



R-Stream Compiler

**Peter Mattson, Richard Lethin, Eric Schweitz,
Allen Leung, Vass Litvinov, Michael Engle,
Charlie Garrett**

Reservoir Labs Inc.



Scope of the problem

- **Automatically parallelize C programs**
 - Source-to-source compilation
 - Initially compile DSP applications for PCA architectures
 - Expand to more general embedded applications and architectures
- **Deliver a reliable compiler this year (version 2.0)**
 - Classic loop transformations
 - Modulo scheduling
 - Phase ordered optimizations
- **Deliver a state of the art compiler next year (version 3.0)**
 - Expose maximum parallelism
 - Schedule at the level of statement instances
 - Unify instruction scheduling and data placement into one mapping algorithm

Demonstration

- **High-Level compiler mapping Lincoln Labs' GMTI demonstration to two Polymorphous Computer Architecture chips**
- **Syntax extensions to C for expressing abstract arrays**
- **Scheduled and Mapped Code**
- **Streaming Virtual Machine Output**

Questions you should be asking

- **How automatic is R-Stream?**
 - May always need programmer assistance to avoid pointers, mark reductions, etc.
- **How good is the generated code?**
 - Too early to say
 - Phase ordered optimizations work for some cases, but have limitations
 - Unified optimizations have theoretical claims of optimality. Will that translate into practice?
- **Will R-Stream work for application X and architecture Y?**
 - I don't know, but let's talk about it