

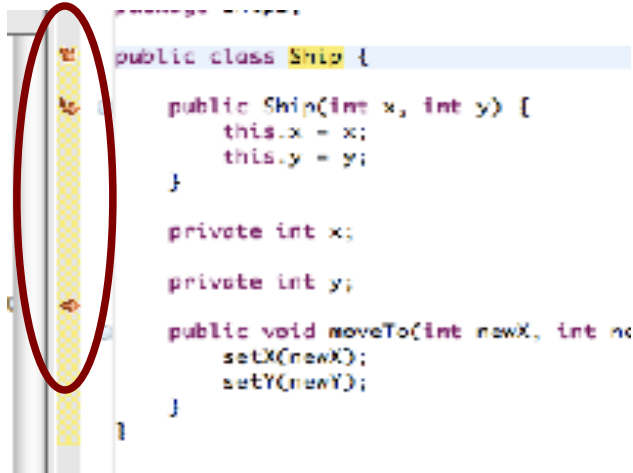
Aspects everywhere

AJDT's deep integration with Eclipse

Andrew Eisenberg

Aspects everywhere?

Aspects are pervasive
In the editor:

A screenshot of an IDE showing a Java class named 'Ship'. The code is as follows:

```
public class Ship {  
    public Ship(int x, int y) {  
        this.x = x;  
        this.y = y;  
    }  
    private int x;  
    private int y;  
    public void moveTo(int newX, int newY) {  
        setX(newX);  
        setY(newY);  
    }  
}
```

The gutter area on the left side of the editor is highlighted with a yellow background and contains several small icons. A red oval is drawn around this gutter area, indicating the presence of aspects.

AJDT 1.6.1



-
- DEMO!

Problem!

- Aspects are hidden from developer
 - searching
 - content assist
 - type hierarchies
- JDT is not properly extensible!

Solution!

- Use AspectJ to make JDT to more extensible

- Weaves into JDT
 - some few, but critical locations
- Exposes pointcuts through extension points
- Uses:
 - Load time weaving
 - Equinox aspects
 - P2 installer

- DEMO

Aspects exposed as extension points



- **CompilationUnitProvider**
 - for adding to the Java model
- **SourceProvider**
 - for indexing
- **ImageDescriptorProvider**
 - for icons
- **ITD awareness**
 - for type hierarchy calculation
 - content assist

It's generic!

- Any editor for a JVM language can use
 - e.g., Scala
- Editor plugins need not be AspectJ aware
 - no dependency on AJDT
 - (only the JDT Weaving plugin)

Aspects Everywhere!



Can I get it?

- Update site (testing only):
 - <http://download.eclipse.org/tools/ajdt/34/dev/weaving>
- In production for AJDT 1.6.2
 - late December

andrew.eisenberg@springsource.com