Model-Based Reuse for Crosscutting Frameworks: Assessing Reuse and Maintainability Effort

Thiago Gottardi*, Rafael Serapilha Durelli†, Oscar Pastor López‡ and Valter Vieira de Camargo*
*Departamento de Computação, Universidade Federal de São Carlos,
Caixa Postal 676 – 13.565-905, São Carlos – SP – Brazil
Email: {thiago_gottardi,valter}@dc.ufscar.br
†Instituto de Ciências Matemáticas e Computação, Universidade de São Paulo,
Av. Trabalhador São Carlense, 400, São Carlos – SP – Brazil
Email: rsdurelli@icmc.usp.br
‡Universidad Politecnica de Valencia, Camino de Vera s/n, Valencia, Spain
Email: opastor@dsic.upv.es

***Abstract*—**

I. Introduction

II. Software Restructuing

Perhaps the most common of all software engineering activities is the modifications of software. Unfortunately, software modification, i.e., software maintenance, often leaves behind software that is difficult to understand for those other than its author. In this context, software restructuring is a field that seeks to reverse these effects on software.

More specifically, software restructuring is the modification of software to make the software easier to understand and to change, or less susceptible to error when future changes are made (ref). In other words, it is the process of re-organizing the logical structure of existing software system to improve specific attributes [1]. Some examples of software restructuring are improving coding style, editing documentation, transforming program components (moving class, creating class, etc). The central idea of restructuring is the action of transformation. According to [2] a transformation can be defined formally as a function that receives a program, P, as input and produces a new program, P'. Thus, P' is said to be functionally equivalent to P $\Leftrightarrow$ P' exhibits identical behavior to P for all defined inputs of P. Finally, T is called a meaning preserving transformation if P' $\equiv$ P.

III. Related Work

IV. Conclusions

Acknowledgments

References

[1] B.-K. Kang and J. M. Bieman, "A quantitative framework for software restructuring," *Journal of Software Maintenance: Research and Practice*, vol. 11, no. 4, pp. 245–284, 1999.

[2] J. Eloff, "Software restructuring: implementing a code abstraction transformation," in *Proceedings of the 2002 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology*, 2002, pp. 83–92.