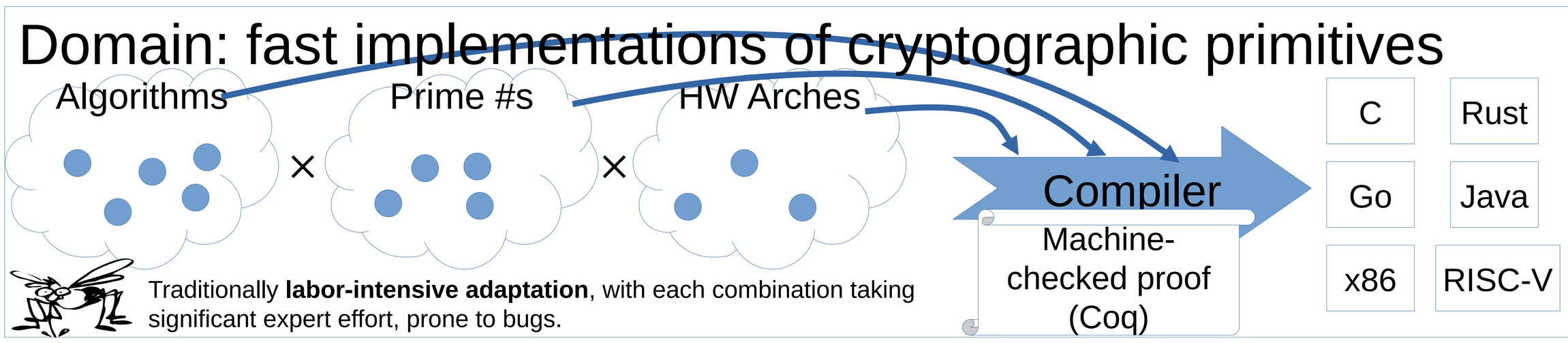
Scaling Correct-by-Construction Code Generation for Cryptography

PI Prof. Adam Chlipala, MIT

https://github.com/mit-plv/fiat-crypto

Domain: fast implementations of cryptographic primitives



Challenges

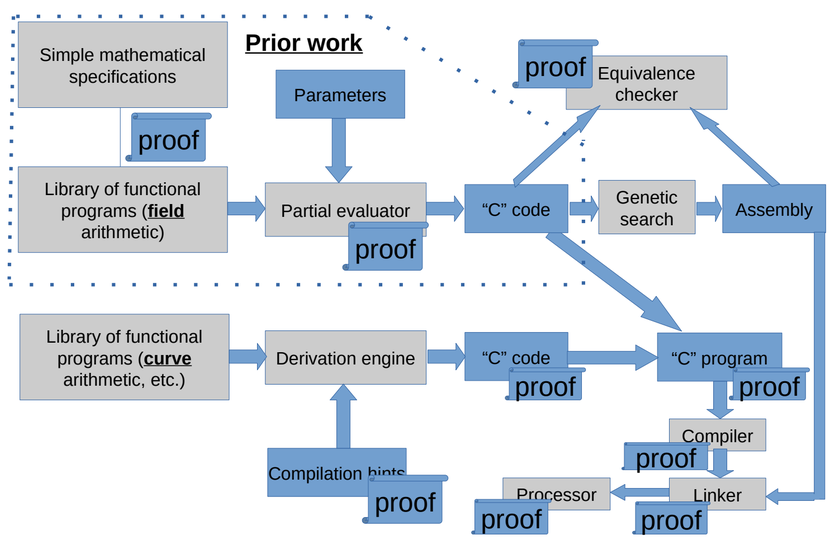
- Maintaining all three of pleasant high-level programming, performance, and correctness/security
- Extension to our C-code-generation method must add support for functions, loops, mutable data structures, and lookup tables.
- Equivalence checker for assembly code should have minimal TCB.

Scientific Impact

- Novel takes on the broad problems of compiler verification and translation validation
- Tools for extracting Coq developments to fast C code should be of interest in other research in formal methods.

Solution [all with Coq proofs!]

- Proofs of new high-level crypto algorithms
- Tool to specialize these routines to fast C code
- Tool to validate compiled assembly code against original algorithms



Real-world impact

- Open-source tool already adopted by all major browsers & mobile platforms.
- Goal is to broaden adoption further by generating more of crypto libraries and with higher-performance code.

Education

- Incorporate ideas in graduate course on program verification.
- Project has already involved many undergrads and other students and will involve more for the new work.

Broader Participation

- Teaching crypto practitioners about formal methods through workshops like High Assurance Cryptographic Software and virtual training sessions