
*The Future and Potential of
Aspect-Oriented Programming*

A Program Algebra for AOP

Christian Lengauer

Department of Informatics and Mathematics



Assertions

Assertion 1:

AOP is a new and fundamental form of modular programming.

Assertion 2:

Its effects are not well understood and its uses may not be pervasive.

Assertion 3:

AOP is more than AspectJ.

Assertion 4:

AOP can be as harmful to the program structure as the GOTO.

- Weaving potentially destroys the structure of a program.
- Any change at any place: around advice without proceed.
- Names can get “captured”: overriding with inter-type declarations.

Assertion 5:

Feature-oriented programming is incremental program development by adding features. FOP can be implemented via AOP, but it can shield the programmer from the dangers of AOP by imposing a discipline.

Assertion 6:

FOP can be reasoned about in a program algebra. The main operations are (1) adding code and (2) applying advice to pointcuts (weaving).

A Program Algebra for FOP

weave :: feature expression $\rightarrow$ module expression

Module Expressions.

- The operands are
 - a base program,
 - introductions (inter-type declarations),
 - advice.
- The operations are
 - introduction sum ::
introduction $\rightarrow$ module expression $\rightarrow$ module expression
 - advice application ::
advice $\rightarrow$ module expression $\rightarrow$ module expression

Feature Expressions.

- A *feature* is an advice/introduction pair.
- A *feature expression* is a sequence of features.

weave :: [feature] $\rightarrow$ base program $\rightarrow$ module expression

Weaving Tactics

Different weaving orders can be defined and compared.

What is $\text{weave } [(a1,i1),(a2,i2)] \ b?$

- $a2(a1(i2 + i1 + b))$

This is the weaving tactic of AspectJ.
It is called *unbounded quantification*

- $a2(i2 + a1(i1 + b))$

This is called *bounded quantification*.

- $i2 + a2(i1 + a1(b))$

This is the weaving tactic of AHEAD.
It uses bounded quantification and
shields introductions from advice.

There is such a thing as higher-order advice:

$\text{2nd-order advice} :: \text{advice} \rightarrow \text{advice}$