

Representation-Independent Program Analysis

Michelle Mills Strout

John Mellor-Crummey (Rice)

Paul Hovland (Argonne and University of Chicago)

PASTE 2005

September 6, 2005



OpenAnalysis



- Problem: Insufficient analysis support in existing compiler infrastructures due to non-transferability of analysis implementations
- Decouples analysis algorithms from intermediate representations (IRs) by developing analysis-specific interfaces
- Analysis reuse across compiler infrastructures
 - Enable researchers to leverage prior work
 - Enable direct comparisons amongst analyses
 - Increase the impact of program analysis research

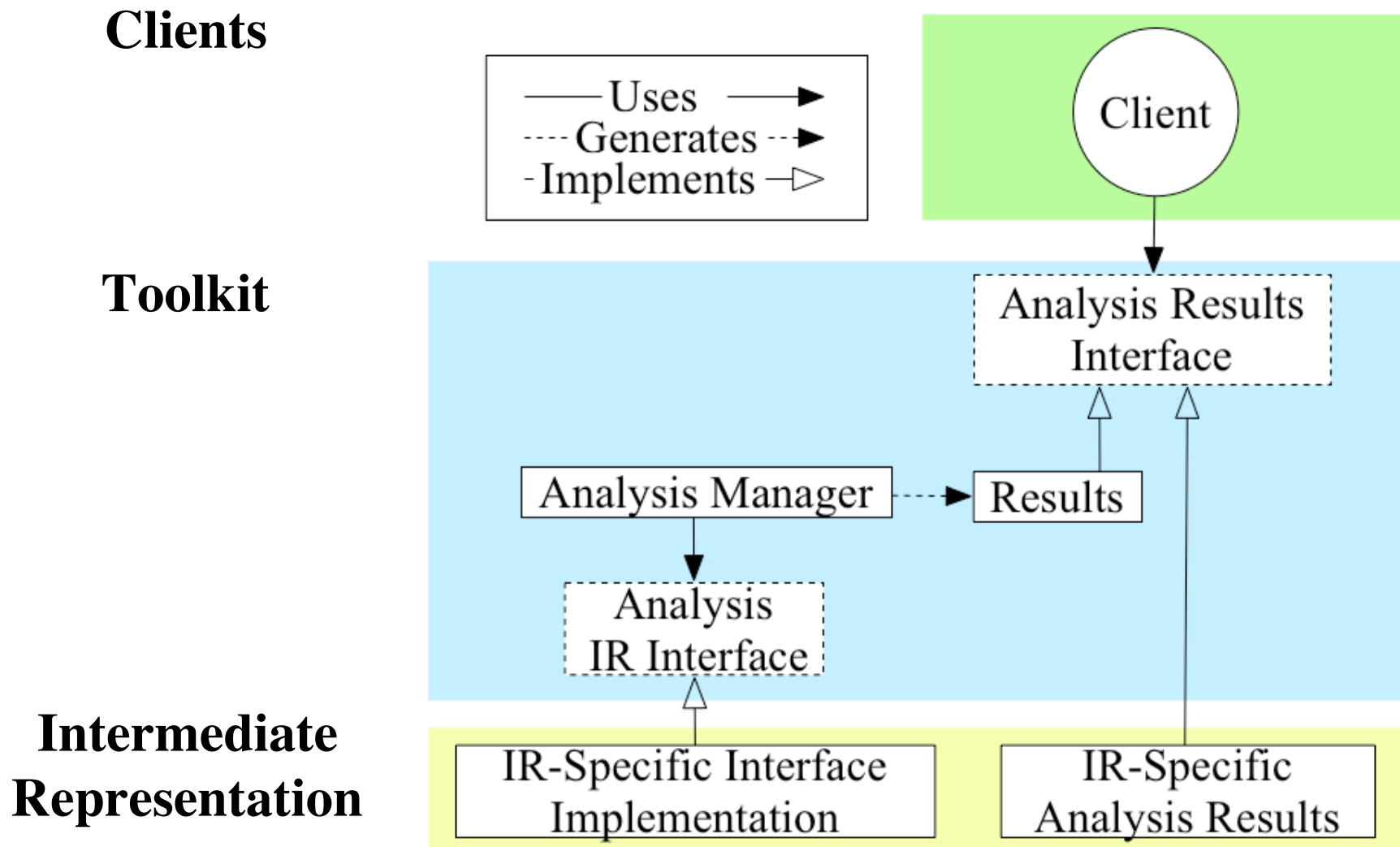
Example Projects that Found Analysis Support Wanting

- Run-time reordering transformations [Strout Thesis Work]
 - improves performance of irregular applications
 - needs data dependence relations with uninterpreted functions in compiler infrastructure for C
- Caching policies in parallel file systems [Vilayannur]
 - estimates memory references in perfect loops with constant loop bounds
 - symbolic analysis would provide a better estimate, but was unavailable in the infrastructure being used
- Hancock at AT&T [Fisher and Rogers]
 - domain-specific language for statisticians to manipulate transactions using a familiar notation
 - two researchers could not do the enormous amount of work to support domain-specific analysis of Hancock

Root Causes of Analysis Support Problem

- Analysis development typically occurs in research compilers that have non-robust language support
- Compiler infrastructures are difficult to support long-term and build robustly in academia
- Some infrastructures have multiple branches because no mechanism for central updates
- Most fundamental problem is that all compiler infrastructures integrate program analysis with program representation

Software Architecture for OpenAnalysis



Analysis-Specific Interfaces in OpenAnalysis

- Represent imperative programming constructs with opaque handles: procedures, statements, memory references, expressions, operations, constants, etc.
- Make queries on handles
- Example: Control-flow graph analysis

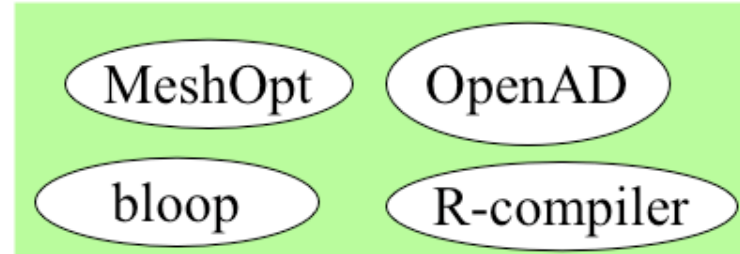
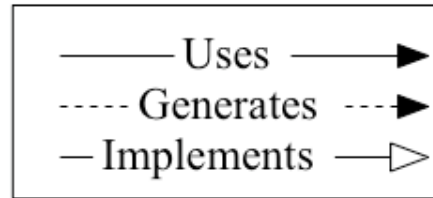
```
IRStmtType getStmtType (StmtHandle)
```

```
SIMPLE, LOOP, STRUCT_TWOWAYCONDITIONAL,
```

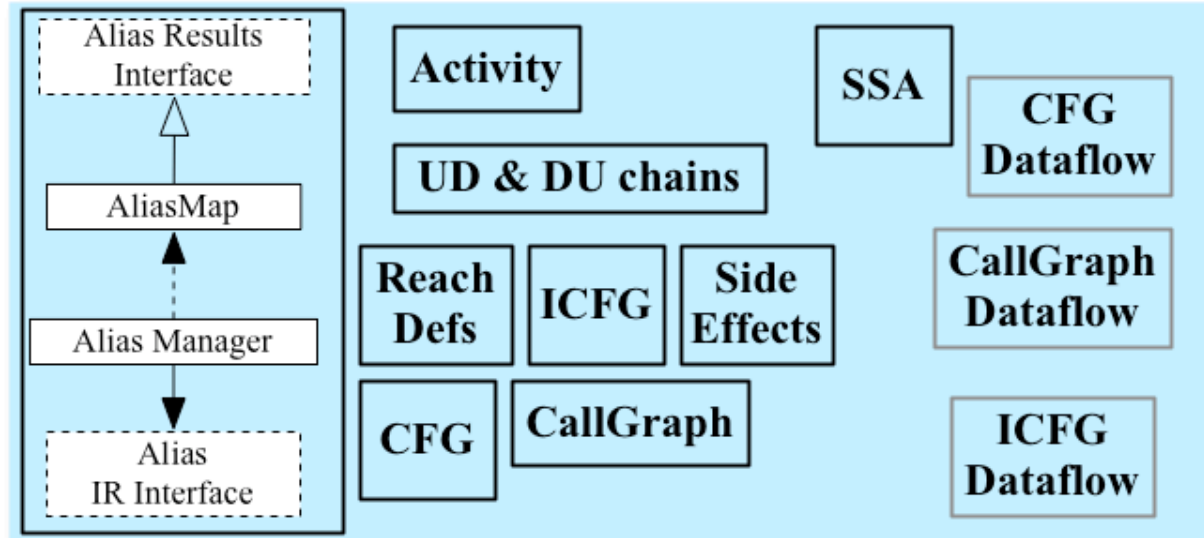
```
...
```

OpenAnalysis Status

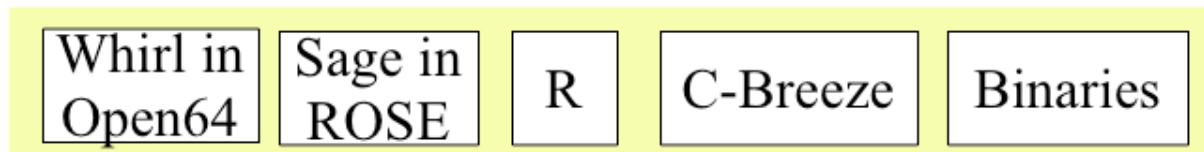
Clients



Toolkit



Intermediate Representation



Key Differences from Related Work

- Analysis frameworks, some IR-specific analysis
 - interprocedural analysis, FIAT [Hall et al 95]
 - data-flow analysis, Sharlit [Tijang & Hennessy 92]
- Data-flow analysis generator, PAG [Alt and Martin 95]
 - must specify how to access entire IR
- Conversion to a canonical IR
 - eg. data-flow analysis, requires aliasing info [Moonen 97]
 - difficult to map results back to source IR
 - must specify a mapping for entire IR
- Analysis tools with adaptation level
 - GENOA [Devanbu 92], monolithic layer, assumes AST
 - StarTool [Hayes et al 00], analysis-centric, assumes AST

Evaluating OpenAnalysis

- **Ease of Use:** how easy is it to ...
 - implement an analysis-specific IR interface?
 - contribute an analysis implementation to the toolkit?
 - use analysis results?
- **Coverage**
 - how many important analysis algorithms are expressible?
 - how many imperative language features can be modeled?
- **Accuracy**
 - how much is lost due to IR independence?
- **Efficiency**
 - what is the performance cost of the extra layer of abstraction?

Conclusions

- Language-independent program analysis enables sharing between and within compiler infrastructures
- Analysis-specific, IR-independent interfaces are the key
 - represent complex language constructs with abstractions that are basic to all imperative programming languages
 - design the interface to satisfy a broad range of implementations
- The OpenAnalysis toolkit is being actively used and further developed within the context of multiple projects (clients) for multiple IRs

Collaborators

- Paul Hovland (Argonne)
- John Mellor-Crummey (Rice University)
- Barbara Kreaseck (La Sierra University)
- Jean Utke (University of Chicago)
- Nathan Tallent (Rice University)
- Beata Winnicka (Argonne)
- Boyana Norris (Argonne)
- Brian White (Cornell)
- Dan Quinlan (LLNL)