Carnegie Mellon University
Computer Science Department

Speaking Skills Talk

Glacier: Transitive Class Immutability for Java

Michael Coblenz

Friday, April 21, 2017
12:00 pm
GHC 8102

Though immutability has been long-proposed as a way to prevent bugs in software, little is known about how to make immutability support in programming languages effective for software engineers. We designed a new formalism that extends Java to support transitive class immutability, the form of immutability for which there is the strongest empirical support, and implemented that formalism in a tool called Glacier. We applied Glacier successfully to two real-world systems. We also compared Glacier to Javas final in a user study of twenty participants. We found that even after being given instructions on how to express immutability with final, participants who used final were unable to express immutability correctly, whereas almost all participants who used Glacier succeeded. We also asked participants to make specific changes to immutable classes and found that participants who used final all incorrectly mutated immutable state, whereas almost all of the participants who used Glacier succeeded. Glacier represents a promising approach to enforcing immutability in Java and provides a model for enforcement in other languages.

Presented in Partial Fulfillment of the CSD Speaking Skills Requirement.