The SLAM Project: Debugging System Software via Static Analysis
Mining Specification
Role Analysis
Analyzing Security Protocols with Secrecy Types and Logic Programs
Types as Models: Model Checking Message-Passing Programs
Lazy Abstraction
Bisimulation Congruences in Safe Ambients
A Uniform Type Structure for Secure Information Flow
Bounding Space Usage of Conservative Garbage Collectors
The Hardness of Cache Conscious Data Placement
Towards Automatic Construction of Staged Compilers
From POPL to the Classroom and Back
CCured: Type-Safe Retrofitting of Legacy Code
An Efficient Profile-Analysis Framework for Data Layout Optimizations
Stochastic Lambda Calculus and Monads of Probability Distributions
Representation Independence, Confinement, and Access Control
Systematic Design of Program Transformations by Abstract Interpretation
Predicate Abstraction for Software Verification
First-Order Theory of Subtyping Constraints
A Type System for Certified Binaries
Functional Logic Overloading
Scalable Formal Design Methods for Asynchronous VLSI
Adaptive Functional Programming
The Strength of Non Size-Increasing Computation
Composing Dataflow Analyses and Transformations
Proving Correctness of Compiler Optimizations by Temporal Logic
Exploiting Prolific Types for Memory Management and Optimizations
Stack Inspection: Theory and Variants
Information Flow Inference for ML
Resource Usage Analysis

Table of Contents provided by Blackwell's Book Services and R.R. Bowker. Used with permission.