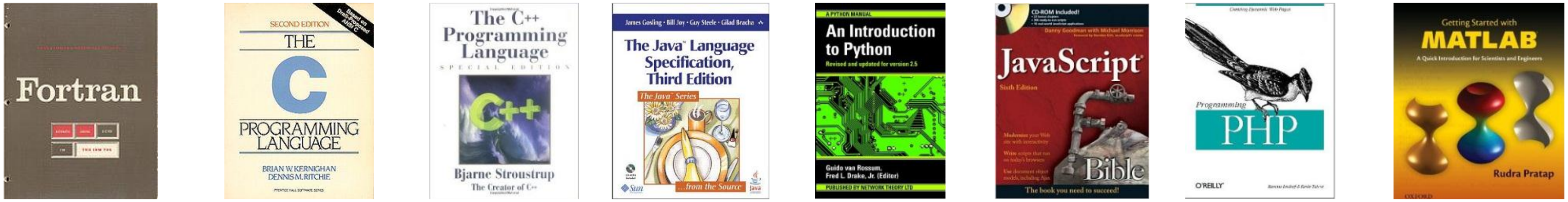
Multilingual Programming Made Easy

이병철

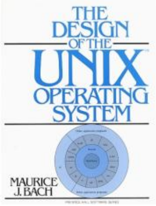
광주과학기술원

Evolving machines, programming languages, and systems

Faster and smaller machines used widely


Fortran
SECOND EDITION
THE C PROGRAMMING LANGUAGE
BRIAN W. KERNIGHAN DENNIS M. RITCHIE
The C++ Programming Language
SPECIAL EDITION
Bjarne Stroustrup
The Creator of C++
James Gosling · Bill Joy · Guy Steele · Gilad Bracha
The Java Language Specification, Third Edition
The Java Series
...from the Source
A PYTHON MANUAL
An Introduction to Python
Revised and updated for version 2.5
Guido van Rossum, Fred L. Drake, Jr. (Editor)
JavaScript
Sixth Edition
Bible
Danny Goodman with Michael Morrison
The book you need to succeed!
Programming PHP
O'REILLY
Getting Started with MATLAB
A Quick Introduction for Scientists and Engineers
Rudra Pratap
OXFORD

Innovative high level programming languages



OpenOffice.org™

mozilla Firefox

Apache Tomcat

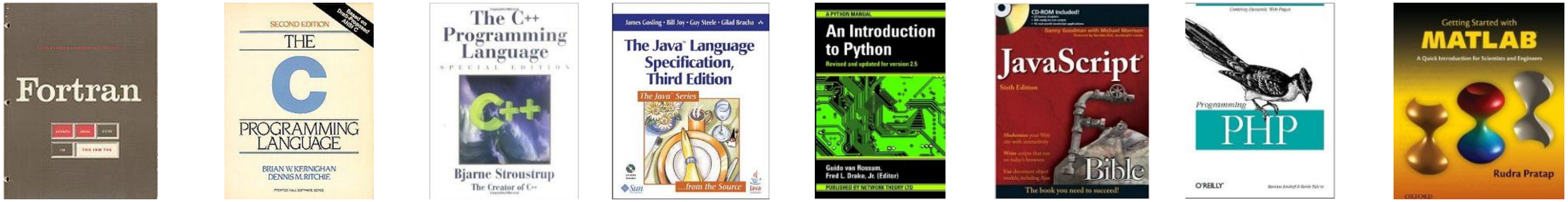
android

Legacy libraries and systems

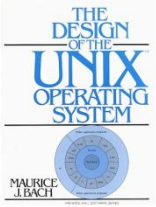
Evolving machines, programming languages, and systems

Faster and smaller machines used widely

Portability


Fortran
THE C PROGRAMMING LANGUAGE
SECOND EDITION
BRIAN W. KERNIGHAN
DENNIS M. RITCHIE
The C++ Programming Language
SPECIAL EDITION
Bjarne Stroustrup
The Java Language Specification, Third Edition
James Gosling · Bill Joy · Guy Steele · Gilad Bracha
An Introduction to Python
Guido van Rossum, Fred L. Drake, Jr. (Editor)
JavaScript Bible
Sixth Edition
Programming PHP
O'REILLY
Getting Started with MATLAB
Rudra Pratap

Innovative high level programming languages



OpenOffice.org™

mozilla Firefox

Apache Tomcat

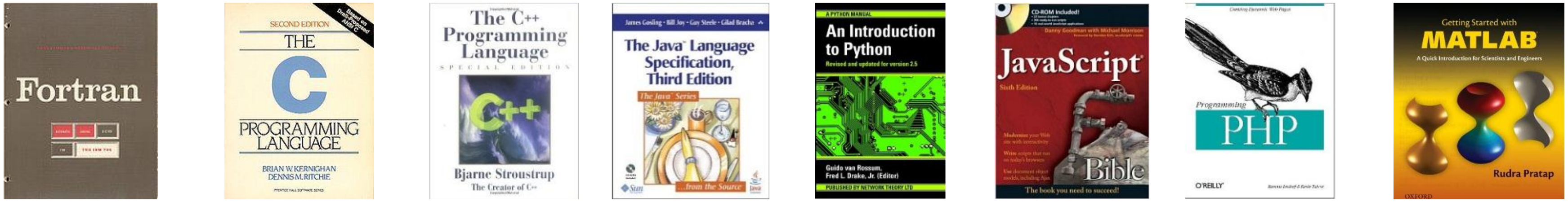
android

Legacy libraries and systems

Evolving machines, programming languages, and systems

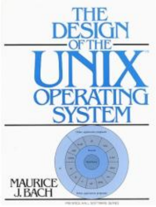
Faster and smaller machines used widely

Portability


Fortran
THE C PROGRAMMING LANGUAGE
The C++ Programming Language
The Java Language Specification, Third Edition
An Introduction to Python
JavaScript Bible
Programming PHP
Getting Started with MATLAB

Innovative high level programming languages

Complexity



OpenOffice.org™

mozilla Firefox

Apache Tomcat

android

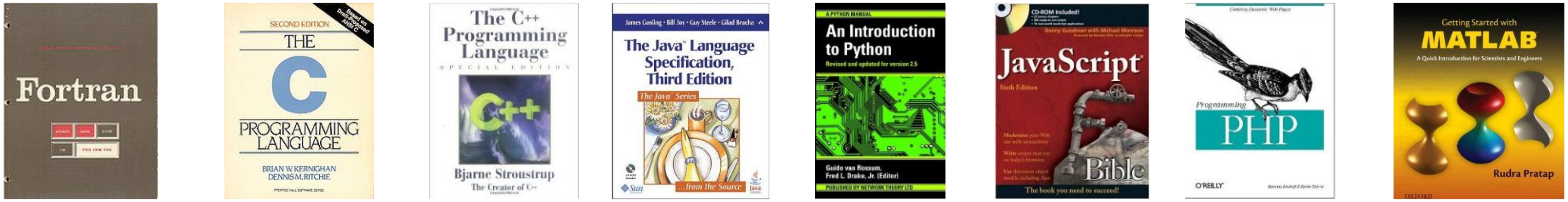
Legacy libraries and systems

Evolving machines, programming languages, and systems

Faster and smaller machines used widely

Portability

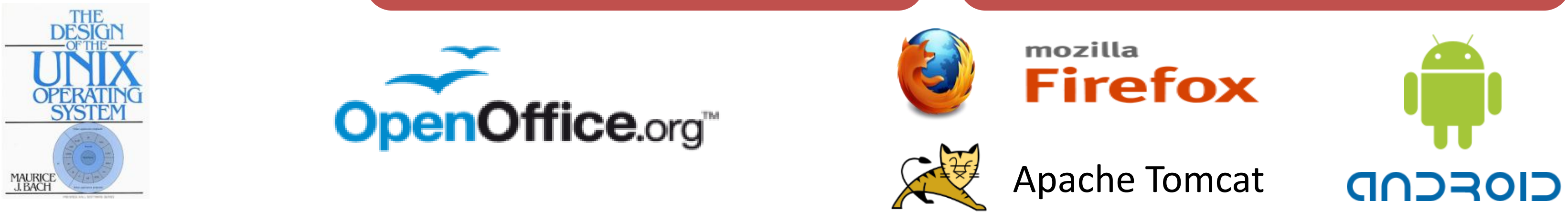
Heterogeneity


Fortran
SECOND EDITION
THE C PROGRAMMING LANGUAGE
BRIAN W. KERNIGHAN
DENNIS M. RITCHIE
The C++ Programming Language
SPECIAL EDITION
Bjarne Stroustrup
The Java Language Specification, Third Edition
An Introduction to Python
JavaScript Bible
Programming PHP
Getting Started with MATLAB
Rudra Pratap

Innovative high level programming languages

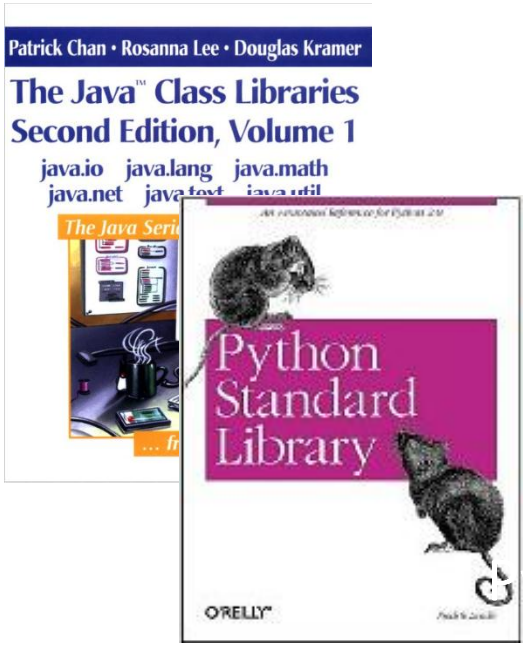
Complexity

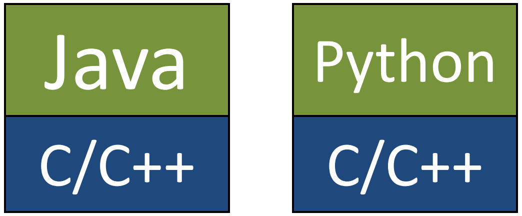
Heterogeneity



Legacy libraries and systems

Multilingual programs


Patrick Chan • Rosanna Lee • Douglas Kramer
The Java™ Class Libraries
Second Edition, Volume 1
java.io java.lang java.math
java.net
Python Standard Library
O'REILLY



Java

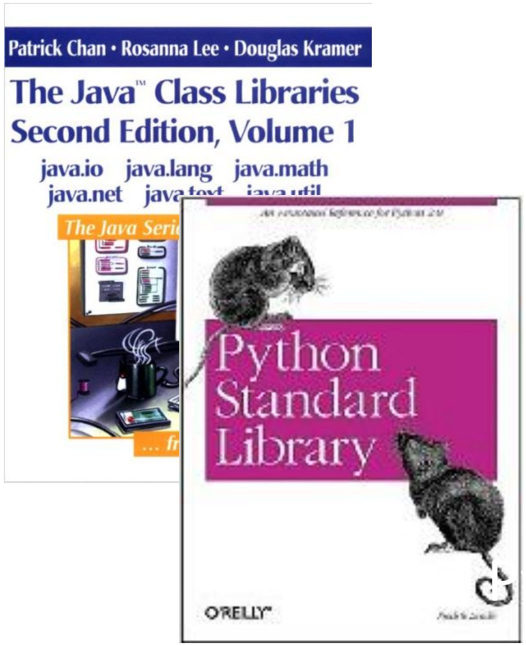
C/C++

Python

C/C++

Standard libraries

Multilingual programs



Patrick Chan • Rosanna Lee • Douglas Kramer

The Java™ Class Libraries Second Edition, Volume 1

java.io java.lang java.math java.net java.text java.util

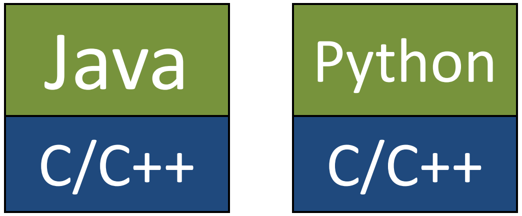
The Java Series

An essential reference for Python 2.0

Python Standard Library

O'REILLY®

Fredrik Lundh



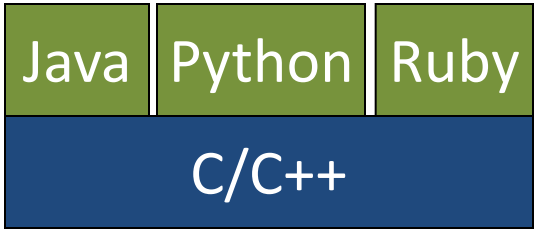
Java

C/C++

Python

C/C++

Standard libraries



Java

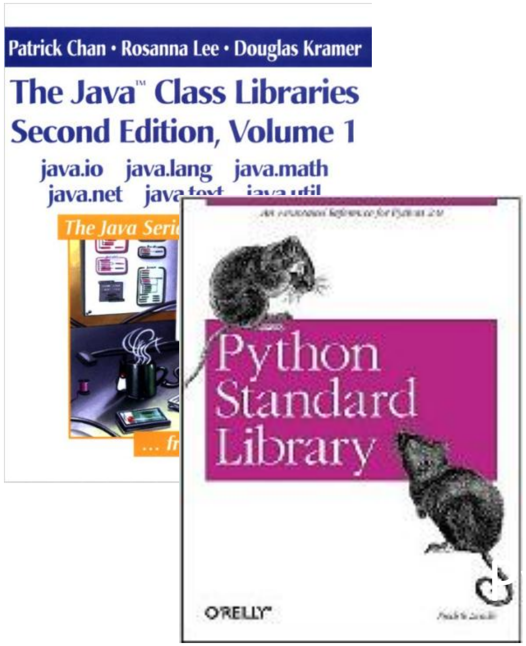
Python

Ruby

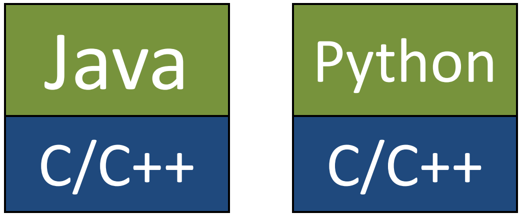
C/C++

Multilingual bindings

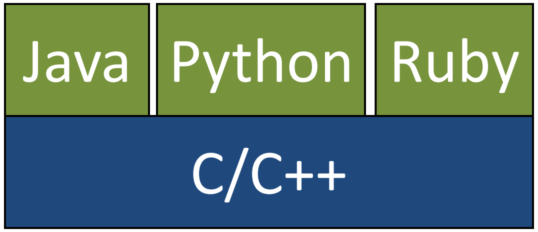
Multilingual programs


Patrick Chan • Rosanna Lee • Douglas Kramer
The Java™ Class Libraries
Second Edition, Volume 1
java.io java.lang java.math
java.net java.text java.util
The Java Series

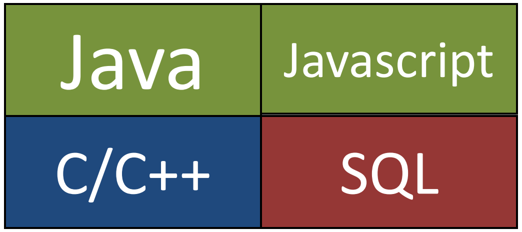
An essential reference for Python 2.0
Python Standard Library
O'REILLY
Fredrik Lundh


Java | Python
C/C++ | C/C++

Standard libraries


Java | Python | Ruby
C/C++

Multilingual bindings


Java | Javascript
C/C++ | SQL

Multilingual components

Active research groups in multilingual programming

	2005	2006	2007	2008	2009	2010	2011	2012
UNIVERSITY OF MARYLAND 1856	PLDI	ESOP	OOPSLA					
VE RI TAS HARVARD			OOPSLA		CCS	CCS	OOPSLA	PLDI
GIST THE UNIVERSITY OF TEXAS AT AUSTIN Microsoft Research IBM NEW YORK UNIVERSITY HARVEY MUDD COLLEGE			OOPSLA		OOPSLA	PLDI		ECOOP

PhD dissertation [Lee 2011]

Static analysis for Ocaml/C and JNI

Static and dynamic analysis for JNI

Composable analysis for language interfaces

Collaborators

GIST

Byeongcheol Lee

Microsoft Research

THE UNIVERSITY OF TEXAS AT AUSTIN

Kathryn S McKinley

IBM

Martin Hirzel

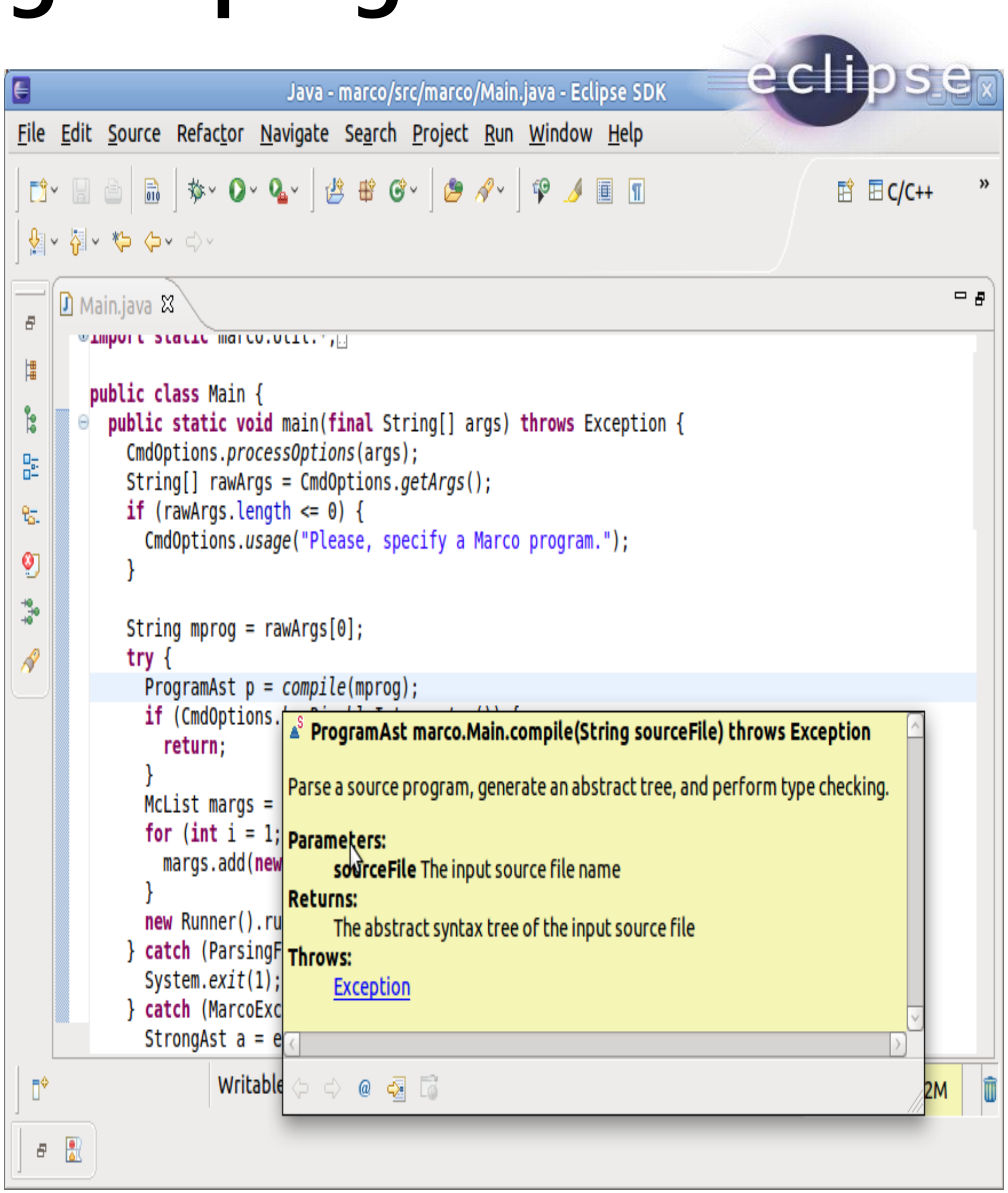
NEW YORK UNIVERSITY

Robert Grimm

HARVEY MUDD COLLEGE

Ben Wiedermann

Multilingual program



eclipse
Java - marco/src/marco/Main.java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run Window Help
C/C++
Main.java
Writable
2M

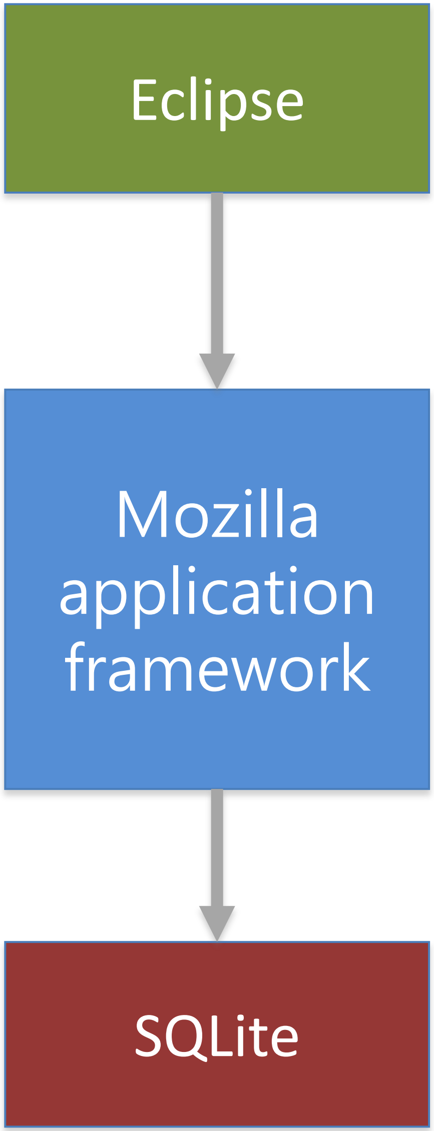
```
import static marco.util.*;

public class Main {
  public static void main(final String[] args) throws Exception {
    CmdOptions.processOptions(args);
    String[] rawArgs = CmdOptions.getArgs();
    if (rawArgs.length <= 0) {
      CmdOptions.usage("Please, specify a Marco program.");
    }

    String mprog = rawArgs[0];
    try {
      ProgramAst p = compile(mprog);
      if (CmdOptions.
        return;
      }
      McList margs =
      for (int i = 1;
        margs.add(new
      }
      new Runner().ru
    } catch (ParsingF
      System.exit(1);
    } catch (MarcoExc
      StrongAst a = e
```

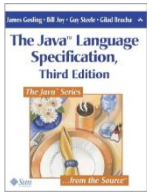
ProgramAst marco.Main.compile(String sourceFile) throws Exception

Parse a source program, generate an abstract tree, and perform type checking.

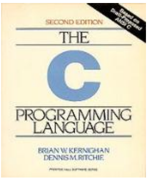
Parameters:
  sourceFile The input source file name
Returns:
  The abstract syntax tree of the input source file
Throws:
  Exception

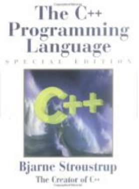
Crossing language boundaries

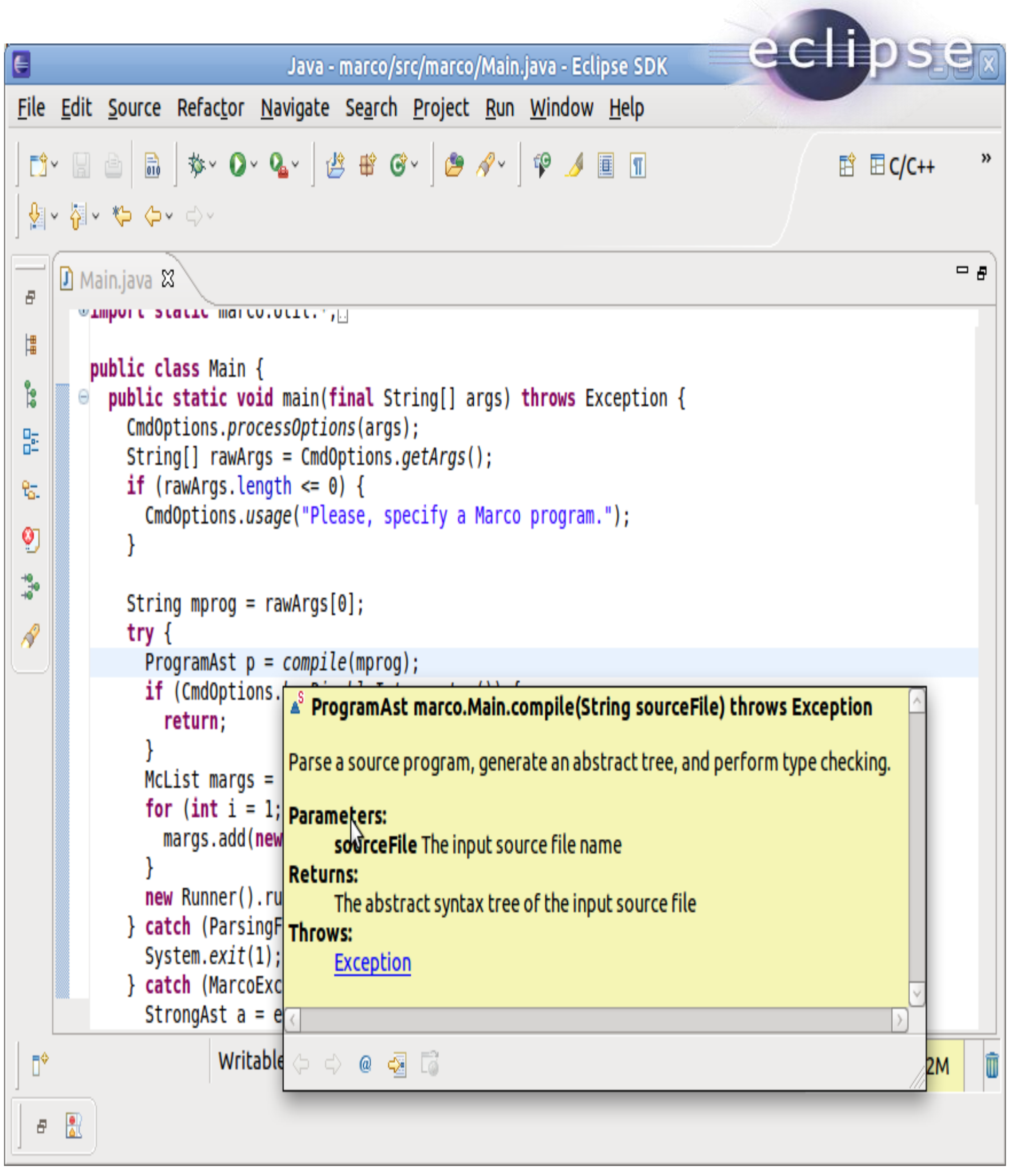



James Gosling · Bill Joy · Guy Steele · Gilad Bracha
The Java™ Language Specification, Third Edition
The Java™ Series
...from the Source

Sheng Liang
The Java™ Native Interface
Programmer's Guide and Specification
The Java™ Series
...from the Source






eclipse
Java - marco/src/marco/Main.java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run Window Help
C/C++
Main.java
Writable
2M

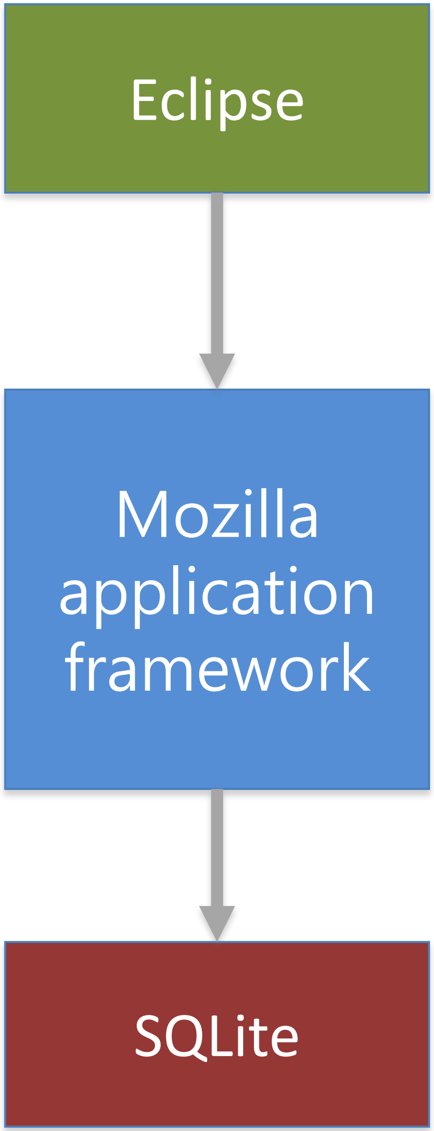
```
import static marco.util.*;

public class Main {
  public static void main(final String[] args) throws Exception {
    CmdOptions.processOptions(args);
    String[] rawArgs = CmdOptions.getArgs();
    if (rawArgs.length <= 0) {
      CmdOptions.usage("Please, specify a Marco program.");
    }

    String mprog = rawArgs[0];
    try {
      ProgramAst p = compile(mprog);
      if (CmdOptions.
        return;
      }
      McList margs =
      for (int i = 1;
        margs.add(new
      }
      new Runner().ru
    } catch (ParsingF
      System.exit(1);
    } catch (MarcoExc
      StrongAst a = e
```

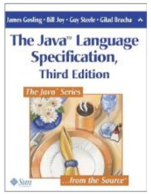
ProgramAst marco.Main.compile(String sourceFile) throws Exception

Parse a source program, generate an abstract tree, and perform type checking.

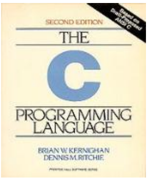
Parameters:
  sourceFile The input source file name
Returns:
  The abstract syntax tree of the input source file
Throws:
  Exception

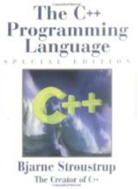
Crossing language boundaries




James Gosling · Bill Joy · Guy Steele · Gilad Bracha
The Java™ Language Specification, Third Edition
The Java™ Series
...from the Source
Sun

Sheng Liang
The Java™ Native Interface
Programmer's Guide and Specification
The Java™ Series
...from the Source



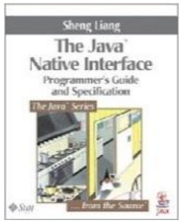


```
class Eclipse {
   native void show(String page);
}
void Java_Eclipse_show(JNIEnv* e,
     jobject jpage) {
   char *cpage =
            (*e)->GetStringUTFChars(e, jpage);
     Mozilla::show(cpage);
}
void Mozilla::show(string cpage) {
  sqlite3_run(
            “select title from bookmark”
            +  “where” + predicate(cpage)
  );  …}
```

Programming pitfalls

```
class Eclipse {
  native void show(String page);
}
```

```
void Java_Eclipse_show(JNIEnv* e,
   jobject jpage) {
  char *cpage =
          (*e)->GetStringUTFChars(e, jpage);
   Mozilla::show(cpage);
}
```



- Is it OK to call a JNI function here?
- Does “jpage” point to a Java string?
- Is “cpage” going to be released eventually?

Synthesizing dynamic FFI bug checker [PLDI’10]

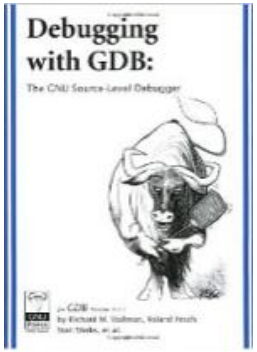
Programming pitfalls

Is the SQL command syntactically correct?

Marco: safe, expressive macros for any language [ECOOP'12]

```
void Mozilla::show(string cpage) {
  sqlite3_run(
        “select title from bookmark”
        +  “where” + predicate(cpage)

  );  …}
```

Programming pitfalls



How to debug this program across Java and C/C++?

Portable mixed-environment debuggers [PLDI'10]

```
class Eclipse {
   native void show(String page);
}
void Java_Eclipse_show(JNIEnv* e,
    jobject jpage) {
   char *cpage =
          (*e)->GetStringUTFChars(e, jpage);
    Mozilla::show(cpage);
}
void Mozilla::show(string cpage) {
  sqlite3_run(
          "select title from bookmark"
          +  "where" + predicate(cpage)
  );  …}
```

PhD Dissertation [Lee 2011]

The Java Native Interface
Programmer's Guide and Specification
Sheng Liang
Python/C API Manual
PYTHON 2.6
Using SQLite
An Introduction to GCC
for the GNU Compilers gcc and g++
Brian Gough
eclipse
Debugging with GDB:
The GNU Source-Level Debugger

Classification and composition

Marco [ECOOP'12], Jinn [PLDI'10], and Blink [OOPSLA'09]

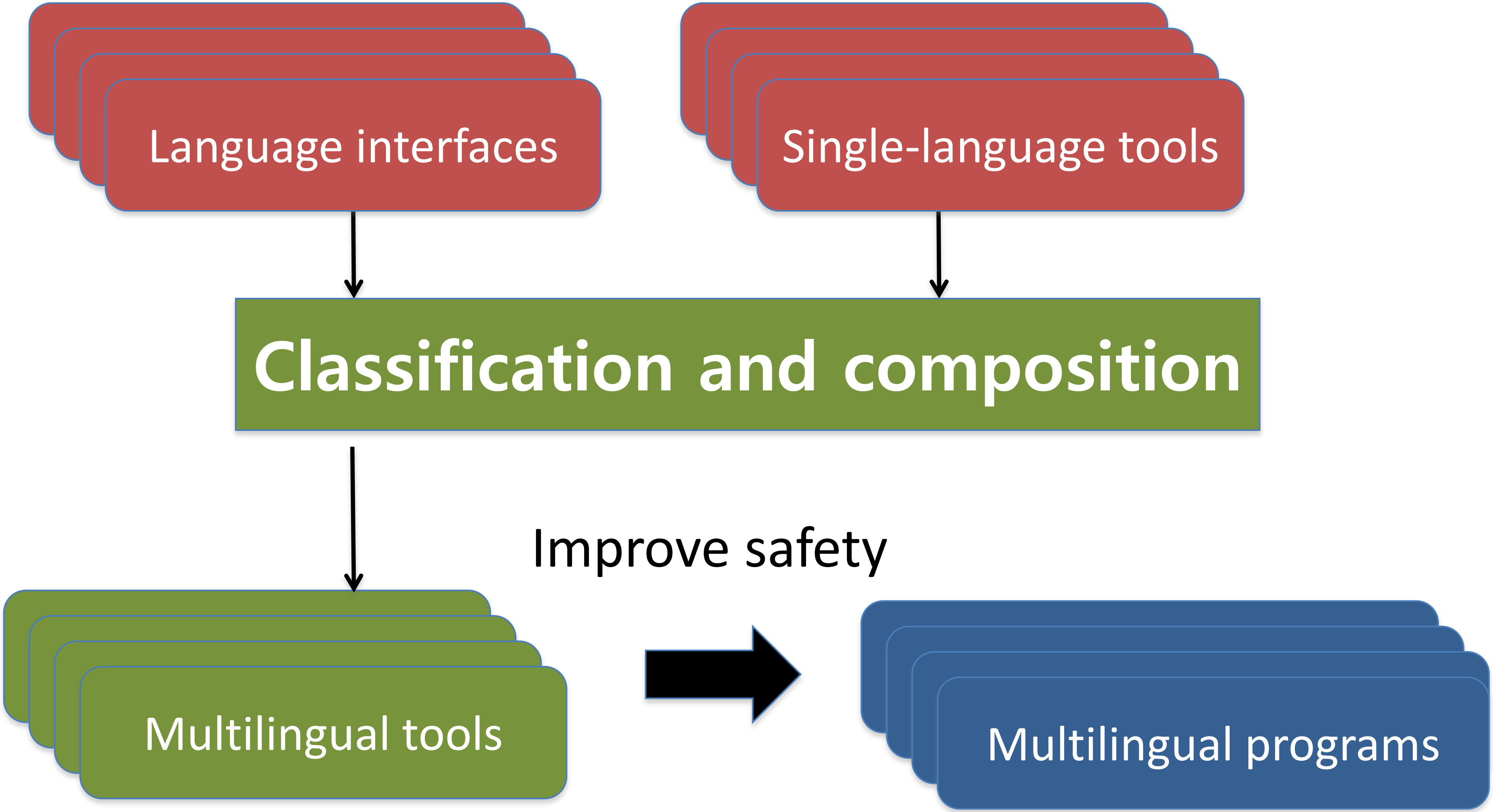
Improve safety

GNOME™

eclipse

SUBVERSION

PhD Dissertation [Lee 2011]



Language interfaces

Single-language tools

Classification and composition

Improve safety

Multilingual tools

Multilingual programs

Future work

- Correctness
  - Program analysis for resource management
  - Empirical studies on real-world bugs
  - Automatically explaining and fixing bugs
- Performance
  - Optimizing away cross-boundary overhead
  - Cooperative memory management for multiple runtime systems
  - Mapping multilingual program fragments to heterogeneous multi-core processors

Thank you

Questions?