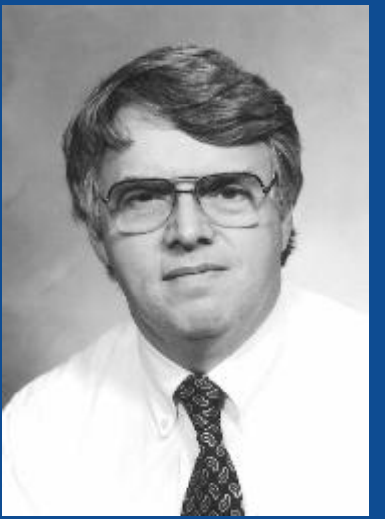


Practical Formal Verification By Specification Extraction

NSF Grant CT-T 0716478

J. Knight, J. Davidson, W. Weimer, A. Nguyen-Tuong
University of Virginia

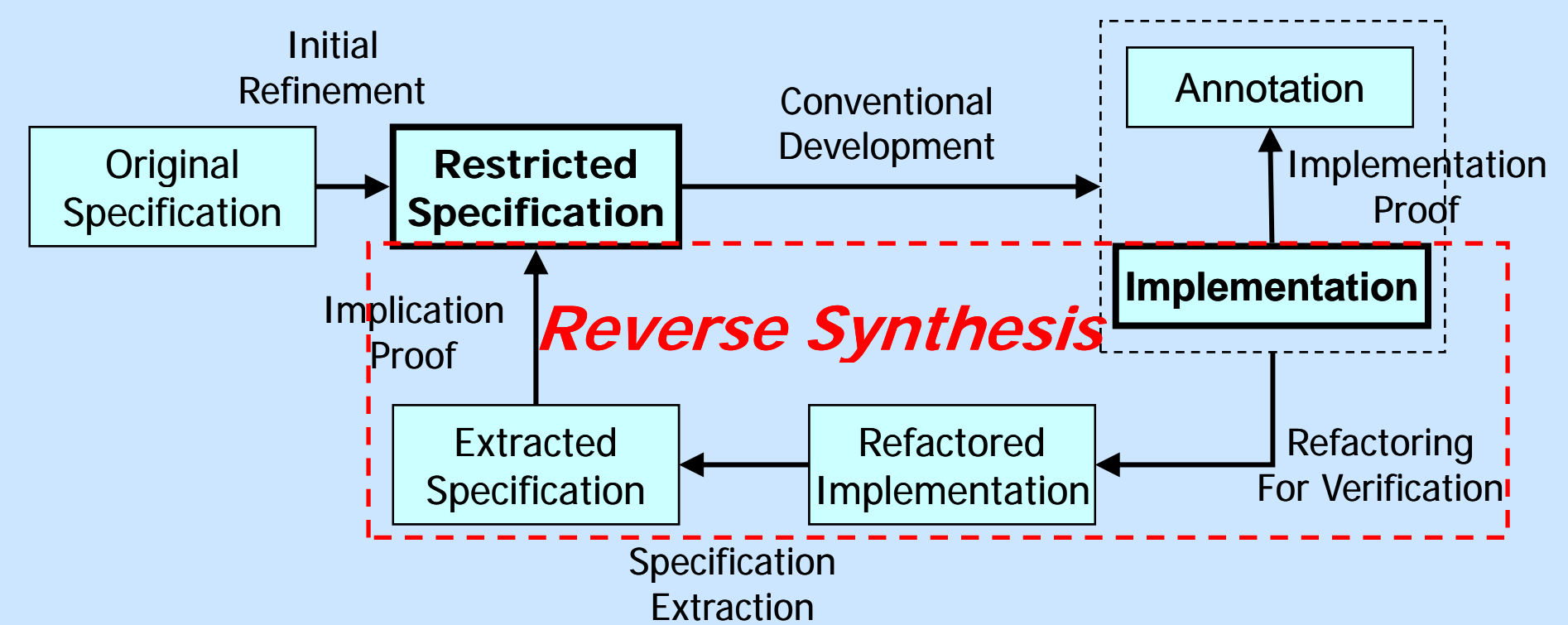
<http://dependability.cs.virginia.edu>



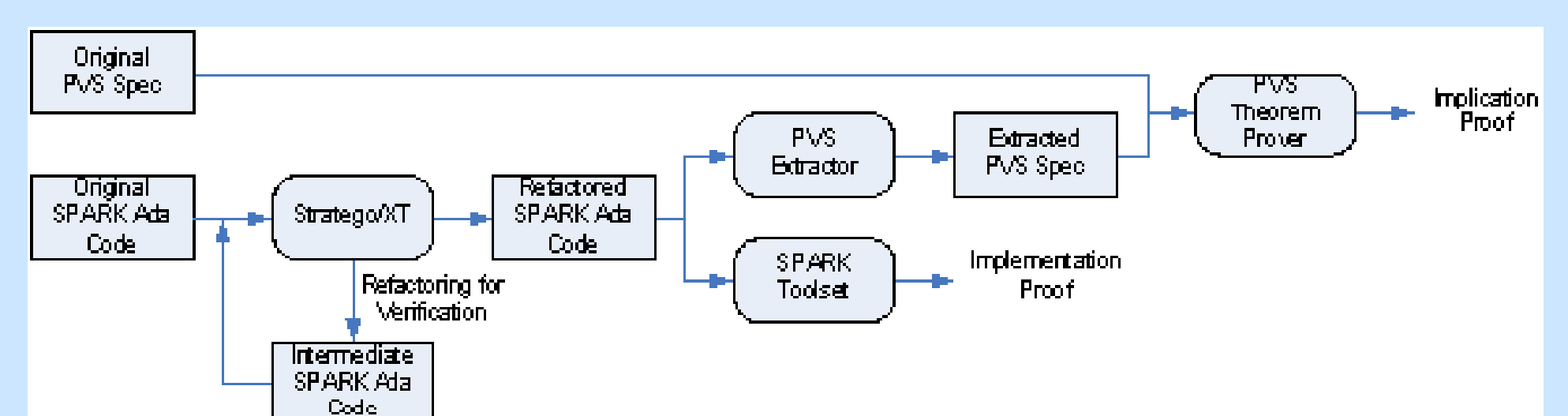
Echo: Verification by Reverse Synthesis

A general mechanism for verifying large software systems

Echo splits verification into two parts. The first verifies an implementation against a low-level specification written using source-code annotations. The second extracts a high-level specification from the implementation with the low-level specification, and proves that it implies the original specification from which the system was built. Semantics-preserving refactorings are applied to the implementation in both parts to reduce the complexity of the verification. Much of the approach is automated. It reduces the verification burden by distributing it over separate tools and techniques, and it addresses both functional correctness and high-level properties at separate levels.



Reverse Synthesis



Echo Prototype Toolset

Approach and Impact

New Approach

- Extract specification from annotated code
- Verify code against annotations
- Verify extracted specification implies original

Research Impact

- Practical formal verification
- Does not interfere with development
- Easier proofs of security properties

Preliminary evaluation by formal verification of a publicly available optimized implementation of the Advanced Encryption Standard (AES):

Artifacts:

NIST official specification translated to PVS.
ANSI C implementation translated to SPARK Ada

Alternatives:

Verify original implementation
Retain just loop rerolling

Refactoring:

- (1) Reverse loop unrolling
- (2) Reverse data word packing
- (3) Reverse table loop-up function implementation
- (4) Reverse function inlining

Equipment:

1.0GHz UltraSparc IIIi
2GB RAM

Implementation Proof:

Completed by SPARK Ada tools.
136 of 144 verification conditions discharged automatically

Original Program:

SPARK toolset ran out of heap space
Failed to complete the proof

Implication Proof:

Completed by PVS theorem prover
More than half the proof obligations discharged by (grind)

Just Loop Rerolling:

15 Mbytes of verification conditions
Two hours of CPU time

For detailed information:

Echo: A Practical Approach to Formal Verification (FMICS 2005)
Formal Verification By Reverse Synthesis (Submitted to SAFECOMP 2008)