Susan L. Graham, Andrew Begel, Marat Boshernitsan

**Harmonia**

An extensible framework for interactive, language-aware programming tools.

## Harmonia Analysis Framework

- **Language Descriptions**
- **Incremental Lexer**
- **Incremental GLR Parser**
- **Semantics**
- **Versioned Syntax Tree**
- **EBNF Grammar**
- **Reflection**
- **XML Serialization of Program Edit History**

## Harmonia Analysis Framework Diagram

- **XEmacs Integration**
- **Semantic Errors While You Edit**
- **OpenGL-based Parse Tree Visualization**
- **Eclipse Integration**
- **Structural Navigation**
- **Online Interactive Parse Tree View**
- **Open GL-based Control Flow Graph Visualization**
- **Structurally-filtered Search**

## Benefits of Eclipse Platform for Harmonia Research

- **Text Editor Framework**
- **Project and Resource Management**
- **Graphical User Interface**
- **3rd Party Tool Integration**
- **Existing Base of Potential Users**

## Future Work Enabled by Eclipse

- **Integrate Eclipse’s User Services with Harmonia’s Analyses**
- **Voice-Based Commenting Environment**
- **Harmonia Language Module Development Perspective**
- **Java/JNI/C Plugin Development Perspective**

## Harmonia + Eclipse Enables New Research in End-Programmer Tools

### Interactive Transformations

**Goal:** Enable the programmer to express operations on program source code at a level above traditional text-based editing.

**Programmers Say**

- “Rename [external variables] to make their first six letters unique.”  
  [Knuth, The errors of TeX]
- “Change BI_* macros to BYTE_*; similarly for bi_* local variables.”  
  [Ben Wing, XEmacs ChangeLog]

**Programmers Now Do**

```bash
sed s/\{\([^\.)]*\)\).take_one(\([^\)]*\))\)/\2 = \1.back();\n\1.pop_back()/g
```

**Programmers Should Do**

### Programming by Voice

**Goal:** Enable a programmer to compose, edit and navigate code in an editor using voice recognition.

**Problems**

- Speech recognizers don’t do Java
- Compilers can’t deal with ambiguities of natural language

<table>
<thead>
<tr>
<th>Homophones</th>
<th>Many Parses</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 bar</td>
<td>foo 4 bar</td>
</tr>
<tr>
<td>for</td>
<td>foo for foo</td>
</tr>
<tr>
<td>fore</td>
<td>foo() . bar</td>
</tr>
</tbody>
</table>

1. **User Speaks in Spoken Java**
   
   ```bash
   if args dot length equals two then
   file two load gets args sub one end if
   ```

2. **Computer Translates to Java**
   
   ```java
   if (args.length == 2)
   filetoload = args[1];
   ```

## Supports

- C
- C++
- Java
- XML
- Scheme
- Common Lisp
- Cobol
- SRCL