



Semantics in Aspectj

Vitus Lorenz-Meyer, 17.10.2005

Overview

- 1) Program Structure
 - Classes
 - Aspects
- 2) Semantics
 - Call
 - Execution
 - Subtype Patterns
- 3) Alternatives
- 4) Conclusion
- 5) Questions

Classes

```
public class A1
{
    public void f() {}
    public void g() {}
}

public class A2 extends A1
{
    public void h() {}
}

public class A3 extends A2
{
    public void f() {}
}
```

Aspects

```
pc c_a1f() : call(void A1.f());
pc c_a1g() : call(void A1.g());
pc c_a3f() : call(void A3.f());
pc c_a3g() : call(void A3.g());
after() : c_a1f(){ sysout(thisJoinPoint); }
...
pc e_a1f() : execution(void A1.f());
pc e_a1g() : execution(void A1.g());
pc e_a3f() : execution(void A3.f());
pc e_a3g() : execution(void A3.g());
...

pc s_a1h() : call(void A1+.h());
pc s_a3h() : call(void A3+.h());
...
```


Semantics

	F			G			H	
	S1	S3	S1d3	S1	S3	S1s3	S3	S1d3
A1.f()	•➤	•➤	•➤					
A1.g()				•➤	•➤	•➤		
A3.f()		•➤	➤					
A3.g()								
A1+.h()							•➤	
A3+.h()								

- call() JoinPoint
- execution() JoinPoint

Semantics – AJ 1.5M4

	F			G			H	
	S1	S3	S1d3	S1	S3	S1s3	S3	S1d3
A1.f()	•➤	•➤	•➤					
A1.g()				•➤	•➤	•➤		
A3.f()		•➤	➤					
A3.g()					•			
A1+.h()							•➤	•➤
A3+.h()							•	•

- call() JoinPoint (AJ 1.5M4)
- execution() JoinPoint (AJ 1.5M4)

Static Semantics

	F			G			H	
	S1	S3	S1d3	S1	S3	S1s3	S3	S1d3
A1.f()	•▶		•					
A1.g()				•▶		•		
A3.f()		•▶						
A3.g()					•▶			
A1+.h()							•▶	•▶
A3+.h()							•▶	•▶

• call() JoinPoint

▶ execution() JoinPoint

Conclusion

- Current AspectJ Semantics unintuitive
- Version 1.5.0 M4 does not fix everything
- Better Alternatives?
- Semantics ultimately depend on what Users want
 - What do you think?

¿ Questions ?