

Automatically identifying changes that impact code-to-design traceability

Maen Hammad
Michael L. Collard
Jonathan I. Maletic

An approach is presented that automatically determines if a given source code change impacts the design (i.e., UML class diagram) of the system. This allows code-to-design traceability to be consistently maintained as the source code evolves. The approach uses lightweight analysis and syntactic differencing of the source code changes to determine if the change alters the class diagram in the context of abstract design. The intent is to support both the simultaneous updating of design documents with code changes and bringing old design documents up to date with current code given the change history. An efficient tool was developed to support the approach and is applied to an open source system (i.e., HippoDraw). The results are evaluated and compared against manual inspection by human experts. The tool performs better than (error prone) manual inspection.

Published in:

[Program Comprehension, 2009. ICPC '09. IEEE 17th International Conference on](#)