

Thanks to:

Java User Group





Only Better.



Modularity with Spring Modulith

Agenda

- What is modularity? (and why should you care?)
- How is modularity (or “logical partitioning”) reflected in the codebase?
- (Simple) Demo (Spring Modulith)

What is Modularity?

In software design, modularity refers to a logical partitioning of the "software design" that allows complex software to be manageable for the purpose of implementation and maintenance. The logic of partitioning may be based on related functions, implementation considerations, data links, or other criteria.

Source: <https://en.wikipedia.org/wiki/Modularity>

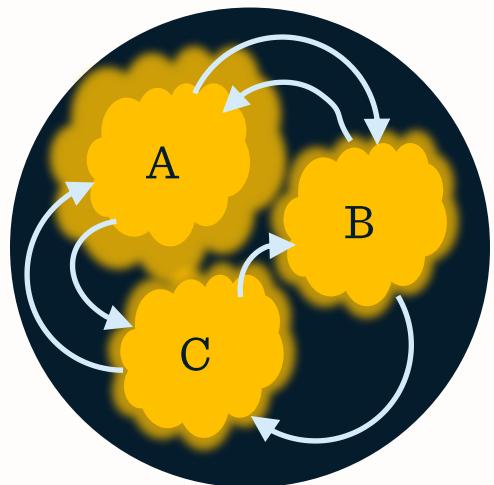
Why Modularity?

“ Good software
architecture enables
agility. ”

Simon Brown

Why Modularity?

Monolith vs Microservice

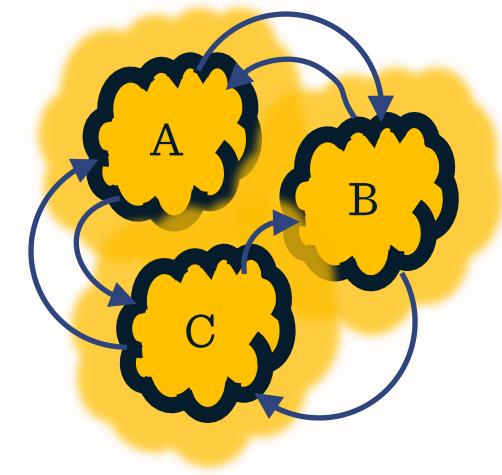


Bounded Context

Deployment Unit

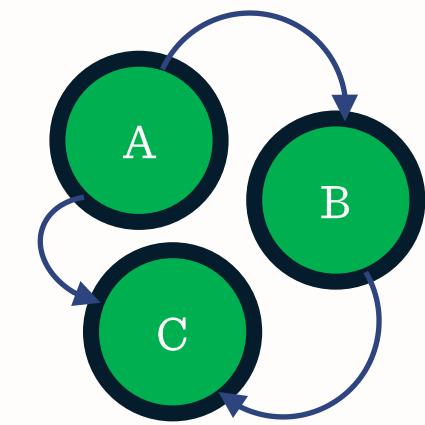
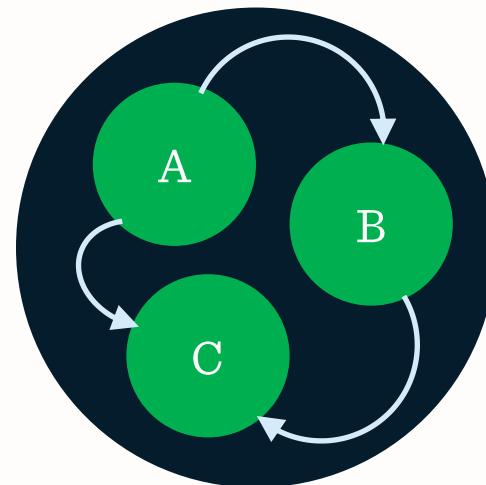
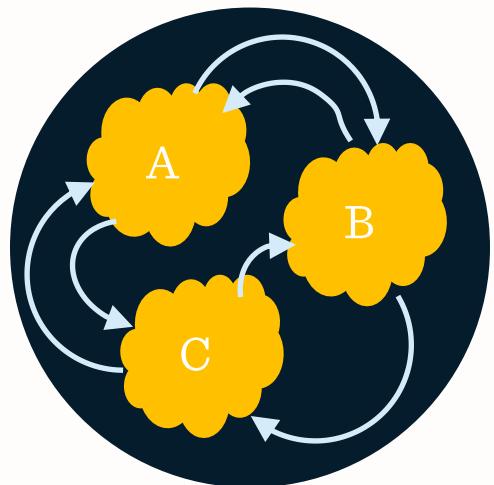
Internal Invocation

External Invocation



Why Modularity?

Monolith vs Microservice



Bounded Context

Deployment Unit

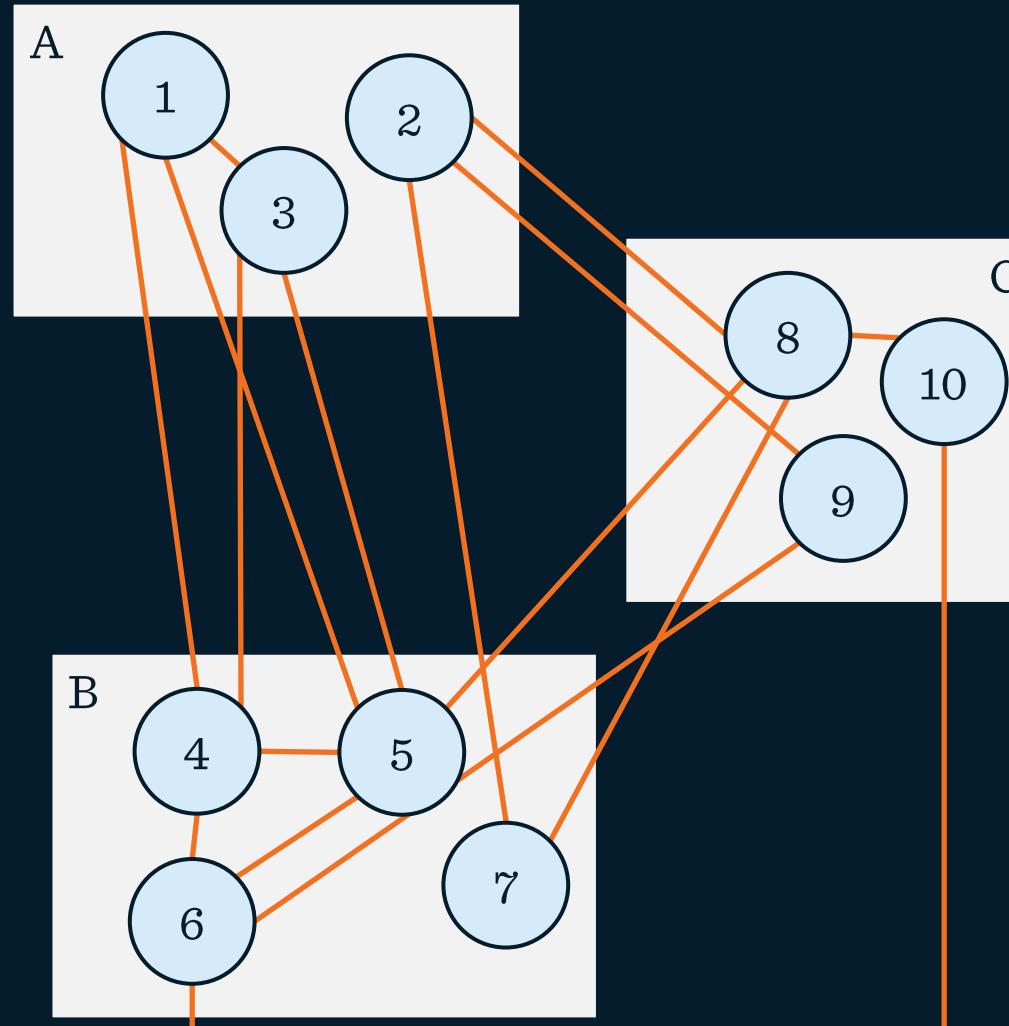
Internal Invocation

External Invocation

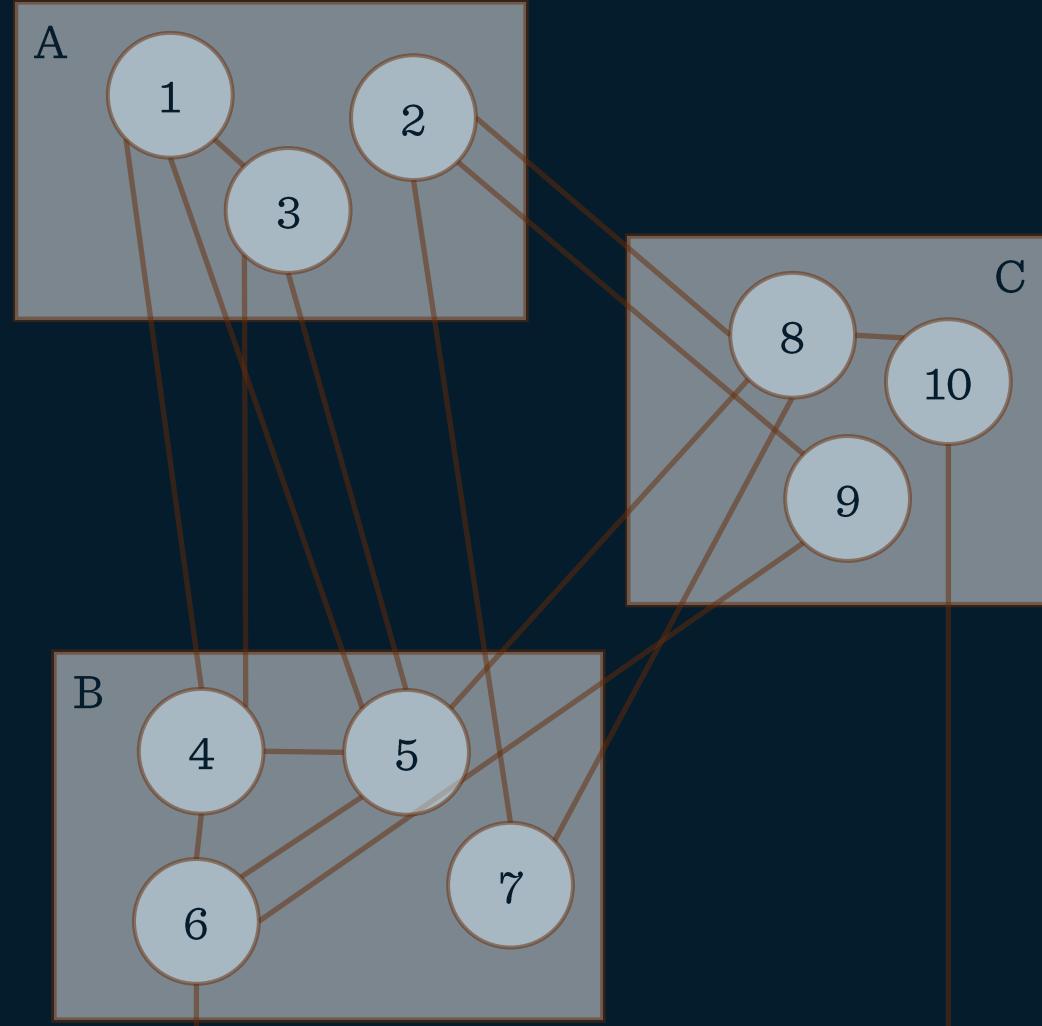
Why Modularity?

- Good software architecture enables agility – Simon Brown
- Good modularity is considered good software architecture.
Therefore, it enables agility. - Me 😊

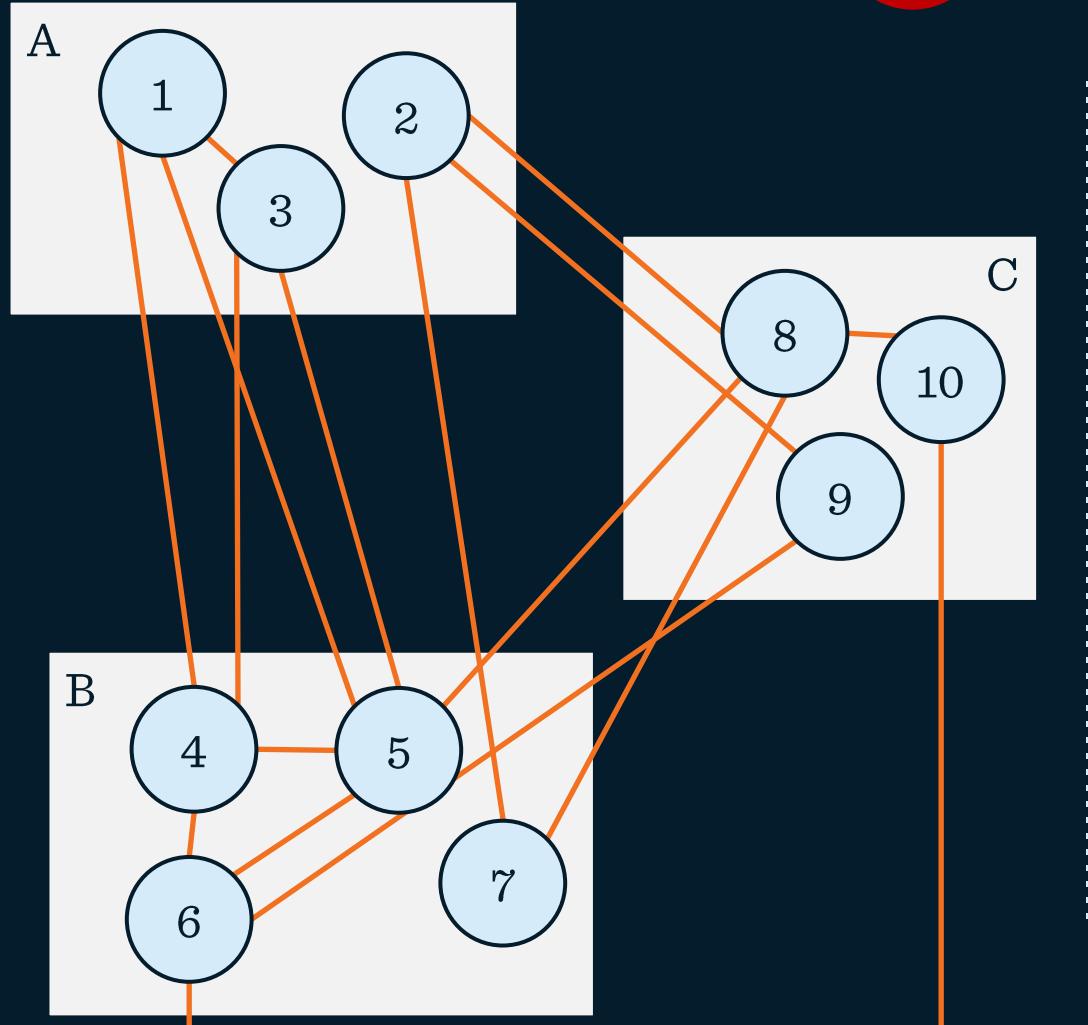
Bad modularization: Low cohesion, high coupling



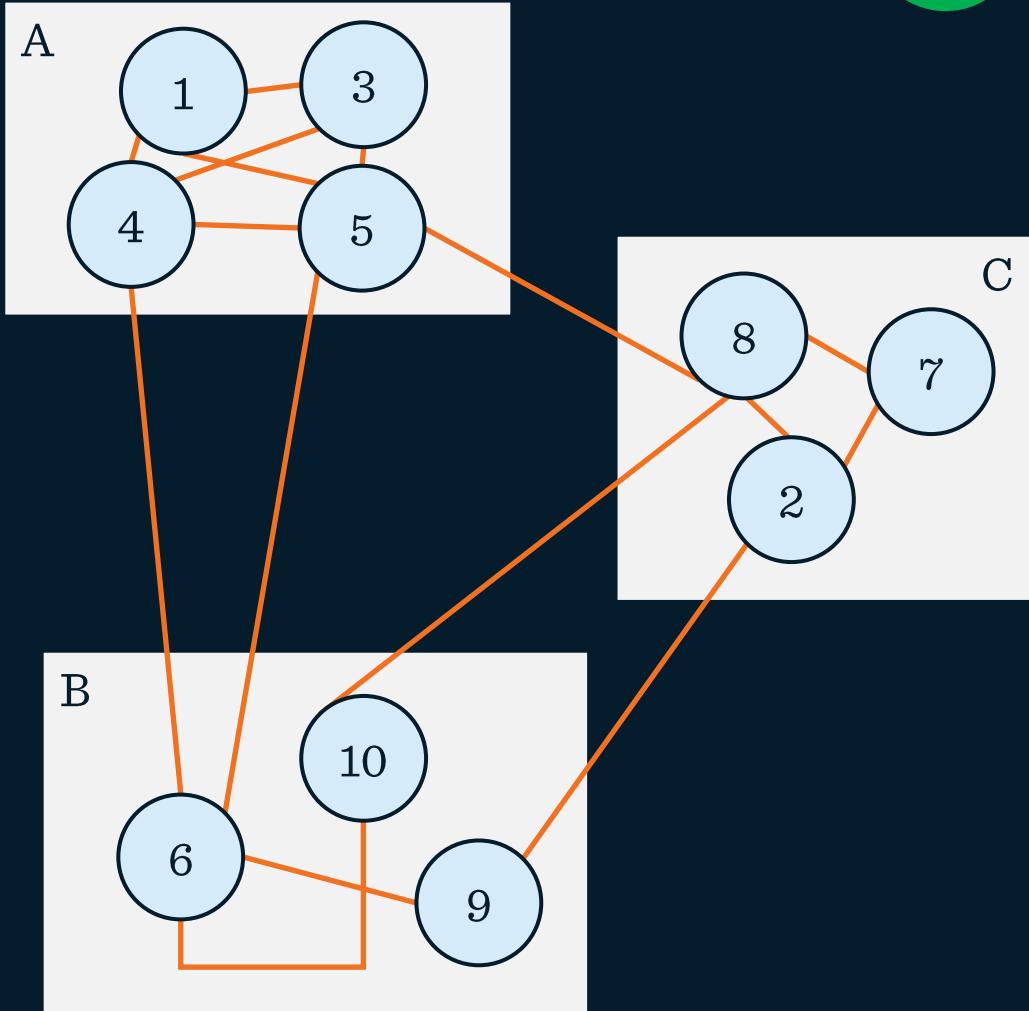
Can we do better?



Bad modularization: Low cohesion, high coupling



Good modularization: High cohesion, low coupling



How is modularity (or “logical partitioning”) reflected in the codebase?



An Example

📁 src/main/java

📦 acme.myapp

📦 acme.myapp.dao

📦 acme.myapp.domain

📦 acme.myapp.service

📦 acme.myapp.web

📁 src/main/java

📦 acme.myapp

📦 acme.myapp.customer

📦 acme.myapp.inventory

📦 acme.myapp.order



An Example

```
src/main/java  
  acme.myapp  
  acme.myapp.dao  
    + CustomerDao  
    + OrderDao  
    + InventoryDao  
  acme.myapp.domain  
    + Customer  
    + Order  
    + OrderItem  
    + Inventory  
  acme.myapp.service  
    + CustomerService  
    + OrderManager  
    + InventoryService
```



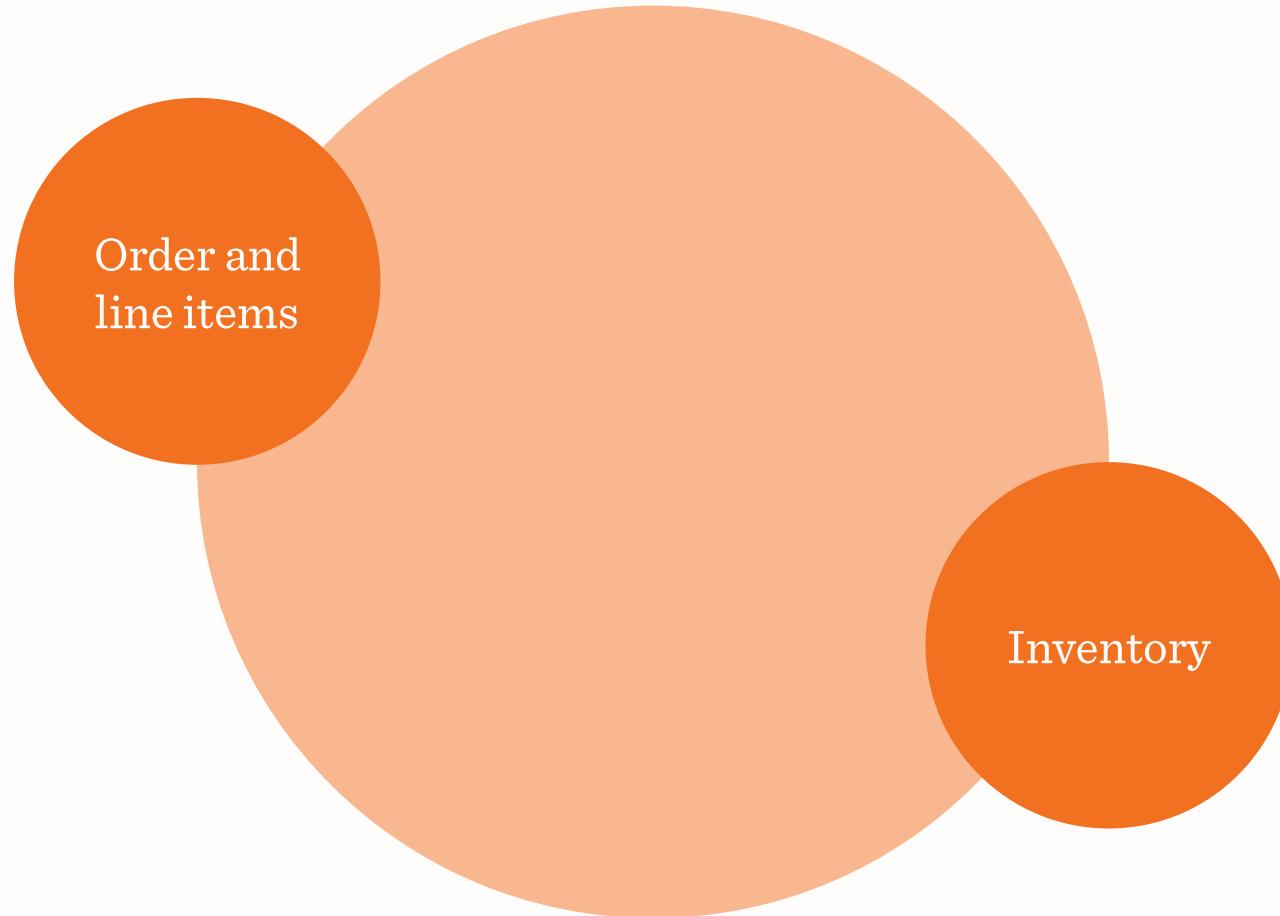
```
src/main/java  
  acme.myapp  
  acme.myapp.customer  
    - Customer  
    - CustomerDao  
    + CustomerService  
  acme.myapp.inventory  
    - Stock  
    - StockDao  
    + Inventory  
  acme.myapp.order  
    - Order  
    - OrderDao  
    - OrderItem  
    + OrderManager
```



Demo

Demo

Sample Domain



Brief Recap

Spring Modulith provides:

- Convention-based module model for Spring Boot applications
- Verification that the functional architecture does not accidentally degrade
- Ability to integration test functional modules individually
- Support to derive architectural documentation from the arrangement
- An opinionated, eventually consistent application module interaction model
- Application observability and actuators driven by the functional architecture

Q&A



Only Better.

Thank you.