

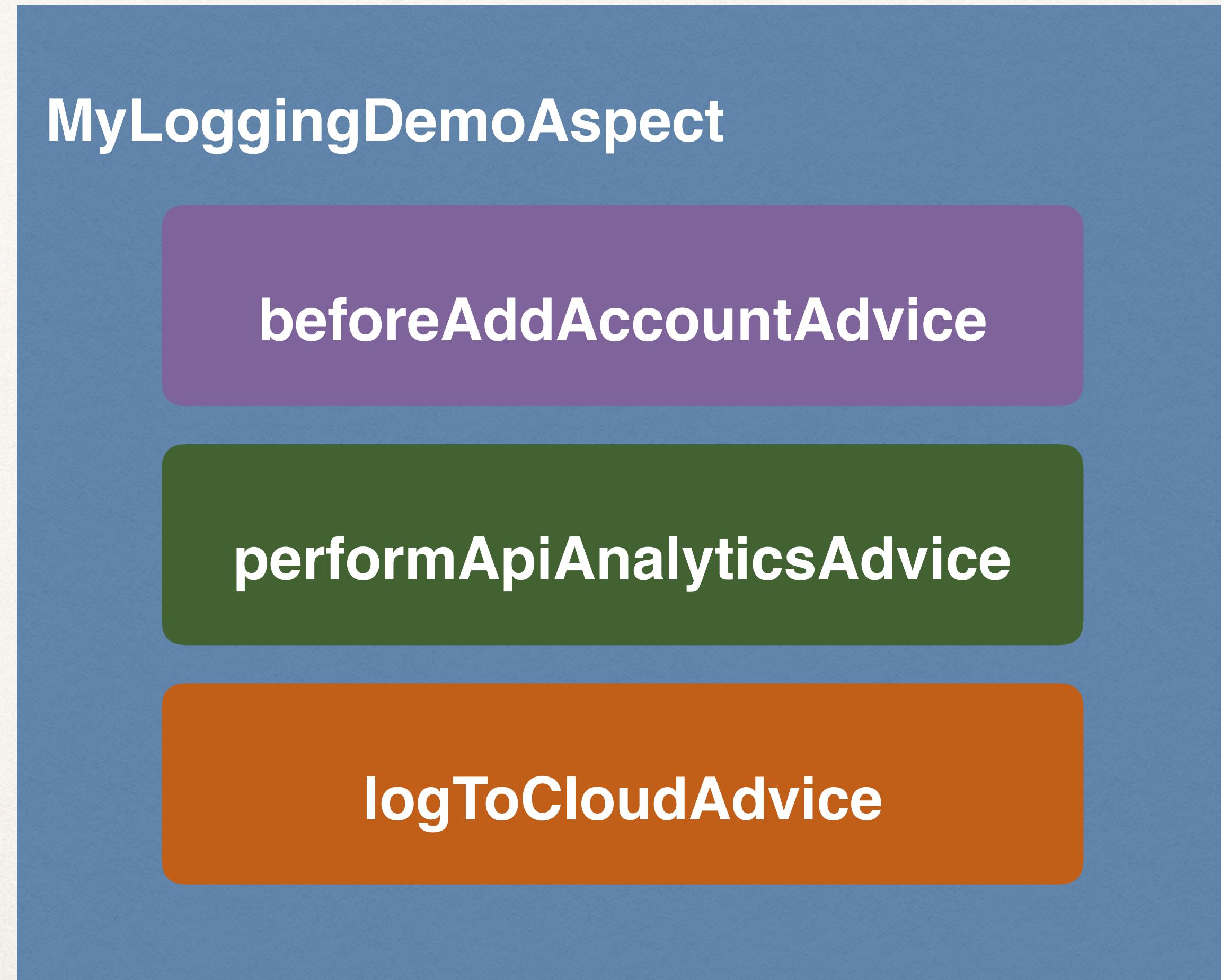
Aspect-Oriented Programming (AOP)

Control Aspect Order



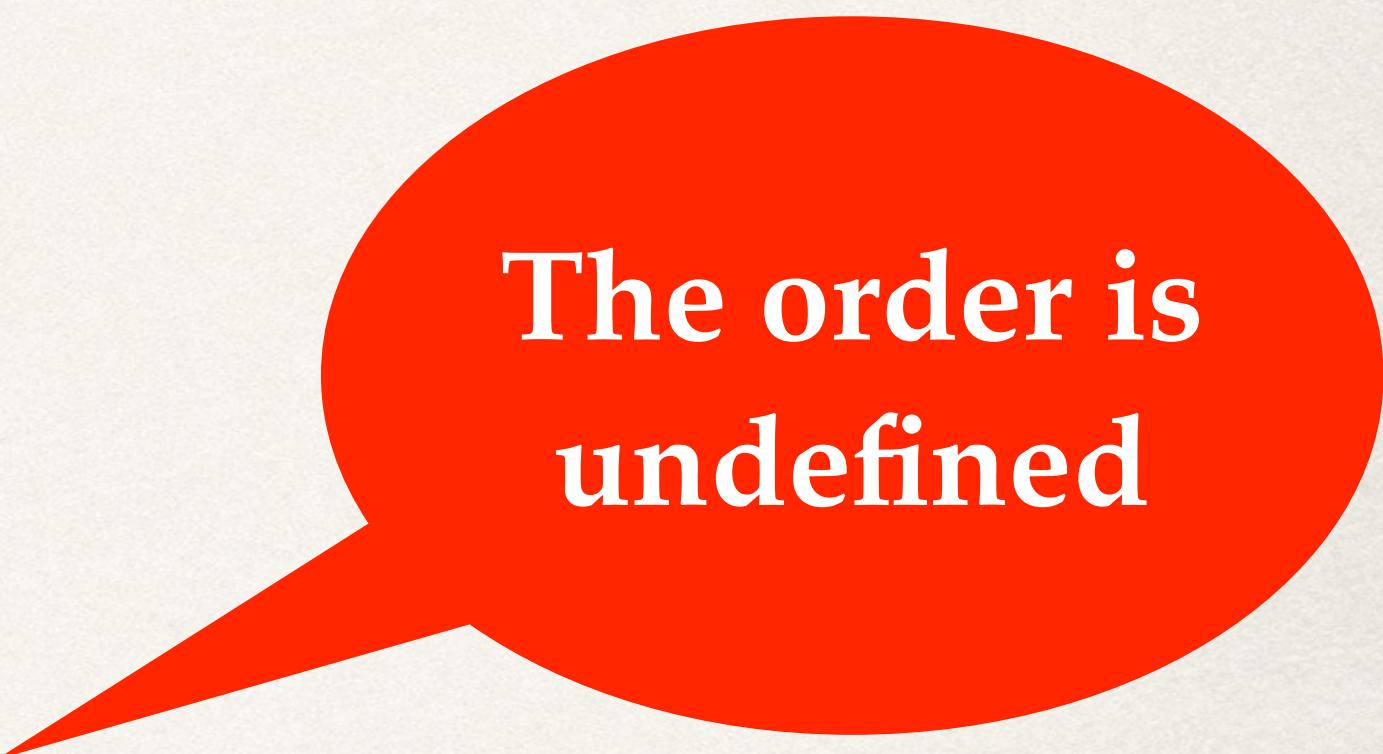
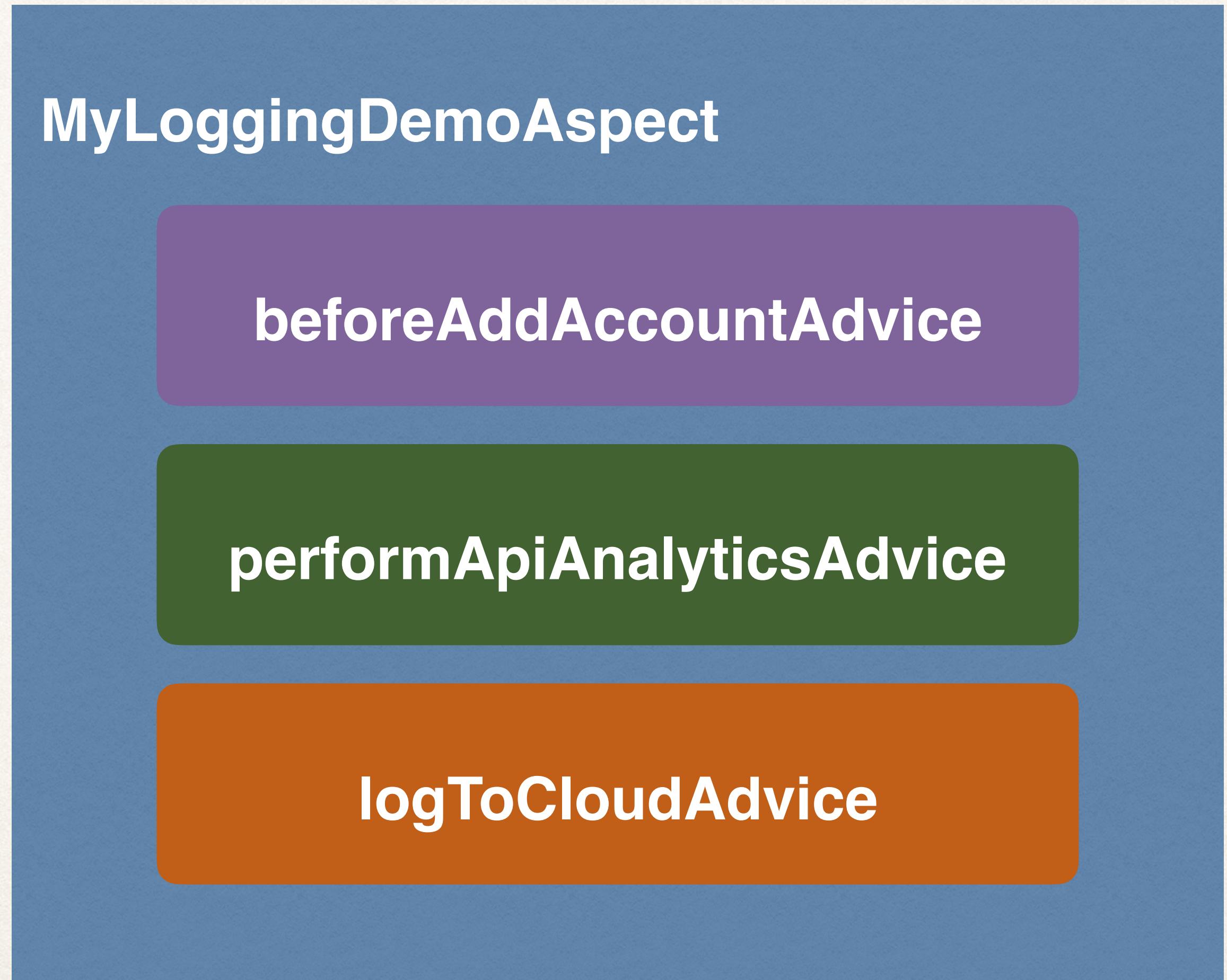
Problem

- How to control the order of advices being applied?



Problem

- How to control the order of advices being applied?



To Control Order

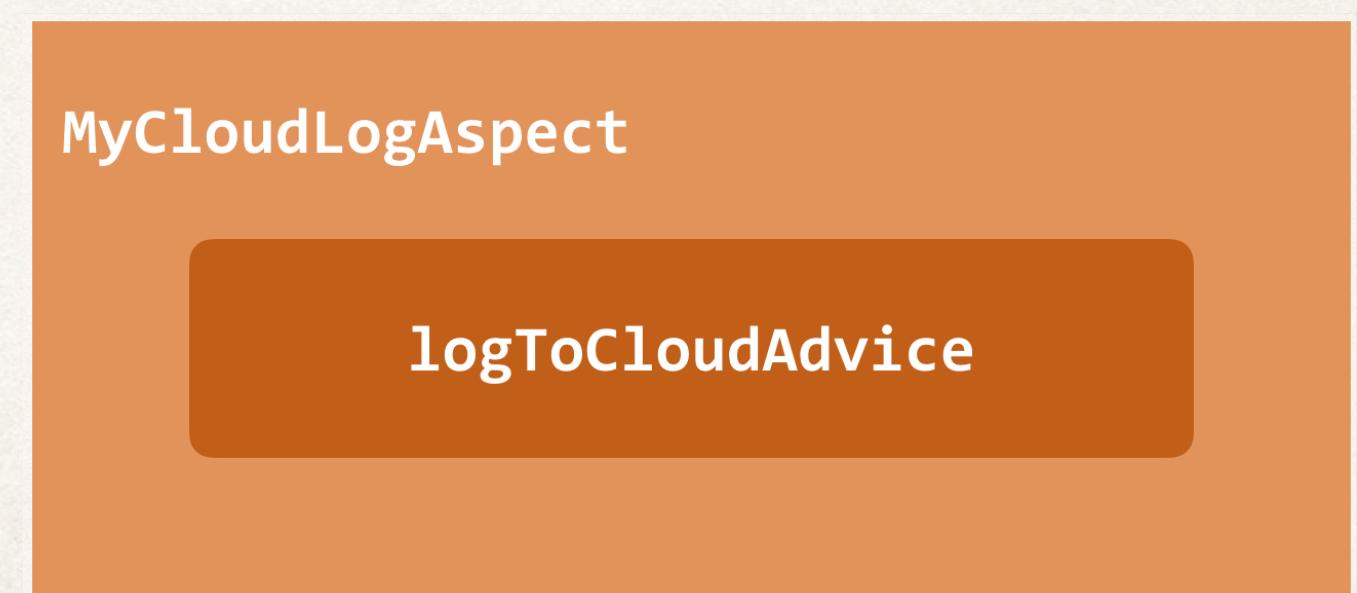
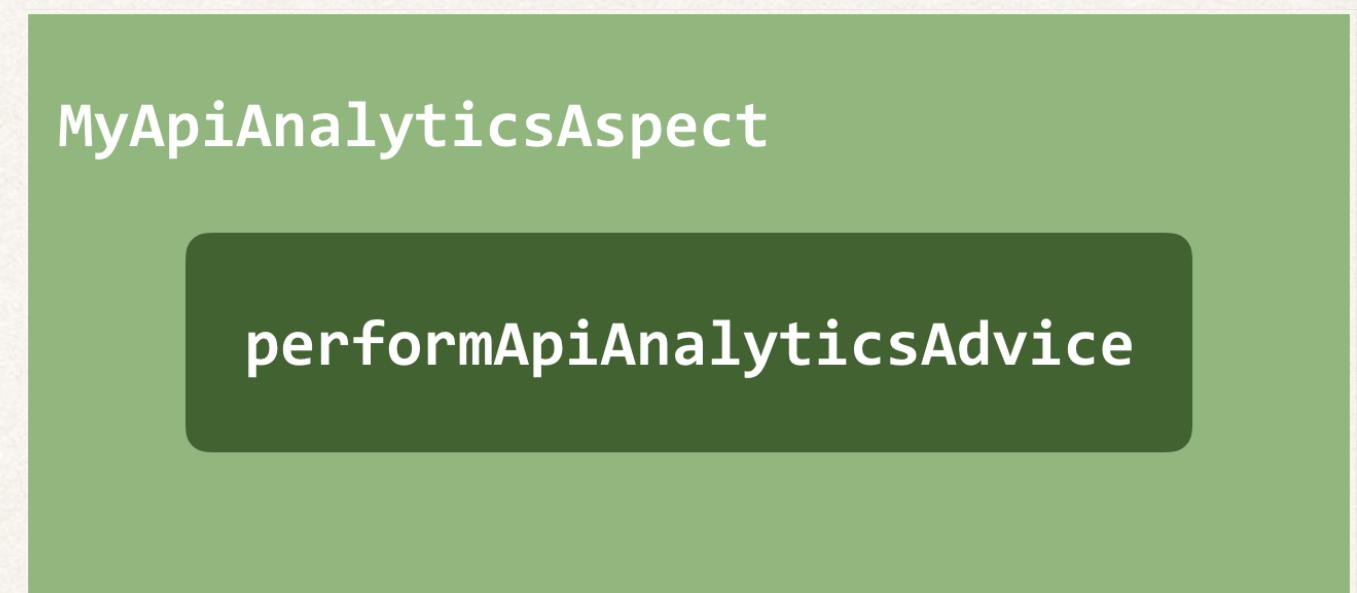
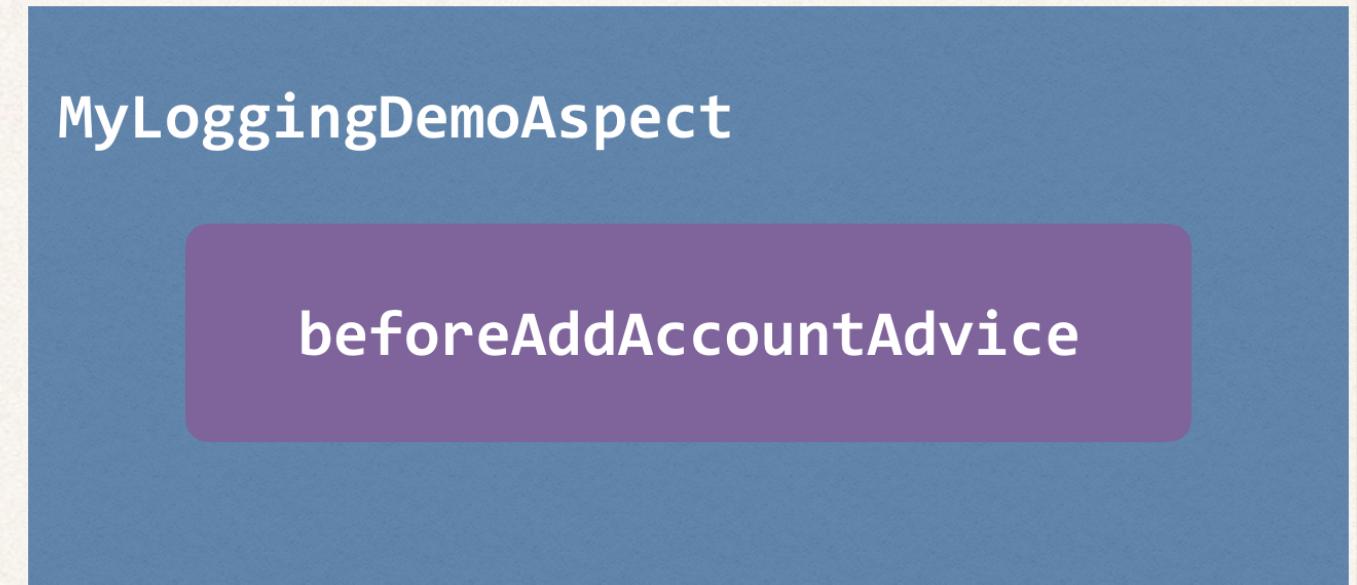
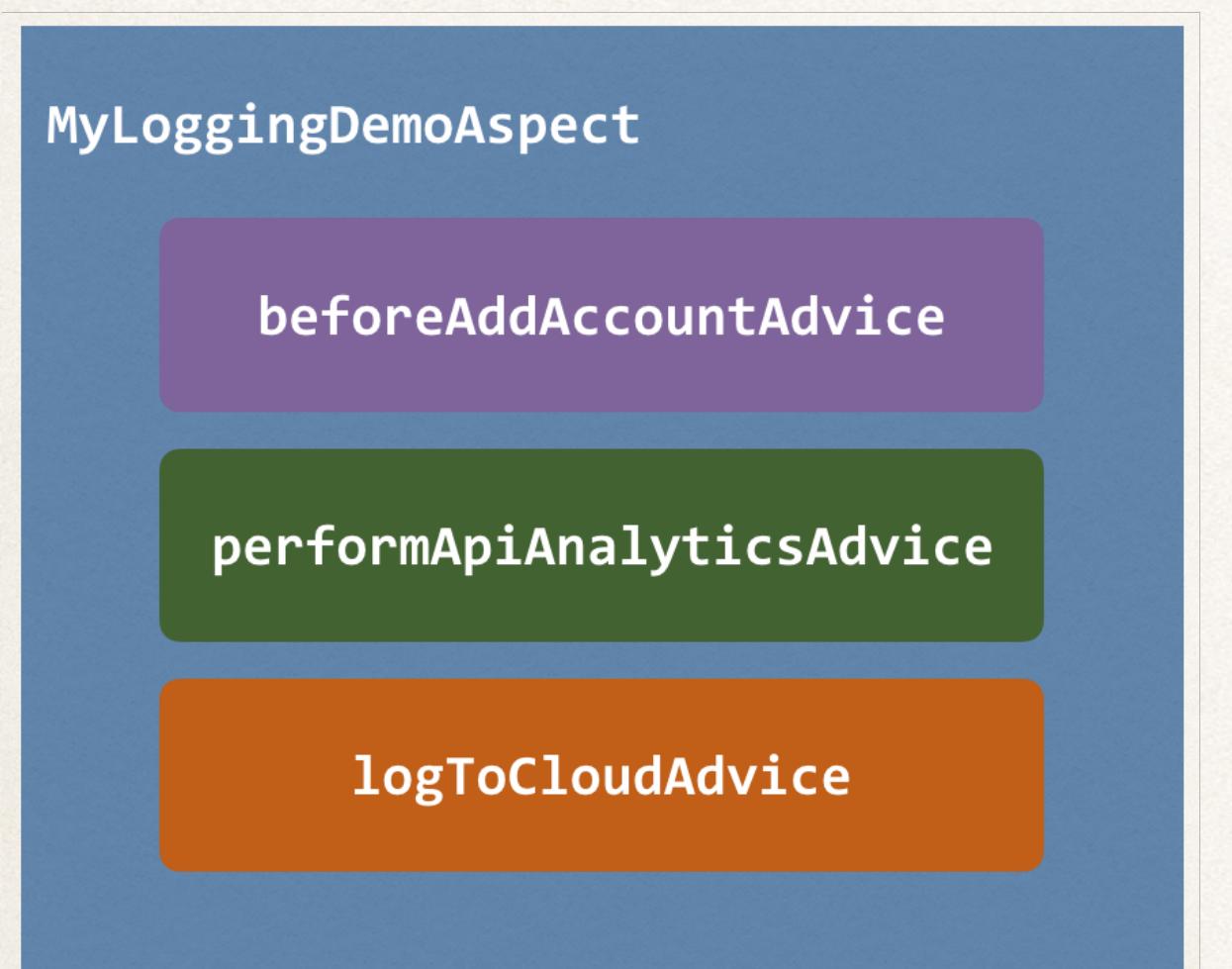
- Refactor: Place advices in separate Aspects
- Control order on Aspects using the @Order annotation
- Guarantees order of when Aspects are applied

Development Process

Step-By-Step

1. Refactor: Place advices in separate Aspects
2. Add @Order annotation to Aspects

Step 1 - Refactor: Place advices in separate Aspects



Step 2 - Add @Order annotation

- Control order on Aspects using the @Order annotation



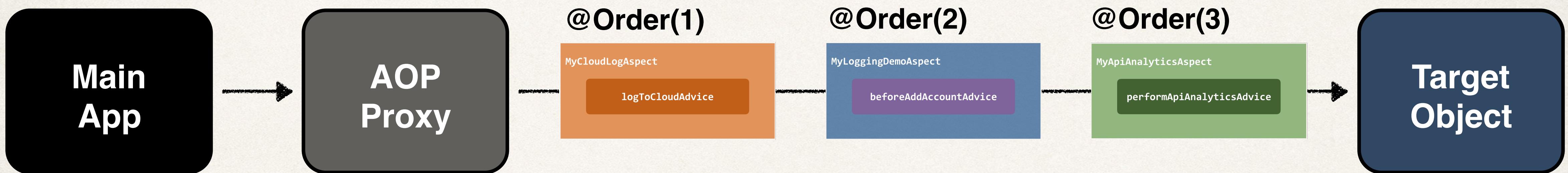
```
@Order(1)
public class MyCloudLogAspect {
    ...
}
```

- Guarantees order of when Aspects are applied
- Lower numbers have higher precedence

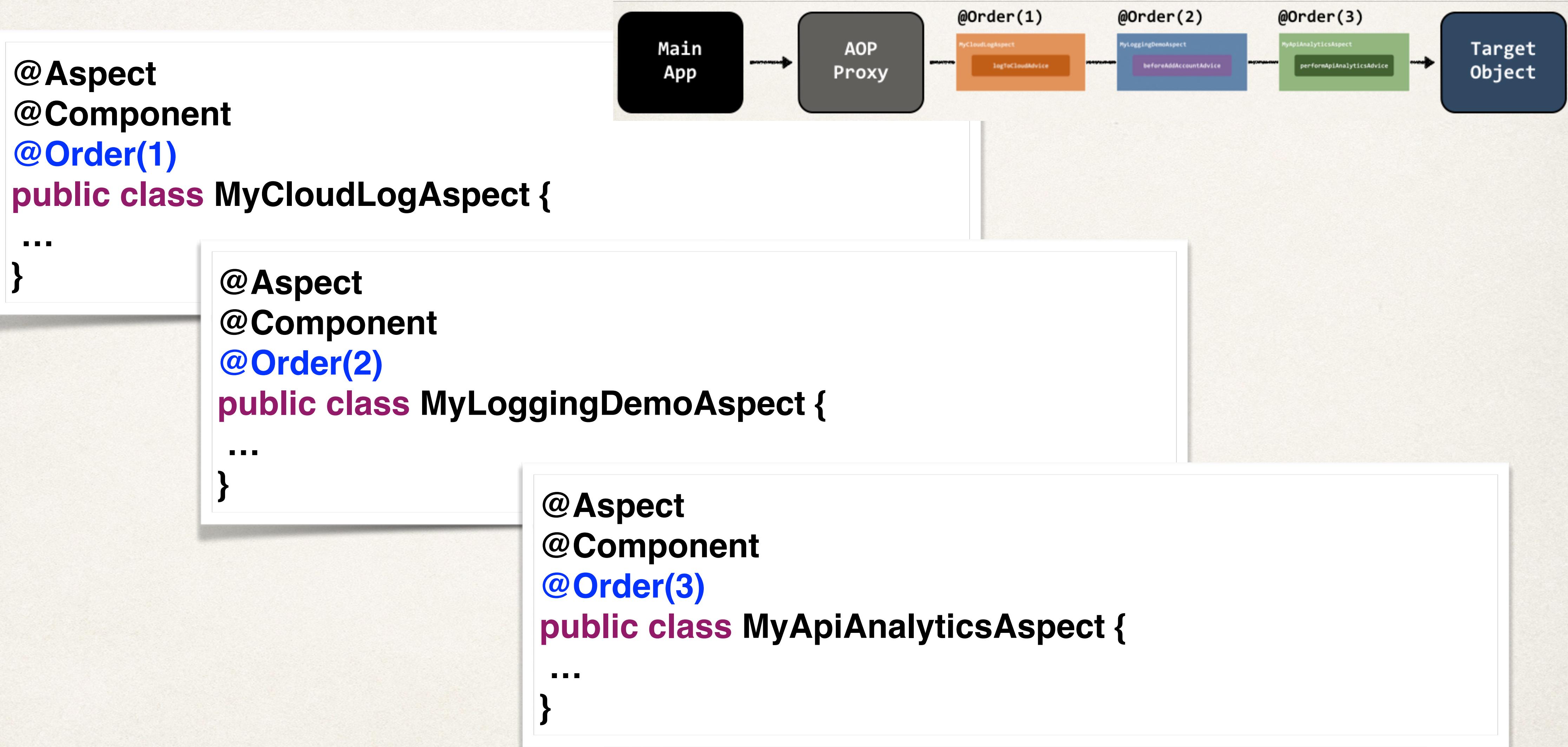
@Order

- We want the following order:

1. MyCloudLogAspect
2. MyLoggingDemoAspect
3. MyApiAnalyticsAspect

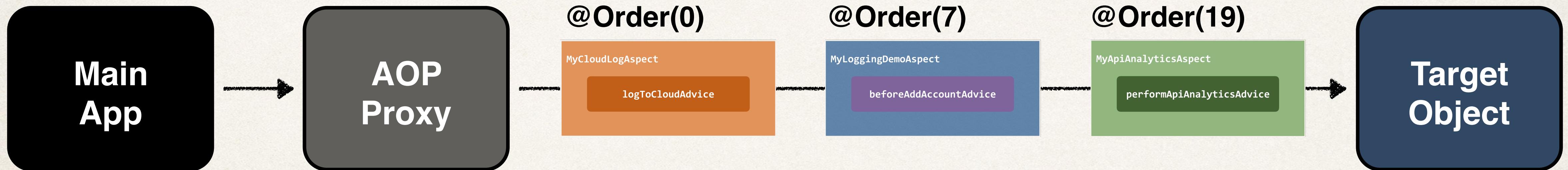


@Order annotation



@Order annotation

- Lower numbers have higher precedence
 - Range: Integer.MIN_VALUE to Integer.MAX_VALUE
 - Negative numbers are allowed
 - Does not have to be consecutive



@Order annotation

- Example with negative numbers

