Title

Rodrigo Souza
rodrigorgs@gmail.com

Dalton Guerrero
dalton@dsc.ufcg.edu.br

Jorge Figueiredo
abrantes@dsc.ufcg.edu.br

Christina Chavez
flach@ufba.br

October 27, 2011

Abstract

As a software system evolves, its actual architecture may deviate from its original reference architecture. A better understanding of the actual architecture can be gained by applying architecture recovery algorithms to the source code of the system. Unfortunately, there is little knowledge regarding the quality of such algorithms, mostly because, in order to assess their accuracy, one needs updated, detailed architectures for a variety of software systems. To overcome such problem, we propose a simulation model that synthesizes implementation-level dependency graphs that conform to the module view of any given reference architecture. We show, using observations from network theory, that the graphs are very similar to dependency graphs extracted from the source code of real software systems. Then, we synthesize thousands of graphs, in a controlled way, and use them as benchmarks for six well-known architecture recovery algorithms: ACDC, Bunch, SL75, SL90, CL75, and CL90. By comparing the given reference architectures with those recovered by the algorithms, we conclude that ACDC and Bunch outperform the alternatives, specially if the architecture contains more than a couple of modules.

1 Introduction

Architectural drift

Reverse engineering / Architecture recovery algorithms

2 Background

Architecture recovery algorithms

Evaluation of architecture recovery algorithms.

3 The BCR+ model

Network theory / complex networks / scale-free networks

BCR+ description [1]

4 Software-Realism (aka "softwareness")

Motifs / Triads. Triad concentration profiles.

Software-realism metric.

Software-realism classifier. Training and testing the classifier.

Software-realism for the BCR+ model. (Cite previous paper, where we also evaluate software-realism for three other models)

5 Evaluation of Architecture Recovery Algorithms

Traditional experimental setup for evaluation of architecture recovery algorithms: given a dependency graph and a reference architecture, run the algorithm on the graph and compare the recovered architecture with the reference architecture.

MoJo metric.

6 Other Possible Sections

Discussion

Limitations

Conclusion

Acknowledgments

References

References

[1] Béla BollobÁs, Christian Borgs, Jennifer Chayes, and Oliver Riordan. Directed scale-free graphs. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms SODA '03*, pages 132–139. Society for Industrial and Applied Mathematics, 2003.