Goal: increase software reliability
Approach: specifications for formal, informal reasoning
Problem: specifications are typically absent
  • programmers are reluctant to write them
  • programmers are not very good at writing them
  • too few tools accept them as input
Solution: automatically infer program specifications
Application: Erdös internet agent system from NTT
Progress Through December 2001

Daikon system (tool for generating specifications)
  • 5 new releases at http://pag.lcs.mit.edu/daikon/
  • improve robustness, scale, documentation

Theorem-proving
  • Java language and ESC/Java static checker
    • generated specifications are 90% accurate
    • user study: humans are aided by Daikon's output
  • IOA language and LP theorem-prover
    • detects invariants necessary for proofs about distributed algorithms
    • collaboration with Theory of Distributed Systems group
Research Plan for the Next Six Months

Support tools and extend previous results

Perform experiments directly on NTT's Erdös agent system from NTT

Detect temporal invariants
• describes properties over time
• example: AG(client.register -> AF (client.buy))
  (If a client registers, it eventually buys.)