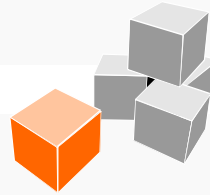
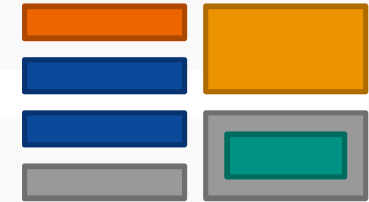




Twente Research and  
Education on Software  
Engineering, Universiteit  
Twente



*Software  
Technology  
Group*  
TU Darmstadt | FB Informatik



www.alia4j.org

# Developing a Generic Debugger for Advanced-Dispatching Languages

Haihan Yin, [h.yin@student.utwente.nl](mailto:h.yin@student.utwente.nl)

Christoph Bockisch, [c.m.bockisch@cs.utwente.nl](mailto:c.m.bockisch@cs.utwente.nl)



# Advanced Dispatching



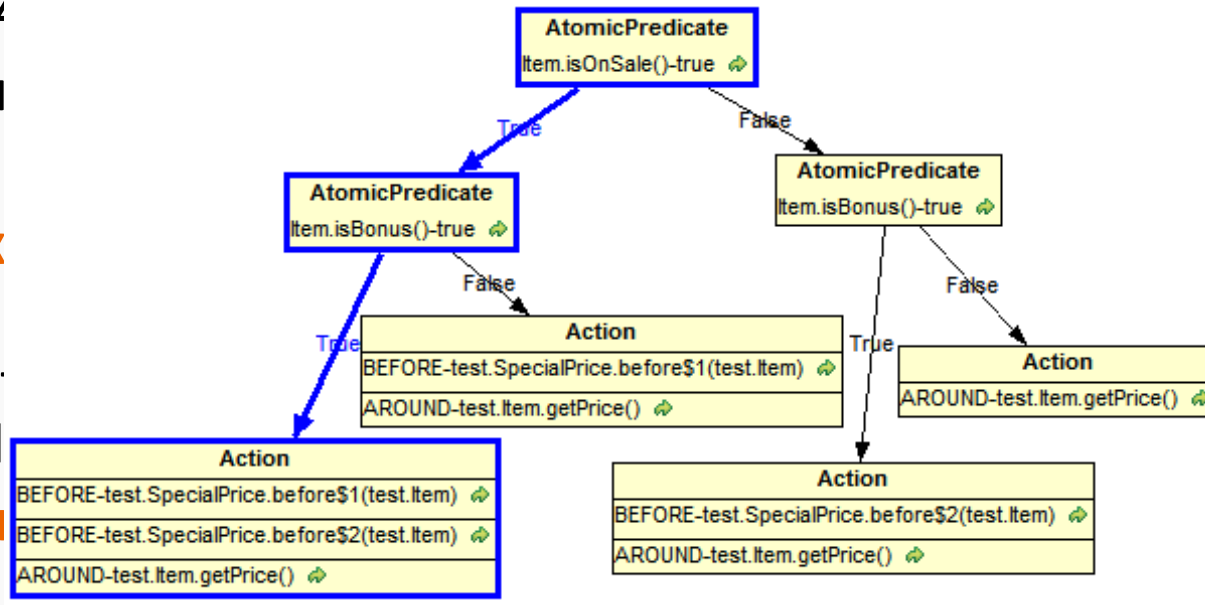
- Dispatching (e.g., virtual method calls)
  - **Late-bind** functionality to dispatch site (invocation)
  - **Alternative** target functionalities (overriding methods)
  - Choose based on **runtime state** (dynamic receiver type)
- Advanced dispatching
  - Composite functionality
  - More complex runtime states
  - **Underlying mechanism of many programming languages**
  - Examples: predicate dispatching, aspect-oriented programming, domain-specific languages

# Language-Implementation Architecture



- ALIA4J

- Support
- Language
- Execution
- Diagnosis
- Support



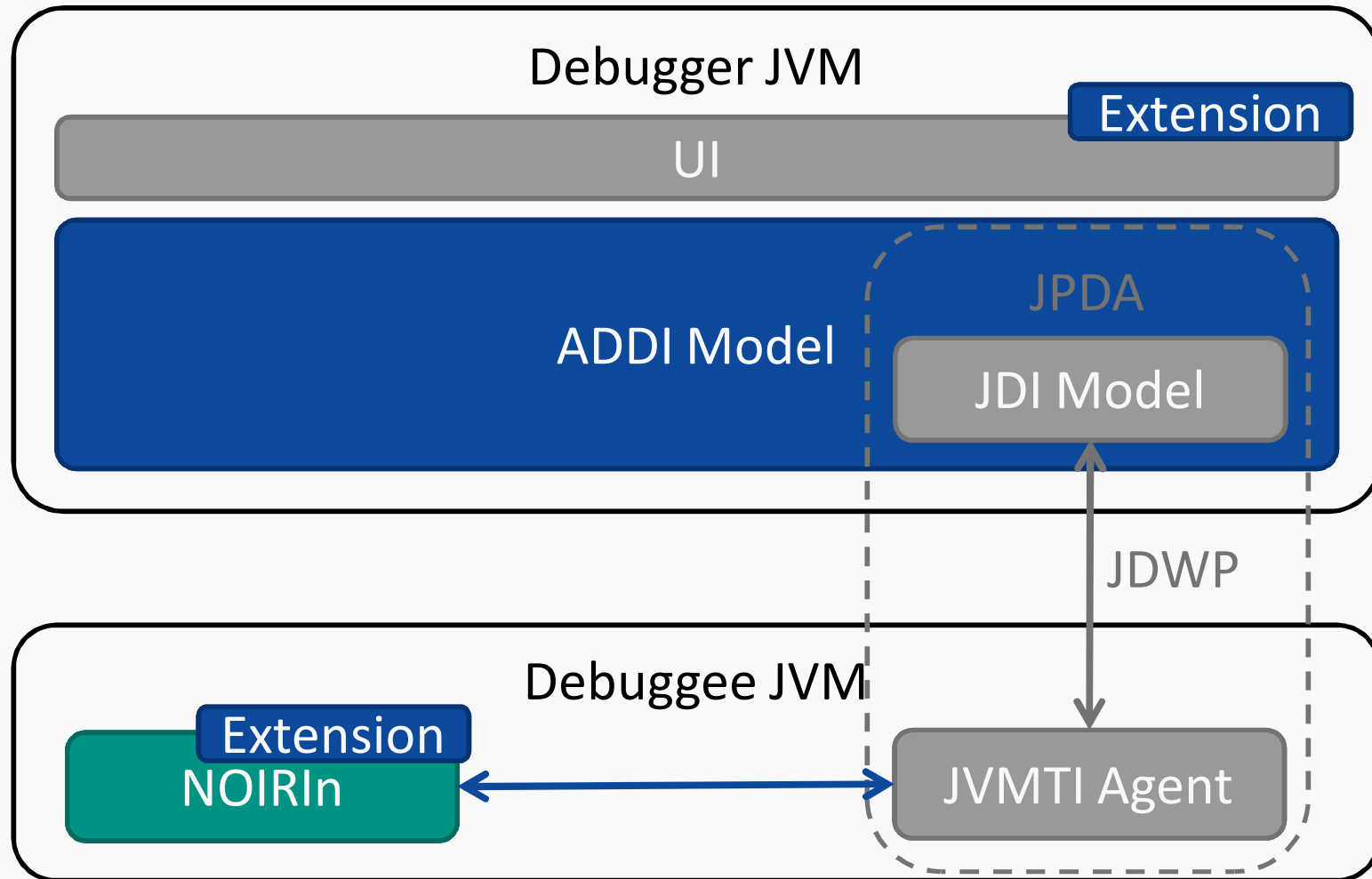
Architecture for Java)

-class at  
preprocessor

- ALIA4J development

- Collaboration with University of Darmstadt, Germany
- Build process using Apache Maven
- Test-case generation (more than 3700 integration tests)
- Eclipse Public License

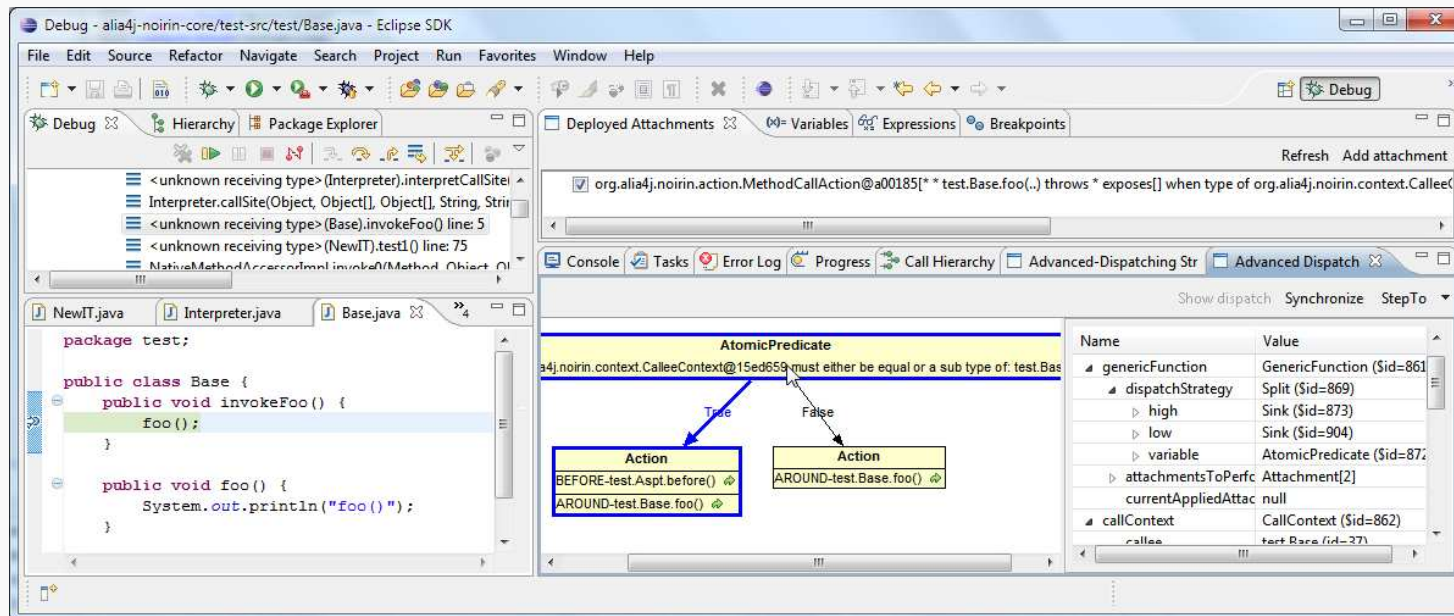
# Advanced-Dispatching Debug Architecture



# Implementation



- Based on **Eclipse Java debugger**
  - Retain functionality for base language
  - Added views
  - **Zest** graph-visualization toolkit

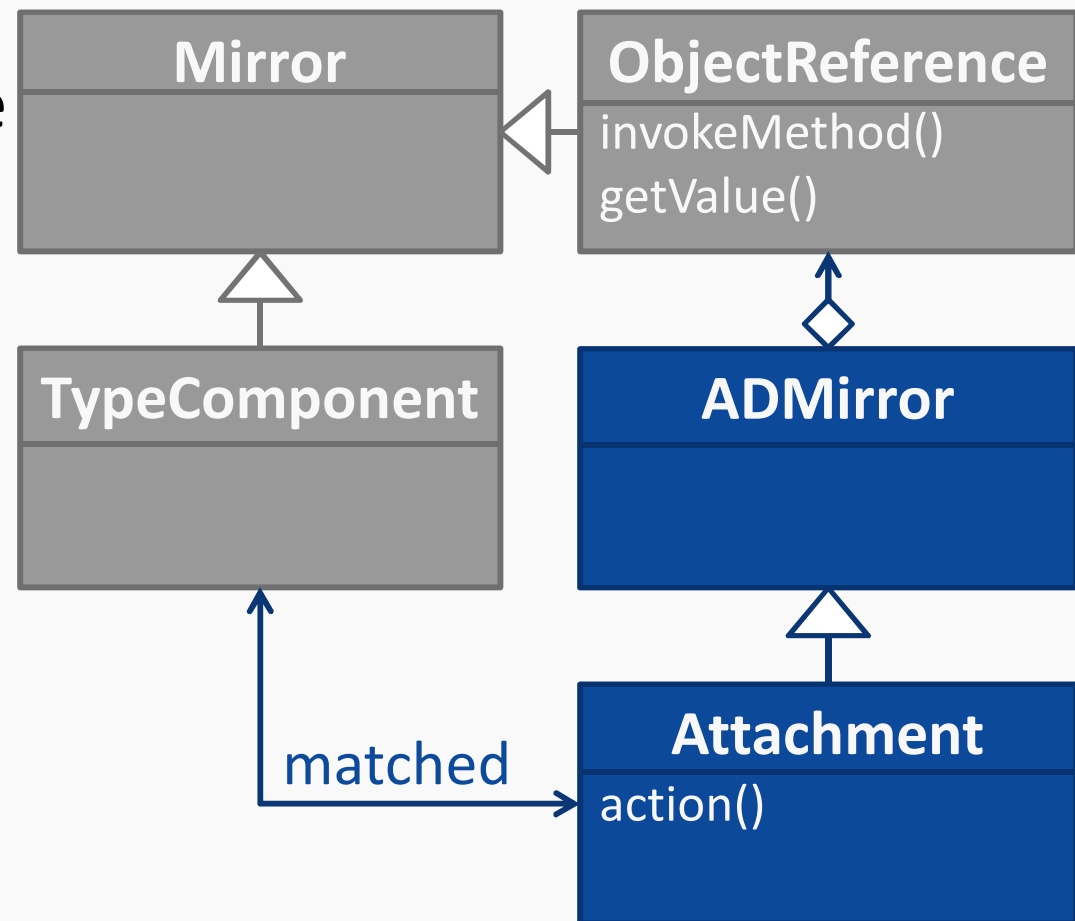


# Implementation – continued



- Advanced-Dispatching Debug Interface

- On top of Java Debug Interface
- ADDI entities** equal to **ALIA4J model**
- New entities **wrap** plain JDI objects
- Added associations** to JDI entities



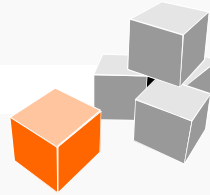
# Future Work (How to Extend Eclipse)



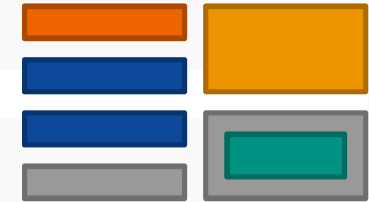
- Currently
  - **Copy and modify** Eclipse's JDI implementation
  - Useful extensions to existing Eclipse UI **omitted**
- Plan to use **Equinox Aspects**
  - Include AspectJ aspect in plug-in to
  - Influence another plug-in
  - **Clear separation**
  - E.g., AspectJ Development Tools integrated with Java Development Tools in this way



Twente Research and  
Education on Software  
Engineering, Universiteit  
Twente



*Software  
Technology  
Group*  
TU Darmstadt | FB Informatik



www.alia4j.org

<http://www.alia4j.org>





# Related Work



- Tools for specific languages
  - AspectJ, CaesarJ, ObjectTeams, JAsCo, JPred
  - **No dynamic debugger**
- IDE Meta-tooling Platform
  - Re-usable infrastructure (includes debugging)
  - **No back-end**
- TIDE
  - Generic debugging framework
  - Must be instantiated per concrete language
- PIL
  - Intermediate layer between DSL and target code