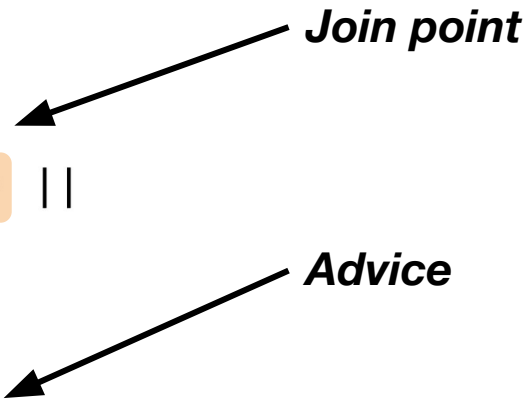
```
aspect Logging {
    pointcut move():
        call(void FigureElement.setXY(int, int) ||
        call(void FigureElement.setX(int) ||
        call(void FigureElement.setY(int) ||

    before(): move() {
        System.out.println("Shape is moving");
    }
}
```



"Where is this code even coming from?

Aspect-Oriented Programming
Kiczales et al., European Conference on Object-Oriented Programming (ECOOP) 1997


But what about cross-cutting concerns?!
LISP is gonna take over the world!
We’re gonna make so much $$$ off the GUI

Why AOP?

- Cross-cutting concerns are **annoying**
  - logging
  - profiling/performance tuning
  - security
- Cross-cutting concerns are **everywhere**
  - code that isn't really related to core business logic is intermingling with business logic **– *tangling***
  - unrelated code appears everywhere in the system **– *scattering***

Why AOP, WHY??

- Cross-cutting concerns are **annoying**
  - logging
  - profiling/performance tuning
  - security
- Cross-cutting concerns are **everywhere**
  - code that isn't really related to core business logic is intermingling with business logic **– *tangling***
  - unrelated code appears everywhere in the system **– *scattering***

Solution: make encapsulating cross-cutting concerns a first-class feature!

Aspects?

Aspects tend not to be units of the system's functional decomposition, but rather to be properties that affect the performance or semantics of the components in systemic ways. Examples of aspects include memory access patterns and synchronization of concurrent objects.

Discussion

- What did we like about this paper?
- What did we we wish this paper did more of?
- What are your thoughts on cross-cutting concerns?
- What are your thoughts on Section 5.4 (Results)?

$$\text{reduction in bloat due to tangling} = \frac{\text{tangled code size - component program size}}{\text{sum of aspect program sizes}} = \frac{35213 - 756}{352} = 98$$

- Is there anything like Aspect-Oriented Programming today?
- Does AOP sound like a decent approach?
  - Pros?
  - Cons?