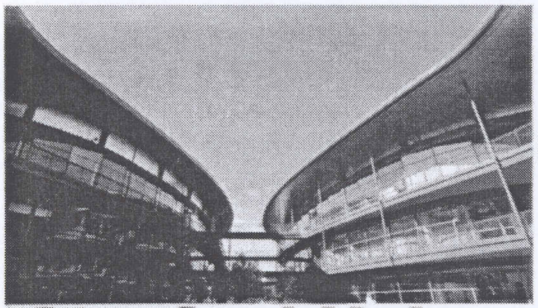
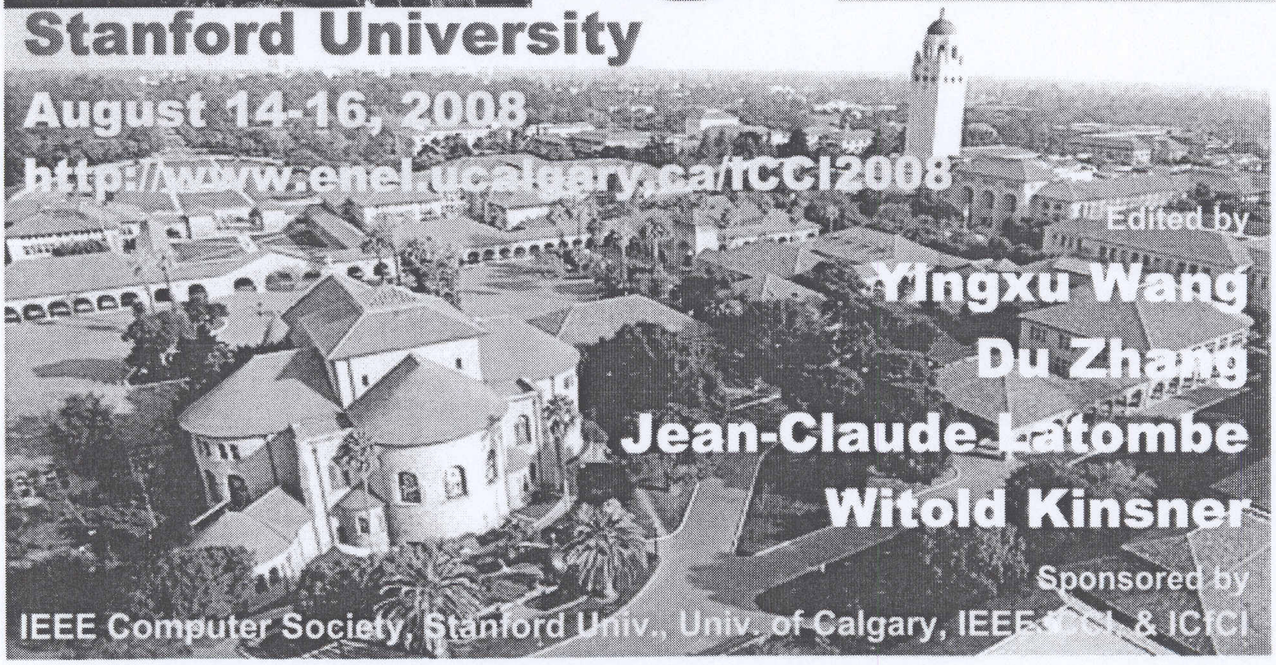
7th IEEE International Conference on Cognitive Informatics (ICCI 2008)



ICCI'08

Stanford University



August 14-16, 2008

http://www.enel.ucalgary.ca/ICCI2008

Edited by

Yingxu Wang

Du Zhang

Jean-Claude Latombe

Witold Kinsner

Sponsored by

IEEE Computer Society, Stanford Univ., Univ. of Calgary, IEEE SCI, & ICfCI

Proceedings of the Seventh IEEE International Conference
on

Cognitive Informatics

ICCI 2008

August 14 – 16, 2008, Stanford University, California, USA

Edited by

Yingxu Wang, Du Zhang, Jean-Claude Latombe, and Witold Kinsner

Supported by

IEEE COMPUTER SOCIETY
IEEE

UNIVERSITY OF CALGARY

IEEE ICCI
IJCINI

IEEE Canada

IEEE COMPUTER SOCIETY

IEEE

Los Alamitos, California
Washington • Brussels • Tokyo

IEEE Computer Society Catalog Number **CFP08312-PRT**
ISBN **978-1-4244-2538-9**
Library of Congress **2008904056**

Additional copies may be ordered from:

IEEE Computer Society
Customer Service Center
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1314
Tel: + 1 800 272 6657
Fax: + 1 714 821 4641
http://computer.org/cspress
csbooks@computer.org

IEEE Service Center
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
Tel: + 1 732 981 0060
Fax: + 1 732 981 9667
http://shop.ieee.org/store/
customer-service@ieee.org

IEEE Computer Society
Asia/Pacific Office
Watanabe Bldg., 1-4-2
Minami-Aoyama
Minato-ku, Tokyo 107-0062
JAPAN
Tel: + 81 3 3408 3118
Fax: + 81 3 3408 3553
tokyo.ofc@computer.org

Individual paper REPRINTS may be ordered at: reprints@computer.org

Editorial production by *Yingxu Wang* and *Du Zhang*

Cover art production by *Yingxu Wang*

Printed in USA

IEEE COMPUTER SOCIETY

IEEE

Table of Contents

ICCI 2008

Session A4: Computational Intelligence

Session B4: Symbiotic Computing (Invited Session)

Session A5: Visual Information Cognition

Preface

Welcome to the *7th IEEE International Conference on Cognitive Informatics* (ICCI 2008)!

Cognitive Informatics (CI) is a cutting-edge and multidisciplinary research area that tackles the fundamental problems shared by modern informatics, computation, software engineering, AI, cybernetics, cognitive science, neuro-psychology, medical science, philosophy, linguistics, life sciences, and many others. CI is the trans-disciplinary study of cognitive and information sciences, which investigates the internal information processing mechanisms and processes of the natural intelligence – human brains and minds – and their engineering applications in computational intelligence.

The development and the cross fertilization among the aforementioned science and engineering disciplines have led to a whole range of extremely interesting new research areas. Following the first six successful conferences on Cognitive Informatics, ICCI'02 (Calgary, Canada), ICCI'03 (London, UK), ICCI'04 (Victoria, Canada), ICCI'05 (Irvine, USA), ICCI'06 (Beijing, China), and ICCI'07 (Lake Tahoe, USA), ICCI'08 focuses on the theme of *Cognitive Computers and Computational Intelligence*. The objectives of ICCI'08 are to draw attention of researchers, practitioners, and graduate students to the investigation of cognitive mechanisms and processes of human information processing, and to stimulate the international effort on cognitive informatics research and engineering applications. The ICCI'08 technical program encompasses 56 regular papers selected from 115 submissions from all over the world based on rigorous reviews by program committee members and external reviewers. The program is enriched by 4 keynotes from prestigious scientists.

The growing field of CI covers many areas as follows in natural intelligence, neural informatics, cognitive computing, computational intelligence, and their engineering applications:

*** Natural Intelligence ***

- Informatics models of the brain
- Cognitive processes of the brain
- Internal information processing mechanisms
- Theories of natural intelligence
- Intelligent foundations of computing
- Denotational mathematics for CI
- Abstraction and means
- Ergonomics
- Informatics laws of software
- Knowledge representation
- Models of knowledge and skills
- Language acquisition
- Cognitive complexity of software
- Distributed intelligence
- Computational intelligence
- Emotions/motivations/attitudes
- Perception and consciousness
- Hybrid (AI/NI) intelligence

*** Computational Intelligence ***

- Imperative vs. autonomous computing
- Reasoning and inferences
- Cognitive informatics foundations
- Robotics
- Informatics foundations of software eng.
- Fuzzy/rough sets/logic
- Knowledge engineering
- Pattern and signal recognitions
- Autonomic agent technologies
- Memory models
- Software agent systems
- Decision theories
- Problem solving theories
- Machine learning systems
- Distributed objects/granules
- Web contents cognition
- Nature of software
- Cognitive computers

*** Neural Informatics ***

- Neuroscience foundations of information processing
- Cognitive models of the brain
- Functional modes of the brain
- Neural models of memory
- Neural networks
- Neural computation
- Cognitive linguistics
- Neuropsychology
- Bioinformatics
- Biosignal processing
- Cognitive signal processing
- Gene analysis and expression
- Cognitive metrics
- Neural signal interpretation
- Visual information representation
- Visual information interpretation
- Sensational cognitive processes
- Human factors in systems

The ICCI'08 program as presented in the proceedings is the result of the great effort and contributions of many people. We would like to thank all authors who submitted interesting papers to ICCI'08. We acknowledge the professional work of the Program Committee, special session organizers, and external reviewers for their effective review and improvement of the quality of submitted papers. Our acknowledgement also goes to the invaluable sponsorships of IEEE Computer Society, The IEEE ICCI Steering Committee, IEEE Canada, and IEEE CS Press, Stanford University, as well as International Journal of Cognitive Informatics and Natural Intelligence (IJCINI) and International Journal of Software Science and Computational Intelligence (IJSSCI). We thank the keynote speakers for presenting their visions and insights on fostering this transdisciplinary area. We acknowledge the organizing committee members, particularly Liliana Rivera, Lisa Doyer, and all student volunteers who have helped to make the event a success.

Jean-Claude Latombe, Yingxu Wang, Witold Kinsner, and Du Zhang

Conference Organization

Conference Co-Chairs

Jean-Claude Latombe, Yingxu Wang, and **Witold Kinsner**

Program Committee Co-Chairs

Yingxu Wang and **Du Zhang**

Organization Co-Chairs

Du Zhang, Liliana Rivera, and **Lisa Doyer**

Program Committee

Anderson, James	USA	Pelayo, F. Lopez	Spain
Barbier, Franck	France	Rajlich, Vaclav	USA
Baciu, George	Hong Kong	Reyes, Spindola J.	USA
Bickle, John	USA	Rubio, Fernando	Spain
Bland, Brian H.	Canada	Ruhe, Guenther	Canada
Chan, Christine	Canada	Sheu, Phil	USA
Chan, Keith	Hong Kong	Shi, Zhongzhi	China
Dawson, Michael R.W.	Canada	Skowron, Andrzej	Poland
Dromey, Geoff	Australia	Sugawara, Kenji	Japan
Fan, Lisa	Canada	Tsai, Jeffrey	USA
Garbajosa, Juan	Spain	Tukiainen, Markku	Finland
Guan, Ling	Canada	Valdes, Julio J.	Canada
Guo, Aike	China	Wang, Deliang	USA
He, Matthew	USA	Wang, Guoyin	China
Henderson-Sellers, Brian	Australia	Wang, Yingxu	Canada
Hou, Zeng-Guang	China	Xu, Zongben	China
Hu, Mou	Canada	Xiang, Yang	Canada
Huang, Bo	Hong Kong	Yang, Fangchun	China
Jin, Yaochu	German	Yang, Qiang	Hong Kong
Kinsner, Witold	Canada	Yao, Yiyu	Canada
Latombe, Jean-Claude	USA	Zhang, Du	USA
Liu, Jiming	Hong Kong	Zhang, Kaizhong	Canada
Lopez, Natalia	Spain	Zhong, Yixin	China
Patel, Dilip	UK	Zhong, Ning	Japan
Patel, Shushma	UK	Zhou, Xiaolin	China
Pedrycz, Witold	Canada	Zhu, Haibin	Canada

Organization Committee

Liliana Rivera, Lisa Doyer, Du Zhang, Yingxu Wang, and Witold Kinsner

Additional Reviewers

Jun Peng, Yousheng Tian, Mehran Mehrandezh, Qi Wang, Vicent Chiew, and Jingzhou Li

The Steering Committee of IEEE International Conference on Cognitive Informatics (IEEE ICCI)

The International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)

Editor-in-Chief: Prof. Yingxu Wang

http://www.enel.ucalgary.ca/IJCINI/

Using cognitive techniques for assessing the influence of coupling on the maintainability of OCL expressions

Luis Reynoso
National University of Comahue, Neuquén Argentina
Lreynoso@uncoma.edu.ar

Marcela Genero, Mario Piattini
University of Castilla-La Mancha, Ciudad Real, Spain
{Marcela.Genero, Mario.Piattini@uclm.es}

Esperanza Manso
University of Valladolid, Valladolid, Spain
manso@infor.uva.es

Abstract

This paper is part of a project we have been developing for the past three years, related to the maintainability of expressions written with the Object Constraint language (OCL). Our objective is to find which elements of OCL influence on the comprehensibility and modifiability of OCL expressions. Coupling is recognized as a key characteristic of software artifacts, which affects comprehensibility and maintainability. In addition, whenever an expression operates in a large context, modelers need to know all the UML artifacts the expression relies on. Therefore, we believe that coupling should also affect to the comprehension of OCL expressions and, consequently, their modifiability. We believe that two main factors of the import coupling are the number of different objects which are coupled to the contextual instance and the depth of coupling. For measuring these factors we have defined two measures (the number of navigated classes –NNC- and the depth of navigation -DN-) based on two cognitive techiniques (chunking and tracing). For testing if both coupling measures (and their interaction) could be indicators of the comprehensibility and modifiability of OCL expressions, we carried out a specific family of experiments which will be carefully described in this paper. The obtained results confirm our hypothesis, revealing that coupling really affects to the maintainability of OCL expressions.

1. Introduction

Cognitive Informatics play an important role in understanding the fundamental characteristics of software [30]. Within this interdisciplinary area, cognitive complexity measures are the human effort needed to perform a task or difficulty in comprehending the software [19]. It is recognized that comprehensibility of software artifacts is the main quality attribute that influences on maintainability [30], [12], [24]. A software artifact must be understood first before any desired changes to it can be identified, designed and implemented. But comprehension should not only be assessed at a code level but also at model level, at early stages of the development. We believe that the comprehension of OCL expressions at early stages of development is essential in order to comprehend the whole UML/OCL model to which expressions are attached [21], [22].

Coupling was empirically proved to be a key characteristic of software artefacts, which affects comprehensibility and maintainability [3] [4]. Therefore, it is advisable to minimize the coupling between software artifacts [20]. We believe that high coupling should also affects the cognitive complexity of modelers, and high cognitive complexity leads OCL expressions to exhibit undesirable external qualities, such as less comprehensibility or reduced modifiability. For example whenever an OCL expression operates in a large context of coupled objects or has many relationships, is likely to be difficult to be understood.

Within OCL there is a concept (which is at the core of the language) defining coupling, the navigation. Coupling is defined between the contextual instance and other objects to which the expression navigates to (within a model) [31] and this influences the OCL expression comprehension.

The inner definition of OCL as a textual add-on to UML models, makes that within an expression being possible to refer to UML artifacts, however it is no possible to refer from a UML model to an OCL expression. So, the locus of impact of coupling within an OCL expression is primary of import coupling, i.e.

Proc. 7th IEEE Int. Conf. on Cognitive Informatics (ICCI'08)
Y. Wang, D. Zhang, J.-C. Latombe, and W. Kinsner (Eds.)
978-1-4244-2538-9/08/$25.00 ©2008 IEEE

if coupling is defined in terms of client-supplier connections, import coupling [4] focused on the client entity of such relationship. Within this context, modelers should be aware of the fact that OCL expressions with high import coupling operate in large context and, to comprehend the meaning of the expression, modelers need to know all the UML artifacts on which OCL expressions rely on.

Two main factors of the import coupling are: the number of different objects which are coupled to the contextual instance and the depth of coupling. For measuring these factors we have defined two measures: the number of navigated classes (NNC) which represents the quantity of coupled objects and the depth of navigation (DN) which stands for the distance of the farthest coupled object from the contextual instance [25]. In addition, we based the explanation of the rationale behind the measures on the cognitive complexity model (CMM) of Cant [8].

The purpose of this article is twofold:

- The rigorous definition of the measures DN and NNC grounded on cognitive techniques of CMM model.
- The description of a specific family of experiments we carried out for testing if both measures (and their interaction) could be indicators of the comprehensibility and modifiability of OCL expressions within UML/OCL models.

The first goal is developed in section 2 whilst the second is developed in section 3, 4 and 5 presenting the experimental design, the data analysis and interpretation and the validity evaluation respectively. Section 6 describes the related work. Finally, section 7 presents the conclusions and outlines the future work.

2. Cognitive Techniques and Measures

Before presenting the proper definition of the coupling measures, DN and NNC, we will introduce the cognitive techniques that will be used for explaining the rationale behind the measures.

We hypothesized that during the comprehension of an OCL expression (considered in our study as a single mental abstraction, a chunk) the modelers apply two cognitive techniques, such as chunking and tracing. These techniques are concurrently and synergistically applied in problem solving and they are part of the Cognitive Complexity Model (CCM model) [8].

Chunking involves the recognition of a set of declarations and the extraction of information from them, which is remembered as a chunk. Tracing involves scanning, in different directions in order to identify relevant 'chunks'. The tracing technique has been observed as a fundamental activity in program comprehension [16], [2], and we believe that this also applies to UML/OCL model comprehension.

We argue that when a modeler is primarily chunking an OCL expression within a UML/OCL model, there are certain dependencies, that he/she needs to understand. For understanding such dependencies, it is necessary that the modeler performs a certain amount of tracing to find the relevant parts of UML models. Having found those parts, modelers will once again chunk them to comprehend the expression. We believe that navigation in OCL expressions guide the process of tracing to different UML artifacts within an OCL expression.

Chunking and tracing techniques are concurrently and synergistically applied, and its effect difficulty on complexity can be graphically modeled using "landscape" models [8]. These models consist of a visual representation applied during program comprehension, which we had applied for the comprehension of OCL expressions.

Example: The upper part of Figure 1 shows an UML/OCL model where an OCL expression named 'flight_capacity', has been defined in the context of the Flight class, meaning that the quantity of passengers of a flight must be lower or equal to the capacity of the plane's type of that flight. The landscape model for the OCL expression is shown in the bottom part of Figure 1. Graphically, at the top-level there is a single visible chunk, the OCL expression, delineated by the two markers (f, g). This chunk is interrupted by four lower-level chunks, the second chunk (the one starting with the mark P) is interrupted by the chunk depicted at the lowest level.

During the comprehension of the OCL expression a modeler must find the rolenames, classes and attributes mentioned in the expression (i.e., trace) and then chunk these entities before returning to the original chunk. The relatively large amount of tracing required causes a disruption in the reading of the superchunks, making them more complex [9]. While reading an upper-level chunk, a dependency requires that the modeler suspend the reading of the original OCL expression because of the need to undertake tracing so as to have a complete understanding of the chunk currently being analyzed.

Graphically, the top-level chunk is interrupted by four lower-level chunks. The first interruption is common to every OCL expression in order to locate the context of the expression (the UML Classifier –a class, interface, etc.- written after the context keyword) within the UML diagram. The second interruption, depicted as the 'vertical drop' x1P visually represents the work required in tracing the relevant features in the

UML diagram. In this case, it implies following a navigation from the Flight class to another class where its opposite-end rolename is defined as 'plane'. Having found this class, the modeler must chunk not only the class but also the cardinality associated to the rolename. Then, the modeler should follow a new navigation from Plane to Type_of_plane by using the 'planetype' rolename, and after chunking the meaning of the latter class, he/she needs to chunk one of its attributes, that is, 'capacity'. The fourth and last interruption during the comprehension of the flight_capacity' OCL expression is during the navigation from Flight to the Passenger class (drop x3P), so as to obtain the size of the set of passengers.

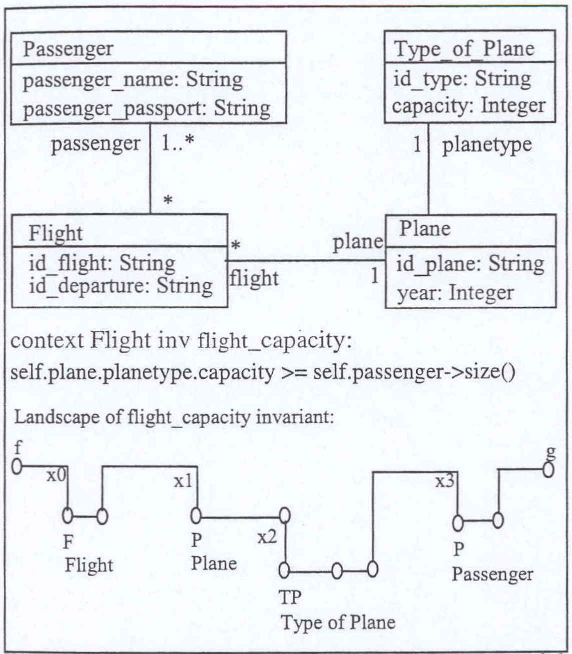


Figure 1. An OCL expression and its landscape model

Cant et al. state that the length of time and amount of work required to resolve the dependency is a function of the aggregate depth and breadth of the dependency "valley" [8]. Landscape models, also called nested valleys, help us to identify the time and amount of work required during chunking and tracing. As Cant et al. argue: *The total depth of the 'nested valley' depends on the length of the chain of dependencies that must be traced, and the difficulty of performing the tracing involved in each link of the chain. The number of 'steps' involved indicates the number of chunks that need to be considered.*

2.1. Measures

In this subsection we will present the definition of the measures NNC and DN.

Both measures were defined in terms of navigation, which means following links from one object to locate (trace) another object or a collection of objects [13].

Number of Navigated Classes (NNC)

- Definition: This measure counts the total number of classifiers (classes, types, interfaces, etc) mentioned in the navigation of an expression. If a class contains a reflexive relation and an expression navigates it, the class will be considered only once in the measure. Similarly, a class used in two (or more) different navigations is counted only once.
- Example: The value of NNC for the expression of Figure 1 is 3 because we have navigated to the Passengers, Plane and Type_of_Plane classes.
- Goal: NNC measures the number of different objects which are coupled to the contextual instance. Warmer and Kleppe [31] argue that any invariant should navigate through the smallest possible number of associations, which also implies that the referred number of classifiers (through navigations) should be kept as lower as possible. A high number of navigated classes will increase the coupling between the objects.
- Cognitive explanation: The first step in constructing a conceptual model is to identify a set of fundamental concepts to describe the domain, and these concepts appear in the model as classes or types [23]. Cant et al. argue that classes are typical examples of chunk during the software comprehension. NNC refers to the different classes that are chunked by the modelers when they have to comprehend or modify an OCL expression. The larger the set of classifiers used in navigations, the greater the context to be comprehended.
  There are many research works dealing with an upper limit of the human capacity to process information on simultaneously interacting object with reliable accuracy. For example, Miller [18] argues that people are able to remember about 7 chunks in short-term memory tasks. Others as Cowan et al. [10] or Broadbent [6] propose an average capacity limit of about 4 chunks.
  Although it is out of scope of this paper to refer to cognitive thresholds for process information, we think that a particular limit for the modelers when comprehending an OCL expression depends on the different chunks involved, and it also depends on the familiarity of the information encountered, i.e. the domain's experience [16].

Depth of Navigations (DN)

- Definition: For graphically representing all navigations expressed in an OCL expression we build a tree. This tree of navigation is built using the class names used in navigations. We will only consider navigations starting from the contextual instance (self). The root of the tree is the class name which 'self' represents. Then we built a branch for each combined navigation, where each class we navigate to is a node in the branch. Nodes are connected by "navigation relations". DN is defined as the maximum depth of the tree.

- Example: The value of DN corresponding to the expression of Figure 1 is obtained by building a navigation tree (see Figure 2). The root of the tree is the class which represents the contextual type of the OCL expression, i.e. Flight. Each path in the tree represents each of the simple or combined navigations used in the expression. These navigations are: self.passengers (a simple navigation because only one relationship is used), and self.plane.planetype (a combined navigation). The largest path from the root to any leaf is the value of DN, i.e. DN = 2.
- Goal: DN was defined for measuring the depth of coupling. A higher depth of navigations may involve a complicated navigation. Warmer et al. [31] suggest avoiding complex navigations expression, they also argue that: "using long navigation makes details of distant objects known to the object where we started the navigation". A high value of this measure will be an indicator of how distant the objects known by the contextual instance are.
- Cognitive explanation: This measure is based on tracing cognitive technique. From a cognitive point of view, DN implicitly shows the length of time and amount required to resolve longest dependency during OCL navigations, that is proportionally dependent of the depth of the dependency valley of landscape models.

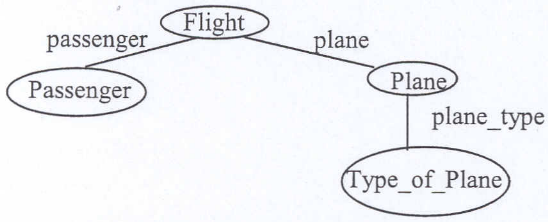


Figure 2. A navigation tree sample

At this moment we are not able to provide thresholds values for DN and NNC; further validation is needed. We can only suggest that its values should be as low as possible, in order not to overload the modelers.

3. A Family of Experiments

The main goal of this section is to carefully describe a family of experiments we carried out for testing if coupling expressed by the depth of coupling (DN) and quantity of coupled objects (NNC) influence on the comprehension and modifiability of OCL expressions. The original experiment was carried out at the National University of Comahue in Argentina (UNC experiment) and its replicas at University of Castilla-La Mancha (UCLM experiment) and University of Alicante (UA experiment) in Spain, respectively.

In order to conduct each experiment we have followed (with minor changes) the experimental process proposed by Wohlin et al. [32].

Hereafter, we will summarize each step of the experimental process, describing the common aspects and the differences between the original experiment and its replicas.

3.1 Experiment Planning

The goal of this family of experiments is defined by using the GQM template [1] as follows:

Analyze *The depth of coupling and quantity of coupled objects in OCL expressions;*
For the purpose of *Evaluating;*
With respect to *The capability to be used as comprehension and modifiability indicators of OCL expressions;*
From the point of view of *researchers;*
In the context of *Undergraduate Computer Science students.*

Subjects. A short description of the subjects and how they were encouraged to participate in each experiment is detailed below:

- UNC Experiment: We motivated a group of students who had taken a semester on System Analysis to take an additional and intensive course of OCL. The students were enrolled in the third year of Computer Science. The course consisted of two sessions of 10 hours each one. Fifteen students who attended the second session participated in the experiment. The experiment was part of the practical exercises of the course, but it was not mentioned that these exercises were related to an experiment. The goal of the experiment was not disclosed to the subjects.
- UCLM Experiment. The subjects were twelve undergraduate students enrolled in the fifth-year of Computer Science. The experiment was run in a practical session of the "Software Engineering II" course. Before the experiment was run the subjects participated in three lectures about OCL (one hour each one). These lectures were given by the same professor who supervised the

experiment. The subjects were motivated to participate in the practical session because we told them that similar exercises could be included in the final exam. Not all the students who participated in the experiment attended all the lectures, having thus different backgrounds in OCL.

- UA Experiment. Twenty nine students enrolled in the third-year of Computer Science participated in this replica. They were students of the first Software Engineering course. We invited the students to participate in a short seminar about OCL and to do a test as part of the seminar. The subjects were motivated to participate in the experiment offering them an extra point in the final score of the Software Engineering course, for completing all the required experimental tasks.

Variable selection. The independent variables (IVs) are the quantity of coupled objects and the depth of coupling. The dependent variables (DVs) are the comprehensibility and modifiability of OCL expressions.

Instrumentation. The experimental material consists of four UML/OCL models with one OCL expression in each one. These models were related to different universes of discourse that were easy enough to be understood by each of the subjects. In order to check the material we performed a pilot experiment. We asked a researcher who has experience on OCL and UML to carry out the experimental tasks. We considered all the modifications she suggested.

The quantity of coupled objects were measured by the Number of Navigated Classes (NNC) measure and the depth of coupling by the Depth of Navigations (DN) measure. The DVs were measured by the following measures:

- COM Eff (Comprehensibility Efficiency) = Number of correct answers/ number of seconds the subject spent carrying out the comprehension tasks (COM Time).
- MOD Eff (Modifiability Efficiency) = Number of correct modifications/ the number of seconds the subject spent performing the modifiability tasks (MOD Time).

Hypotheses formulation. We formulated the hypotheses:

- Hypotheses 1, $H_{0,1}$: There is no effect of the depth of coupling (measured by DN) on the Comprehensibility Efficiency (COM Eff) of OCL expressions // $H_{1,1}$: $\neg H_{0,1}$
- Hypothesis 2, $H_{0,2}$: There is no effect of the quantity of coupled objects (measured by NNC) on the Comprehensibility Efficiency (COM Eff) of OCL expressions. // $H_{1,2}$: $\neg H_{0,2}$
- Hypothesis 3, $H_{0,3}$: There is no interaction effect between depth of coupling (measured by DN) and the quantity of coupled-objects (measured by NNC) on the Comprehensibility Efficiency (COM Eff) of OCL expressions. // $H_{1,3}$: $\neg H_{0,3}$
- Similar hypotheses were defined for MOD Eff ($H_{i,j}$, i=0,1 ; j=4,5,6)

Experiment design. We considered a 2x2 factorial design (see Table 1). We considered two factors: NNC and DN with two levels each one (low, high), 1 and 3 for DN, and 2 and 4 for NNC.

For each group we designed one UML/OCL model composed of one UML class diagram with one OCL expression. We selected a within-subject design experiment, i.e. all the tasks of the four models had to be solved by each of the subjects. The four models were randomly assigned in different order to the subjects to avoid learning effects.

A general description of the four different tests is:

- G1 test is the simplest OCL expression, few objects are coupled within the expression and they belong to the context surrounding of the contextual instance.
- G2 test is similar to G1 test but the number of objects is higher than in G1 test. Nevertheless the objects are not far away from the contextual instance.
- G4 test includes navigations to distant objects through the deepest navigations.
- G3 test does not involve a high number of different objects as G4 does, but includes a high depth of navigations which involves the use of reflexive navigations.

Table 1: A 2X2 crossed factorial design

		DN	
		Low	**High**
NNC	**Low**	2,1 (G1)	2,3 (G3)
	High	4,1 (G2)	4,3 (G4)

Experimental Tasks. Each UML/OCL model had an enclosed test that included two types of tasks:

- Comprehension tasks (COM Tasks): Consist of four questions about the meaning of the OCL expression within the UML/OCL model. These questions reflected whether the subjects had understood each OCL expression or not. The subjects also had to write down the initial and end time for each test, obtaining thus what we called COM Time.

- Modification Tasks (MOD Tasks): Each subject had to modify the OCL expression according to three new requirements. The modifications to each test were similar, including defining new navigations, attributes referred through navigations, etc. They also had to record the initial and end time for each test obtaining, thus, what we called MOD Time.

Execution. The experiments were run in one session. The subjects were given all the materials previously described. We explained to them how to carry out the tests, asking them to carry out the test alone, and using unlimited time to solve it. An instructor supervised the experiment.

Data validation. Once the data were collected, we checked them and noted down the different times and the number of answered (right and wrong) questions. COM and MOD Efficiency were also calculated.

4. Data Analysis and Interpretation

All the data analysis was carried out by means of SPSS [29]. We did a Shapiro-Wilk test for testing the normality of the three sample data (Com Eff and MOD Eff data). Since all the data followed a normal distribution, we carried out ANOVAs with repeated measures in order to test our hypotheses presented in section 3.1 (with a level of significance $\alpha = 0.10$). ANOVAs allowed us to analyze the interaction between the independent variables under study when the measurement of the dependent variable is repeated.

4.1. Testing Hypotheses

First we checked for outliers related to the COM Eff and MOD Eff data. Table 2 shows the outliers found in each sample data.

For the sake of brevity, we summarize the ANOVA results for the three experiments in Table 3. The column titled 'p-value' represents the significance level which will allow us to reject or accept the null hypotheses (significant coefficients at level 0.10 are shown in bold font). The column titled as 'observed power' indicates the chance of detecting a genuine main effect in a factor. For example in the UNC experiment data, the power of the depth of navigations (measured by DN) main effect was 0.813 which indicates that the chance of detecting a genuine main effect there is fairly high, that is, 81.3%.

Table 3 reveals that:

UNC experiment: DN, NNC and their interactions seem to affect COM Eff (as the p-value is less than 0,10). The value of DN seems to affect the MOD Eff.

UCLM experiment: DN affects on COM Eff and NNC on MOD Eff.

UA experiment: DN and the interaction between the measures affects on COM Eff, whilst DN and NNC affect on MOD Eff.

4.2. Conclusions of the family of experiments

Looking at Table 4 which summarizes the results, the general findings obtained through the family of experiments are:

- DN seems to affect COM Eff in the three experiments. The interaction of DN and NNC also seems to affects the COM Eff.
- The interaction between the factors seems not to affect MOD Eff.

Regarding the results of the three experiments we decided to analyze the data of the family altogether. We added a precedence factor (PR) when we gathered all the information (PR = 1 for UNC data, PR = 2 for UCLM data, and PR = 3 for UA data). As a result, we obtained the following: DN, NNC and their interactions seem to affect the COM Eff, and DN and NNC seem to affect the MOD Eff but not their interaction.

The fact that the interaction affects the COM Eff means that in order to estimate the comprehensibility of an OCL expression we need to know the number of classes and the depth of navigation of the navigation. In other words, the way the number of classes affects the comprehensibility depends of the depth of navigations and vice versa. In terms of a landscape model of an expression the depth and the number of steps influence its comprehension, meaning that tracing as well as chunking affect the comprehension.

Figure 3 depicts the estimated marginal means in a profile plot for COM and MOD Eff. The vertical axis of a profile plot represents the dependent variables (COM or MOD Eff). As we proceed in the plots of Figure 3, DN factor was selected to be represented as the horizontal axes, whereas NNC factors will be displayed as separate plots. A simple overview of the plot clearly shows that the two lines are not parallel in COM Efficiency (interaction of DN and NNC seems to affect COM Eff), and indeed they are parallel in the MOD Eff (indicating no interaction between DN and NNC). Here we summarize the conclusions extracted from Figure 3:

- In connection to COM Eff, we can say that if the depth of the navigation is low, better COM efficiency is obtained if the quantities of classes involved are low (NNC= 2) rather than high (NNC= 4). However, if the depth of navigation is high, better COM Eff is obtained if the quantities of classes involved are high (NNC= 4) rather than low (NNC= 2). The last case takes place when DN

= 3 and NNC = 2, and represents navigations using reflexive relationships in a class which seems to be more difficult to comprehend than simple navigations involving two different classes.

- Related to MOD Eff, whichever the plots (representing the NNC = 4 or NNC =2) may be, they have a negative slop in both experiments, also their slops seem to be proportional. Moreover, considering the plot, the larger the DN factor the worse the MOD Eff. Without considering the quantity of classes, that is, high (NNC = 4) or low (NNC = 2), if classes are loosely coupled (DN is low) the better the MOD Eff and if classes are highly coupled the worse MOD Eff.

5. Validity Evaluation

Next we will discuss how we tried to alleviate such issues that could threaten the validity of the family experiments:

Threats to conclusion validity. In the conclusion validity we want to make sure that there is a statistical relationship, i.e. that our conclusions are statistically valid. The only issue that could affect the statistical validity of this study is the size of the sample data which is perhaps not enough for parametric statistic tests. We are aware of this, so we will consider the results of the experiment only as preliminary findings.

- Low statistical power. The power of a statistical test is the ability of the test to reveal a true pattern in the data. If the power is low, there is a high risk that an erroneous conclusion will be drawn. We have considered the observed power of the ANOVAs in our analysis, and when we evaluate the results for the whole family we also add a PR factor as an inter-subject factor.
- Violated assumption of statistical tests. Violating the assumptions of a test may lead to wrong conclusions. In our case we used a robust test, the ANOVA with repeated measures..
- The error rate. The error rate is concerned with the actual significance level. In order to improve the power of the test we have selected $\alpha = 0.10$ significance level which is common in ANOVA tests.

Threats to construct validity. This validity is concerned with the relationship between theory and observation. It defines the extent to which the variables successfully measure the theoretical constructs in the hypothesis. We proposed an objective measure for the DVs, the comprehension/ modifiability efficiency. The construct validity of the measures used for the IVs is guaranteed by Briand et al.'s framework used to validate them [25]. An issue we also dealt with is:

- Confounding constructs and levels of constructs. In some relations they are not primarily related to the presence or absence of a construct, but to the level of the construct which is of importance to the outcome [32]. We selected subjects having knowledge in UML language, but the years of experience with UML is different: UNC students have two years of experience, UCLM students have one year of experience, and UA students have less than a year.

Threats to internal Validity. The internal validity is the degree of confidence in a cause-effect relationship between factors of interest and the observed results. We have dealt with the following issues:

- History. An existent risk is that history affects the experimental results, since the circumstances are not the same on the experiments [32]. This is the situation in the UA experiment, in which case we ran the experiment in the final lecture before Easter Holidays, and we realized that during the experiment they were eager to finish the exercise. We also became aware of that because, before the seminar started, many students had asked us about the ending time of the seminar. We think that this could have affected the results obtained in this replica because in general, the tests' correctness was low.
- Maturation. Subjects may react differently as time passes by. For example, the subjects are affected negatively during the experiment if they get tired or bored [32]. We dealt with this issue and we ran a pilot experiment in order to estimate the average time the subject will spend performing the four tests. The estimation was of one hour and we believe that it would not produce a boredom effect [14]. In fact, after running the experiment we proved that the estimated time was right.
- Selection. This is the effect of natural variation in human performance. Depending on how the subjects are selected from a larger group, the selection effects may vary [32]. The selection of subjects was not the same in the three experiments as described in section 3.
- Mortality. This effect is due to the different kinds of persons who drop out of the experiment. We think the UCLM experiment was affected by this issue. The experiment was run in a practical session of a course. Although as a general rule a reduced number of students participate in a

practical session, even less students than usual took part of the session when the experiment was run.

Table 2. Oultiers found in each experiment

	UNC	UCLM	UA	UNC+UCLM+UA
# Outliers of COM Eff	3	1	1	3
# Outliers of MOD Eff	5	4	2	7

Table 3. ANOVA results for COM Eff and MOD Eff

		UNC			UCLM			UA		
		# subjects	p-value	observed Power	# subjects	p-value	observed Power	# subjects	p-value	observed Power
COM Eff	DN	12	**0.021**	0.813	**11**	**0,079**	0.569	**25**	**0.004**	0.918
	NNC	12	**0.000**	0.998	11	0,479	0.179	25	0.173	0.391
	Int.	12	**0.036**	0.724	11	0.103	0.511	**25**	**0.003**	0.939
MOD Eff	DN	10	**0.091**	0.542	8	0.873	0.104	**24**	**0.001**	0.985
	NNC	10	0.488	0.175	**8**	**0.002**	0.995	**24**	**0.002**	0.967
	Int.	10	0.447	0.190	8	0.786	0.111	24	0.287	0.323

Table 4. Summary of findings

Exp	COM Eff		
	DN	NNC	Inter.
UNC	Yes	Yes	Yes
UCLM	Yes		
UA	Yes		Yes
	MOD Eff		
	DN	NNC	Inter.
UNC	Yes		
UCLM		Yes	
UA	Yes	Yes	

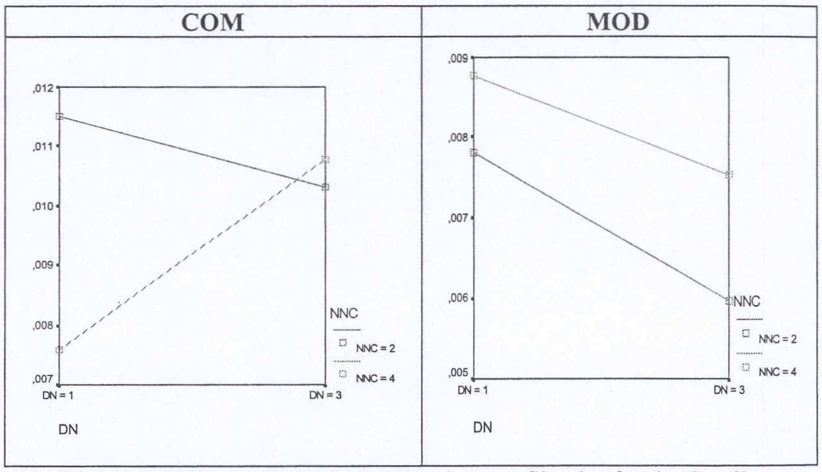


Figure 3. Estimated marginal means in a profile plot for the Family

Threats to external validity. The greater the external validity, the more the results of an empirical study can be generalized to actual software engineering practice. Two threats of validity have been identified which limit the possibility of applying any such generalization:

- Interaction of selection and treatment. The most important threat affecting this validity is the experimental subjects. We are aware that experiments with practitioners and professionals should be carried out in order to be able to generalize the results although there is no industrial experience in modeling with UML and OCL. So, in this case in particular, as the tasks to be performed do not require high levels of experience, experiments with students could be appropriate [15]. Probably as Briand et al. argue in [5] the chosen student (as well as those of their experiments) are better trained in modeling with UML and OCL than most software professionals.
- Interaction of setting and treatments. This is the effect of not having representative experimental setting or material. In the experiment we used UML/OCL models which can be representative of real cases. Moreover, we give a course on OCL using the same terminology as its last version (2.0)
- Interaction of history and treatment. The aforementioned issue related to history in the internal validity could affect the result of UA experiments. Students were eager to finish the

exercises in the last day before Easter Holidays, and by finishing them they could obtain an extra point and this point was only dependent on finishing the exercise, not on how the exercise was done.

6. Related Work

Although it was empirically proved that OCL has the potential to significantly improve UML-based model comprehension and maintainability [5], OCL expressions themselves are difficult to comprehend and, consequently to maintain. As we noted above, the lack of empirically validated measures that can be used as early indicators the comprehensibility and modifiability of OCL expressions was the motivation for a project we have been pursuing for the last three years. For that reason, all the related work focuses on our own research.

In [25] we define and theoretically validate a set of measures[1] for OCL expressions, considering the most frequently used elements of the OCL metamodel. In [26] we present a family of experiments we run in order to empirically validate a set of measures using a correlational analysis. We found that there is a statistically significant relationship between many metrics, especially those related to tracing, and the comprehensibility and modifiability efficiency. For example, the number of classes used in navigations, the number of collection operations and the number of collection operation's iterator variables influences the subjects' comprehensibility efficiency. The number of navigations used in navigation and the length of navigations has a correlation with the cognitive load when subjects rate modifiability Tasks. This fact was also empirically validated using verbal protocols [27] using thinking aloud [28] methods. However we think that an important aspect is that cognitive techniques are concurrently applied by modelers (as we graphically show through landscape models) and we must evaluate whether the interactions of measures related to tracing and chunking (such as the interaction of NNC and DN) affects also the comprehension/modification of OCL expressions. So, the experiment presented in this paper is specific due to the fact it analyze the effect of two important measures and their interactions in the comprehension and modification of OCL expressions.

7. Conclusions

Through a family of experiments, we confirm our hypotheses. We found that to certain extent coupling affects on two main characteristics of the maintainability of OCL expressions, comprehensibility and modifiability. More specifically, the comprehensibility and modifiability of OCL expressions are dependent on how many different objects are coupled to the contextual instance (measured through the NNC measure) and the depth of coupling (i.e. farthest distance of any coupled object to the contextual instance -measured through the DN measure). Moreover, the interaction of DN and NNC affects the comprehensibility of an OCL expression but not the modifiability. We believe that the main fact that influencing that the interactions of the measures affects the comprehension is that reflexive relationships are more difficult to understand than simple relationships. The DN measure seems to affect comprehension in a stronger way rather than NNC probably due to the effort of doing tracing instead of chunking UML features. Thus, through experimentation we confirm what Warmer et al. state in [31]: it is recommended to limit the object's knowledge of the contextual instance to only its direct surroundings. However we claim that navigation using reflexive relationships should be carefully analyzed.

Further validation is need for confirming or disconfirming the current findings and to finding thresholds values. At this point we can only suggest to decrease as much as possible the values to DN and NNC. The findings of this study can be useful in many aspects, such as: to support the modeler in recognizing OCL specifications with high import coupling [11] and specifying good practices and guidelines, to use the measures in modeling and refactoring a UML/OCL design.

We are planning to carry out more experimentation taking "real cases" from industry, improving the experimental tasks, through qualitative methods such as thinking aloud [27] tasks or interviews.

Acknowledgments

This research is part of the ESFINGE project supported by the "Ministerio de Educación y Ciencia (Spain)" (TIN2006-15175-C05-05) and the COMPETISOFT project (506PI0287) financed by "CYTED (Programa Iberoamericano de Ciencia y Tecnología para el Desarrollo)".

References

[1] Basili, V. R. and Rombach, H. D.: The TAME project: Towards Improvement-oriented Software Environments. IEEE Trans. on Softw. Eng.Vol.14 N° 6, 1998, pp. 758-773.

[2] Boehm-Davis, D.A., Fox, J. E. and Philips, B.: Techniques for Exploring Program Comprehension. Empirical Studies of Programmers, Sixth Workshop. Norwood, NJ: Ablex, 1996, pp. 3-37.

---

[1] The measures were defined following a rigorous process [7], which mainly consists of three steps: definition, theoretical and empirical validation.

[3] Briand, L. C., Bunse, L. C. and Daly, J. W.: A Controlled Experiment for evaluating Quality Guidelines on the Maintainability of Object-Oriented Designs. IEEE Trans. on Softw. Eng., Vol. 27 Nº 6, 2001, pp. 513-530.
[4] Briand, L. C., Daly, J. W. and Wüst, J.: A Unified Framework for Coupling Measurement in Object-Oriented Systems. IEEE Trans. on Softw. Eng., Vol. 25 Nº 1, 1999, pp. 91-121.
[5] Briand, L. C., Labiche, Y., Yan, H. D. and Di Penta, M.: A Controlled Experiment on the Impact of the OCL in UML-based Maintenance. IEEE Int. Conference on Softw. Maintenance, 2004, pp. 380-389.
[6]Broadbent, D.E.: The magic number seven after fifteen years. In A. Kennedy and A. Wilkes (Eds) Studies in long-term memory. pp 3-18. London: Wiley, 1975.
[7] Calero, C., Piattini, M. and Genero, M.: Method for Obtaining Correct Metrics. Proc. of 3rd Int. Conf. on Enterprise and Information Systems ICEIS 2001,pp.779-784.
[8] Cant, S. N., Henderson-Sellers, B. and Jeffery, D.R.: Application of Cognitive Complexity Metrics to Object Oriented Programs. Journal of Object Oriented Programming, Vol. 7 Nº 4, 1994, pp. 52-63.
[9] Cant, S. N., Jeffery, D. R. and Henderson-Seller, B.: A Conceptual Model of Cognitive Complexity of Elements of the Programming Process. Information and Softw. Technology, Vol. 37 Nº 7, pp. 351-362. 1995.
[10] Cowan, N. The magical number 4 in short-term memory: A reconsideration of Mental Storage Capacity. Behavioral and Brain Sciences 24 (1), 2001.
[11] Chiorean, D., Bortes, M. and Corutiu, D.: Good practices for creating correct, clear and efficient OCL specifications. In 2nd Nordic Workshop on the UML. pp. 127–142, 2004.
[12] Gold, N. and Bennett, K.: A flexible method for egmentation in concept assignment. Proc. of the 9th International Workshop on Program Comprehension, page 135, Washington, DC, USA, 2001. IEEE Computer Society.
[13] Hamie, A., Howse, J., and Kent, S.: Interpreting the Object Constraint Language. Proceeding of Asia Pacific Softw. Engineering Conference. pp. 288-295. 1998
[14] Juristo, N. and Moreno, A.: Basics of Software Engineering Experimentation. Kluwer Academic.. 2001.
[15] Kitchenham, B., Pflegger, S. and Fenton, N.: Towards a Framework for Softw. Measurement Validation. IEEE Trans. of Softw. Eng.,Vol. 21 Nº 12, 1995, pp. 929-944.
[16] Klemola, T.: A cognitive model for complexity metrics. 4th International ECOOP Workshop on Quantitative Approaches in Object-Oriented Softw. Engineering. Sophia Antipolis and Cannes, France. 2000.
[17] Knight, C. and Munro, M.: Program comprehension experiences with gxl; comprehension for comprehension. Proc. of the 10th Int. IEEE Workshop on Program Comprehension, pp 147-156, Washington, DC, USA, 2002.
[18] Miller, G.A.: The magical number 7, plus or minus two: some limits on our capacity of processing information. Vol 63, No. 2. pp 81-97. 1956
[19] Misra, S.: Cognitive Program Complexity Measure. pp: 120-125. 6th IEEE ICCI. CA, USA, 2007.
[20] Nunes, I.: An OCL Extension for Low-Coupling Preserving Contracts. UML 2003. pp. 310-324. pp: 120-125. 6th IEEE ICCI. CA, USA, 2007.
[21] Object Management Group. UML 2.0 , OMG Document. Available at http://www.omg.org, 2005.
[22] Object Management Group. UML 2.0 OCL, ptc/05-06-06, OCL FTF report. OMG Document. Available at http://www.omg.org, 2005.
[23] Parsons, J. and Wand, Y.: Choosing Classes In Conceptual Modeling. Communications of the ACM. June 1997. Vol. 40, No. 6. 1997.
[24] Rajlich, V. and Wilde, N.: The role of concepts in program comprehension. Proc. of the 10th Int.Workshop on Program Comprehension, page 271, Washington, DC, USA, 2002. IEEE Computer Society.
[25] Reynoso, L., Genero, M. and Piattini, M.: Measuring OCL Expressions: An approach based on Cognitive Techniques. Pp: 161-206.Imperial College Press, UK. 2005.
[26] Reynoso, L., Genero, M. and Piattini, M.: Assessing the impact of coupling on the Understandability and modifiability of OCL expressions within UML/OCL combined models. pp.14. 11th IEEE International Software Metrics Symposium. 2005.
[27] Reynoso, L., Genero, M. and Piattini, M.: Using Verbal Protocols to Assess the Influence of Import-Coupling on the Comprehensibility of OCL Expression. pp: 440-449. 6th IEEE ICCI. CA, USA, 2007.
[28] Someren M., Barnard Y. and Sandberg J: The Think Aloud Method: a Practical Guide to Modelling Cognitive Process. Academic Press, London. 1994.
[29] SPSS, 2002 SPSS 11.5. "Syntax Reference Guide". Chicago. SPSS Inc., 2002.
[30] Y. Wang, and J. Shao. Measurement of the Cognitive Functional Complexity of Software, Cognitive weights. IEEE ICCI '05. pp.4-5,2005.
[31] Warmer, J. and Kleppe, A.: The Object Constraint Language. Second Edition. Getting Your Models Ready for MDA. Addison-Wesley, Massachusetts, 2003.
[32] Wohlin, C., Runeson, P., Höst, M., Ohlson, M., Regnell, B. and Wesslén, A.: Experimentation in Softw. Eng.: An Introduction, Kluwer Academic Publishers, 2000.