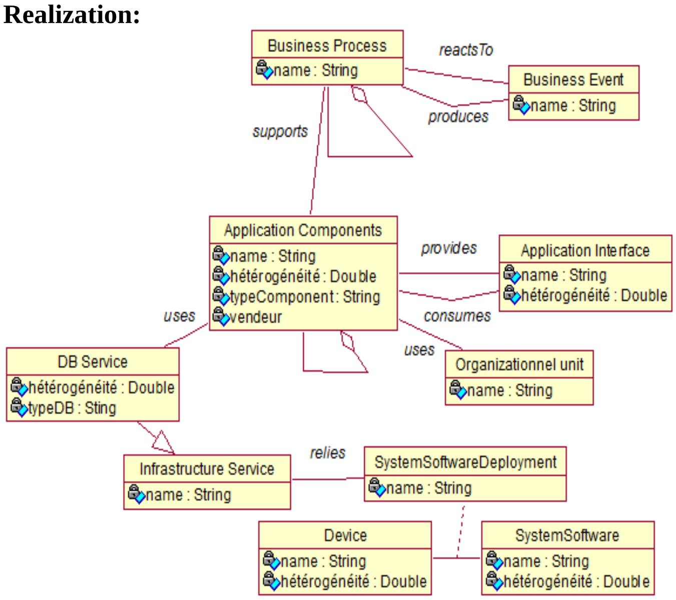

Fig. 6. The I-Pattern for the visualization of EA landscape

IJCSI International Journal of Computer Science Issues, Volume 13, Issue 6, November 2016
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org https://doi.org/10.20943/01201606.1524

We have already defined and discussed all of the EA components in the paper [38].

Relation:

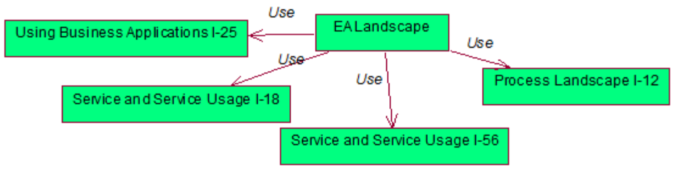

Fig. 7. The relations between the I-pattern EA Landscape and the I-Pattern of EAM pattern catalogue

- EA Structure I-pattern

In this pattern we describe how to present the different EA components using the network graph and how to calculate the structure using the network formula.

We propose to model EA landscape as a network G(N,M) consisting of N nodes and M links between these nodes, where links or arcs present the flow of the information. Our network is undirected because we do not consider the directions of the information flow. The network compromises several types of nodes.

Overview:

TABLE III. THE OVERVIEW OF THE EA STRUCTURE I-PATTERN

Classification	EA Structure
Context	Use {EA landscape}
Layer	Business, Application, Technology
Problem	This pattern shows how to present the components of EA but don't explain how we can analyze
Force	This pattern presents the EA structure landscape
Context	Use {EA landscape}

Realization:

The EAMetaModel represents the enterprise architecture models which are composed of components represented by EAMetaModelComposants and relations represented EAMetaModelRelations. In the model we can have three types of nodes: MetaNodeMetier, MetaNodeApplication and MetaNodeInfrastructure. The MetaNode and MetaEdge have properties defined in the MetaProprites

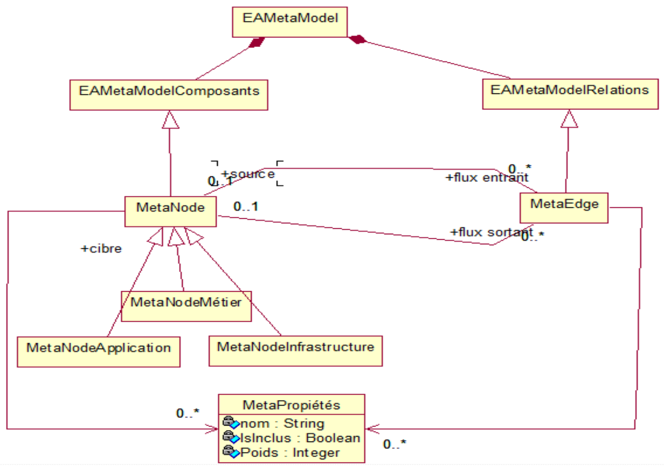

Fig. 8. The model of enterprise architecture presented as graph

Relations:

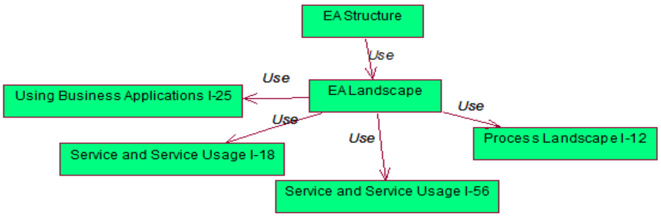

Fig. 9. The relations between the I-pattern EA structure and the I-Pattern of EAM pattern catalogue

4.4 V-patterns for analyzing enterprise architecture landscape

In the standard ArchiMate, there are 13 views available for the different layers. In the contribution we will use existing points of view and we will add other views to complement the methodology

Realization:

The figure below presents the V-pattern of EA landscape

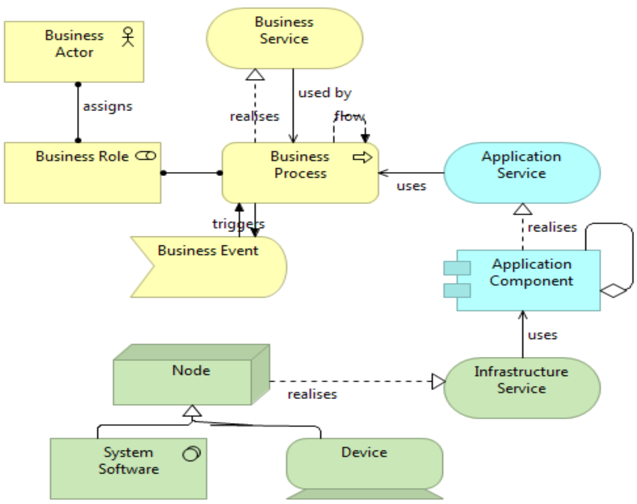

Fig. 10. The archimate diagram proposed to modelise EA landscape

IJCSI
www.IJCSI.org

IJCSI International Journal of Computer Science Issues, Volume 13, Issue 6, November 2016
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org https://doi.org/10.20943/01201606.1524

Relations:

The figure 11 below presents the dependences between the proposed patterns and the patterns of EAM pattern catalogue.

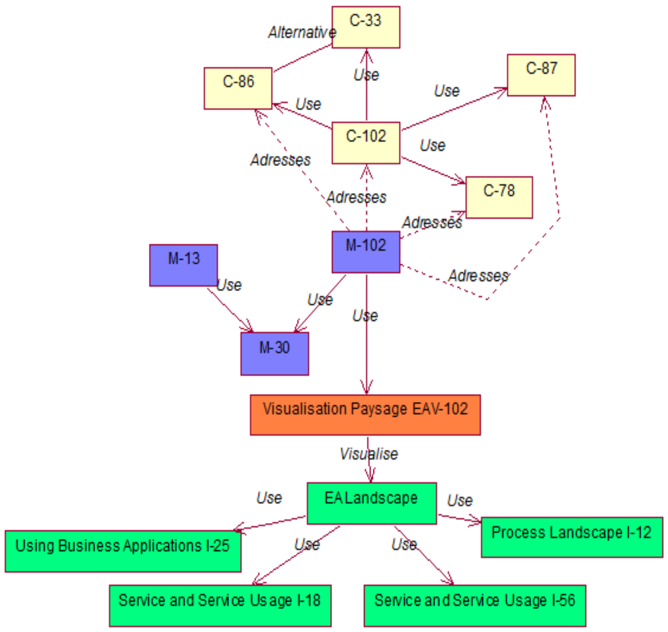

Fig. 11. The relations between the V-pattern EA landscape and the Patterns of EAM pattern catalogue

4.5 M-pattern for analyzing enterprise architecture

In this pattern we will present the different steps to achieve for measuring the complexity of enterprise architecture. We will detail realization part using an algorithm and application example applied to information system of a private university.

Realization

The algorithm of methodology is presented in the figure 12**.**

Example of application:

The case study which is used in this paper is an abbreviated version of a study under development in a private university. This is done to give a more comprehensive presentation of how the method can be used and to demonstrate the efficacy of our approach. In our case study, we chose the case of a university. The university has an enormous amount of information.

We decided to use the existing data that we could recover from existing software applications. We have refined the information necessary to provide a basis for the landscape of business processes and applications enabling better management of future information system. Also, we used surveys among administrators using different applications to help us improved modeling.

In this paper, we present in the figure 13 an extract of applications landscape and in the figure 14 an extract of business process and we discuss the results of analyzing the landscape. After studying the landscape we detected around 338 nodes and 859 edges, which represent a 0.015 density and 0.49 of modularity. About the average degree centrality, it was 4.5.
On this basis, we calculated the degree of centrality CD, centrality betweenness CB and centrality spectral CE in the applications network (The organization asked to focus on some measures using the AHP method detailed in the paper [37]). From the measurements, we invested discussions with the architect responsible of enterprise architecture and software architects that created the landscape of enterprise architecture, at this meeting we asked architects to propose some applications that are most critical, prone to failure, complex, expensive and difficult to adapt, and later after presenting the ten best applications according to each metric. We compared the measurement results with the intuitive notions.

Step 1: Collect data and documents necessary to apply the methodology
IF we want to model each layer separately
Step 2: Apply the point of vue of Archimate
Business Process Viewpoint
Application Structure Viewpoint
Infrastructure Usage Viewpoint
Else
Step 2: Apply Step3
End IF
Step 3: Apply the I-Pattern EA landscape proposed in our approach**.**
Step 4: Apply the V-Pattern Visualization EA landscape proposed in our approach.
If we want to analyze the structure of enterprise architecture
Step 5: Apply the I-Pattern EA Structure to transform the models of Enterprise architecture on graph.
Step 6: Measure the macro-view metrics of the graph.
Step 7: Measure the micro-view metrics of the graph.
End IF

Fig. 12. The methodology of enterprise architecture landscape

IJCSI
www.IJCSI.org

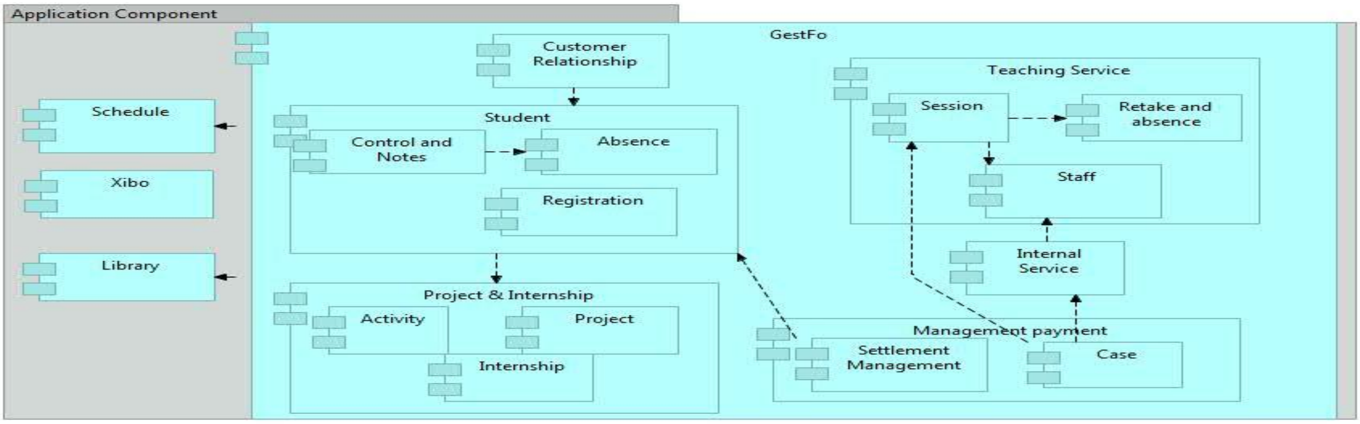

Fig. 13. ISGA information system applications components

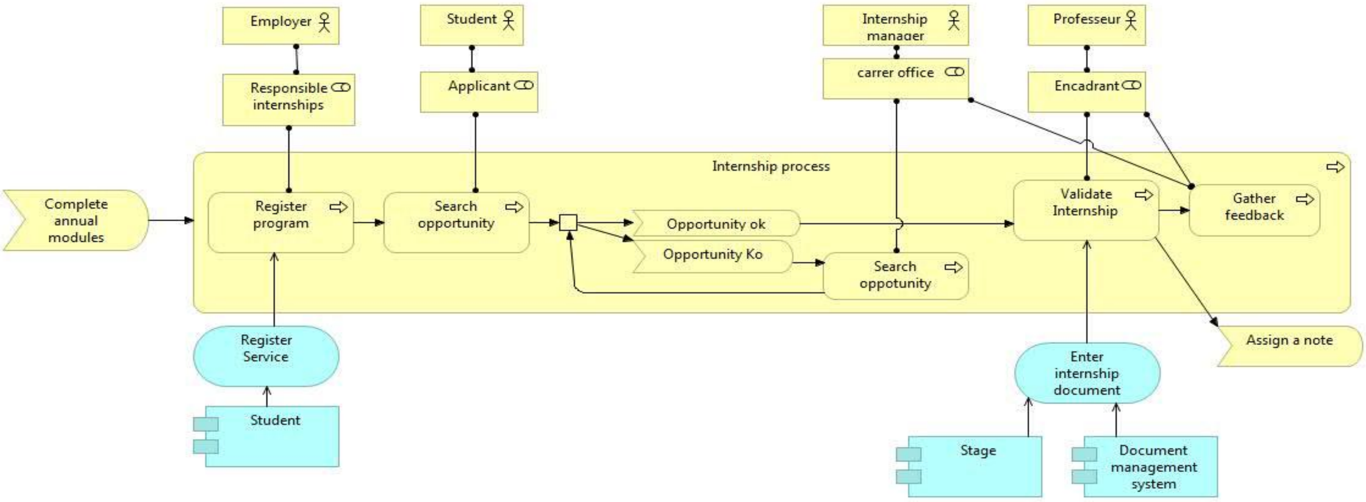

Fig. 14. ISGA information system business process components

Some figures were evident for them, especially with respect to applications that represented the largest measure of degree centrality, she was the one that represent key applications. In fact, all indicators point to the same four applications to be more central. However, other applications have not been raised as important when measuring the degree of centrality, by far against the betweenness centrality CB showed a significant value, for example the application of management absent (CB = 0.2), which did not have a great degree of centrality. The measurements obtained from the betweenness centrality also identified a central accounting application (CB = 0.14) which was considered especially valuable by the architects themselves. However, other applications have not been raised as important when measuring the degree of centrality, by far against the betweenness centrality CB showed a significant value which did not have a great degree of centrality.

5. Conclusion

Enterprise architecture (EA) is the process of translating business vision and strategy into effective enterprise change by creating, communicating and improving the key principles and models that describe the enterprise's future state and enable its evolution. The scope of the enterprise architecture includes the people, processes, information and technology of the enterprise, and their relationships to one another and to the external environment. Definitely, it allows optimizing the alignment, business strategy, organizational culture, business, people, process and technology. In this article, we proposed to use enterprise architecture management patterns to overcome the problem of generic approaches and enterprise architecture frameworks. We proposed patterns of information, visualization and methodology to analyze the enterprise architecture landscape using a new formalization also proposed in this paper.

References

[1] C. Lucke, S. Krell, and U. Lechner, "Critical Issues in Enterprise Architecting – A Literature Review," in Proceedings of the 16th Americas Conference on Information Systems (AMCIS), 2010.

[2] MM. Lehmann, "Laws of software evolution revisited. In: Montangero C (ed) Software process technology", proceedings of the 5th European workshop on software process technology, Nancy, Oct 1996 (Lecture notes in computer science), vol 1149. Springer, Berlin, pp 108–124.

[3] C. Schmidt, P. Buxmann, "Outcomes and success factors of enterprise IT architecture management: empirical insight from the international financial services industry". Eur J Inf Syst 20(2): 2011, 168–185

[4] IEEE Standards Board, "IEEE Standard Glossary of Software Engineering Technology", Technical report, the Institute of Electrical and Electronics Engineers, Sep. 1990.

[5] J.S. Davis,and R.J. LeBlanc, A Study of the Applicability of Complexity Measures. IEEE Transactions on Software Engineering, 14 (9), 1366,

[6] Schneberger S., McLean E., 2003, "The complexity cross: implications for practice", Commun ACM 46(9):216–225.

[7] Mocker M., 2009, "What is complex about 273 applications? Untangling application architecture complexity in a case of European investment banking". In: Proceedings of the 42nd Hawaii international conference on system sciences (HICSS), IEEE Computer Society, Washington, DC.

[8] Widjaja T., Kaiser J., Tepel D., Buxmann P., 2012, "Heterogeneity in IT landscapes and monopoly power of firms: a model to quantify heterogeneity". The international conference on information systems (ICIS), Orlando, FL.

[9] Jacquemin, A.P., and Berry, C.H. 1979. "Entropy Measure of Diversification and Corporate Growth," The Journal of Industrial Economics (27:4), pp 359-369.

[10] Schutz, T. Widjaja, J. Kaiser, 2013, "Complexity in enterprise architectures—conceptualization and introduction of a measure from a system theoretic perspective". the European conference on information systems (ECIS), Utrecht, The Netherlands.

[11] Saat, J., Aier, S., and Gleichauf, B. 2009. „Assessing the Complexity of Dynamics in Enterprise Architecture Planning-Lessons from Chaos Theory" in Proceedings of the 15th Americas Conference on Information Systems, San Francisco, California.

[12] Lagerstrom R., Baldwin CY., Maccormack AD., Aier S., 2013, "Visualizing and measuring enterprise application architecture: an exploratory telecom case". School working paper 13-103. Harvard Business, Cambridge, MA.

[13] Alexander, C., Ishikawa, S., And Silverstein, M. 1977. A Pattern Language. Vol. 2. Oxford University Press, New York. About the Portland Form, http://c2.com/ppr/about/portland.html, 1994

[14] Buckl S., Ernst A.M., Lankes J., Schneider K., and Schweda C.M.. A Pattern based Approach for constructing Enterprise Architecture Management Information Models. In In Proceedings of the IEEE Enterprise Distributed Object Computing Conference, 2009. IEEE.

[15] Buckl, S., ERNST, A. M., MATTHES, F., AND LANKES, J. 2008b. Enterprise Architecture Management Pattern Catalog.

[16] Buckl, S., Ernst, A. M., Matthes, F., Ramacher, R., And Schweda, C. M. 2009. Using enterprise architecture management patterns to complement TOGAF. In Proceedings of the IEEE Enterprise Distributed Object Computing Conference, 2009. IEEE.

[17] Ernst, A. M. 2008. Enterprise Architecture Management Patterns. In Proceedings of the 15th Conference on Pattern Languages of Programs, J. Yoder and A. Aguiar, Eds. ACM.

[18] Mili H., Mcheick H., Sadou S., « CorbaViews – Distributing Objects that Support Several Functional Aspects », in Journal of Object Technology, vol. 1, no. 3, Special issue : TOOLS USA 2002 proceedings, pp. 207-229, 2002.

[19] Nassar M., « Analyse/conception par points de vue : le profil VUML », Thèse de l'Institut National Polytechnique de Toulouse, Septembre 2005.

[20] Lankes J. and Schweda C.M.. Using Metrics to Evaluate Failure Propagation and Failure Impacts in Application Landscapes. In Multikonferenz Wirtschaftsinformatik, Berlin, Germany, 2008. GITO-Verlag.

[21] Gamma E., Helm R., Johnson R.E., Vlissides J., Design patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1995.

[22] Coad P., Object Models : Strategies, Patterns and Applications, Yourdon Press Computing Series, 1995

[23] Hagen, M., Gruhn, V., "Process Patterns - a Means to Describe Processes in a Flexible Way" . ProSim 2004, Edinburgh, United Kingdom, 24.05.2004 - 25.05.2004;Kwoka Jr, J.E. (1985). The Herfindahl Index in Theory and Practice. Antitrust Bull., 30, 915-947.

[24] Garrison, C.B. and Paulson, A.S. (1973). An Entropy Measure of the Geographic Concentration of Economic Activity. Economic Geography, 49 (4), 319-324. Berlinger, E. (1980). An Information Theory Based Complexity Measure. Proceedings of the National Computer Conference. ACM.

[25] Harrison, W. (1992). An Entropy-Based Measure of Software Complexity. IEEE Transactions on Software Engineering, 18 (11), 1025-1029.

[26] Henningsson, S. and Hanseth, O. (2011). The Essential Dynamics of Information Infrastructures. International Conference on Information Systems. Shanghai, China.

[27] Jonkers H., Groenewegen L., Bonsangue M., and van Buuren R.. A language for Enterprise Modelling. In Lankhorst Marc., editor, Enterprise Architecture at Work. Springer, Berlin, Heidelberg, New York, 2005.

[28] UML, Specification UML, version 2.4.1, 2011.

[29] C. Lucke, S. Krell, and U. Lechner, "Critical Issues in Enterprise Architecting – A Literature Review," in Proceedings of the 16th Americas Conference on Information Systems (AMCIS), 2010.

[30] A. Schneider, T. Reschenhofer, A. Schutz, F. Matthes, "Empirical Results for Application Landscape Complexity",48th Hawaii International Conference on System Sciences, 2015

[31] S.L Schneberger, S.L. and E.R McLean. The Complexity Cross: Implications for Practice. Communications of the ACM, 46 (9), 2003, 216-225

[32] Y.Y. Yusuf, E.O Adeleye,"A comparative study of lean and agile manufacturing with related survey of current practices in the UK", International Journal of Production Research 40 (17), 2002, pp 4545–4562.

[33] S. Buckl, A.M. Ernst, J. Lankes, K. Schneider, and C.M. Schweda. A Pattern based Approach for constructing Enterprise Architecture Management Information Models. In A. Oberweis, C. Weinhardt, H. Gimpel, A. Koschmider, V. Pankratius, and Schnizler, editors, Wirtschaftsinformatik 2007, pages 145{162, Karlsruhe, Germany, 2007. University atsverlag Karlsruhe.

[34] J. Zachman. Extending and Formalising the Framework for Information Systems architecture. IBM Systems Journal, Vol. 31(3), 1992.

[35] The Open Group. TOGAF (The Open Group Architecture Framework). Version 8.1 "Enterprise Edition". The Open Group, 2003.

[36] Conte, A., Fredj,M. Giraudin, JP. Rieu, D., "P-Sigma : un formalisme pour une représentation unifiée de patrons", Inforsid, Genève, 2001

[37] Lakhrouit J., Baïna K., 2015 Evaluating enterprise architecture complexity using fuzzy AHP approach: Application to university information system. AICCSA 12th IEEE/ACS International Conference of Computer Systems and Applications, AICCSA 2015, Marrakech, Morocco, November 17-20, 2015. IEEE 2015, ISBN 978-1-5090-0478-2

[38] Lakhrouit J., Baïna K., Enterprise Architecture Complexity Component Based on Archimate Language. Advances in Ubiquitous Networking - Proceedings of the UNet'15, Casablanca, Morocco, September 8-10, 2015.Lecture Notes in Electrical Engineering 366, Springer 2016, ISBN 978-981-287-989-9

[39] Arti Chhikara and R.S.Chhillar Analyzing the Complexity of Java Programs using Object-Oriented Software Metrics, International Journal of Computer Science Issues, Vol. 9, Issue 1, No 3, January 2012.

[40] F. Sabhia., Md. U. Ashraf., A Survey on Clustering Algorithms and Complexity Analysis, JCSI International Journal of Computer Science Issues, Volume 12, Issue 2, March 2015.

[41] Babak Darvish-Rouhani, Sadegh Kharazmi, " Presenting New Solution Base On Business Architecture For Enterprise Architecture", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 3, No 1, May 2012, ISSN.

[42] Babak Darvish-Rouhani, Fatemeh Nikpay "Aagent-Oriented Enterprise Architecture: new approach for Enterprise Architecture" IJCSI International Journal of Computer Science Issues, 9 (6) (2012 Dec)