

Program Comprehension and Reliability Analysis

Atlas Tool and Query Model Refine Technology

EnSoft and Iowa State University

Abstract

Software reliability analysis can be dauntingly difficult for problems such as dead lock detection. Human experts solve these problems by analyzing software artifacts. It requires tracing volumes of code, extracting the important relationships, and filtering out unnecessary details – tasks which are very difficult to perform manually. Atlas is a tool that automates these difficult manual tasks. Atlas is based on the powerful Query-Model-Refine (QMR) program comprehension technology we are developing for retrieving, abstracting, and analyzing complex relationships between program artifacts. The QMR technology provides a flexible and powerful mechanism to design and execute software analysis strategies. Atlas implements the QMR technology by providing a SQL-like syntax for queries, presenting complex relationships as graphs, and enabling their refinement through a variety of graph transformations. Powerful queries can be composed and combined with graph transformations to perform highly useful analysis of software.

Applications Of QMR Technology

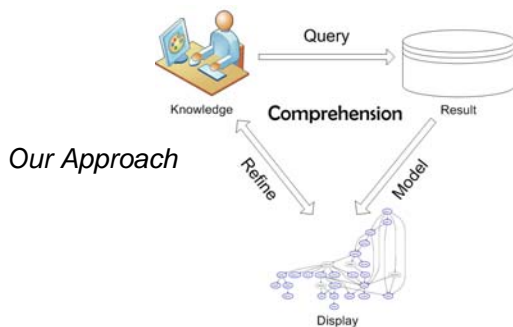
Architecture extraction, auditing safety-critical software, defect analysis, interactive automatic parallelization, and business rule extraction.

Intelligence Amplifying

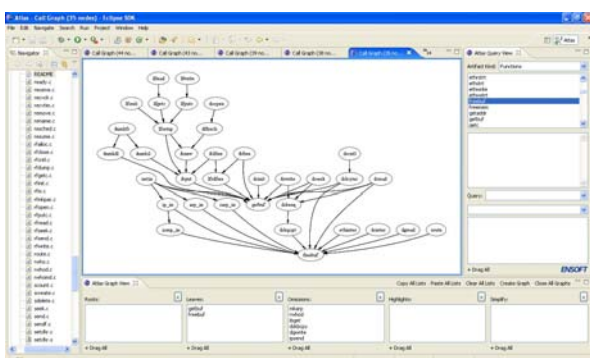
If indeed our objective is to build computer systems that solve very challenging problems, my thesis is that IA > AI, that is, that intelligence amplifying systems can, at any given level of available systems technology, beat AI systems. That is, a machine and a mind can beat a mind-imitating machine working by itself.

--Fred Brooks

Program Comprehension Process



Tool Interface



Why QMR Technology

- The time spent on program comprehension can be drastically reduced by using a QMR tool.
 - In industry, software engineers spend a lot of time on program comprehension due to factors such as inadequate documentation and code decay from software aging.
 - QMR technology can be used to improve the efficiency of testing and requirements gathering activities.
 - QMR technology can be used for training software engineers.
- Contact: Suraj Kothari, kothari@iastate.edu

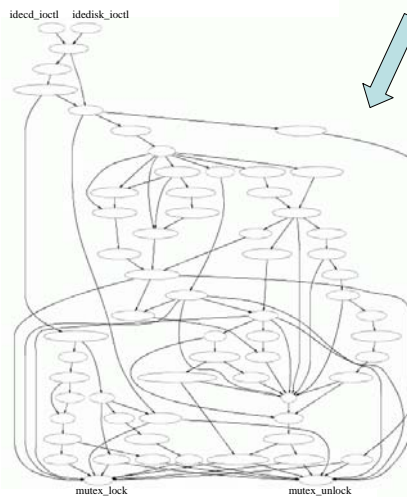
Deadlock Detection using Atlas – an analysis of Linux code

Queries:

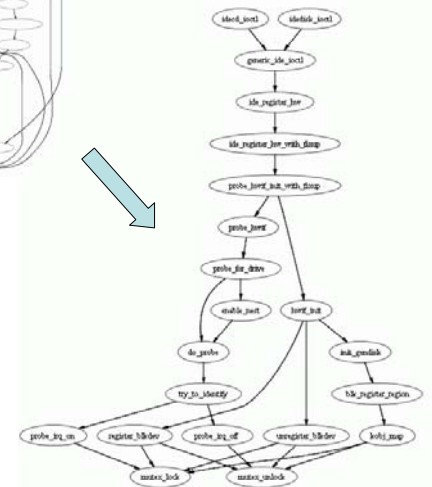
```
#x = call(mutex_lock)
#y = call(mutex_unlock)
#z = #x minus #y
#a = roots(#z)
```

For a subsystem analysis, choose “idecd_ioctl” & “idedisk_ioctl” as the roots “mutex_lock” & “mutex_unlock” as the leaves
Atlas create the graph on the right

#a, #x, #y, #z are set of functions



Omit the “device_add” case, Atlas simplifies graph



Omit “ide_unregister” operation
Atlas further simplifies the graph

Brief History

- ParAgent, an interactive automatic parallelization tool at ISU – 1996
- TotalInsight, a COBOL tool for business rule extraction at EnSoft – 2004
- EnSoft built a tool for Rockwell for auditing avionics software – 2005
- C-Vision a research prototype at ISU – 2006
- Atlas built at EnSoft – 2007

An Outsider’s View

□ “I think this is a real breakthrough – and I am now a confirmed advocate of program reading. I am hoping to work with Prof Kothari to do some more stuff with this ...” – a blog by a Microsoft manager
<http://port25.technet.com/archive/2006/11/16/learning-to-read.aspx>