Thinking Outside the □: 
Verified Compilation of ML5 to JavaScript

Joomy Korkut
Advisor: Daniel R. Licata
Wesleyan University

Abstract
Curry-Howard correspondence describes a language that corresponds to propositional logic. Since modal logic is an extension of propositional logic, then what language corresponds to modal logic? If there is one, then what is it good for? Murphy’s dissertation[1] argues that a programming language designed based on modal type systems can provide elegant abstractions to organize local resources on different computers. In this thesis, I limit his argument to simple web programming and claim that a modal logic based language is correct for all world, and the □ connective, which means a proposition is correct for all world, and the ∪ connective, which means a proposition is correct for some world. We replace them with the hybrid connectives □ω for all world and △ω for some world. If there is one, then what is it good for? Murphy’s dissertation[1]

Type-Directed Translation
Our compiler has 5 conversion steps before JavaScript:
1. ML5: an Agda formalization of Lambda 5
2. Continuation-passing style: Considering that most actions in JavaScript are run through callbacks, this process is necessary to move us closer to JavaScript, our final target language.
3. Closure conversion: We eventually want to hoist all lambdas in a program to the top, so that we can call them by their names during network communication. However, this is not possible because these functions contain bound variables from previous definitions. That is why we create closures to get rid of these bound variables.
4. Lambda lifting: Now that functions do not have any other bound variables other than the argument of the function they are in, we can hoist the functions.
5. Monomorphic: Before conversion to JavaScript, we have to monomorphize valid values into values in specific worlds.

Conversion to JavaScript
We are defining functions to convert continuation expressions and expressions to JavaScript expressions and function statements.

References