



# Klaas Pruiksma

Adjoint Logic with Applications

**Monday, May 10, 2021 – 11:00 a.m. – REMOTE**

Many different logics are used in computer science, and in particular in programming language theory. Likewise, many different programming languages are used, often with different sets of features for making different tasks easier. Unsurprisingly, it is also often useful to work with features from multiple logics or multiple languages, and this is generally done in an ad-hoc way, creating a new logic/language with all of the desired features. This approach does not generalize well and, moreover, adding additional features to a programming language can make it harder to reason about the behavior of programs, and as a result makes it harder to perform optimizations in an implementation.

I propose to resolve this issue using a uniform framework for constructing logics, called Adjoint Logic. Adjoint Logic is a schema for combining logics with various structural properties. By taking suitable instances of this schema, possibly with some additional restrictions, we can reconstruct a variety of different logics used in computer science, such as the modal logic S4, lax logic, linear logic with the exponential modality  $!$ , and more. Moreover, the uniform nature of these constructions means that they can be combined cleanly. Likewise, when we interpret Adjoint Logic as a programming language via a Curry-Howard interpretation, we can reconstruct several language features, particularly dealing with concurrency, and combine them in a uniform way.

Building on this, I plan to study the modularity and isolation properties that hold in Adjoint Logic, with a goal of being able to seamlessly combine different logical or language features while still maintaining enough separation to reason about and implement portions of the language independently. An ideal end result would be a general theorem specifying how the semantics of different portions of the logic interact and affect each other, and so what constraints we need in order to make the whole system coherent.

**Thesis Committee:**

**Frank Pfenning, Chair**

**Bob Harper**

**Stephanie Balzer**

**Henry DeYoung**

**Andy Pitts, Darwin College, Cambridge**

**Thesis Summary: [www.andrew.cmu.edu/user/kpruiksm/proposal.pdf](http://www.andrew.cmu.edu/user/kpruiksm/proposal.pdf)**