

Technique of Adding A New Functionality in A Package with An Approach Based on Formal Concept Analysis.

Lala Madiha Hakik

*Faculty of Science and Techniques,
University Hassan I, BP 577. Settat, Morocco,*

Rachid El Harti

*Faculty of Science and Techniques,
University Hassan I, BP 577. Settat, Morocco,*

Abstract

Software development is studying, Design, build transforming, develop, maintain and improve the software.

In this paper we are looking for an improvement reconstruction remodularized architecture by applying a technique of adding a new functionality with business classes and utilities classes in a package with an Approach Based on Formal Concept Analysis.

Keyword: Reconstruction Remodularized Architecture, Formal Concept Analysis.

1. Introduction

A software project is to group in packages, folders, business classes and utilities classes corresponding to the entities from directly functional needs. The attributes of these classes contain the specific data to be processed by the system. The methods correspond to different features associated with the entity represented by the class¹.

Use a business architecture in a software project responds to a need of factorization and differentiation of the Code, providing an additional level of abstraction¹.

Our principle design based on the idea of adding business classes and utilities classes in a software architecture by using a technique based on a method of formal concept analysis FCA providing expert software means for performing an addition of class in a package.

We illustrate our proposal with a theoretical example (Figure1) [11][12].

Section 2 presents our example, then we describe the approach in Section 3. Related work is presented in Section 4, and then we conclude in Section 5.

2. Illustration

In this section we suggest contexts of use for the addition of new classes in a software (eg. new functionality with business classes and utilities classes) and it is desired to store classes somewhere in existing packages in a remodularized architecture.

Our choice was focused on a remodularised architecture 1 (Figure 1) extracted from papers [11][12], consisting of 4 packages [A, B, C, D] connected between them without being shown by simplicity.

These packages are expected to contain more classes that are not shown also for reasons of simplicity.

This architecture includes classes with relationships:

- Internal-between classes in the same package.
- External-between classes in different packages.

We believe that the Formal Concept Analysis (FCA) can bring interesting ways to solve the problem of introduction business classes and utilities classes in one of the packages of architecture to rebuild.

This technical method provides one solution to an expert.

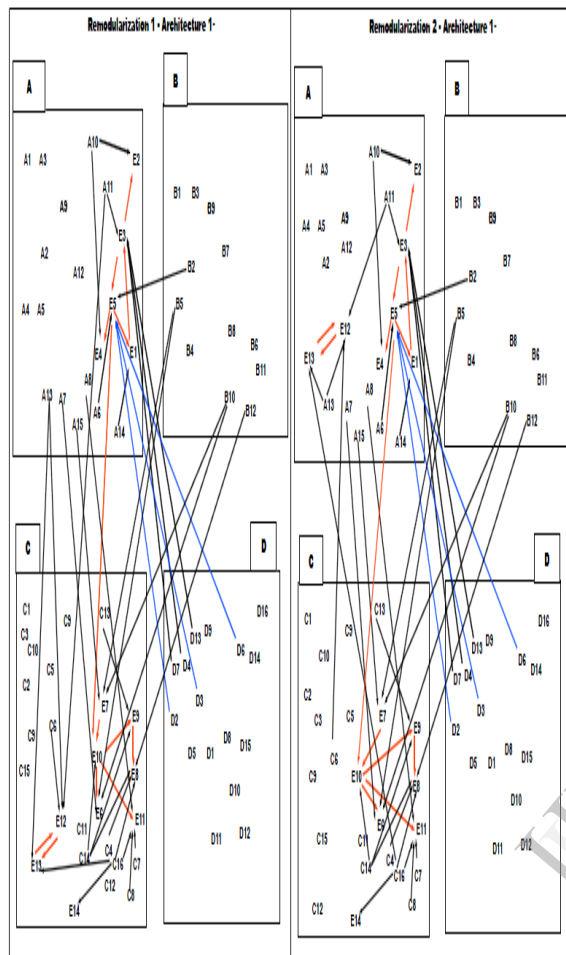


Figure 1. Remodularization 1 and 2
-Architecture 1-.

3. Proposed approach

The Formal Concept Analysis (FCA) [2] [4] [11] [12] is a technical data analysis that allows you to group entities with common characteristics. A concept is a maximal set of entities (extension of the concept) sharing a maximal set of characteristics (intension of the concept). The AFC is used in software engineering for solving several problems [9].

Configurations In the context of our problem, we studied three different configurations with FCA. We present all of them.

The configuration with FCA is to define a formal context C: the set O of entities studied (or formal objects). Set A of characteristics (or formal attributes) and the relationship

$$R \subseteq O \times A.$$

Our approach is introducing two new business classes CM1, CM2 and two utilities classes CUT1, CUT 2.

To facilitate the addition of these classes, we opted for the formal contexts C1, C2 and C2 cited below:

-The first formal context (figure 2):

Context (formal context C1).

- O1 is the set of classes of packages in connection with the business classes
- A1 business classes and utilities classes to introduce.
- R1 is the relation "Target is access from the class with a relationship of dependency."
- $(CM, x) \in R1$ if there is an arrow x to CM or
- $(CUT, x) \in R1$ if there is an arrow x to CUT.

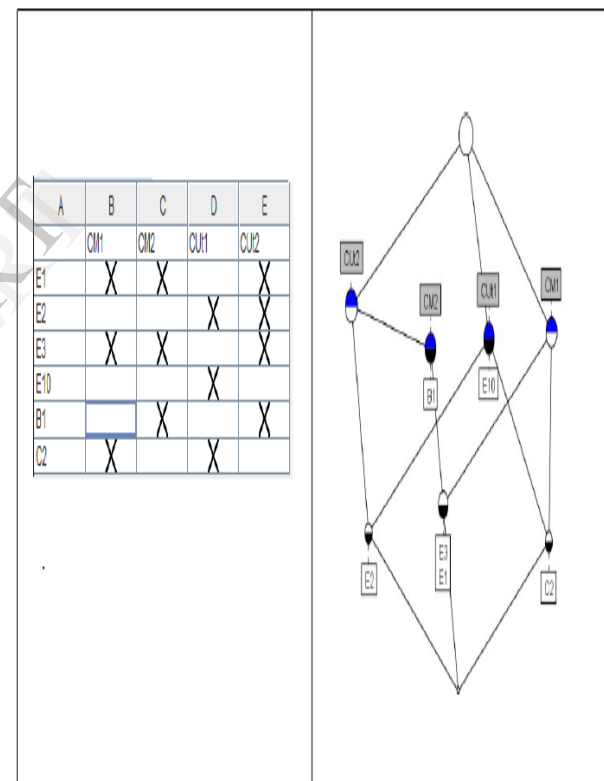


Figure 2. Formal context 1 and lattice T(C1)
-Remodularization 1-

The second formal context (figure 3):

Context (formal context C2).

- O2 business classes and utilities classes to introduce.
- A2 is the set of classes of packages in connection with the business classes.
- R2 is the relation "Target is access from the class with a relationship of dependency."
- $(CM, x) \in R2$ if there is an arrow x to CM or
- $(CUT, x) \in R1$ if there is an arrow x to CUT.

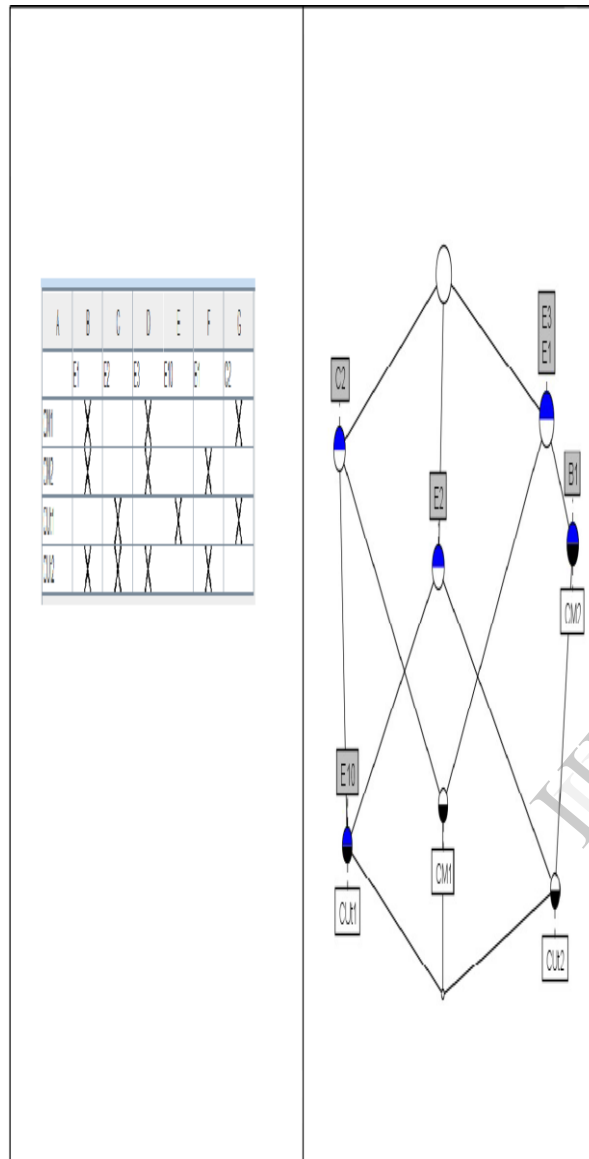


Figure 3. Formal context 2 and lattice T(C2)
- Remodularization 1 -

The third formal context (figure 4):

Context (formal context C3).

- O3 is the set of classes of packages of remodularization 1.
- A3 packages of remodularization 1.
- R3 is the relation "Packages of remodularization 1 with their classes."

Note that for a concern of simplicity and given the size table of a context C3, it was limited only to the presentation of an overview to communicate information about our idea.

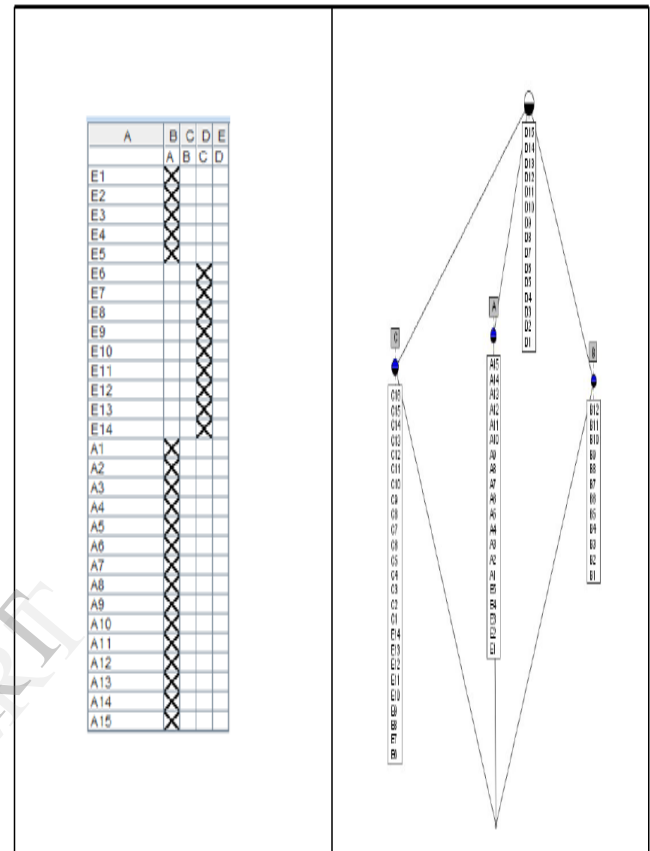


Figure 4. Formal context 3 and lattice T(C3)
- Remodularization 1 -

Example of exploration The exploration is to navigate the three lattices T(C1), T(C2) and T(C3) to identify opportunities in order to add classes. We partially detail an example of analysis to explain the principle.

Ordinary case: From Concepts $\{\{E1, E3\}, \{CM1\}\}$, $\{\{C2\}, \{CM1\}\}$ of the lattice T(C1), our methodology is searching to have a confirmation of the assignment of the business class CM1 in the lattice T(C2) to know exactly the final destination.

CM1 is related to E1, E3 and C2 are respectively the classes in packages A, A and C in the lattice T(C3) will be added in the dominant package A. It is the same for the business class CM2 and utility class CUT1 assigned in order to dominant package A for CM2 and dominant package C for CUT1.

Special case: According to the lattice T (C1), utility class CUT2 is related to classes E2, B1 respectively located in Packages A and B.

To know the destination of the addition of this class we will check its presence in the lattice T (C2) showing its connection, by inheritance, with classes E1, E2, E3 and B1 designed by the lattice T (C3) respectively to packages A, A, A and B where the addition of the class CUT2 to the dominant package A.

Figure 5 materializes adding of these classes.

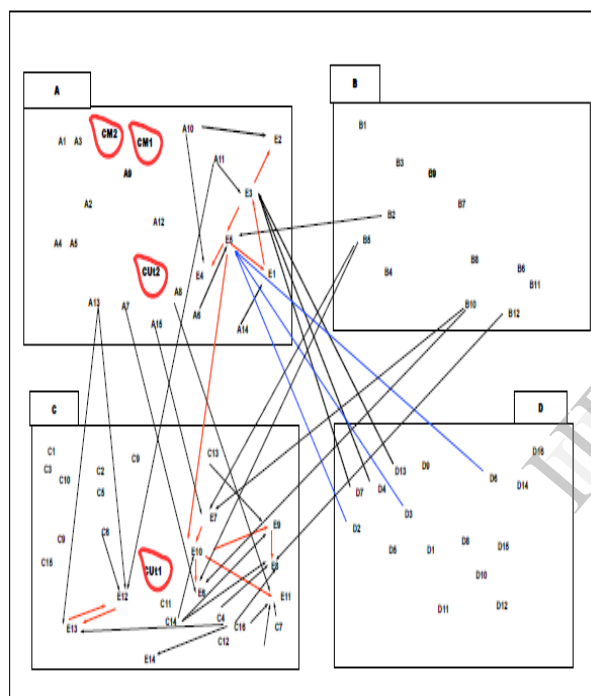


Figure 5. Remodularization 1 with added business and utilities classes.

4. Related Work

Different automated approaches have been proposed to restructure object systems. We cite three: the clustering algorithms, algorithms based on meta-heuristics and those based on the FCA. The first aim to restructure system by the distribution of some elements (e.g. classes, methods, attributes) in groups such that the elements of a group are more similar to each other with elements of other groups [1] [5] [3]. Approaches to restructuring based on meta-heuristic algorithms [7] [6] are generally iterative stochastic algorithms, progressing towards a global optimum of a function by

evaluating a certain objective function (e.g. characteristics or quality metrics). Finally, the approaches based on FCA [8] [10] provide an algebraic derivation of hierarchies of abstractions from all entities of a system. Reference [4] presents a general approach for the application of the FCA in the field of object-oriented software reengineering. Recently, we added the dimension of exploration using the FCA [11] [12]. In our approach, we have extended our research to introduce a technique of adding a new functionality in a package with FCA.

6. Conclusion and discussion

In this article, we present and illustrate a theoretical case and propose a technique of adding a new functionality in a package with an approach based on formal concept analysis.

Our approach has generated a unique solution for the realization the addition of business classes and utilities classes in an architecture consisting of classes and packages.

This method will be evaluated by metric validation on real cases.

7. References

- [1] F.B. Abreu, G. Pereira, and P. Sousa. A coupling-guided cluster analysis approach to reengineer the modularity of object-oriented systems. In Proceeding of the conference on Software Maintenance and Reengineering. CSMR '00, pages 13-, Washington, DC, USA, 2000. IEEE Computer Society Press.
- [2] G. Arévalo, S. Ducass, and O. Nierstras. Lessons learned in applying formal concept analysis to reverse engineering. In Proceeding of the Third international conference on Formal Concept Analysis, ICFA'05, pages 95-112, Berlin. Heidelberg, 2005. Spinge-Velag.
- [3] M. Bauer and M. Trifu. Architecture-aware adaptive clustering of oo s systems. In Pceedings of the Eighth Euromicro Working Conference on Software Maintenance and Reengineering (CSMR '04), CSMR '04, pages 3-, Washington, DC, USA, 2004. IEEE Computer Society.
- [4] B. Ganter and R. Wille. Formal Concept Analysis. Mathematical Foundations. Spinge. 1999.
- [5] B.S. Mitchell and S. Mancoridis. Compaing the decompositions produced by software clustering algorithms using similarity measurements. In ICSM, pages 744-753, 2001.

- [6] M.O'Keeffe and M. i Cinneide. Search-based refactoring for software maintenance. J. Syst. Softw., 81(4): 502-216, April 2008.
- [7] O. Seng, J. Stammel and D. Burkhart. Search- based determination of refactorings for improving the class structure of object-oriented systems, In Mike Cattolico, edito. GECCO, pages 1909-1916. ACM, 2006.
- [8] G.Snelting. Software reengineering based on concept lattices. In CSMR, pages 3-10, 2000.
- [9] T. Tilley, R. Cole, P. Becker, P.W. Eklund. A survey of formal concept analysis support for software engineering activities. In Int. Conf. Formal Concept Analysis (ICFCA 2005), pages 250-271, 2005.
- [10] P. Tonella. Concept analysis for module restructuring. IEEE Trans. Software Eng., 27 (4): 351-363, 2001.
- [11] Lala Madiha Hakik, Marianne Huchard, Rachid El Harti et Abdelhak Djamel Seriai. Exploration de la redistribution des classes d'un package par des techniques d'Analyse Formelle de Concepts. The first conference in software engineering (CIEL 2012), France, 2012.
- [12] Lala Madiha Hakik, Rachid El Harti . " *Exploring the Redistribution Classes of a Package with an Approach Based on Formal Concept Analysis* ", Vol.2 - Issue 12 (December - 2013), International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, www.ijert.org