The presence of aliasing makes modular verification of object-oriented code difficult. If multiple clients depend on the properties of an object, one client may break a property that others depend on.

In this thesis I present a modular verification approach based on the novel abstraction of object propositions, which combine predicates and information about object aliasing. In my methodology, even if shared data is modified, an object invariant specified by a client holds. My permission system allows verification using a mixture of linear and nonlinear reasoning. This allows it to provide more modularity in some cases than competing separation logic approaches, because it can more effectively hide the exact aliasing relationships within a module.

I have implemented the methodology into my tool Oprop, that can be accessed online. I have used Oprop to automatically verify a number of examples, thus showing the applicability of this research. I have verified simple examples that show particular features of the Oprop language, and also instances of different design patterns, such as the composite, state, proxy and flyweight design patterns.

Thesis Committee:
Jonathan Aldrich, Chair
Frank Pfenning
David Garlan
Matthew Parkinson, Microsoft Research Cambridge