

Name:

Paul V. Gestwicki

E-mail address:

pvg@cse.buffalo.edu

Dissertation Proposal Title:

Interactive Visualization of Object-Oriented Programs

Committee:

Chair: Dr. Bharat Jayaraman

Members: Dr. Ashim Garg, Dr. Bina Ramamurthy

Dissertation Proposal Abstract:

The goal of this dissertation is to develop effective and efficient methods for interactive visualization of object-oriented programs. Interactive visualization serves three purposes: (i) as a pedagogic tool for program understanding; (ii) as an aid to graphical debugging of programs; and (iii) as a visual operational semantics of the language. In this dissertation, we significantly extend a traditional graphical notation for procedural languages to provide an effective visualization for the object-oriented language Java. Our notation makes clear the important fact that objects are environments within which method activations take place, and it also clarifies the semantics of classes, inheritance, exceptions, threads, etc. We describe an interactive tool for visualizing Java programs that supports the simultaneous visualization of object structures and method activations. The tool provides multiple levels of granularity (such as method-level and statement-level) for forward execution as well as reverse execution. Additionally, the tool runs on existing Java Virtual Machines and does not require a separate interpreter to be implemented. We discuss algorithms for rendering our visualizations; standard graph drawing algorithms cannot be directly used due to the special properties of the visualizations. We describe the aesthetic criteria for our visualizations along with the complications of satisfying these criteria. Related work in program visualization, reverse execution, and graph drawing are discussed, along with how it can apply to the proposed research. The proposed research falls into four main categories: (i) finding efficient and effective algorithms for automated, aesthetic drawing of our visual semantics; (ii) developing a useful model for interactive reversible execution; (iii) extending our notation to cover the advanced features of Java; and (iv) an empirical study of the visualization of Java so as to discover relationships between diagrams and the programs that produce them.