Sistema de Gestão de Arquitecturas Empresariais
Especificação de Viewpoints

Pedro Mbote Zanani Panzo

Instituto Superior Técnico (IST)

mbotepm@gmail.com

Summary: Organizations like any other complex system can best be understood through a set of viewpoints of artifacts that make up the organization and its relationships. Viewpoints specify how to build a view. And they provide a means to focus on particular aspects of a description of the architecture. These aspects are determined by the concerns of stakeholders with whom communication takes place. Enterprise Architectural Management tools allow you to specify the viewpoints. But you can only specify one that has mastery of the meta-model. And once the viewpoints are specified it is necessary to create queries to obtain the data of the architectural repository to be presented in the views. These queries are done manually, because these tools do not allow you to create automated queries. And this has made it difficult for users who need to make quick decisions based on the data stored in the architectural repository. In order to help these users, the present dissertation aims to develop *"Q-EAMS"* software based on an Enterprise Architecture (EAMS) management tool that will allow them to navigate the architectural repository in order to specify the viewpoints that Intended to view the repository using the *"AndarPor"* algorithm. And it will also allow you to create automated queries on the viewpoint artifacts specified by "AndarPor" using the ERML query language specifications, making the process easier and simpler to do without manual modeling effort.

.**Key words**: Viewpoints, visual queries, metamodel, EAMS.

1. Introduction

1.1 Enterprise Architecture

An Enterprise Architecture can be defined briefly as an organized collection of plans and models that seek to describe and represent the knowledge of an organization. Organizations are constantly evolving entity, either due to internal stimuli or external stimuli (1) and it is necessary also to Enterprise Architecture can keep up this momentum, not only supporting the change, but also encouraging her and guiding her .

Enterprise Architecture is "a well-defined practice for conducting business analysis, project planning and implementation, always using a holistic approach for the successful development and implementation of the strategy.

The enterprise architecture applies principles and practices to guide organizations through four areas (as shown in figure 1): business, information, technological processes and changes needed to execute their strategies. These practices use the various aspects of a company to identify, motivate and achieve these changes. "[4] Each domain is described by various artifacts such as models, blueprints, processes, capabilities, standards and reference architectures that describe the next big step in the evolution of an organization, often called the "future state". It also analyzes the gaps between the future status and the current status, and provides maps that support the company's evolution to the future state, closing gaps or breaks. EA artifacts are used to define target solutions, resources, etc. [4]

Figure 1 - EA domain [4].

1.2 Problem

Companies or organizations, like any other complex system, can be better understood through a set of viewpoints of the artifacts that make up the organization and their relationship. The viewpoints provide a means to focus on particular aspects of a description of the architecture. These aspects (objects and relationships that should appear in a view) are determined by the concerns of stakeholders with which communication takes place. What should and should not be visible from a particular point of view is therefore totally dependent on the argument with regard to a stakeholder concerns. Relevance to the concerns of the stakeholders, therefore, is the selection criteria that is used to determine which objects and relationships must appear in a view [11]

An architectural view is a representation of the architecture from the perspective of a related set of concerns [11].
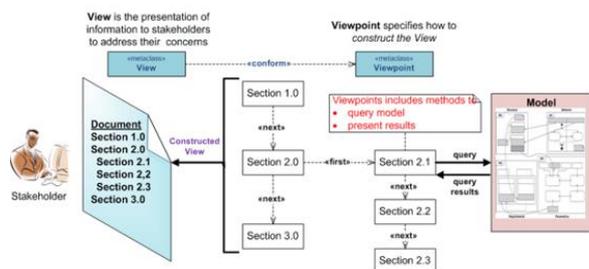

Figure 2 - Overview of a viewpoint

The contents of the different architectural views are obtained through consultations in architectural repository as we see in Figure 2 above. A query is a specification that gets a result set that includes content repository. Consultations are used alone or can support the Blueprint project and Charts in enterprise architecture management tools (EAMS [8] Vitech [9] MAM-EA [10] and ARIS-EA [11]) to

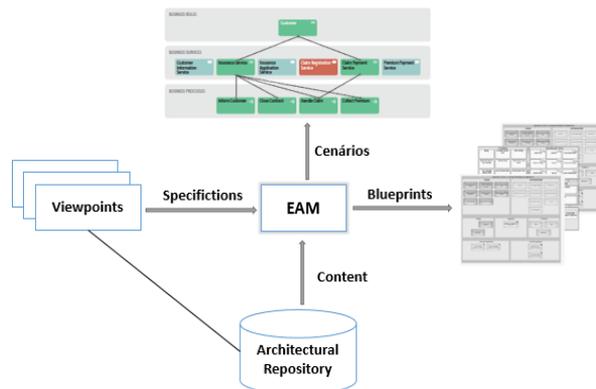answer some needs of users and stakeholders. As shown in Figure 3 below.

Figure 3 - Using queries in an EAM.

These tools let you specify automatically viewpoints, but can only specify one who has the metamodel domain or one that has the metamodel side (printed or digital). And once specified the viewpoints you need to create queries for seeking or obtaining the data to be displayed in the views. And these consultations are done manually by architects or other users who have the knowledge or domain architectural repository (existing artifacts and the relationships between them) and the query language concerned that these tools do not allow the user to create automated queries. And in many lines of business, people often need to make quick decisions based on data stored in the architectural repository. And yet the majority are not database experts and often have difficulties using the query language to express their needs. And even for those users who already have a certain domain or knowledge of query language and architectural repository (metamodel), there are situations where the viewpoints specification and creating queries become even more difficult and complicated when it involves many artifacts requiring them much concentration and effort. And the likelihood of making mistakes are many, and the other side, it takes them too long to respond to a specific need

or request of the stakeholders of the system or the user.

1.3 Objective

In order to help business users to specify the viewpoints and create their own data queries in order to make quick decisions in their daily tasks. And help users already with some experience to create complex queries easily. This thesis aims to develop a software **"Q-EAMS"**, based on the Architectures Corporate management tool (EAMS) that will allow the user to navigate the architectural repository to specify the viewpoints you want to act or view the repository using the algorithm **"AndarPor"**. And it will allow you to create automated queries on the artifacts of viewpoints specified by "AndarPor", making the creation process easier and simple queries to be done without manual.

1.5 Organization of the Dissertation

Our dissertation summary has five chapters. The first chapter is this introduction, which introduces the theme of this work and the others chapters are listed below:

Chapter 2. Solution Proposal. In this chapter, we begin to present our proposed solution the "Q-EAMS".

Chapter 3. Implementation of the proposal. This chapter aims to show and explain step by step how we got to implement our proposed solution.

Chapter 4. Evaluation. This chapter aims to show and explain how we evaluate or how it was validate our solution.

Chapter 5. Conclusions. The last chapter of this thesis presents the main contributions of the approach as well as its limitations and suggests some directions for future work.

2 Proposal

2.1 Solution Architecture

Our objective proposal make the process of creating queries, easy and simple to be done for both experienced users as for the less experienced users, added a module (Q-EAMS) to EAMS [16] that will allow them to specify the viewpoints and create queries automatically, without little effort as possible, based on the query language ERML [16] and metamodel EAMS. The figure 4 shows our architectural solution.
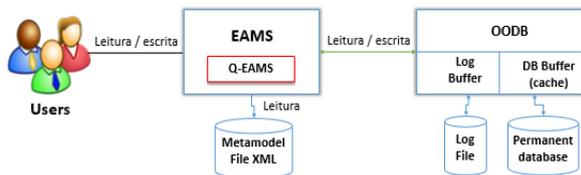

Figure 4 **-** Architectural solution

2.1.2 Q-EAMS

"Q-EAMS" is a software that lets you browse the architectural repository to specify the viewpoints you want to view in the repository using the **algorithm "AndarPor"**. And allows you to create automated queries on the artifacts of viewpoints specified by "AndarPor", making the creation process easier and simple queries to be done without manual modeling effort.

Algorithm "AndarPor".

"Andar Por" "AndarPor" is an exhaustive search algorithm that allows you to navigate the architectural repository so the specific viewpoints (identify paths) of a particular area of the repository that it intends to operate or view.

Once defined artifacts, started to show all artifacts that are related directly and inversely to the original artifact using adjacent list similar to the graphs. The distance that can exist between the

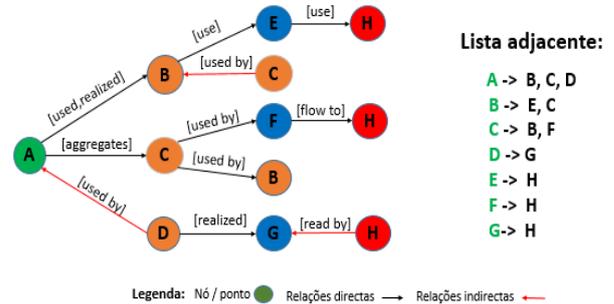relative related articles and final artifact is also shown.

Figure 5 - Representation of algorithm Andar Por

Looking at Figure 7 the initial artifact is A and the ending is H. The first list contains all the artifacts that are directly and inversely related to the initial artifact and the distance of this artifact in the final artifact {B-1,C-1,D-1}. Then, the user will choose the list {B-1,C-1,D-1}. one of the articles, for example B, C or D and will select the property and the type of relationship. Once you have chosen one of the artifacts will again be shown a list with all directly related artifacts and indirectly to the chosen artifact, in this case will be the list {E-0, C-0}, {F-0,C-B}. Or {B-0}. Thus successively until the user finds the final artifact.

We consider define or the distance as the number of possible artifacts that may exist to leave an artifact other. And the maximum distance that can exist between two artifacts are two, as shown below.
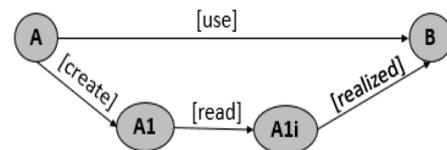

Figure 6 - Maximum distance between two artifacts

3. Implementation

This section describes in detail the implementation of a functional prototype that supports the creation

of automated queries easily and interactively previously described.

The general idea of the prototype is shown in Figure 7. The software solution begins by loading all artifacts (datatypes) of our architectural repository (metamodel) XML. This information is processed in order to identify the relationships existing between the articles in the Enterprise Architecture. The user then identify which architectural view you want to analyze in the context of existing relations between the articles. Starting with select or set two artifacts (a start and end datatypes) and intermediate artifacts (artifacts needed to reach the final artifact). As user is defining or choosing, intermediate artifacts must also select the property and the relationship.

Consultations will be made in each set artifact, using the specifications of ERML query language [8]. After consultations created, the software solution will generate an XML file that will contain all queries.
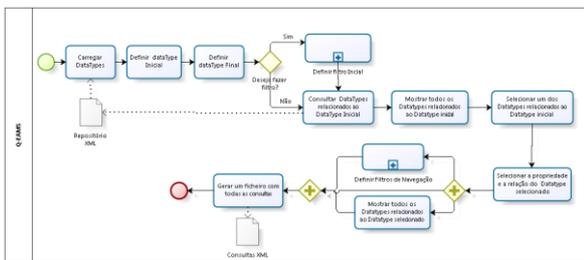

Figure 7 - general concept of the implemented solution

The Q-EMAS is an application that lets you browse the architectural repository to specify or define the area or the view that will put up with the repository. And create automated queries on the specified view. Queries are created based on specifications or syntax and semantics of ERML query language [8]. and consults and that need data to make decisions in their daily tasks.
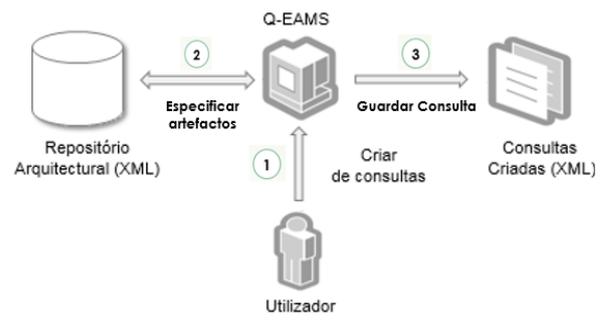
Figure 8 - Features of Q-EAMES

The Q-EAMS, has three stages and particularly the interaction approaches location, manipulation and display, according to the query formulation structure.

3.1 Implementation of the *"AndarPor"*

This step is the path ID that allows the user to specify the region (path), or regions of interest database that he wishes to operate. This phase is achieved by our algorithm "AndarPor" described in the proposal, which seeks to link all existing paths from the two selected points; such paths are shown to the user to decide whether to include them in their scheme of interest or not.

Figure 9 - Sequence Diagram path creation

Integration Repository

When information is requested to EAMS the *Load ()* method is invoked to read the relevant information from the XML file and returns it to the EAMS that deliver to the recipient. The reading of the data in the XML file is done sequentially, in

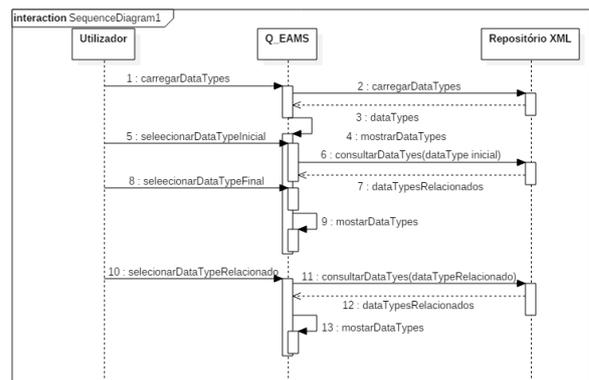each request for access to the file information is read from the start thereof until the information is found. Even if the requested information is not in the XML repository, a request involves the reading of all of that file.

Manipulation

At this stage, the user will build or develop appropriate consultations for each view of the artifacts defined in the seethe. The formulation of queries must obey the syntactic and semantic rules ERML query language [16]. At this stage, the user may or may not also apply filters (initial, navigation and final) as seen in the proposal in artifacts.
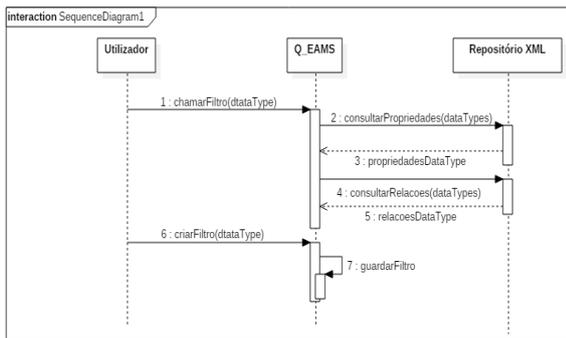

Figure 10 - Sequence diagram creation paths

At this stage, the Q-EAMES does not interact with the repository, but rather reads the memory artifacts created at the stage of location. The methods used at this stage are described in the table below.

Methods	Purpose
createQuery	Create a query in a particular artifact
addItemLists	Adds a query created in the list

Table 1- Methods of creating queries

Preview

At this stage, the Q-EAMES calculates the query result and represents. The query result is displayed in a file or XML file.
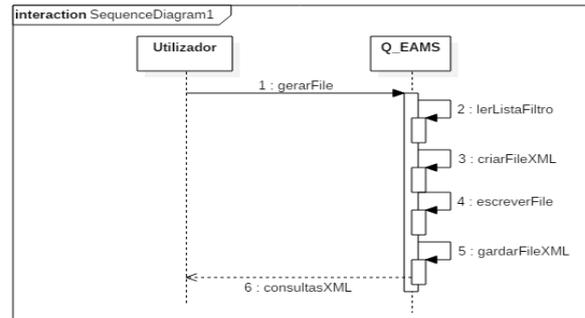
Figure 11 - generation sequence diagram of the Query File

At this stage, the Q-EAMES also does not interact with the repository, but rather reads the memory artifacts list created at the stage of location.

Methods	Purpose
generateQuery	Generate a file with all created filters

Table 2 - Methods of generating XML queries

3.2. User Interface

The user interface designed for identification or definition of paths and creating queries is based on the specification described Viewpoint, according to ANSI / IEEE Std 1471-2000.

Upload file

The user can then make the loading of articles you want to scan using the button labeled "Load". After identifying the location of the file, the software solution will make loading even in memory and the user interface will be updated starting the comboBox subject to the articles. Once initialized the comboBoxs the user can now create their paths starting by defining the start and end point.

Figure 12 - Definition of points (initial and final)

Definition of Path

Once the user has set the start and end point will pop up a gridview showing all artifacts that are directly and inversely related to the initial artifact. In this case is the "Application Collaboration". Also shown is a number in front of the article which means the same distance with the final artifact.

Figure 13 – Gridview for path definition

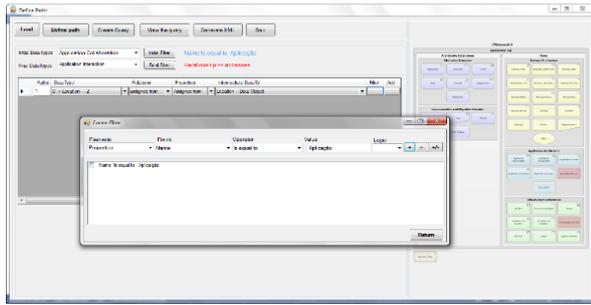Then the user will select the comboBox "dataType" the artifact he needs to reach the end point. After selecting the artifact, the user must also select the relationship and the property. It will also be shown a list of all possible paths through the selected artifact.

Figure 14 – Gridview for path definition.

If the distance of the selected artifact is zero, then the user end here, and may use or not a filter on the article. Otherwise, you should add another artifact to reach the expected artifact with a distance equal to zero. For this, click the "Add" button that will get all the artifacts that are related directly and inversely to the selected artifact. And the user must also select the relationship and the property. And apply or no filter.

Figure 15 – Gridview for path definition

And the user must also select the relationship and the property. And apply or no filter.

Figura 16 – Gridview para definição de caminho

Filter Creation

You can make or apply a filter in the initial articles, intermediate and / or end. A form will be / is opened to create a filter. A filter may have a elements, a field, an operator and a value. The value and the operator are not mandatory in some cases because not all filter elements such relations do not need or value operators.

You can create a more complex filter using logics operators (AND, OR, UNION, AS). As well as add, change or remove one or more operations of a filter using the commands (+, -, +/-). It was created three lists one for storing the filters created in the original artifact, another to store the filter through artifacts and other save for the final artifact filters.

Figure 17 - Filter creation screen.

4 Analysis of Results Achieved

This section identify the methodology used to evaluate the approach and test the prototype implemented and the usefulness of its results.

4.1 Performance evaluation of the algorithm "AndarPor"

What about performance analysis, "AndarPor" algorithm requires a time O (n) at startup. The loop (cycle) is performed n times 3. In the first iteration, they are visited all the artifacts that are related directly and inversely to the selected artifact. If the final artifact is not on this list, then the second interaction is to visit all related directly or inversely artifact every artifact of the first interaction is performed. If the final artifact is not on this list then the third interaction, also comfortable visit all directly related artifacts or conversely every artifact of the second interaction is performed, and so on.

We can then deduce that the running time is O (n), for the best case, when the final device is in the first iteration; It is O (n²) and for the second iteration is O (n³) for the third interaction our worst case. In short, the complexity of our algorithm and the execution time increases as the distance between two point increases.

4.2. Usability Evaluation

The central objective of this work is to provide user interaction data queries environments based on

metamodel, it is important to get feedback from users on the use of the features available in the system. To evaluate the usability of the features 10 users were asked to carry out some experiments with the use of Q-EAMS interface. The guest users are divided into two groups as follows: the first group consists of six volunteers, they have knowledge about metamodel; and the second group is composed of four elements and they do not have a knowledge of metamodel.

The collection of opinions and suggestions from users aims to gain an appreciation of the same on the system. Normally, you want to identify the level of user satisfaction with the system, which includes aspects such as: the aesthetic appearance of the system is satisfactory, if the system does what they want, if they had any problems while using it, and / or if they would like to (or plan to) use it again. The technique used in our thesis to collect opinions and suggestions from users was the use of a questionnaire.

Charts shown in the Figures 18 and 19 show the results of these experiments to expert users, not experts, respectively. The graphs tell the percentage of responses (Bad, Good and Excellent) for each of the questions, where each of these is related to a system's functionality.

Looking at the charts, we see that the majority of users, experts and non-experts, the questionnaire indicating that they were satisfied with the features available in the system. And most of the questions responses were obtained between Good and Excellent, confirming that the interface provides the user with a simple and objective interaction with this type of system.
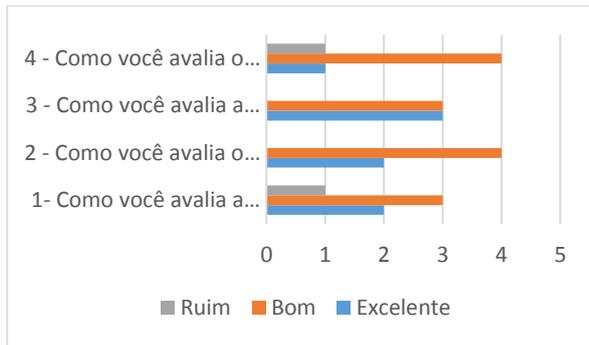
Figure 18 - Chart with percentage evaluation of the features users by experts

Figure 19 - Chart with percentage assessment of functionality for users not experts

5 Conclusion

5.1 Summary

With this thesis, it was found that, with the aid of visualization techniques can improve understanding of the systems that are or have been developed.

5.2 Contributions

The scientific contributions that this work arise from the need to help mainly the non-specialist business users, in those who do not dominate the architectural repository and query language, creating an application "Q-EAMS" that will allow them will able to express or create your own queries or not, to take quick decisions of their daily tasks. We developed also an exhaustive search algorithm "AndarPor" that lets you browse the architectural repository in order to specify viewpoints (identify paths) of a particular area of the repository that it intends to operate or view.

5.3 Solution Limitations

- **Consultation accuracy**: Our solution not correctly returned data from some complex queries created in the test phase;

- **User Satisfaction**: subjective evaluations user satisfaction with the tool was not great but rather good.

- **Algorithm "andar por" Performance**, is very efficient in the ways of detection at level three (for articles of greater than or equal to 3)

5.4 Lessons Learned

Throughout the development of this work have been identified various topics and issues associated with the problem and the solution developed on which it is worth pondering.

- Approach is dependent on the quality of information - As indicated in the initial phase of work, the company is perceived by the information maintained by the repository. The quality of information is determined by the quality and accuracy of the data that support If the repository does not reflect reality, consultations will be useless.

5.5 Future Work

Performing any work always leaves development opportunities in the future; in particular, in the future it is intended:

- Improve the performance of our algorithm *ANDAR POR* in identifying possible ways given two articles and an initial or final.

- Allow our Q-EAMS solution not only can create queries based on a metamodel but also based on other metamodel.

- Improve in order to be able to interface any user to create their own component in the application.

- Extend the views, adding new views and new features the tool.

References

[1] Porter, M.E. (2008). The Five Competitive Forces That Shape Strategy. Harvard Business Review, 1, 79–93.

[2] Federation of EA Professional Organizations, Common Perspectives on Enterprise Architecture, Architecture and Governance Magazine, 1, 9-4.

[3] Lapalme, J.(2012). Three Schools of Thought on Enterprise Architecture, IT Professional, 14, 37–43. doi:10.1109/MITP.2011.109

[4] Enrique Lobo Cruz. [Online]. ENTERPRISE ARCHITECTURE MANAGEMENT. https://uk.boc-group.com/consulting/enterprise-architecture-management, acedido a 10 de Fevereiro de 2016.

[5] Ahlemann. F. (2012). Strategic Enterprise Architecture Management: Challenges, Best Practices, and Future Developments, Springer-Verlag Berlin Heidelberg.

[6] Trustradius. P. (2016). Enterprise Architecture Management Tools. http://trustradius.com/architecture-management, acedido a 10 de Março de 2016.

[7] Lankhorst, M., et al., *"Enterprise Architecture at Work: Modeling, Communication and Analysis"*, Springer, Germany (2005).

[8] Link Consulting. [Online]. Enterprise Architecture Manager Software. http://www.link.pt/Conteudos/Artigos/detalhe_artigo.aspx?idc=661&idsc=1652&idl= 1, acedido a 16 de Dezembro de 2015.

[9] Vitech. [Online]. Enterprise Architecture Software.http://www.vitechcorp.com/solutions/Enterprise-Architecture-Software.shtml, acedido a 16 de Dezembro de 2015.

[10] Vizrt. [Online]. MAM Enterprise Solution. http://www.vizrt.com/solutions/mam-solution/, acedido a 15 de Abril de 2016.

[11] Software AG. [Online]. ARIS Enterprise Architecture.http://www.softwareag.com/corporate/products/aris_alfabet/ea/products/architecture/capabilities/default.asp, acedido a 15 de Abril de 2016.

[12] Enterprise Architecture Management System EAMS Client Application Manual, by Link

[13] Tiziana Catarci, Giuseppe Santucci, Michele Angelaccio. Fundamental Graphical Primitives for Visual Query Languages1 (Accepted for publication on Information Systems Vol. 18 No. 2)

[14] T. Catarci, et.al. *Visual Query Systems for Databases: Analysis and Comparison* – Journal of Visual

[15] M. Ahmed and A. Lubiw, "Shortest Paths Avoiding Forbidden Subpaths", Proceedings of the 26th International Symposium on Theoretical Aspects of Computer Science, Freiburg, Germany, (2009) February 26-28.

[16] IEEE Computer Society, *"IEEE Std. 1471-2000: IEEE Recommended Practice for Architecture Description of Software-Intensive Systems",* IEEE, USA (2000).