

# PerformanceTacticIndex

## compiler tactics

- delayed compilation
- tiered compilation
- on-stack replacement
- delayed reoptimization
- program dependence graph representation
- static single assignment representation

## speculative (profile-based) techniques

- optimistic nullness assertions
- optimistic type assertions
- optimistic type strengthening
- optimistic array length strengthening
- untaken branch pruning
- optimistic N-morphic inlining
- branch frequency prediction
- call frequency prediction

## proof-based techniques

- exact type inference
- memory value inference
- memory value tracking
- constant folding
- reassociation
- operator strength reduction
- null check elimination
- type test strength reduction
- type test elimination
- algebraic simplification
- common subexpression elimination
- integer range typing

## flow-sensitive rewrites

- conditional constant propagation
- dominating test detection
- flow-carried type narrowing
- dead code elimination

## language-specific techniques

- class hierarchy analysis
- devirtualization
- symbolic constant propagation
- autobox elimination
- escape analysis
- lock elision
- lock fusion
- de-reflection

## memory and placement transformation

- expression hoisting
- expression sinking
- redundant store elimination
- adjacent store fusion
- card-mark elimination
- merge-point splitting

## loop transformations

- loop unrolling
- loop peeling
- safepoint elimination
- iteration range splitting
- range check elimination
- loop vectorization

## global code shaping

- inlining (graph integration)
- global code motion
- heat-based code layout
- switch balancing
- throw inlining

### **control flow graph transformation**

- local code scheduling
- local code bundling
- delay slot filling
- graph-coloring register allocation
- linear scan register allocation
- live range splitting
- copy coalescing
- constant splitting
- copy removal
- address mode matching
- instruction peepholing
- DFA-based code generator