Bauhaus
–
A Tool Suite for Program Understanding and Reverse Engineering

Department of Programming Languages
Institute for Software Technology
University of Stuttgart

07/07/06

Content

1. Introduction
2. Infrastructure
3. Analyses and Tools
4. Summary

Bauhaus - Introduction

The Bauhaus Project

Goal:

- Provide methods, techniques and tools for program understanding on all levels of abstraction
  - source code $\longleftrightarrow$ architecture level
- Improve quality and efficiency of software development processes

The Bauhaus Project

Goal:

- Provide methods, techniques and tools for program understanding on all levels of abstraction
  - source code $\longleftrightarrow$ architecture level
- Improve quality and efficiency of software development processes

The Bauhaus Project

Goal:

- Provide methods, techniques and tools for program understanding on all levels of abstraction
  - source code $\longleftrightarrow$ architecture level
- Improve quality and efficiency of software development processes

Bauhaus - Infrastructure

Infrastructure

Key idea: Source code as most important source of information

$\longrightarrow$ Compiler technology

Infrastructure

Key idea: Source code as most important source of information

$\longrightarrow$ Compiler technology

Applications:

- Source code navigation
- Anomaly detection
- Architecture recovery and validation
- Quality assessment

Infrastructure

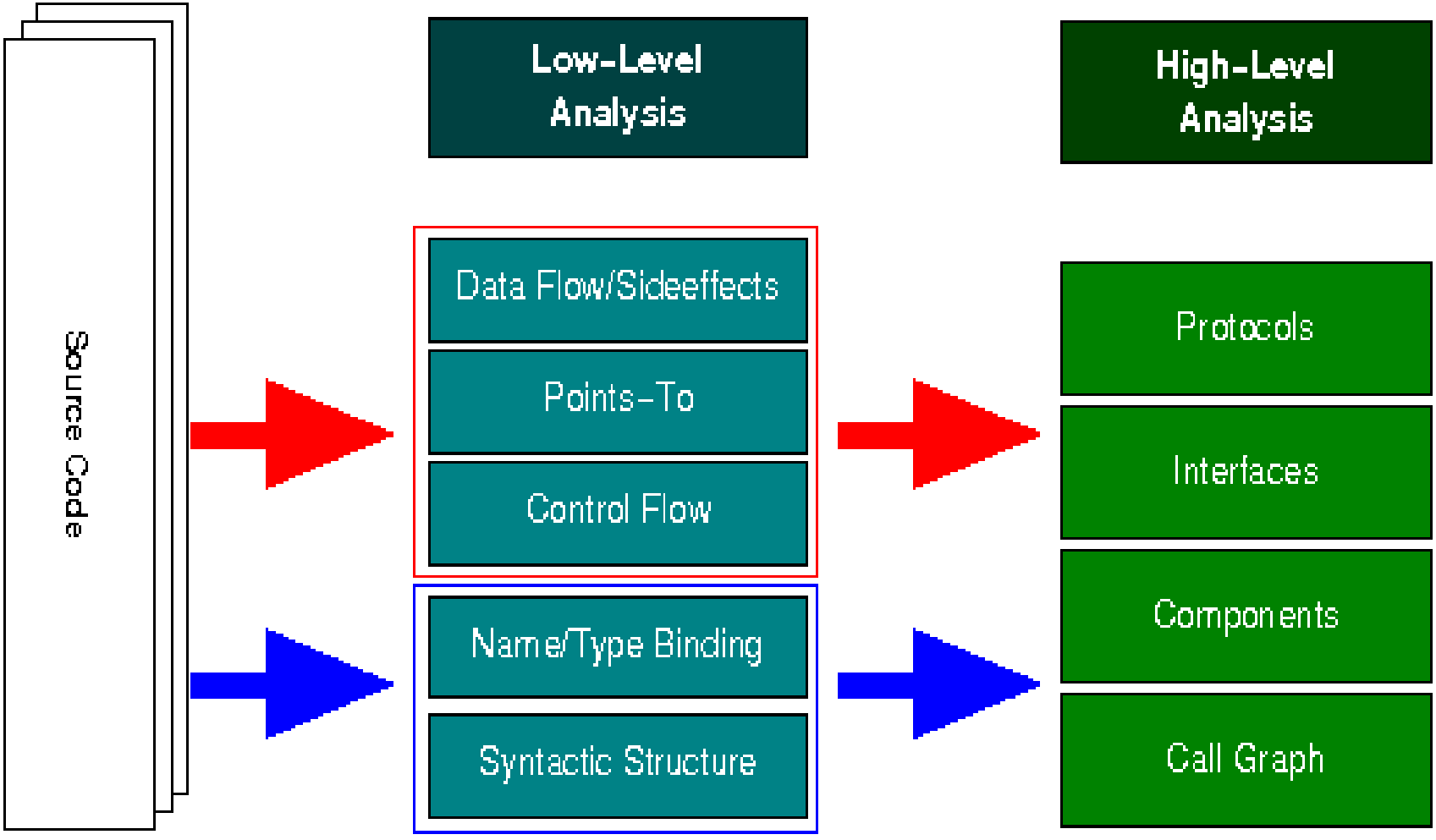
Key idea: Source code as most important source of information

$\longrightarrow$ Compiler technology

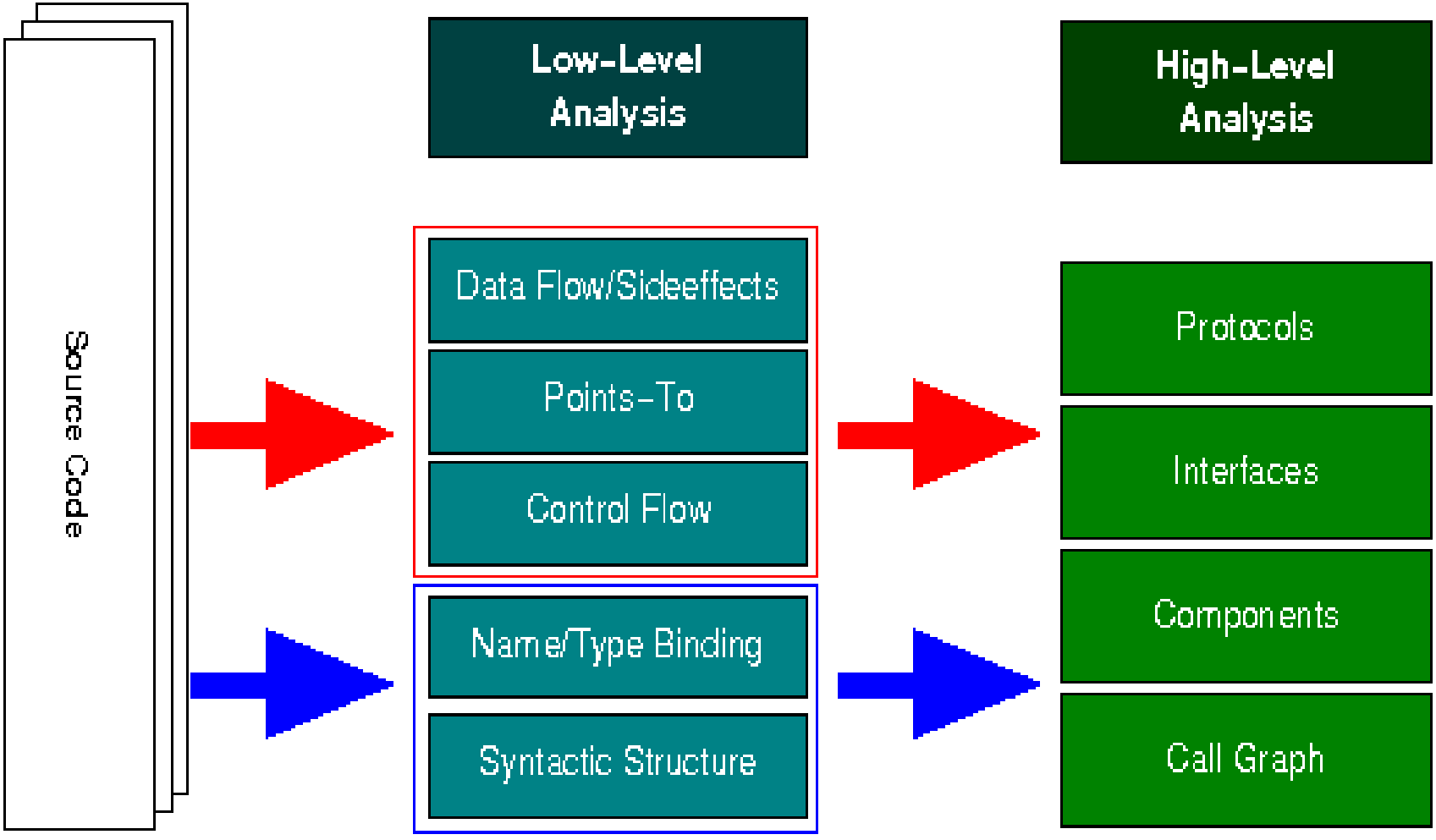
Applications:

- Source code navigation
- Anomaly detection
- Architecture recovery and validation
- Quality assessment

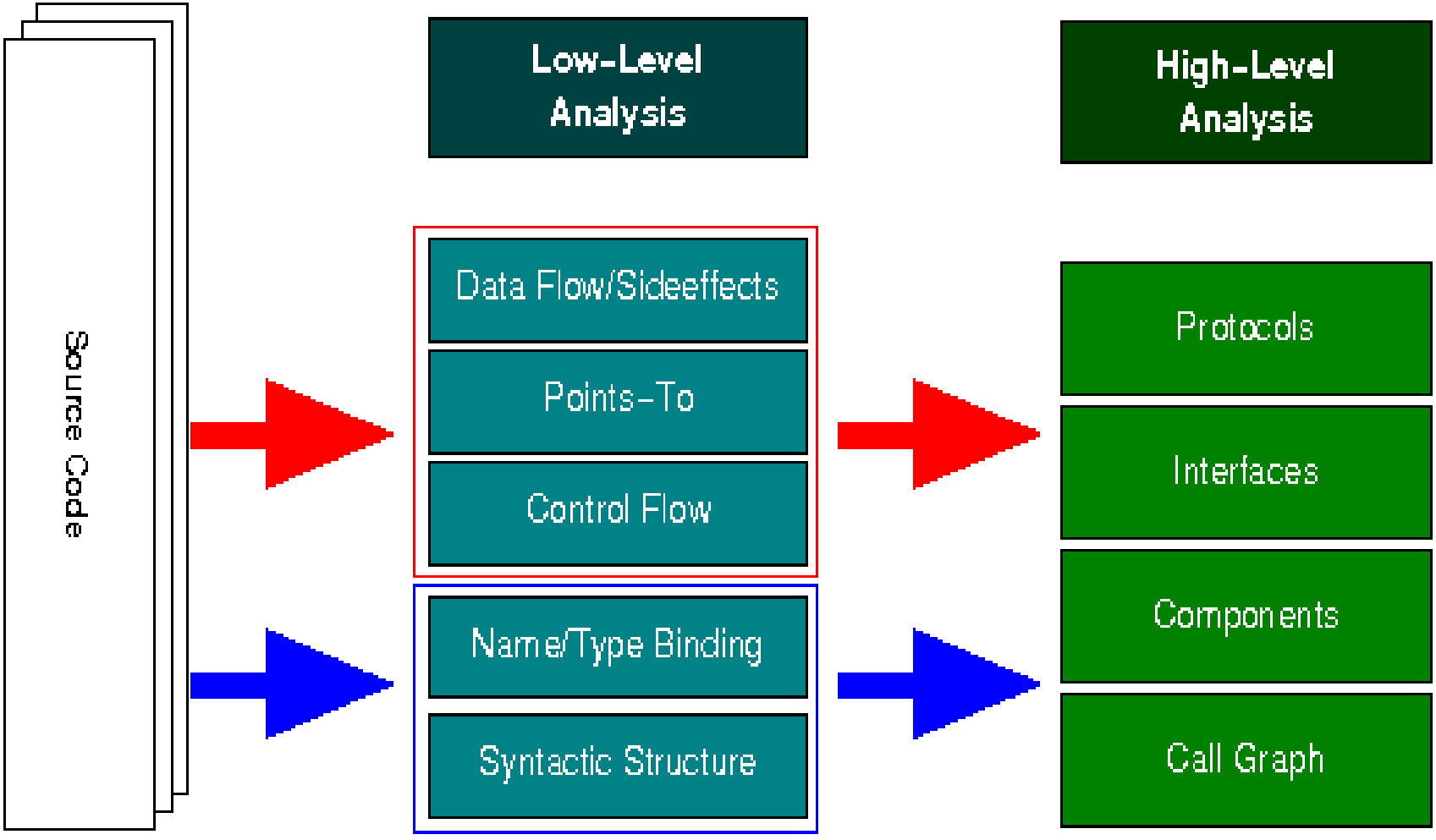
Infrastructure



Infrastructure



Infrastructure



Infrastructure – Technical Details

Tools:

- 124 tools and analyses
- Wide range of high quality tools for program understanding and reverse enginnering
- Mature front ends for the analysis of C and C++
- Research prototypes for Java, Ada and Cobol
- 2 intermediate representations for high and low level program understanding
- Special for tools for visualisation

Infrastructure – Technical Details

Tools:

- 124 tools and analyses
- Wide range of high quality tools for program understanding and reverse enginnering
- Mature front ends for the analysis of C and C++
- Research prototypes for Java, Ada and Cobol
- 2 intermediate representations for high and low level program understanding
- Special for tools for visualisation

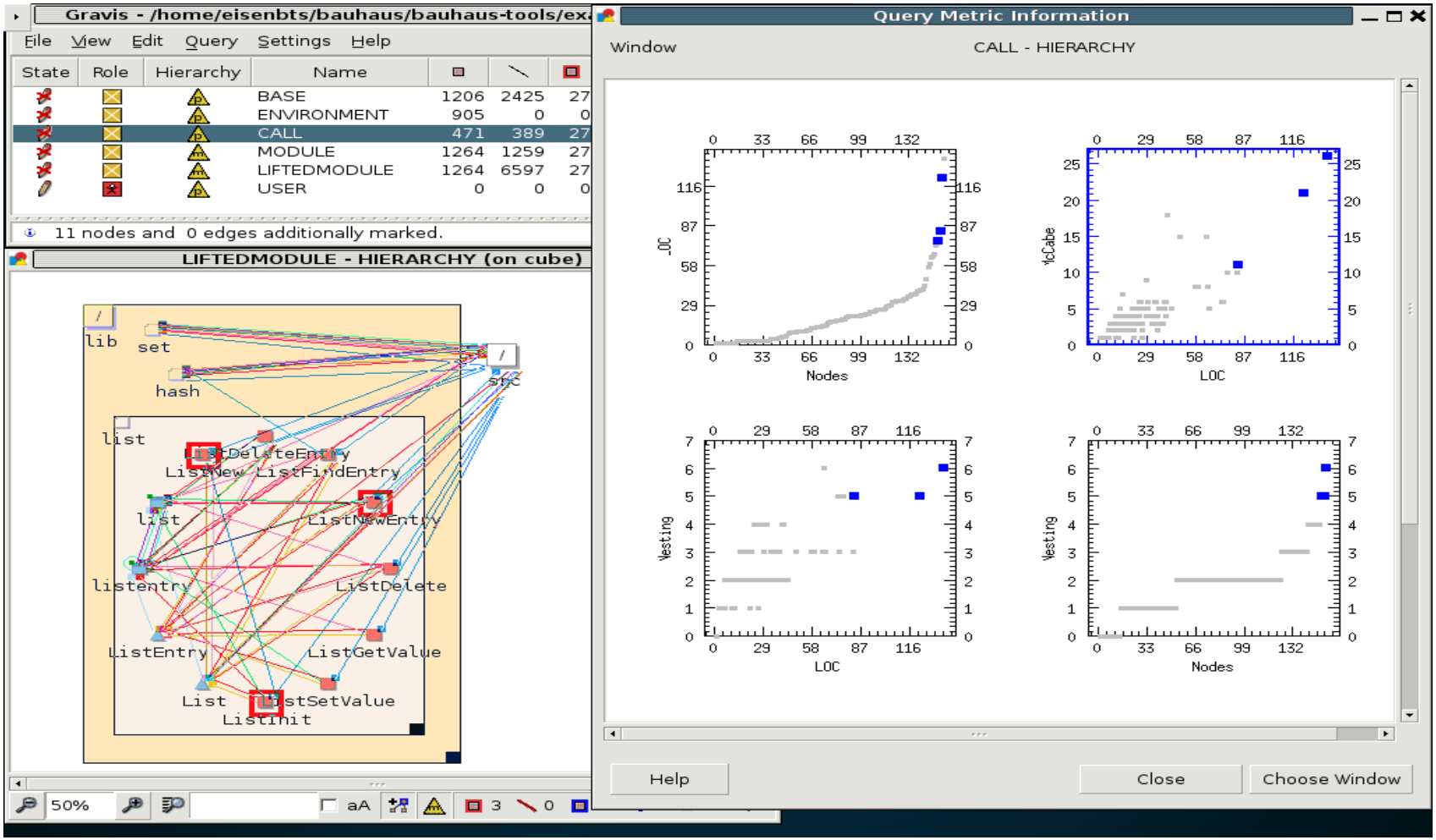
Infrastructure – Technical Details

Tools:

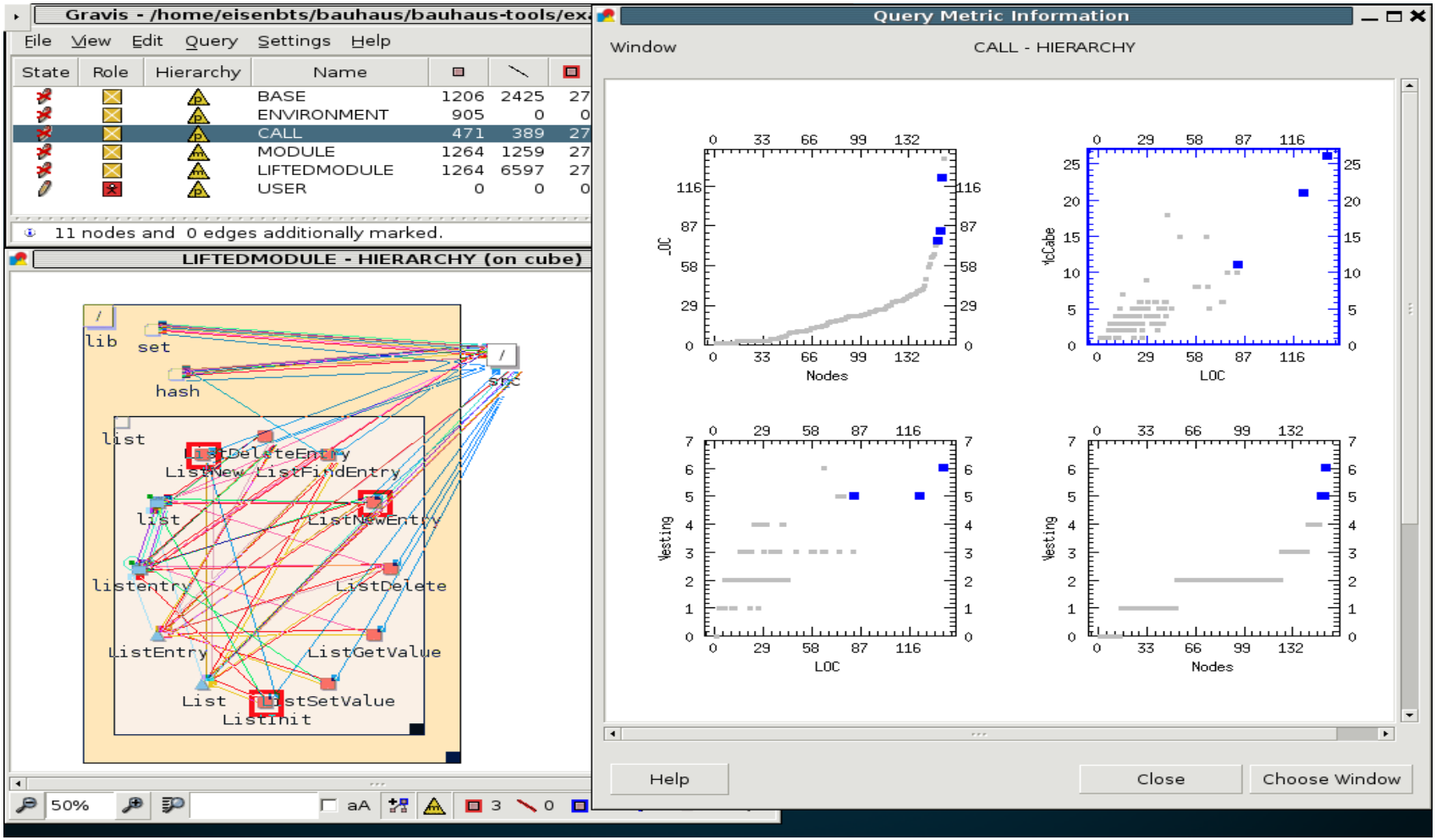
- 124 tools and analyses
- Wide range of high quality tools for program understanding and reverse enginnering
- Mature front ends for the analysis of C and C++
- Research prototypes for Java, Ada and Cobol
- 2 intermediate representations for high and low level program understanding
- Special for tools for visualisation

Software Quality Assessment

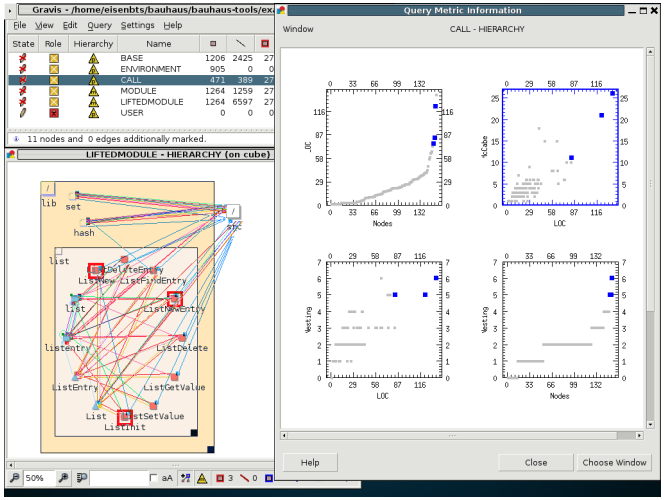
Software Quality – Metrics



Software Quality – Metrics

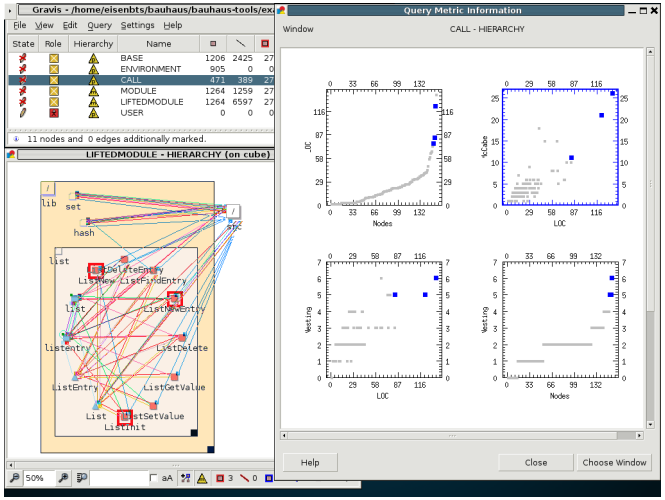


Software Quality – Metrics



- Code level: lines of code, Halstead, maximum nesting, cyclomatic complexity
- Architecture level: number of methods, classes, and units, coupling, cohesion
- Derived metrics (Python Scripting): average number of methods per class, classes per unit, maintainability index (Coleman, Oman)

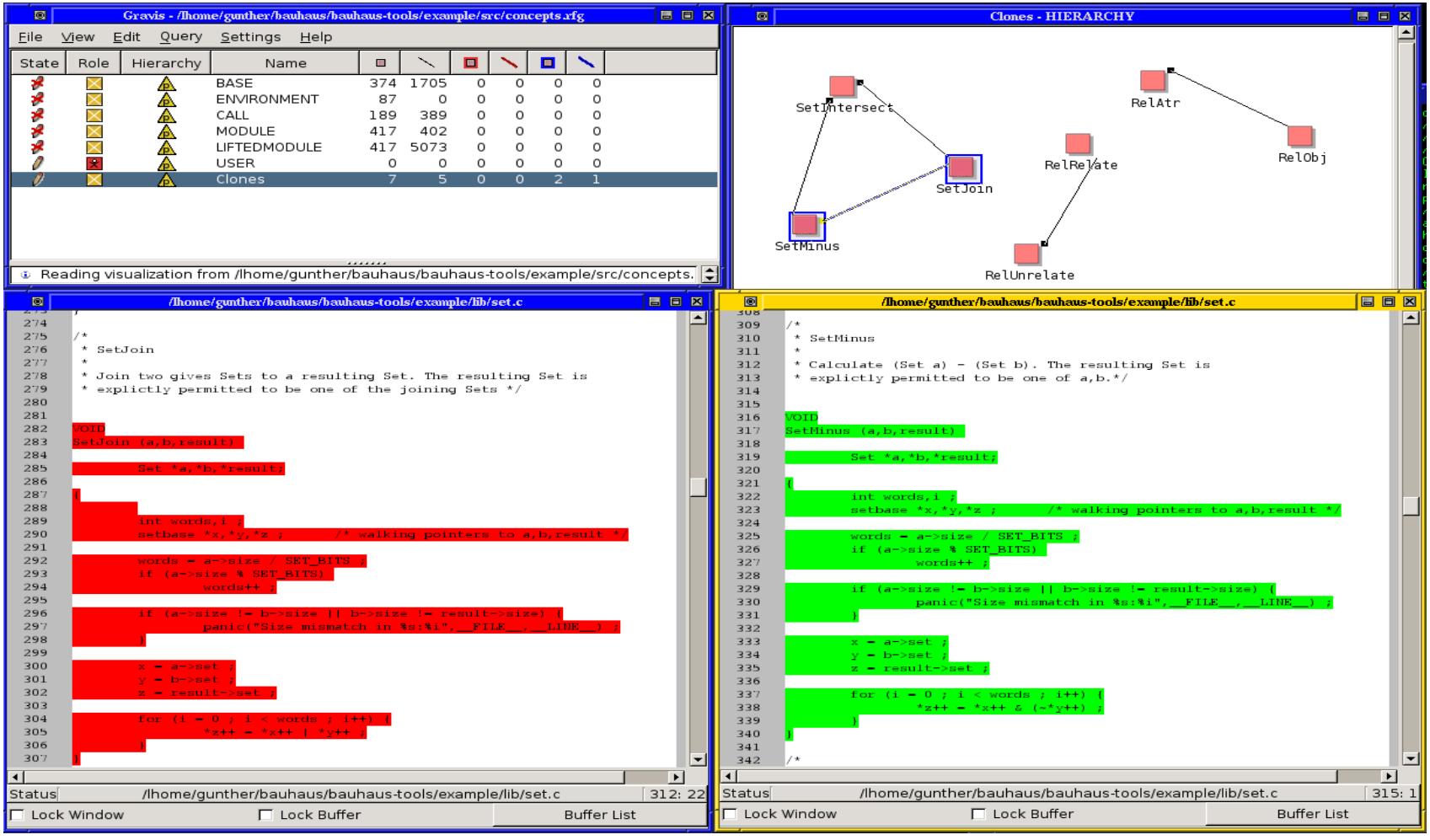
Software Quality – Metrics



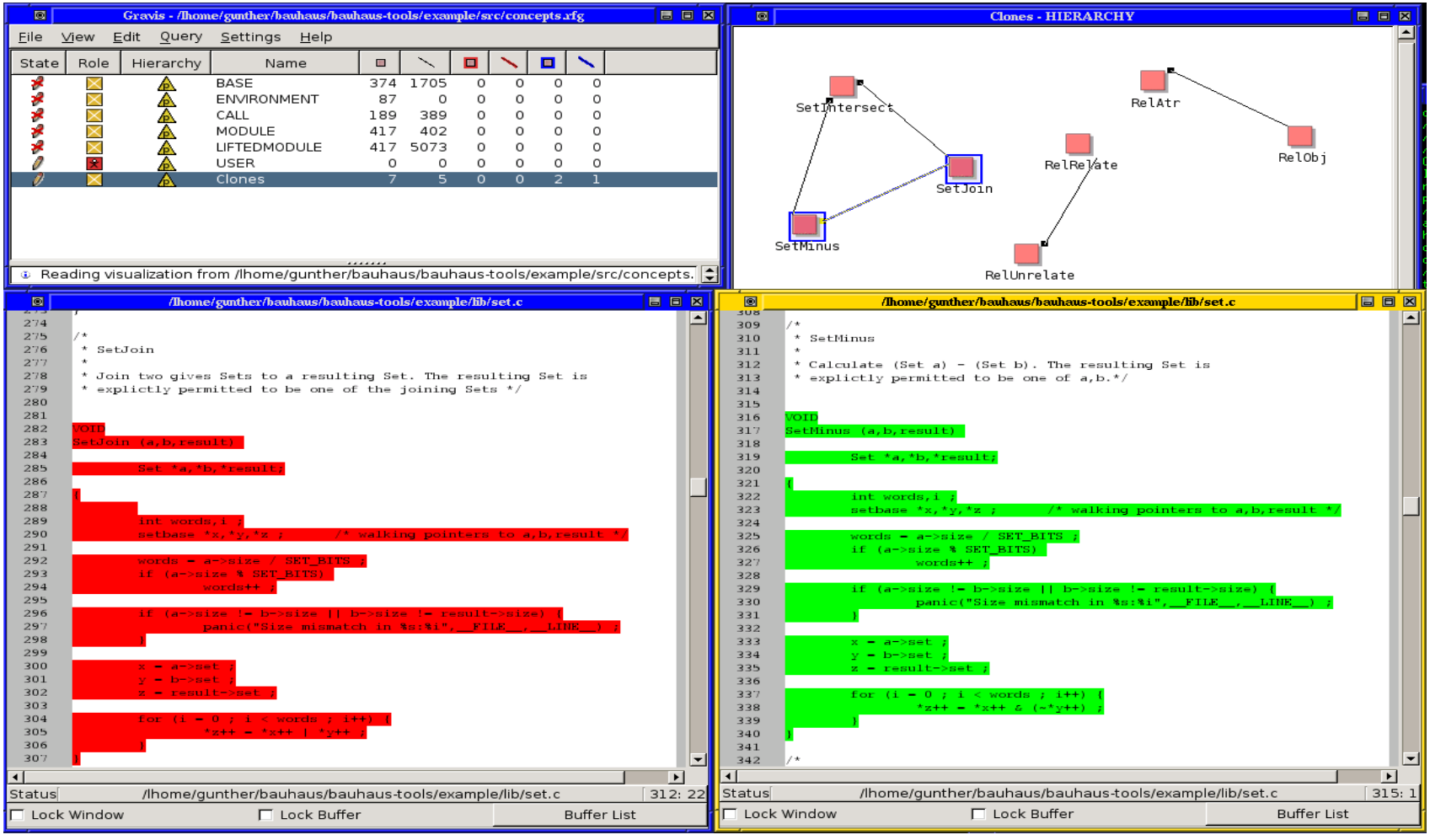
- Code level: lines of code, Halstead, maximum nesting, cyclomatic complexity
- Architecture level: number of methods, classes, and units, coupling, cohesion
- Derived metrics (Python Scripting): average number of methods per class, classes per unit, maintainability index (Coleman, Oman)

Find Code Duplication

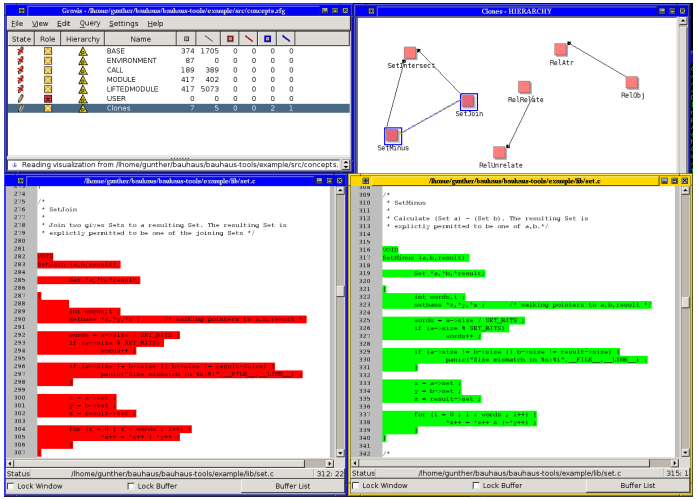
Clones


Gravis - /lhome/gunther/bauhaus/bauhaus-tools/example/src/concepts.rfg
File View Edit Query Settings Help
State Role Hierarchy Name
BASE 374 1705 0 0 0 0
ENVIRONMENT 87 0 0 0 0 0
CALL 189 389 0 0 0 0
MODULE 417 402 0 0 0 0
LIFTEDMODULE 417 5073 0 0 0 0
USER 0 0 0 0 0 0
Clones 7 5 0 0 2 1
Reading visualization from /lhome/gunther/bauhaus/bauhaus-tools/example/src/concepts.
Clones - HIERARCHY
SetIntersect
SetJoin
SetMinus
RelAtr
RelRelate
RelObj
RelUnrelate
/lhome/gunther/bauhaus/bauhaus-tools/example/lib/set.c
/*
* SetJoin
*
* Join two gives Sets to a resulting Set. The resulting Set is
* explictly permitted to be one of the joining Sets */
VOID
SetJoin (a,b,result)
Set *a,*b,*result;
{
int words,i ;
setbase *x,*y,*z ; /* walking pointers to a,b,result */
words = a->size / SET_BITS ;
if (a->size % SET_BITS)
words++ ;
if (a->size != b->size || b->size != result->size) {
panic("Size mismatch in %s:%i",__FILE__,__LINE__) ;
}
x = a->set ;
y = b->set ;
z = result->set ;
for (i = 0 ; i < words ; i++) {
*z++ = *x++ | *y++ ;
}
}
Status /lhome/gunther/bauhaus/bauhaus-tools/example/lib/set.c 312: 22
Lock Window Lock Buffer Buffer List
/lhome/gunther/bauhaus/bauhaus-tools/example/lib/set.c
/*
* SetMinus
*
* Calculate (Set a) - (Set b). The resulting Set is
* explictly permitted to be one of a,b.*/
VOID
SetMinus (a,b,result)
Set *a,*b,*result;
{
int words,i ;
setbase *x,*y,*z ; /* walking pointers to a,b,result */
words = a->size / SET_BITS ;
if (a->size % SET_BITS)
words++ ;
if (a->size != b->size || b->size != result->size) {
panic("Size mismatch in %s:%i",__FILE__,__LINE__) ;
}
x = a->set ;
y = b->set ;
z = result->set ;
for (i = 0 ; i < words ; i++) {
*z++ = *x++ & (~*y++) ;
}
}
/*
Status /lhome/gunther/bauhaus/bauhaus-tools/example/lib/set.c 315: 1
Lock Window Lock Buffer Buffer List

Clones


Gravis - /lhome/gunther/bauhaus/bauhaus-tools/example/src/concepts.rfg
File View Edit Query Settings Help
State Role Hierarchy Name
BASE 374 1705 0 0 0 0
ENVIRONMENT 87 0 0 0 0 0
CALL 189 389 0 0 0 0
MODULE 417 402 0 0 0 0
LIFTEDMODULE 417 5073 0 0 0 0
USER 0 0 0 0 0 0
Clones 7 5 0 0 2 1
Reading visualization from /lhome/gunther/bauhaus/bauhaus-tools/example/src/concepts.
Clones - HIERARCHY
SetIntersect
SetJoin
SetMinus
RelAtr
RelRelate
RelObj
RelUnrelate
/lhome/gunther/bauhaus/bauhaus-tools/example/lib/set.c
/*
* SetJoin
*
* Join two gives Sets to a resulting Set. The resulting Set is
* explictly permitted to be one of the joining Sets */
VOID
SetJoin (a,b,result)
Set *a,*b,*result;
{
int words,i ;
setbase *x,*y,*z ; /* walking pointers to a,b,result */
words = a->size / SET_BITS ;
if (a->size % SET_BITS)
words++ ;
if (a->size != b->size || b->size != result->size) {
panic("Size mismatch in %s:%i",__FILE__,__LINE__) ;
}
x = a->set ;
y = b->set ;
z = result->set ;
for (i = 0 ; i < words ; i++) {
*z++ = *x++ | *y++ ;
}
}
Status /lhome/gunther/bauhaus/bauhaus-tools/example/lib/set.c 312: 22
Lock Window Lock Buffer Buffer List
/lhome/gunther/bauhaus/bauhaus-tools/example/lib/set.c
/*
* SetMinus
*
* Calculate (Set a) - (Set b). The resulting Set is
* explictly permitted to be one of a,b.*/
VOID
SetMinus (a,b,result)
Set *a,*b,*result;
{
int words,i ;
setbase *x,*y,*z ; /* walking pointers to a,b,result */
words = a->size / SET_BITS ;
if (a->size % SET_BITS)
words++ ;
if (a->size != b->size || b->size != result->size) {
panic("Size mismatch in %s:%i",__FILE__,__LINE__) ;
}
x = a->set ;
y = b->set ;
z = result->set ;
for (i = 0 ; i < words ; i++) {
*z++ = *x++ & (~*y++) ;
}
}
/*
Status /lhome/gunther/bauhaus/bauhaus-tools/example/lib/set.c 315: 1
Lock Window Lock Buffer Buffer List

Clones

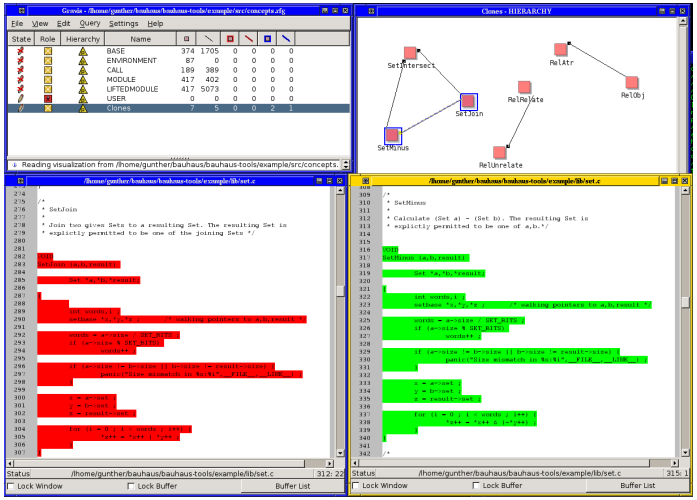


Token-based vs IML-based clone analysis

Types of duplication:

- Type 1: exact copy
- Type 2: copy with consistent substitution
- Type 3: additional insertions and deletions

Clones



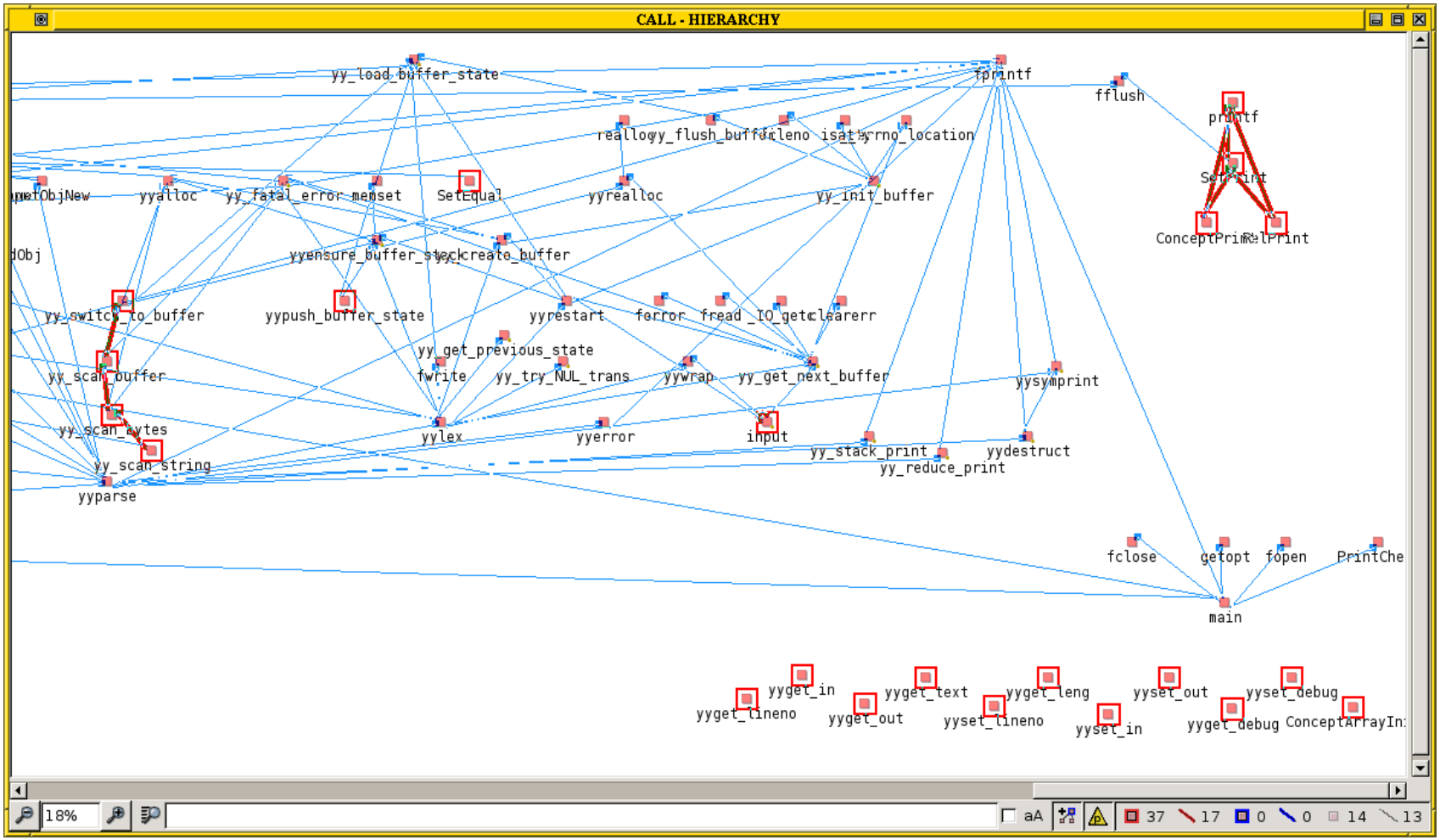
Token-based vs IML-based clone analysis

Types of duplication:

- Type 1: exact copy
- Type 2: copy with consistent substitution
- Type 3: additional insertions and deletions

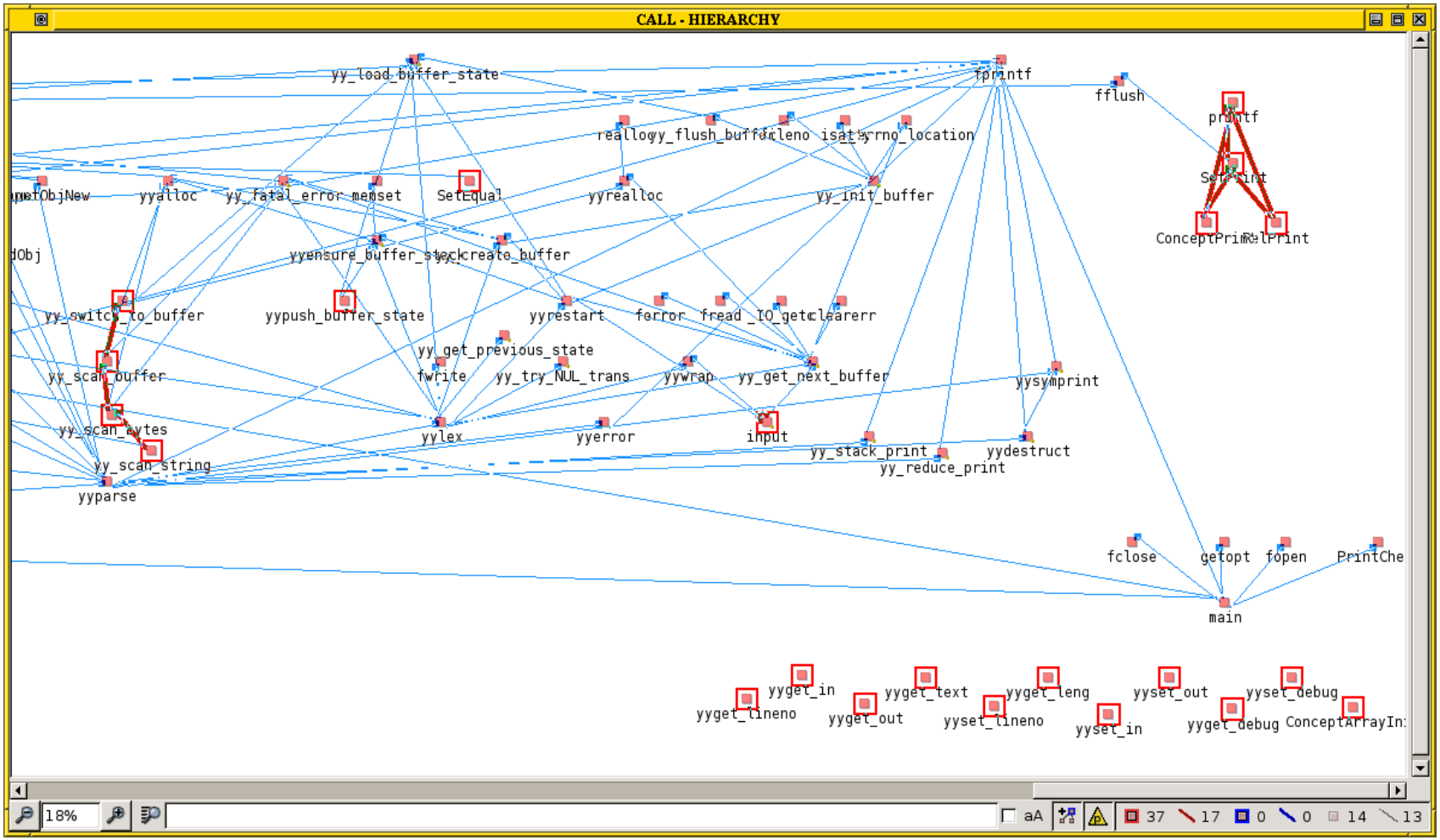
Find Dead Code

Dead Code



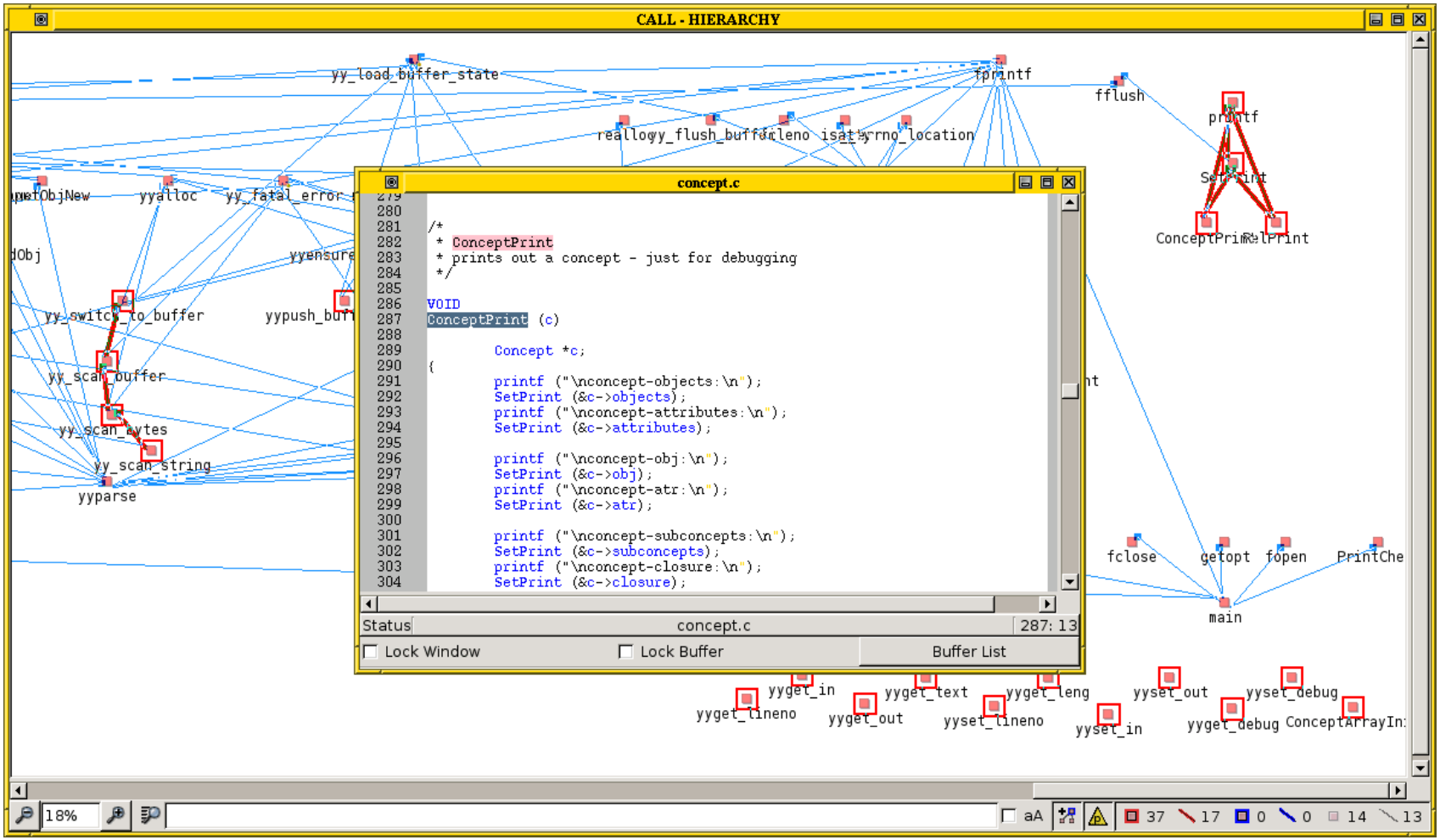
Dead code: Routines that are not reachable from subroutine main

Dead Code



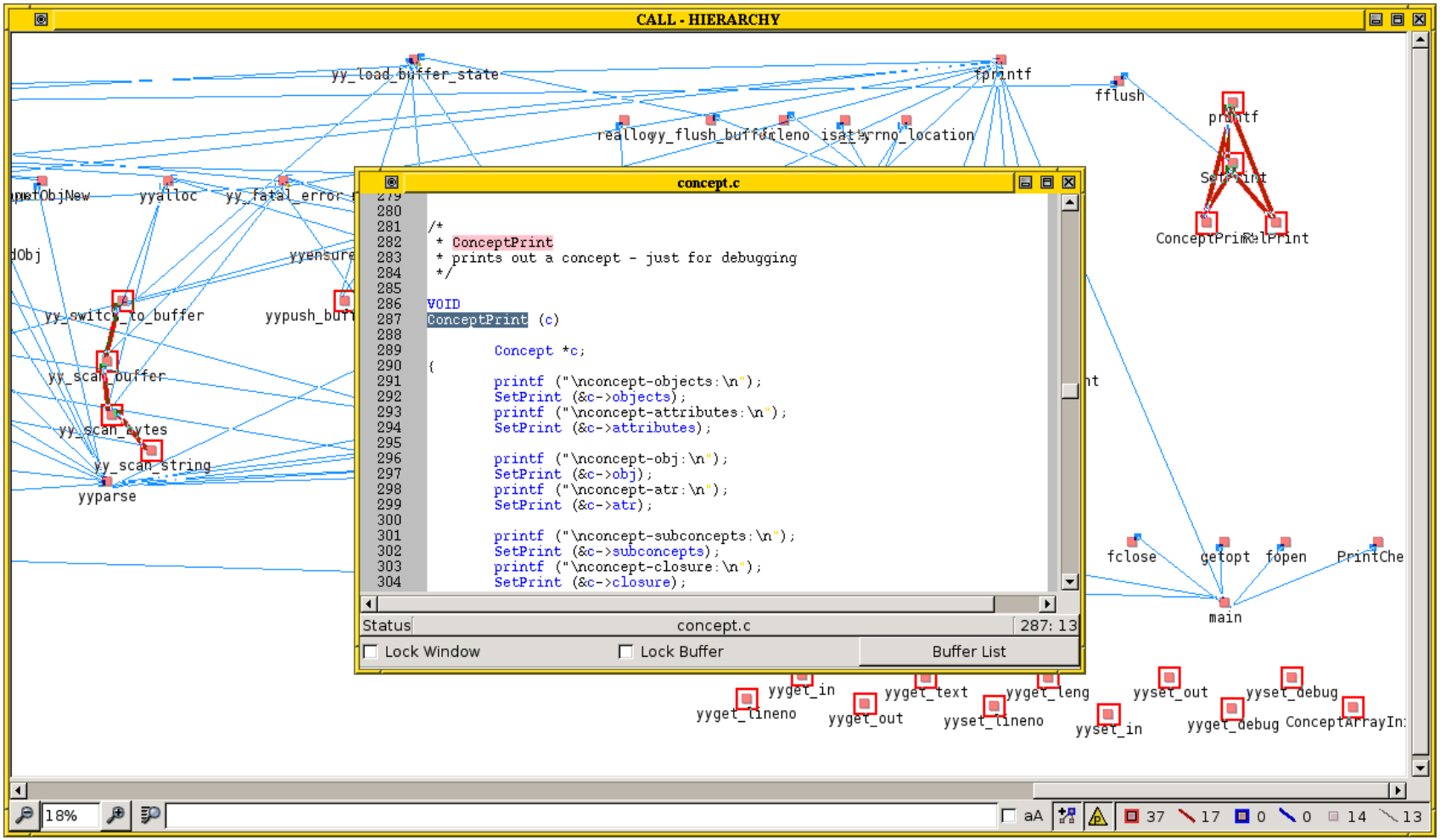
Dead code: Routines that are not reachable from subroutine main

Dead Code



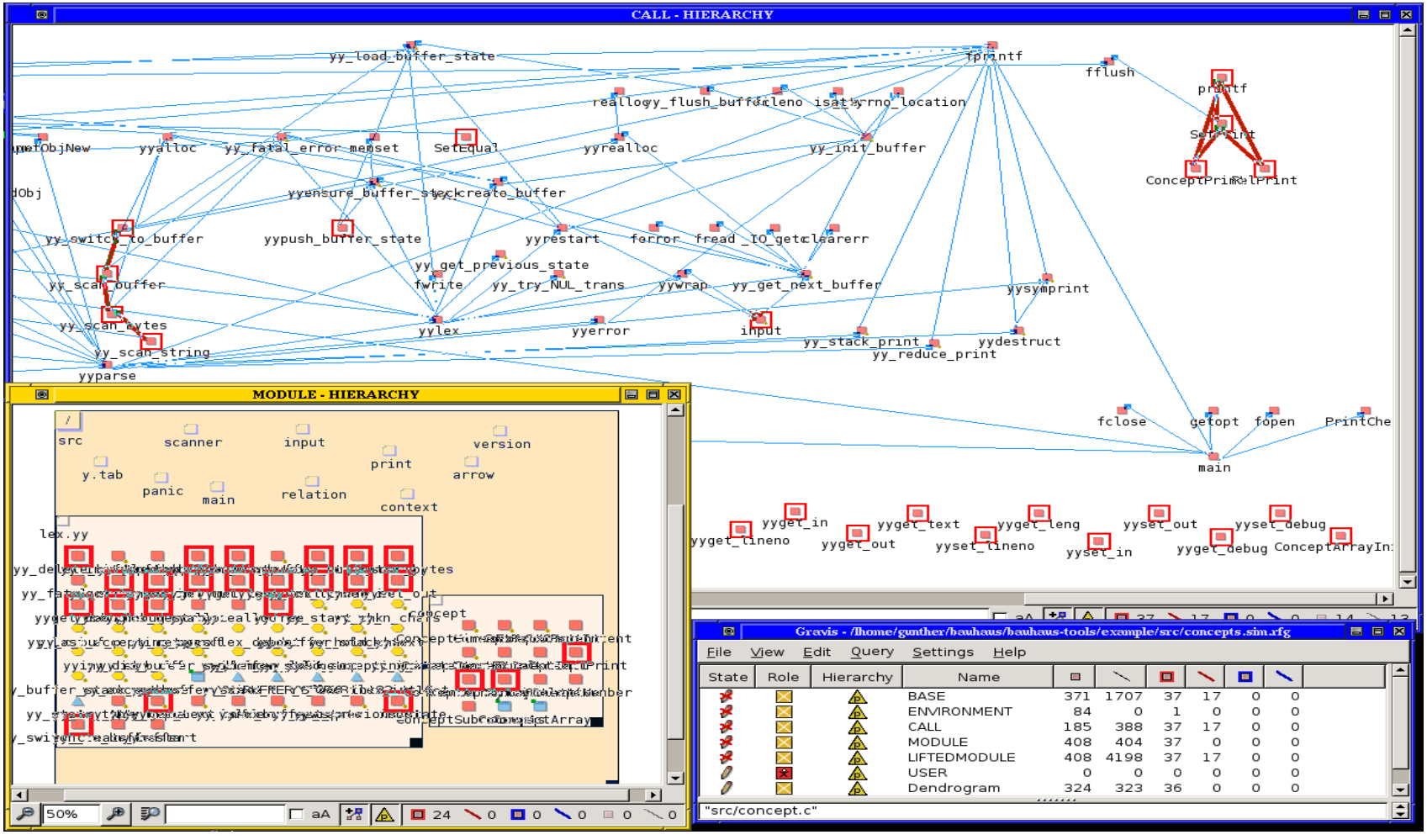
Dead code: Routines that are not reachable from subroutine main

Dead Code



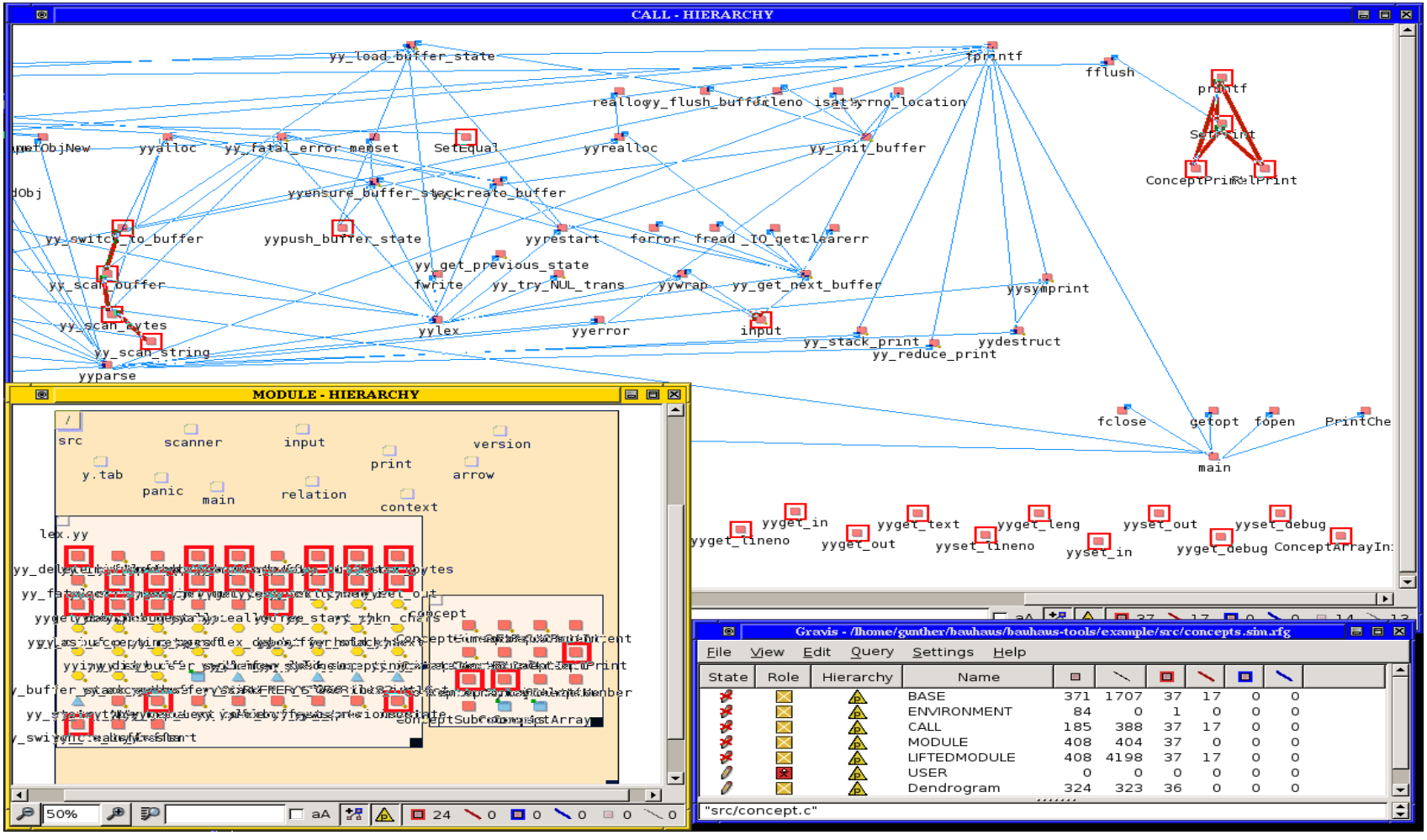
Dead code: Routines that are not reachable from subroutine main

Dead Code



Dead code: Routines that are not reachable from subroutine main

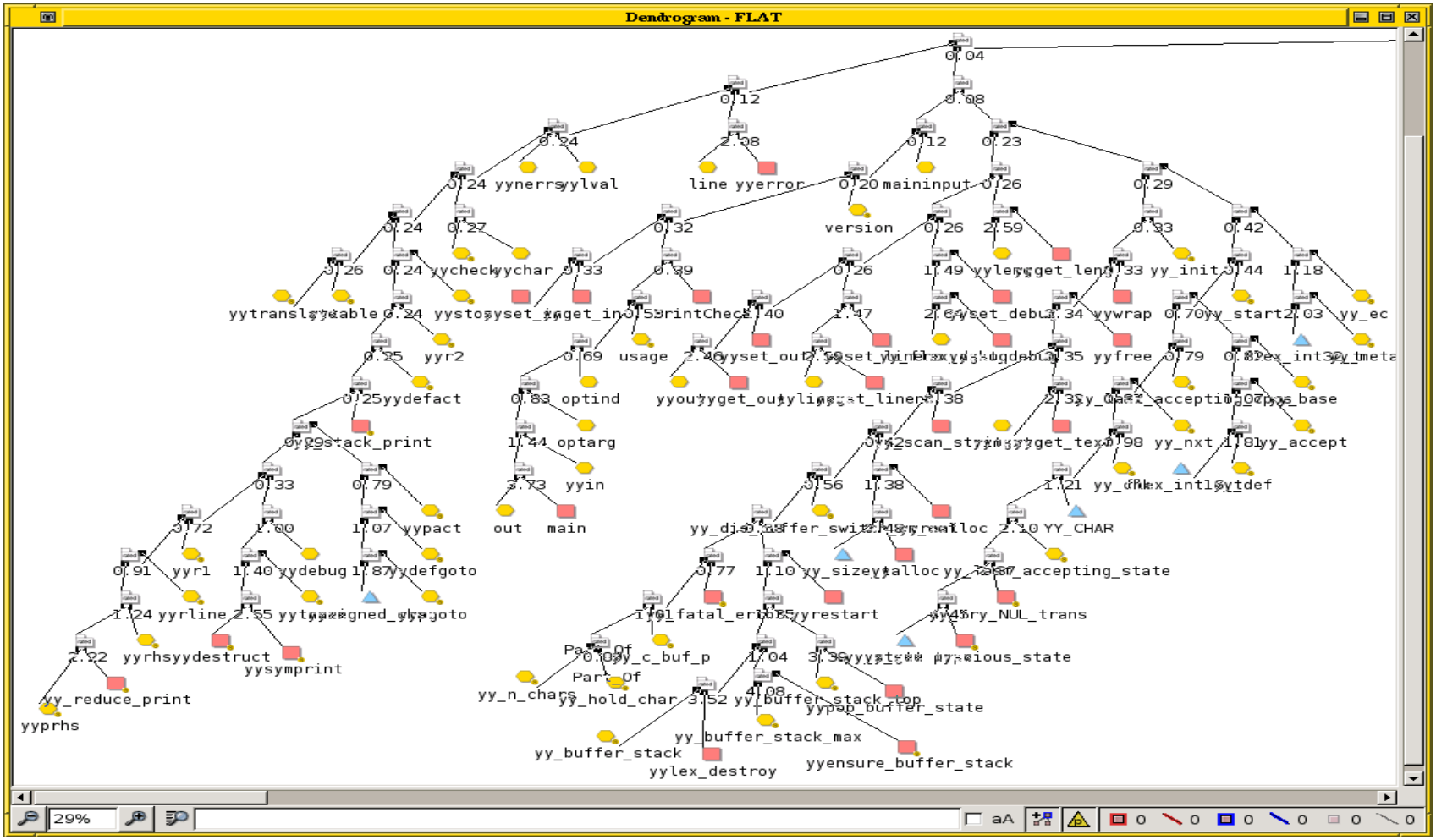
Dead Code



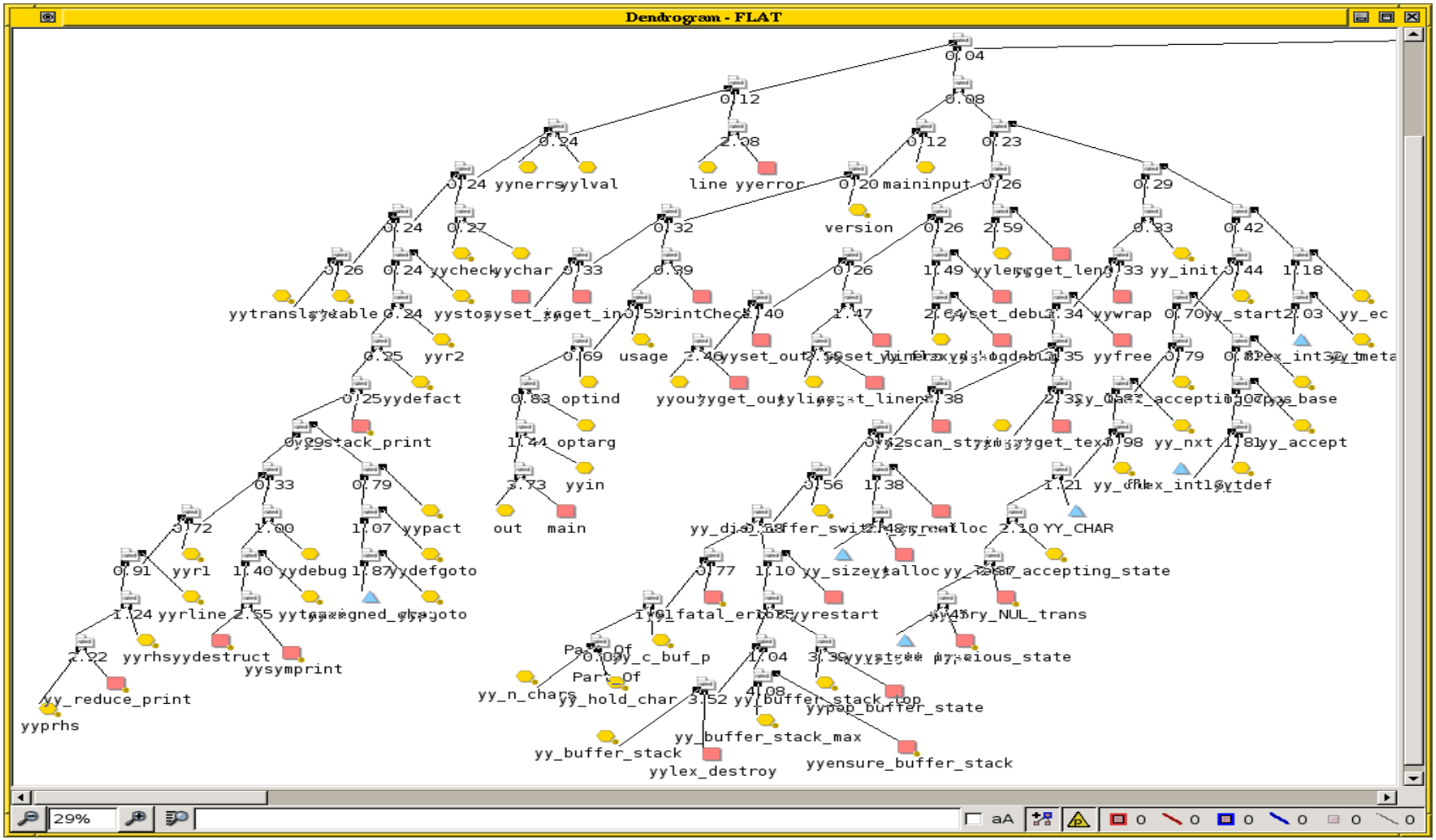
Dead code: Routines that are not reachable from subroutine main

Architecture Recovery

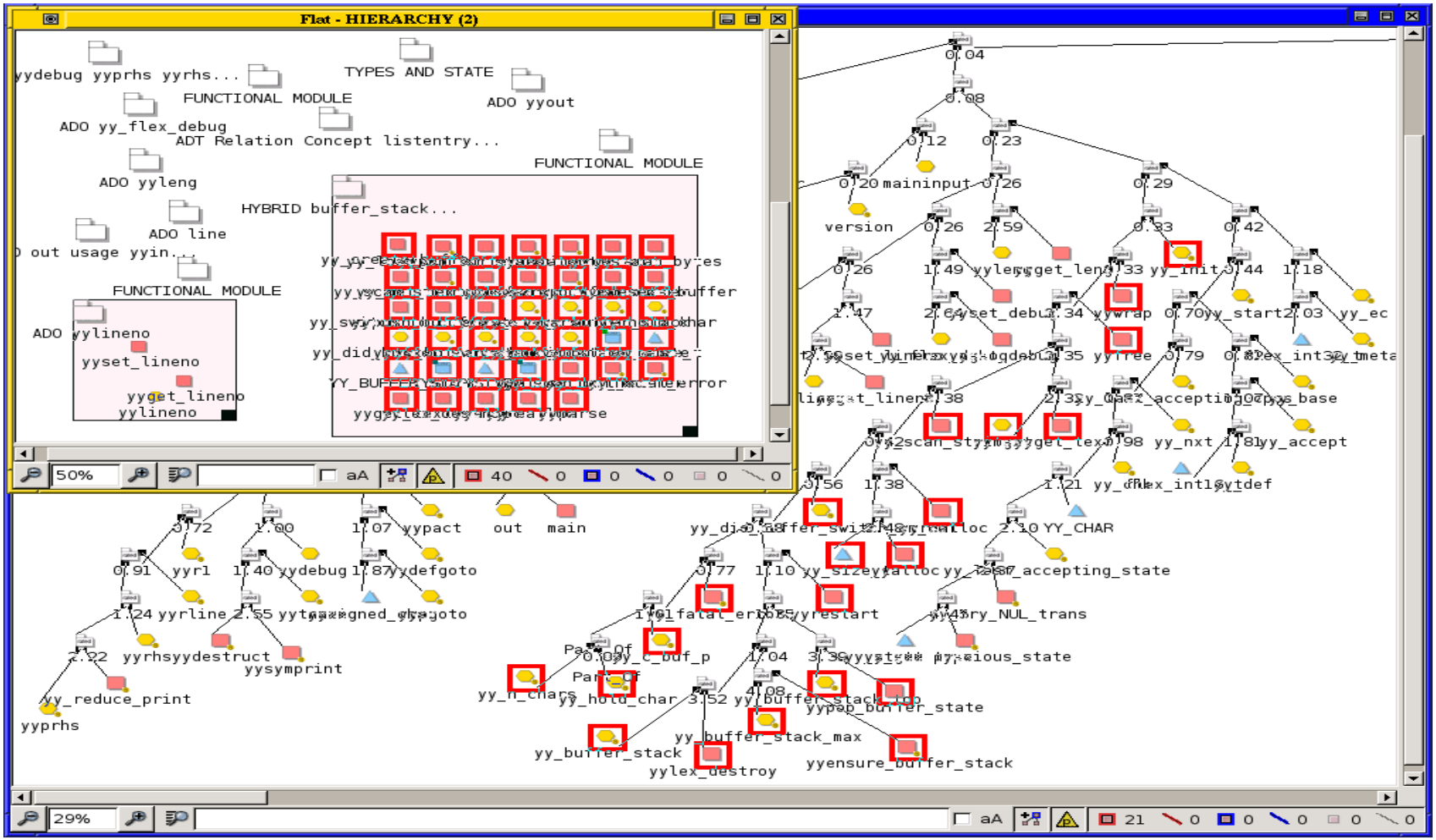
Architecture Recovery



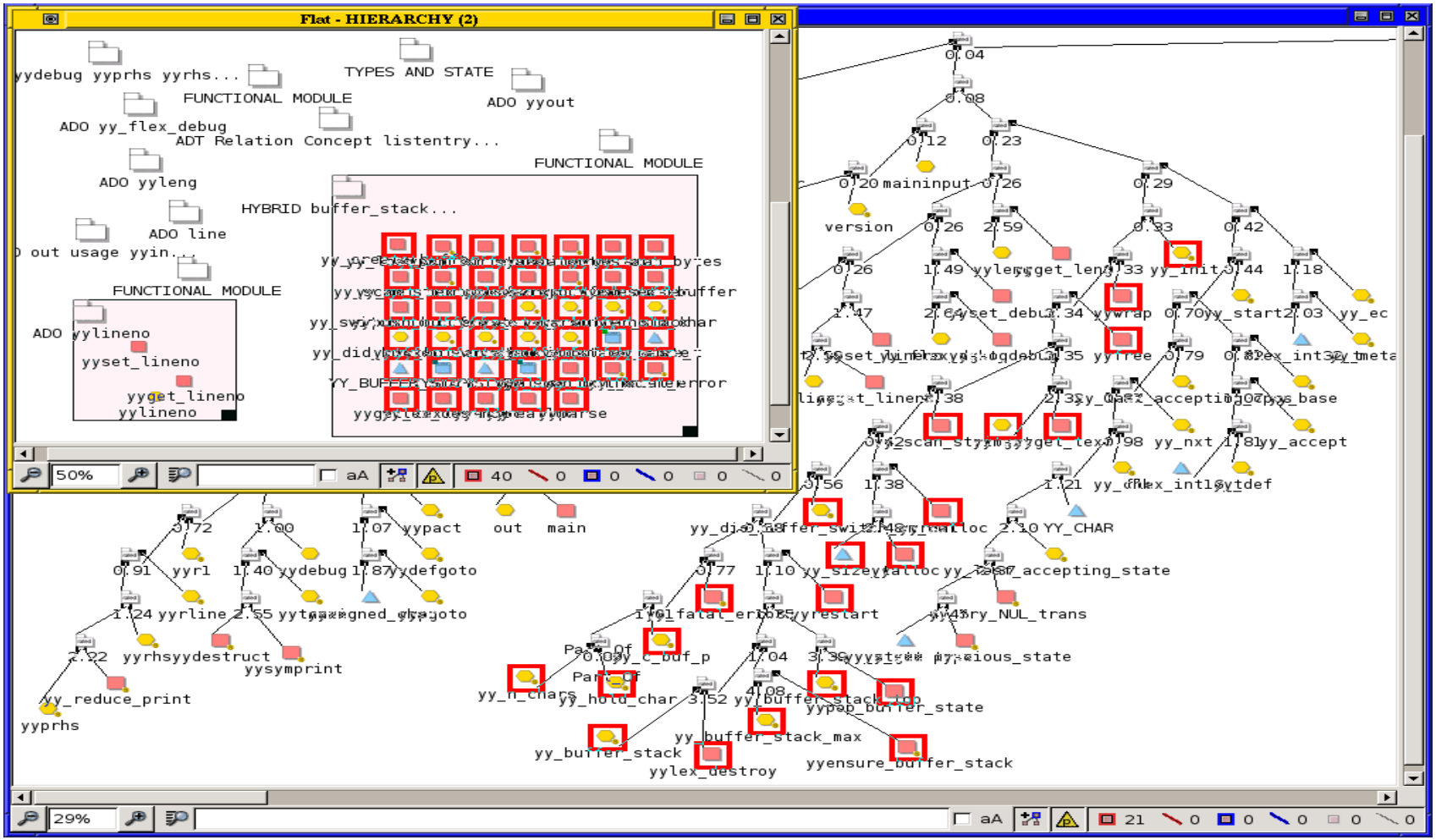
Architecture Recovery



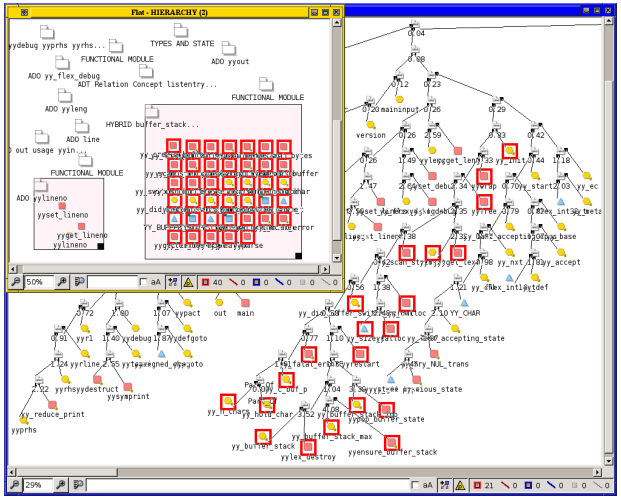
Architecture Recovery



Architecture Recovery



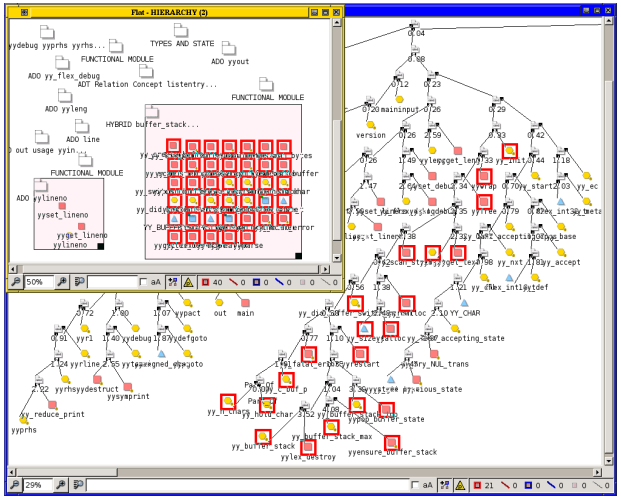
Architecture Recovery



- 14 automatic recovery techniques
- 7 categories of components: ADT, ADO, Function Library, ...
- 1 iterative semiautomatic recovery process

- Validation of hypothetical architectures – Reflexion method

Architecture Recovery



- 14 automatic recovery techniques
- 7 categories of components: ADT, ADO, Function Library, ...
- 1 iterative semiautomatic recovery process

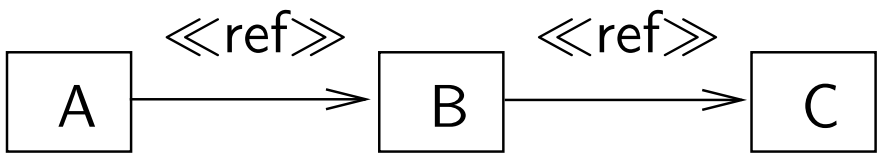
- Validation of hypothetical architectures – Reflexion method

Architecture Validation

Validation of hypothetical Architectures

Reflexion Method:

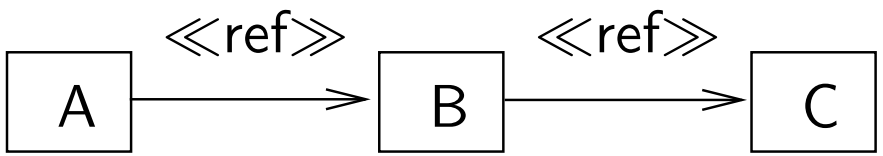
1. Draw hypothetical model



Validation of hypothetical Architectures

Reflexion Method:

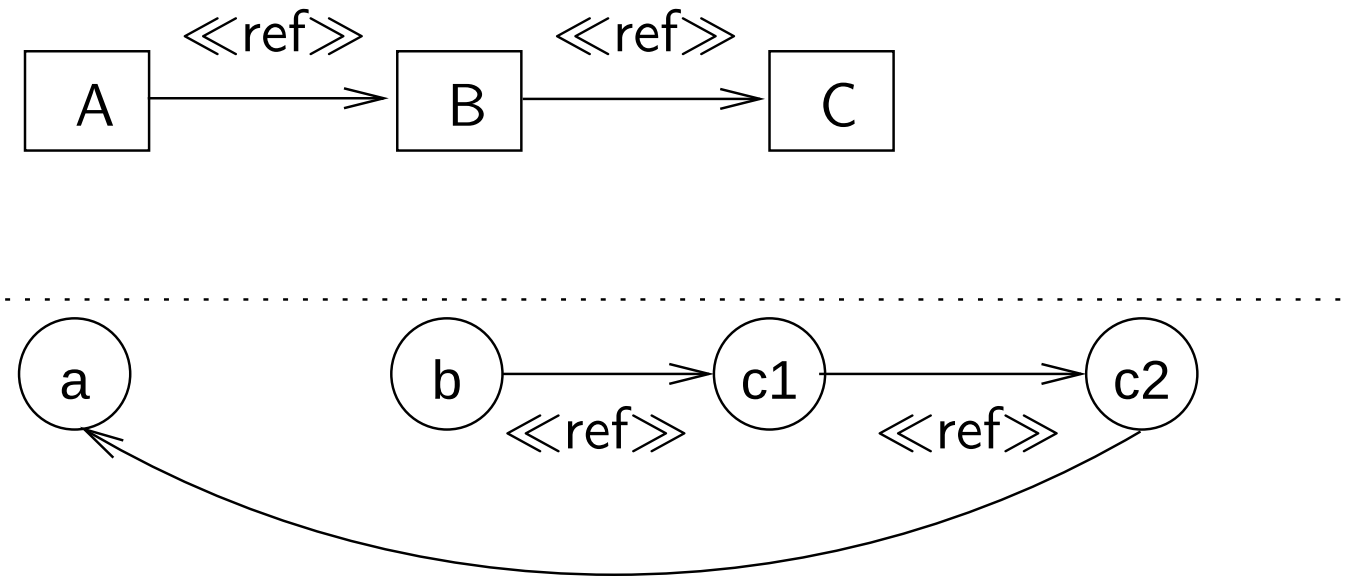
1. Draw hypothetical model



Validation of hypothetical Architectures

Reflexion Method:

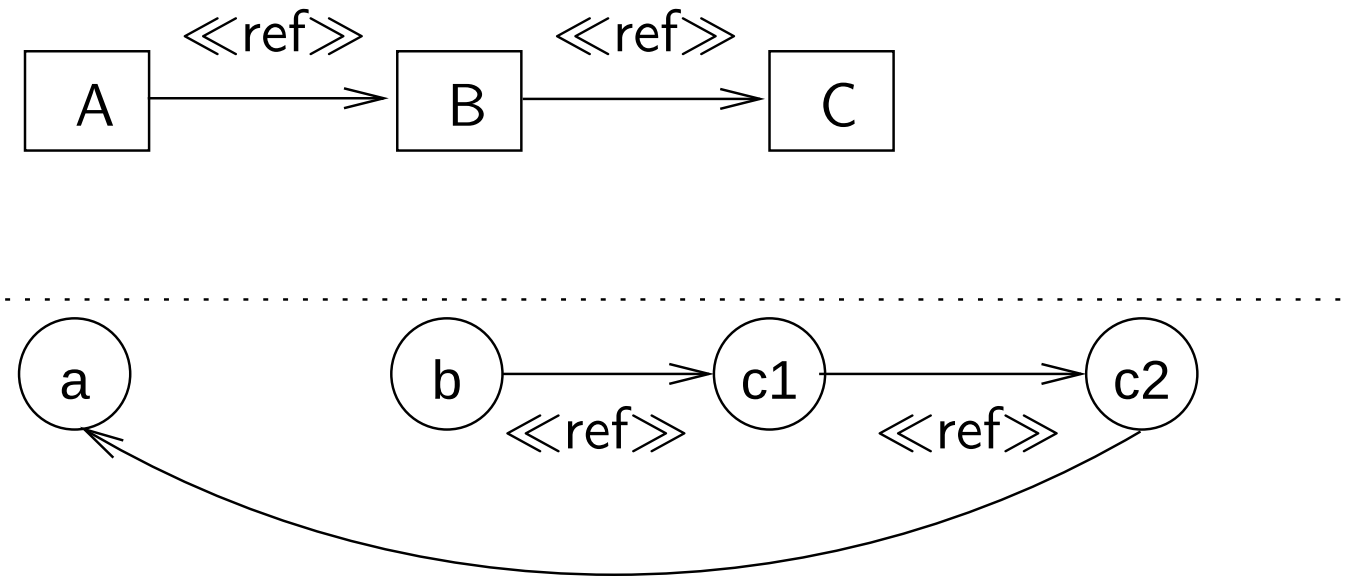
2. Extract concrete model



Validation of hypothetical Architectures

Reflexion Method:

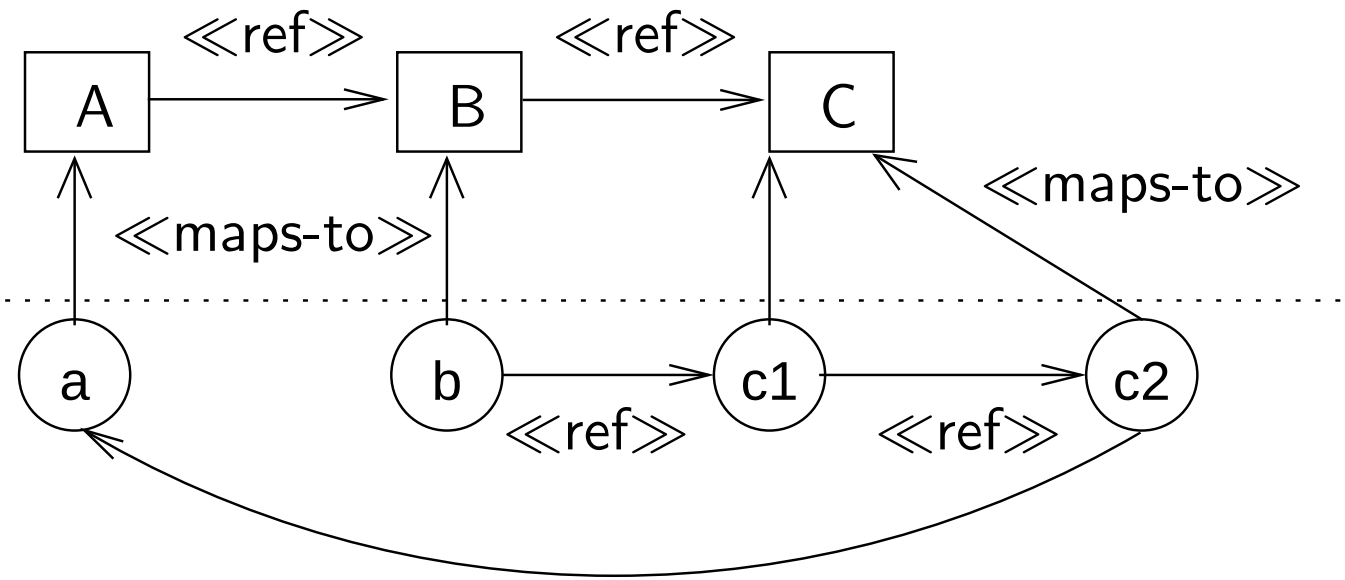
2. Extract concrete model



Validation of hypothetical Architectures

Reflexion Method:

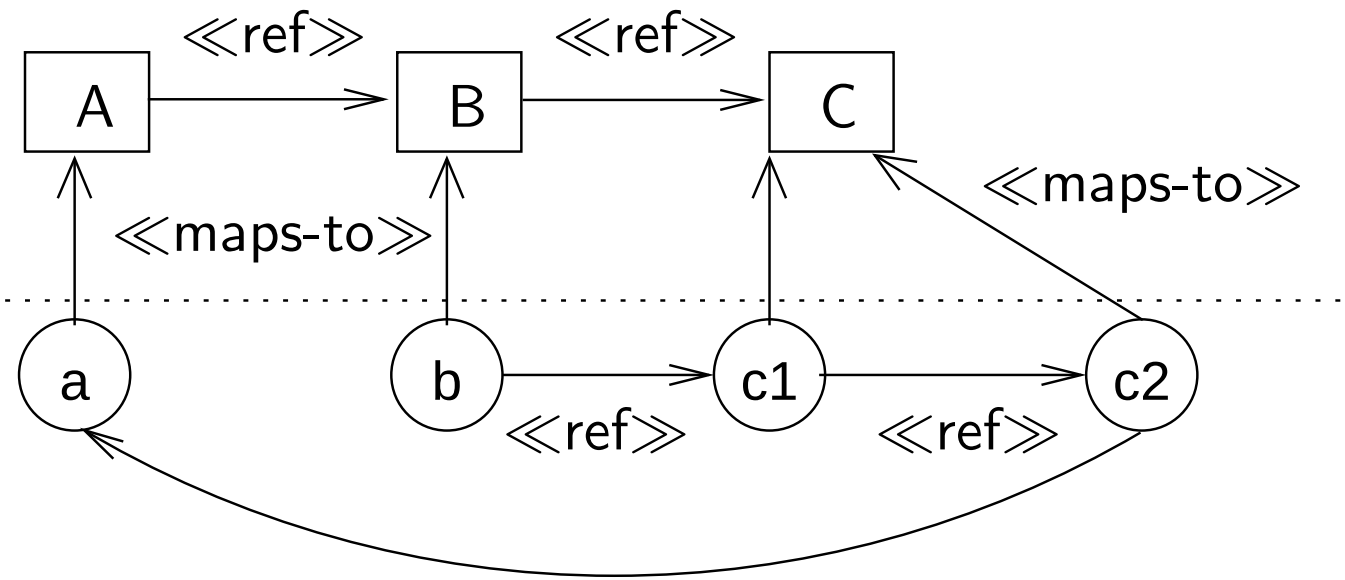
Construct Mapping



Validation of hypothetical Architectures

Reflexion Method:

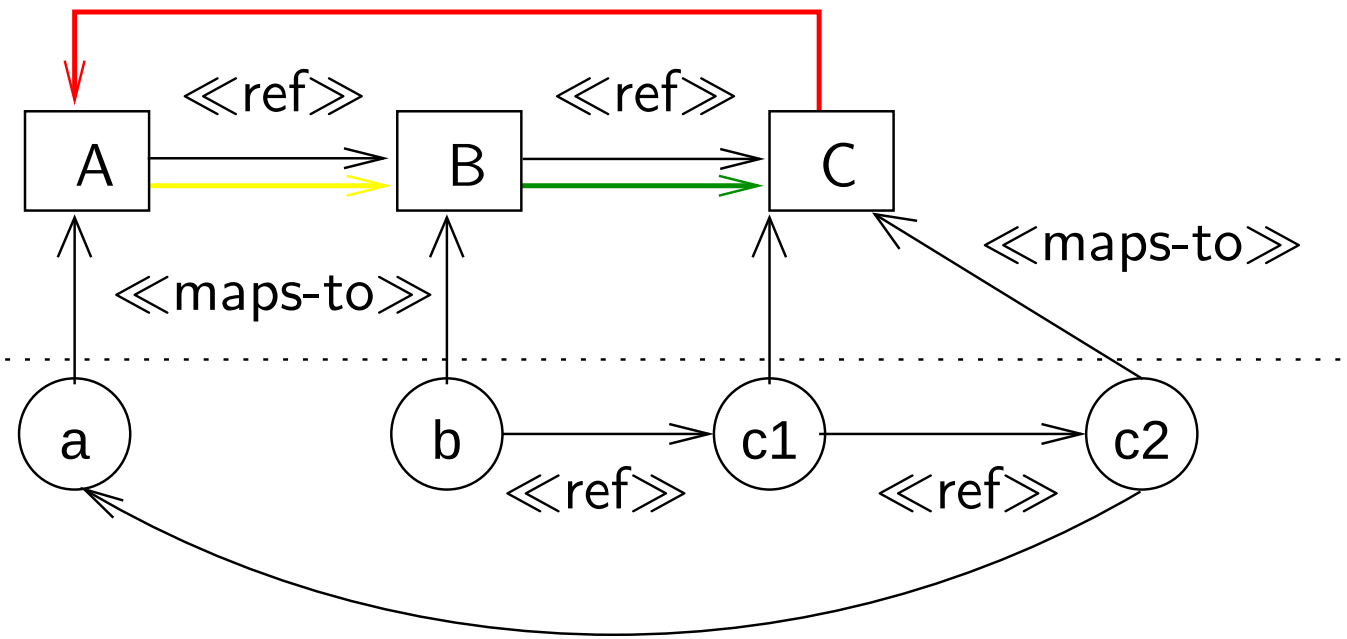
Construct Mapping



Validation of hypothetical Architectures

Reflexion Method:

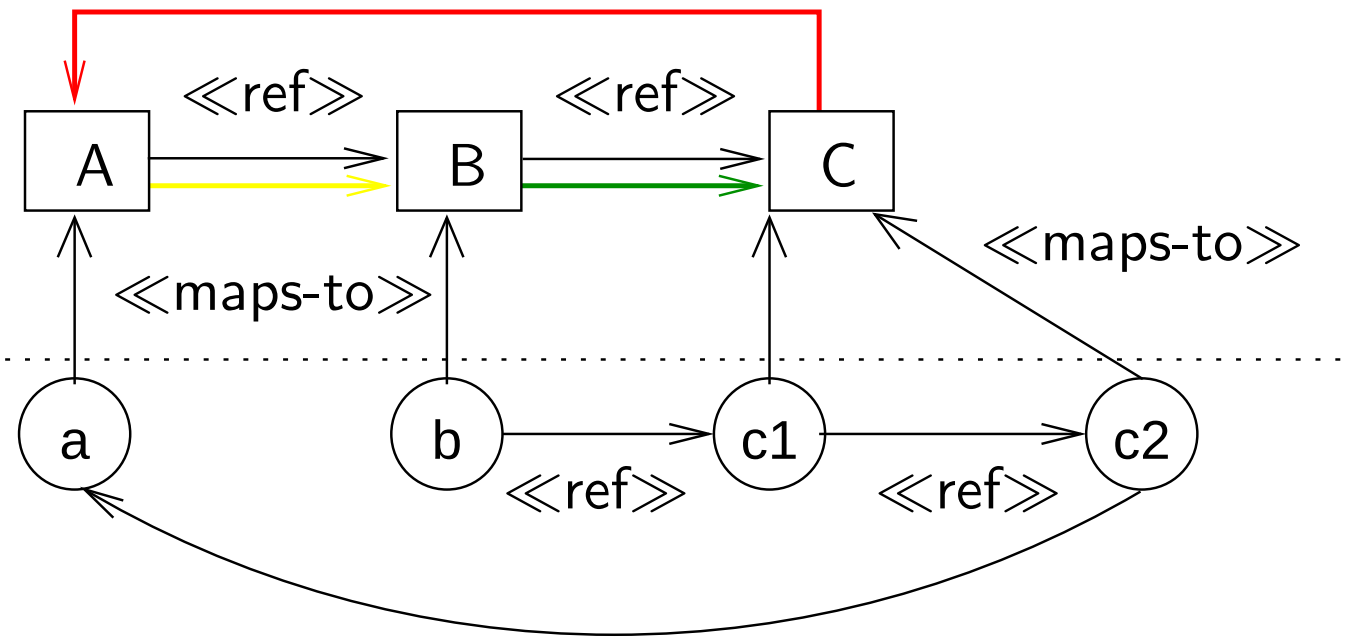
Compute Reflexion



Validation of hypothetical Architectures

Reflexion Method:

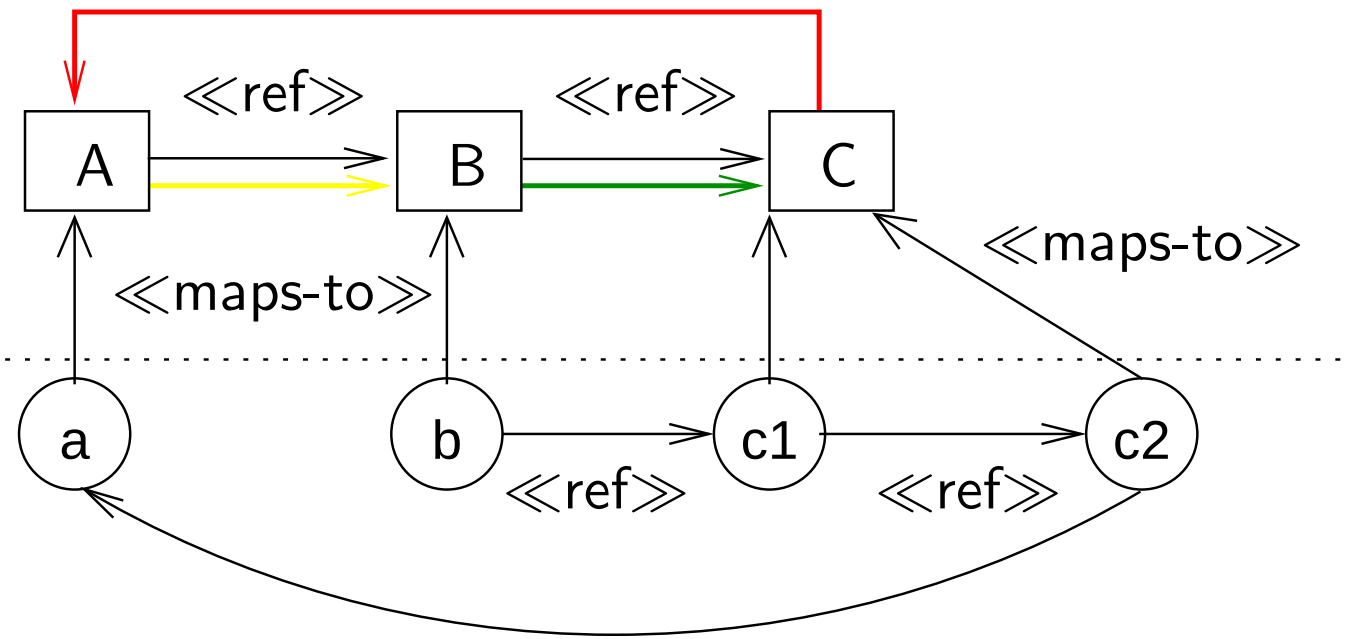
Compute Reflexion



Validation of hypothetical Architectures

Reflexion Method:

Compute Reflexion



Protocol Analysis

Protocol Analysis

Examples

- Variables must be defined before read
- Sequence constraints for operations on objects
- Interactions between software components
- Network protocols

Protocol Analysis

Examples

- Variables must be defined before read
- Sequence constraints for operations on objects
- Interactions between software components
- Network protocols

Protocol Analysis

Definition

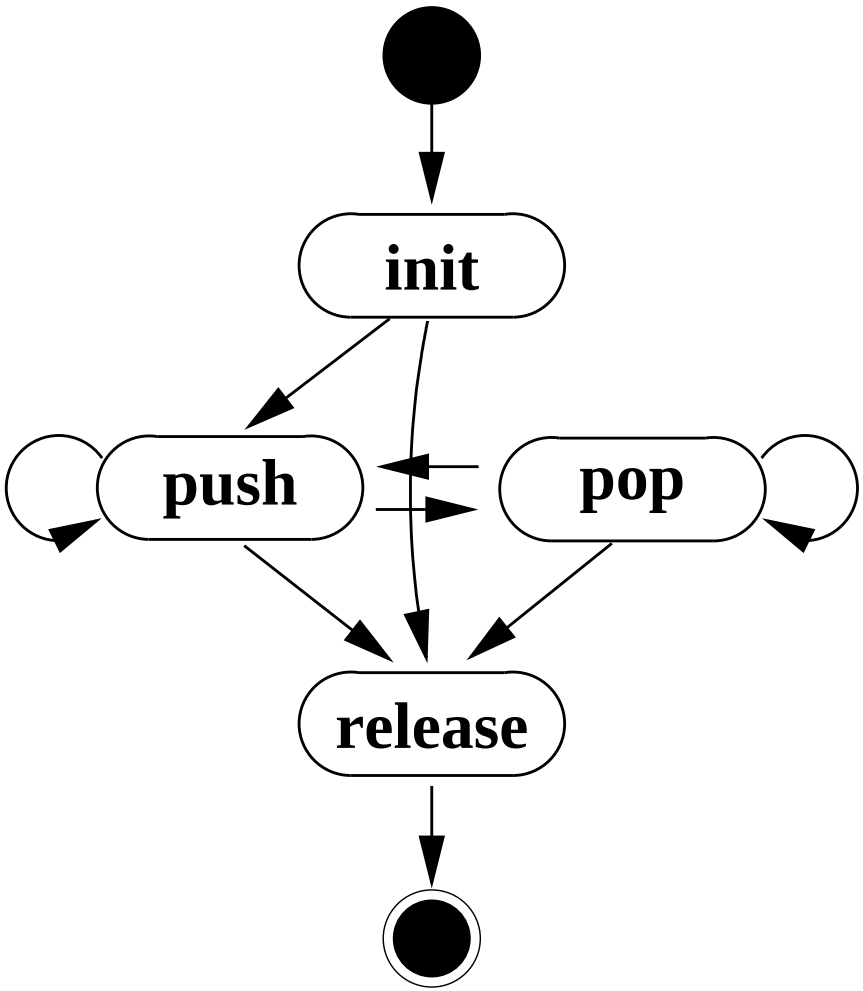
A protocol is a set of rules and conventions for program execution sequences

Applications:

- Program understanding
- Verification

Protocol representation:

- Control flow graphs
- Finite automatons



Protocol Analysis

Definition

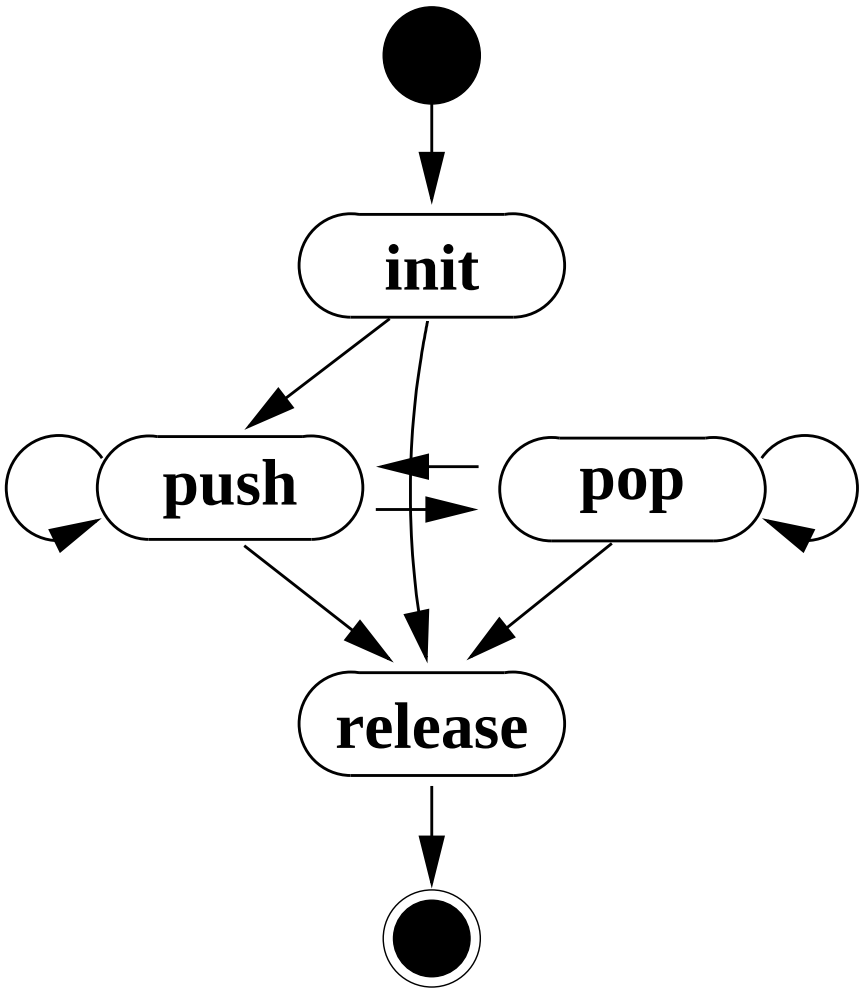
A protocol is a set of rules and conventions for program execution sequences

Applications:

- Program understanding
- Verification

Protocol representation:

- Control flow graphs
- Finite automatons



Protocol Analysis

Definition

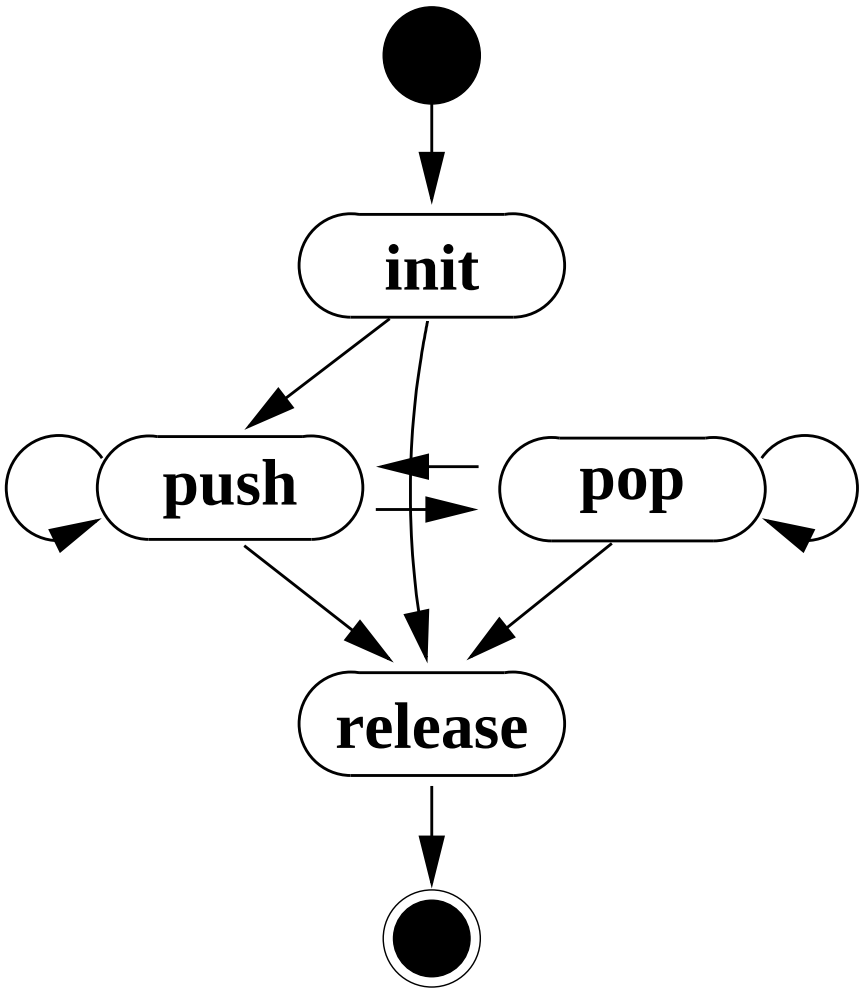
A protocol is a set of rules and conventions for program execution sequences

Applications:

- Program understanding
- Verification

Protocol representation:

- Control flow graphs
- Finite automatons



Protocol Analysis

Definition

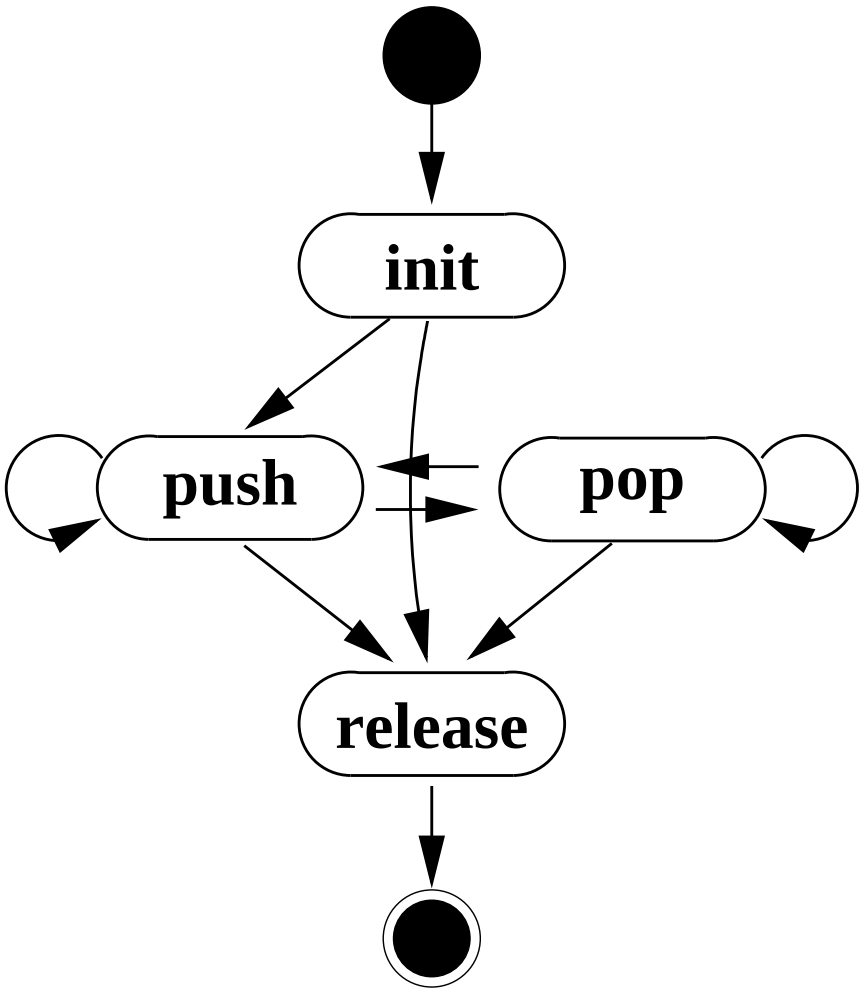
A protocol is a set of rules and conventions for program execution sequences

Applications:

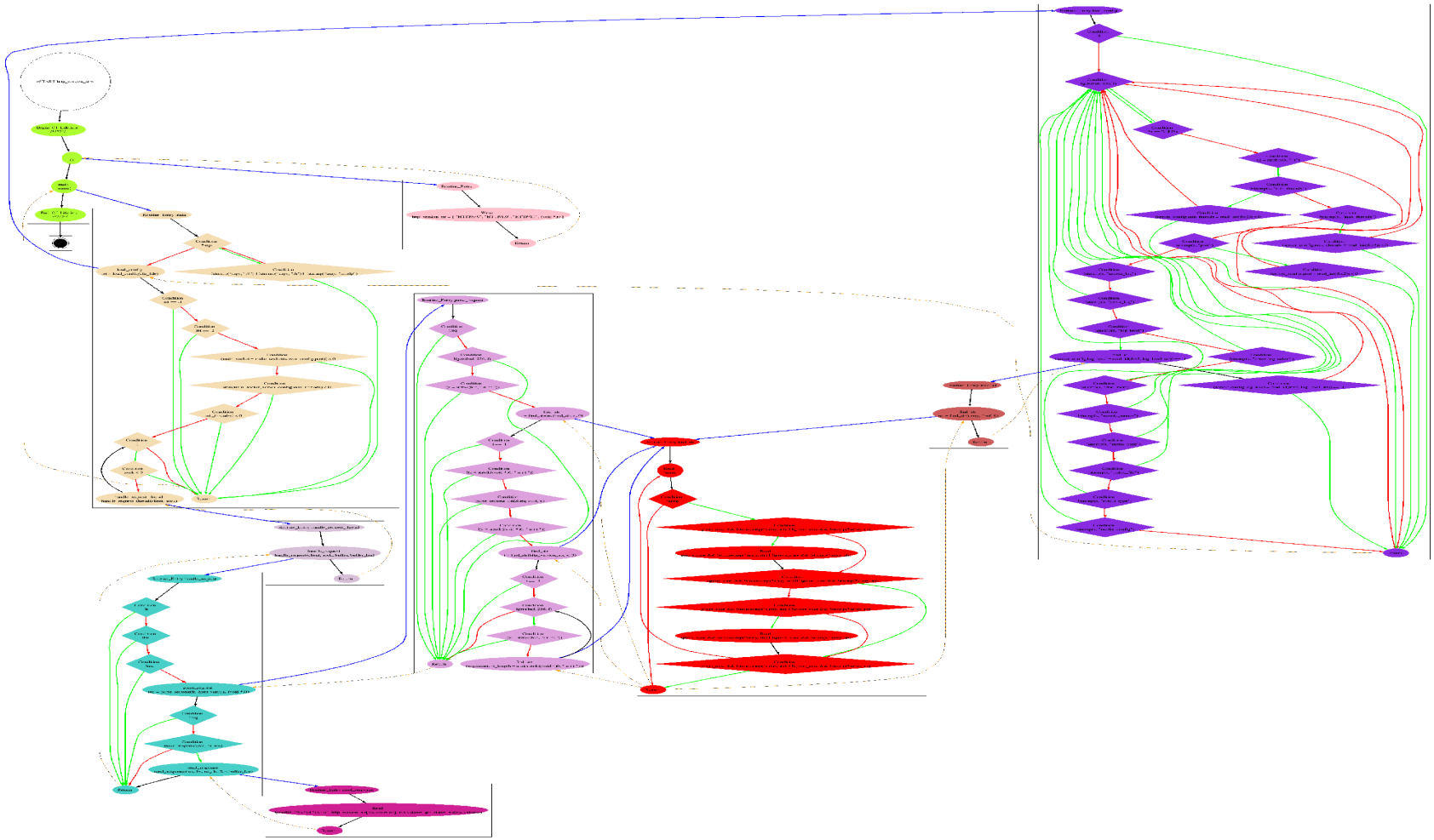
- Program understanding
- Verification

Protocol representation:

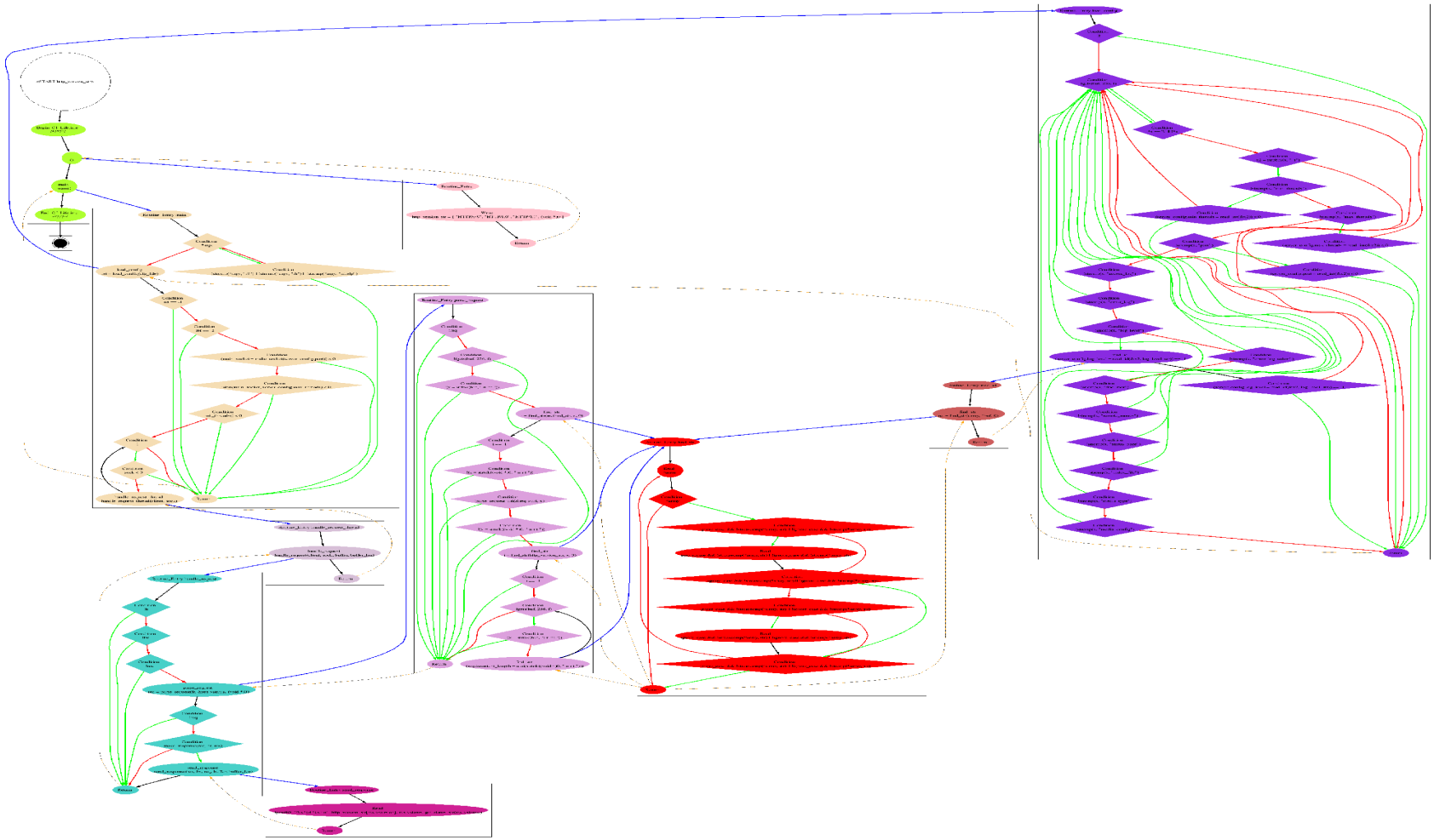
- Control flow graphs
- Finite automatons



Example: Global Variable in a Web Server



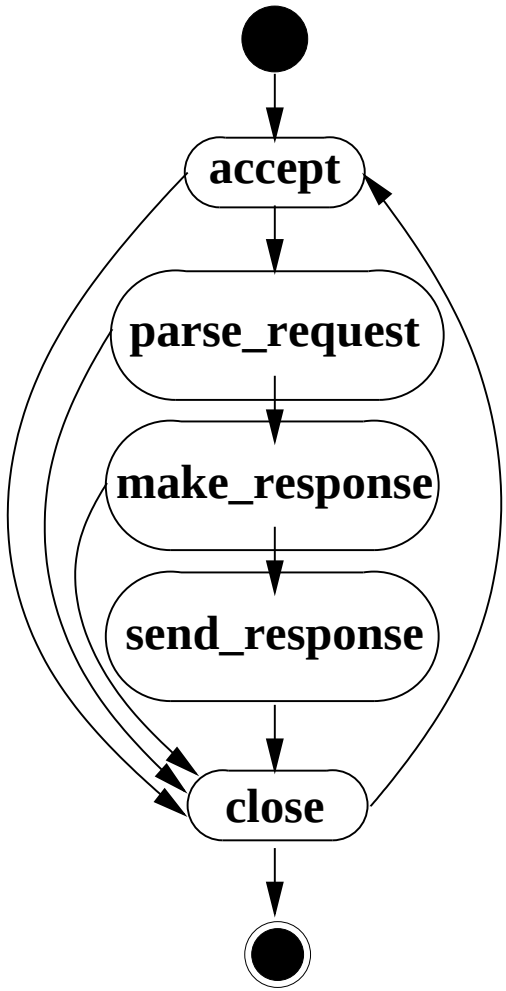
Example: Global Variable in a Web Server



Web Server – Network Communication

Implementation of HTTP:

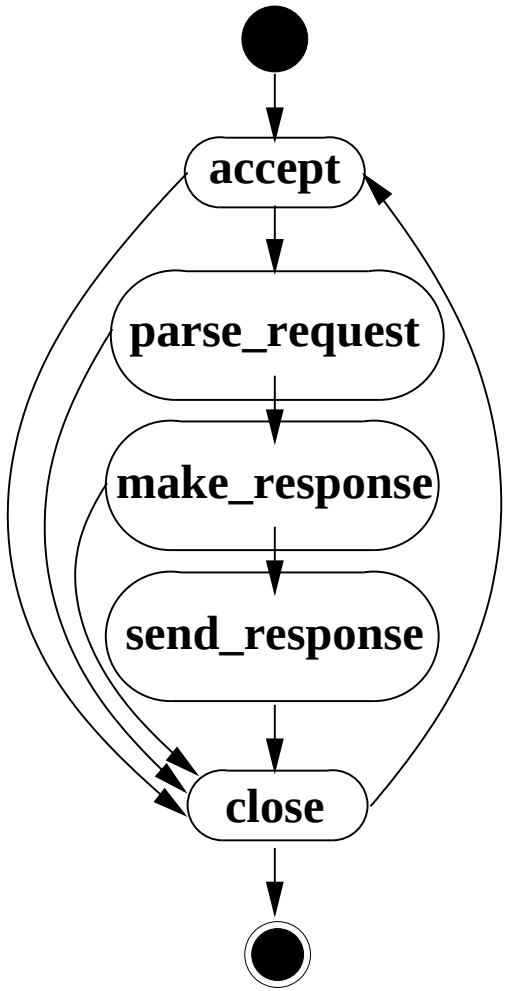
- Sequence of actions is correct
- Connections can be closed at arbitrary times
- But: Only one request per connection



Web Server – Network Communication

Implementation of HTTP:

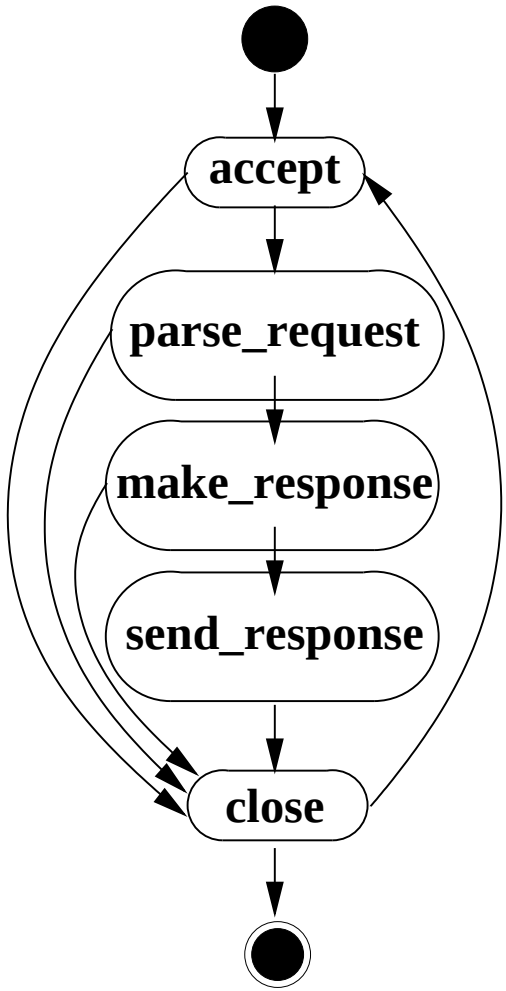
- Sequence of actions is correct
- Connections can be closed at arbitrary times
- But: Only one request per connection



Web Server – Network Communication

Implementation of HTTP:

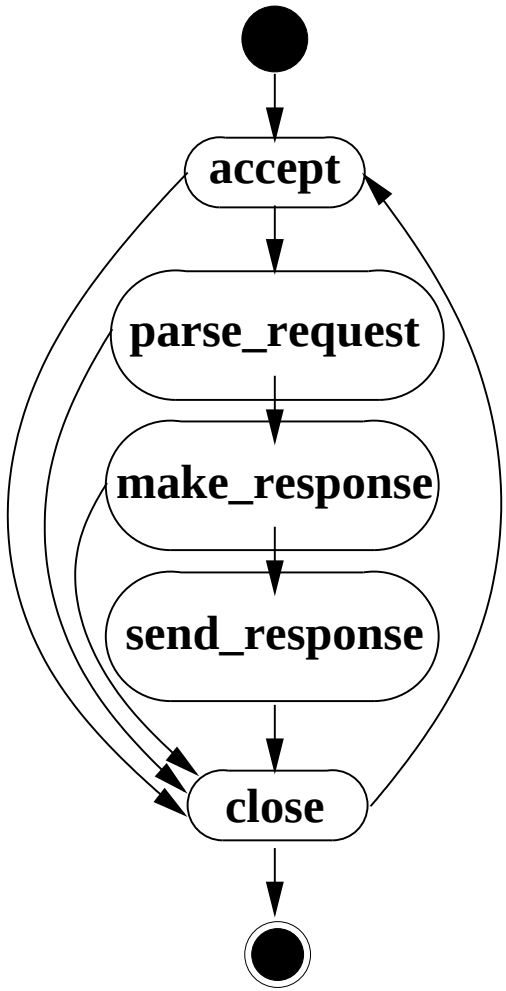
- Sequence of actions is correct
- Connections can be closed at arbitrary times
- But: Only one request per connection



Web Server – Network Communication

Implementation of HTTP:

- Sequence of actions is correct
- Connections can be closed at arbitrary times
- But: Only one request per connection



More Information

More Information

- http://www.bauhaus-stuttgart.de
- http://www.bauhaus-bremen.de
- http://www.axivion.de

More Information

- http://www.bauhaus-stuttgart.de
- http://www.bauhaus-bremen.de
- http://www.axivion.de

More Information

- http://www.bauhaus-stuttgart.de
- http://www.bauhaus-bremen.de
- http://www.axivion.de

More Information

- http://www.bauhaus-stuttgart.de
- http://www.bauhaus-bremen.de
- http://www.axivion.de