

Reactive Streams Using Spring WebFlux 6

Experiencing Reactive Programming



Bogdan Sucaciu

Tech Lead

@bsucaciu | bsucaciu.com

Reactive Streams Using Spring WebFlux 6

Version Check



Version Check



This course was created by using:

- Spring Boot 3.0.7
- Spring WebFlux 6.0.9
- Java 17



Version Check

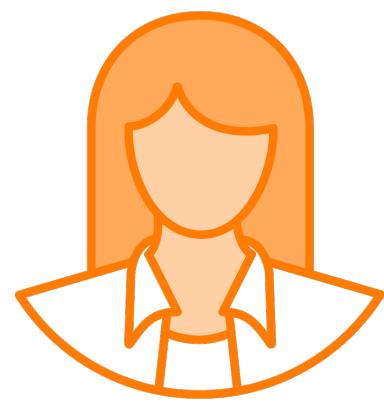


This course is 100% applicable to:

- Spring Boot 3
- Spring WebFlux 6
- Java 17



Globomantics



User



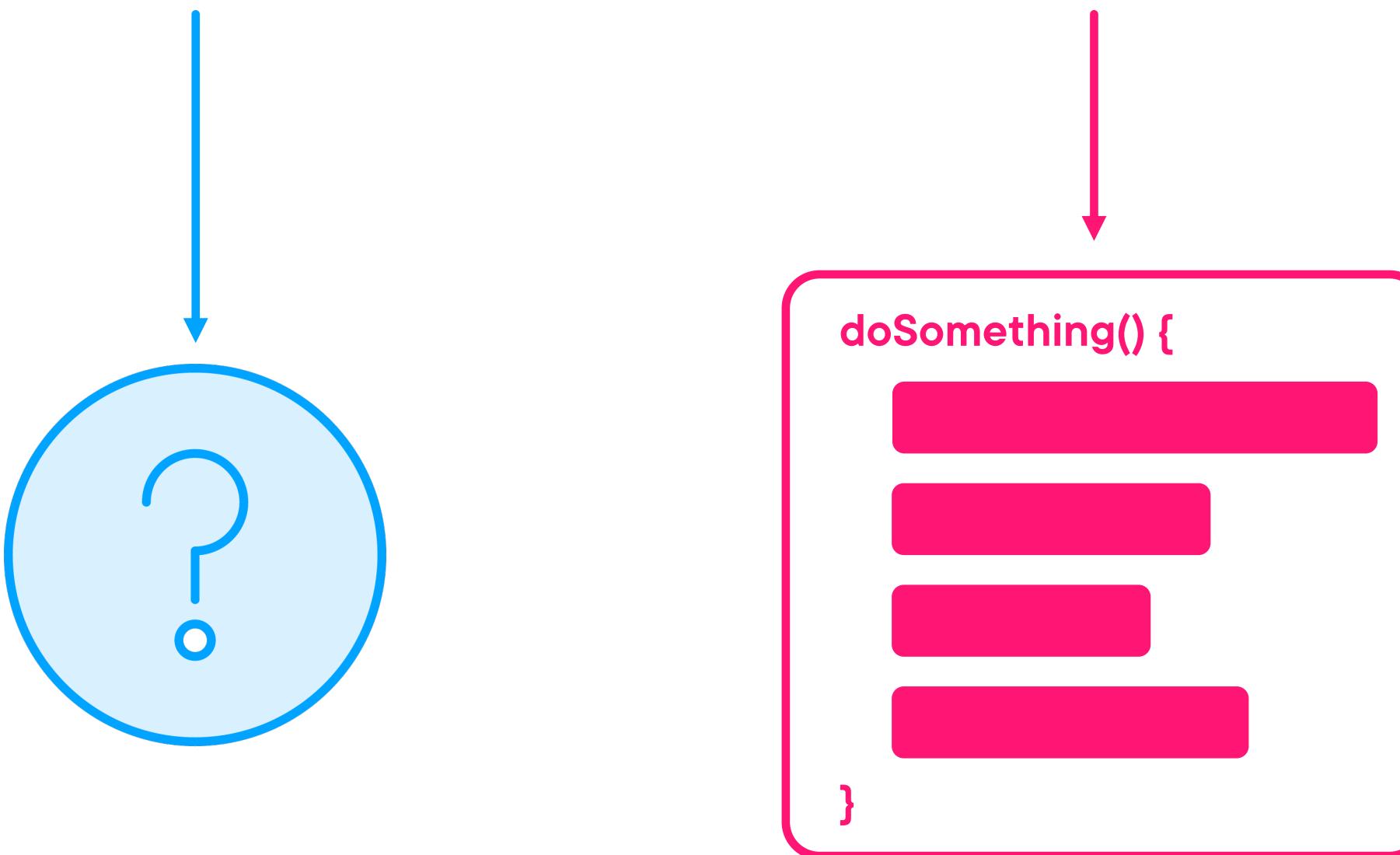
Stock trading
application



Prerequisites



Reactive Programming



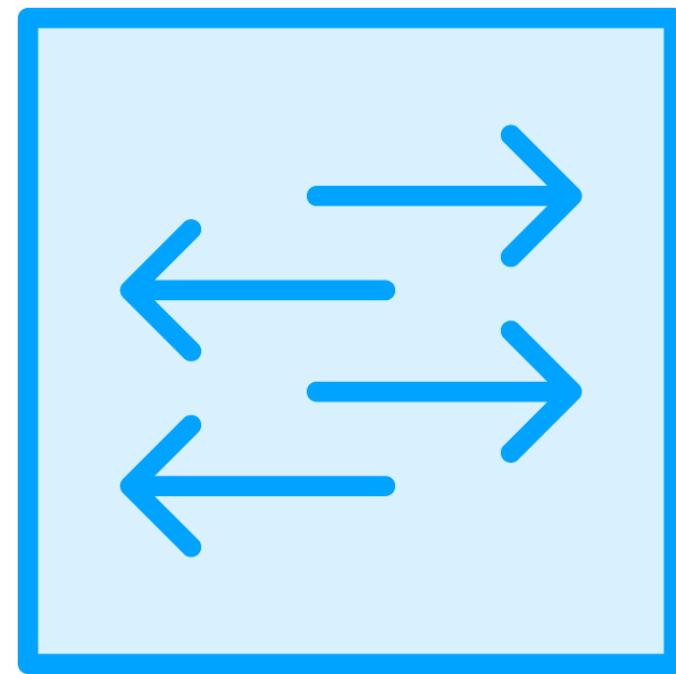
Imperative Programming



```
getData() {  
    .....> filterData() {  
        .....> processData() {  
            .....  
        }  
    }  
}
```



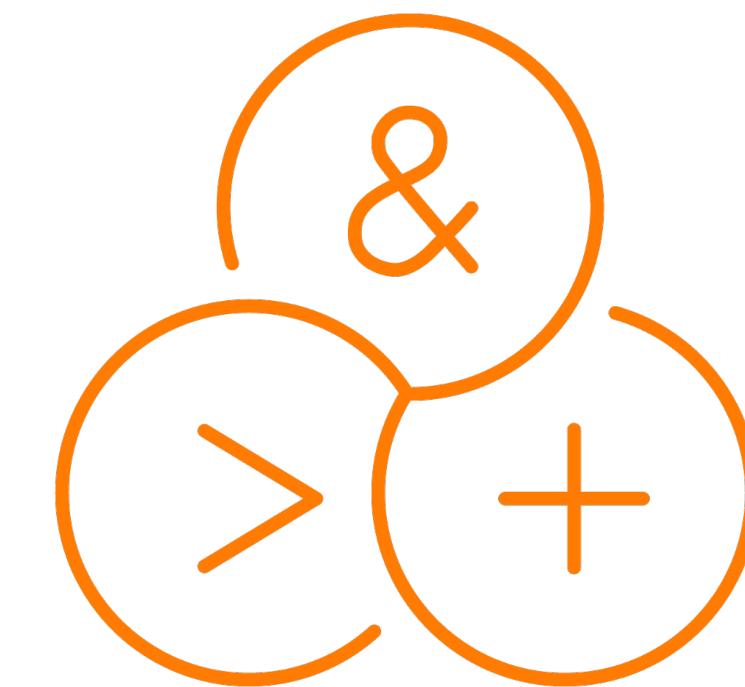
Reactive Programming



**Non-blocking
asynchronous
components**



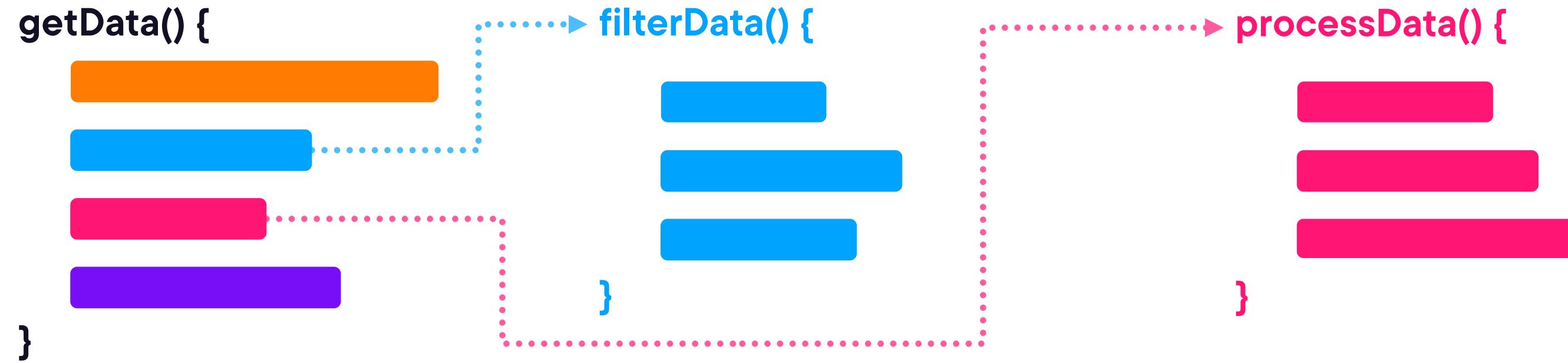
Data streams



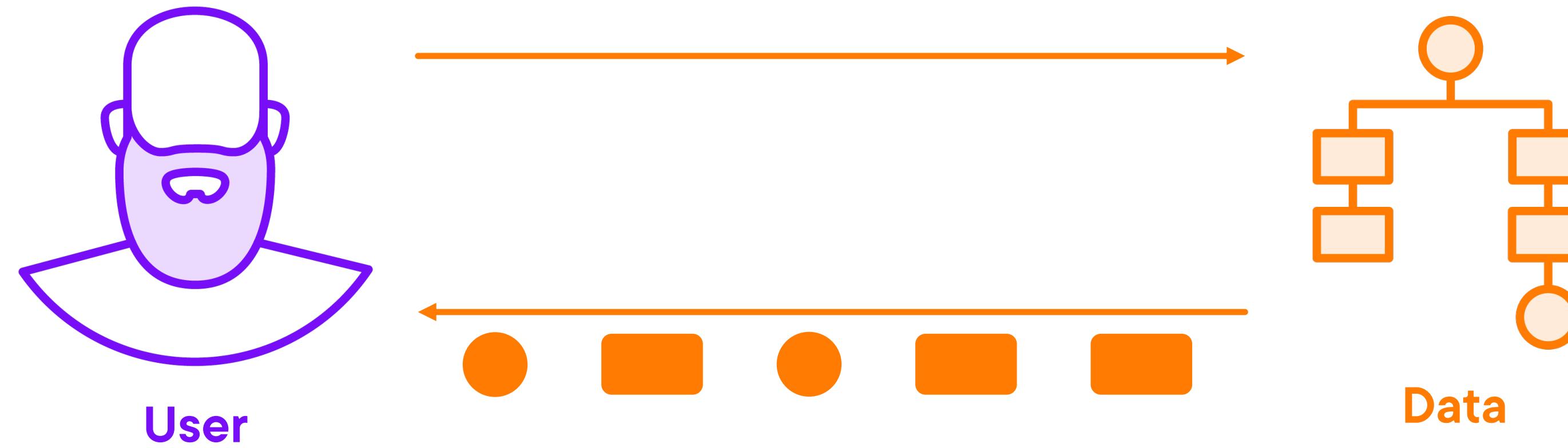
**Functional
Programming**



Reactive Programming



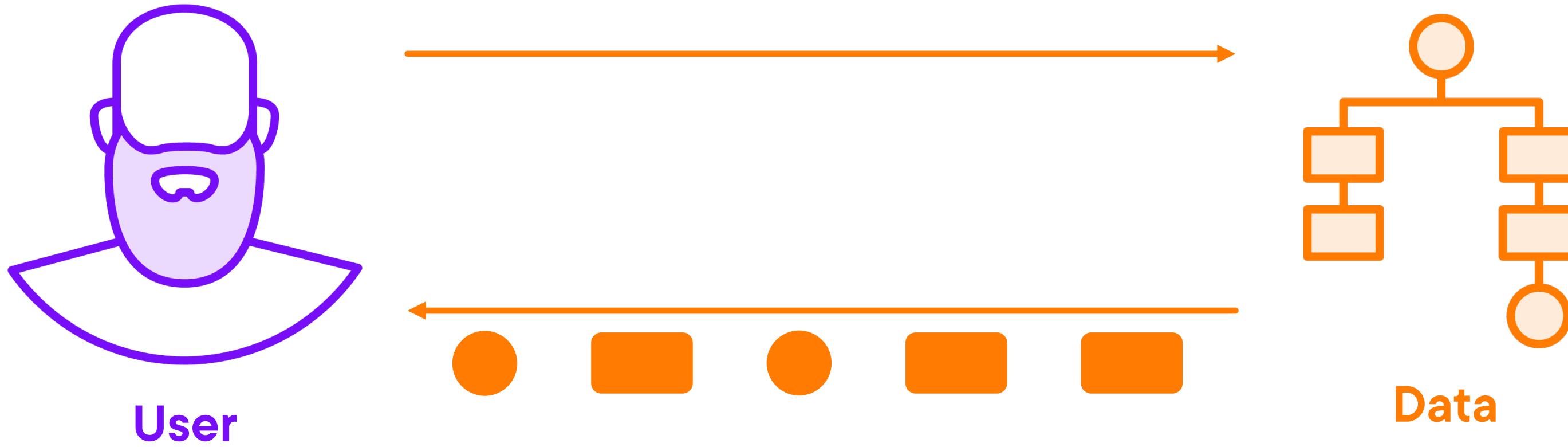
Reactive Programming



```
getData() {  
    .....> filterData() {  
        .....> processData() {  
    }  
}
```



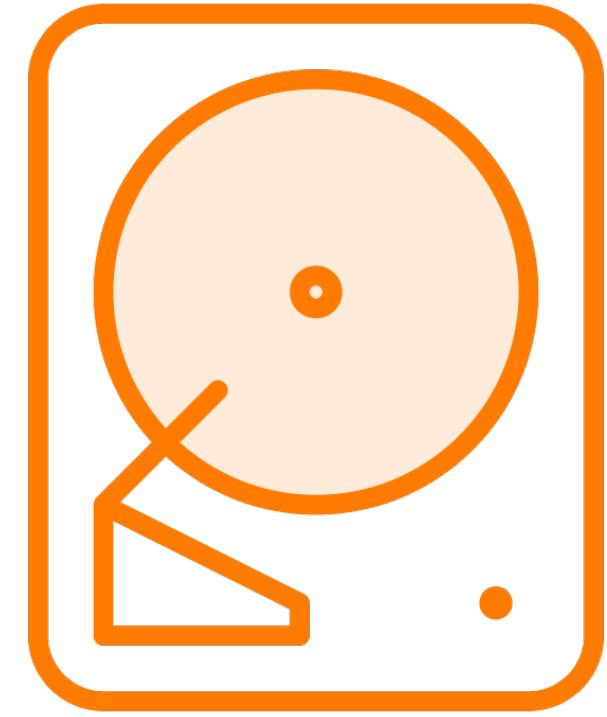
Reactive Programming



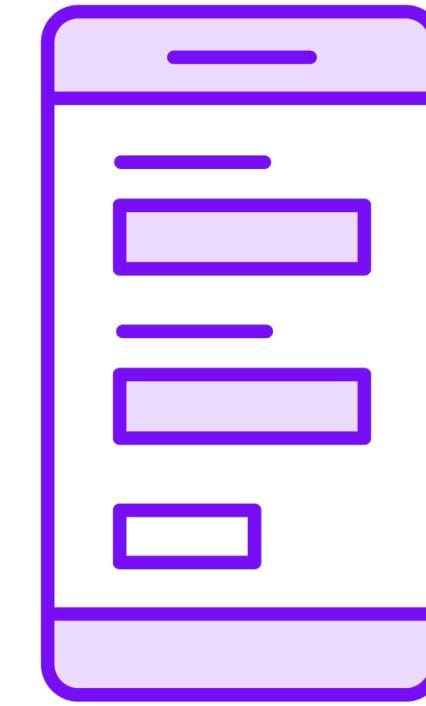
```
getData() {  
    //  
    .filter( )  
    .process( )  
    .return( )  
}
```



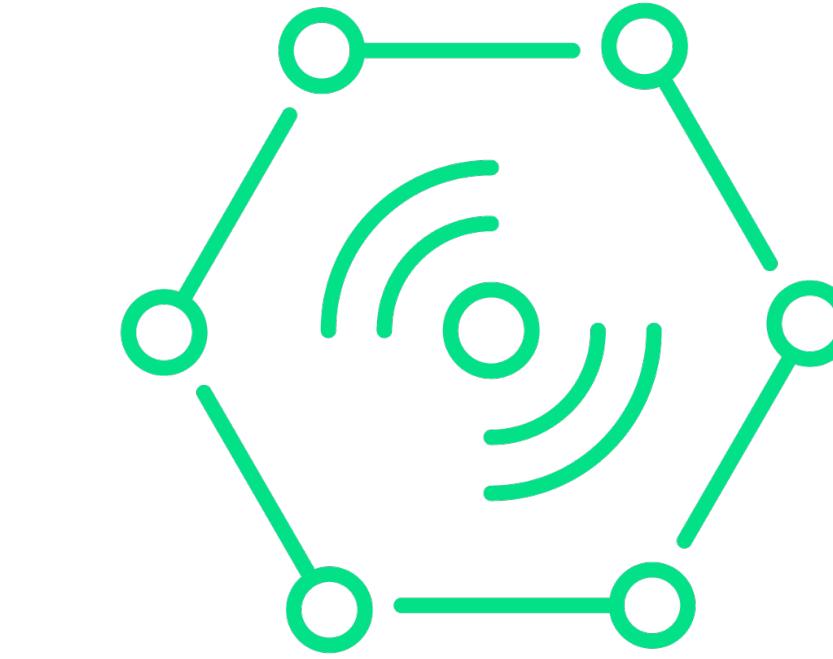
Reactive Programming Use Cases



I/O Operations



UI Controllers



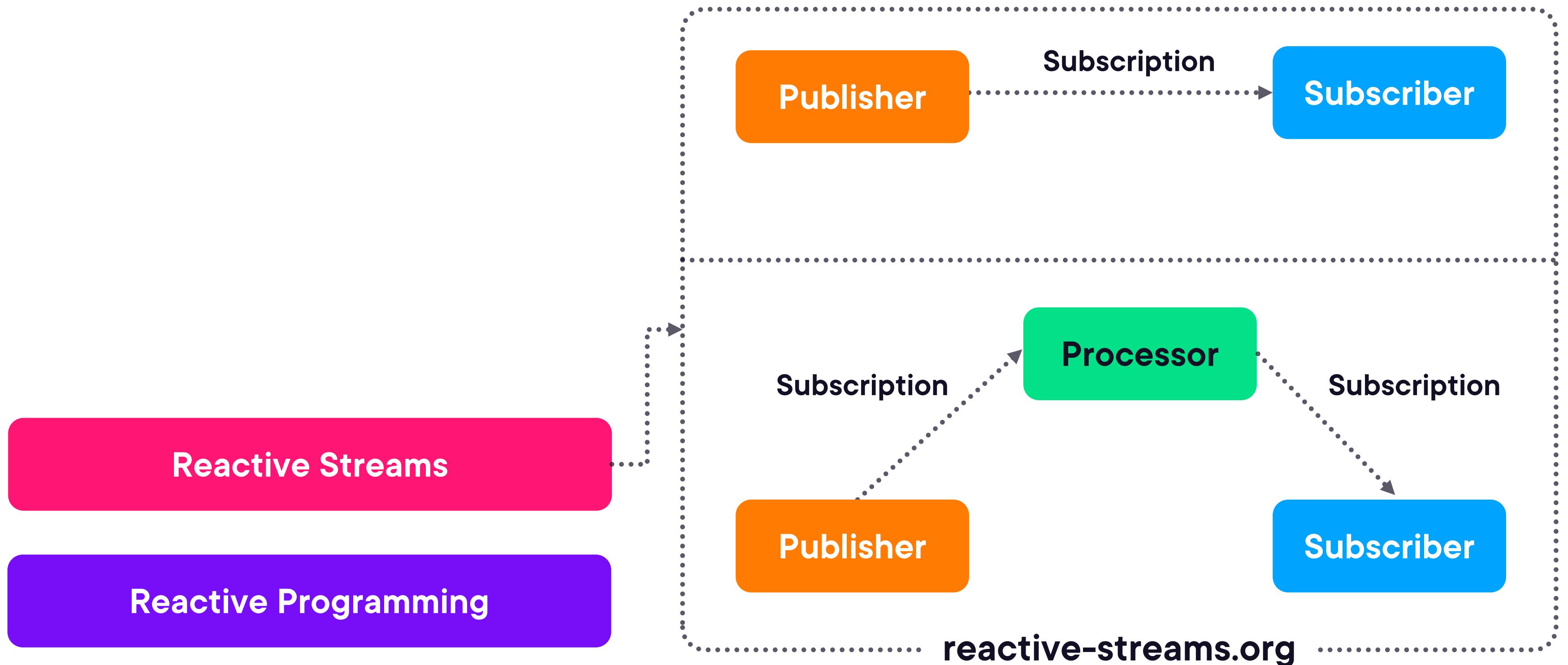
Internet of Things



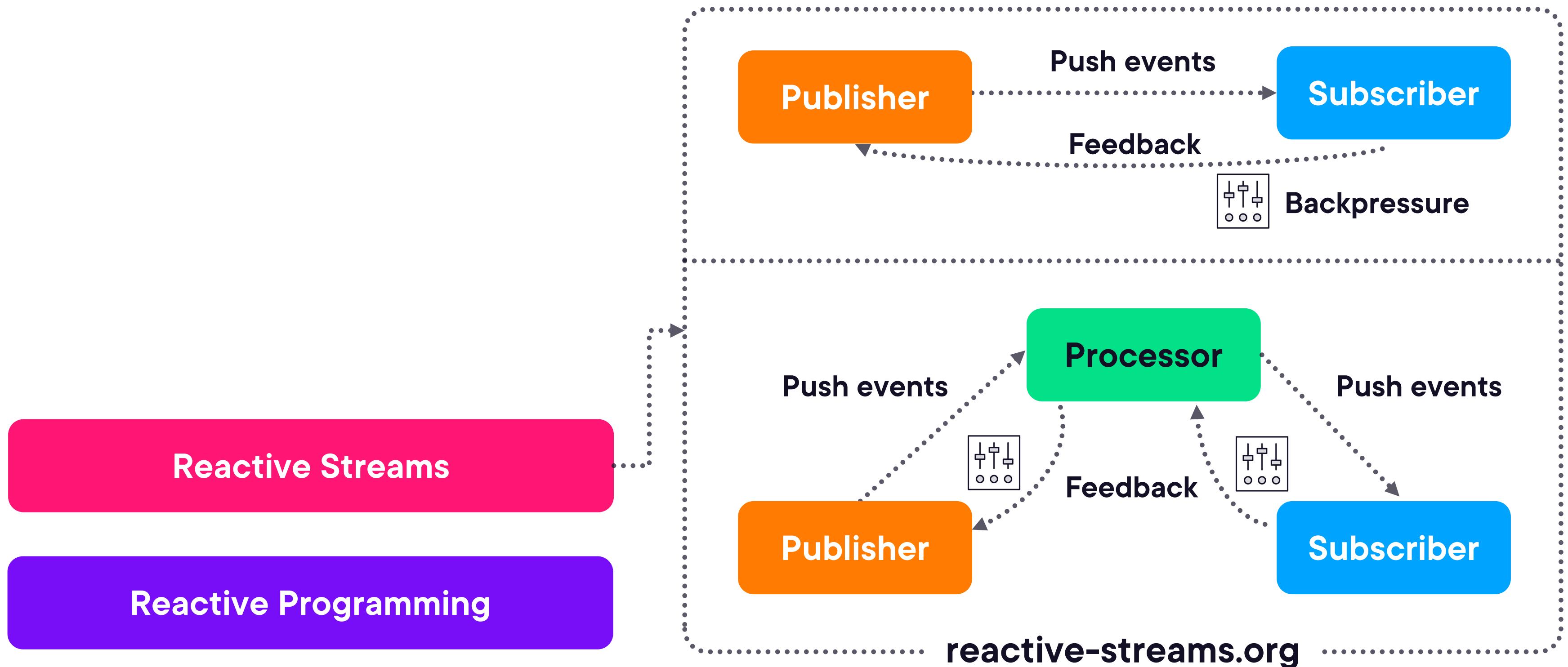
Reactive Streams



Reactive Streams



Reactive Streams



Reactive Streams

Project Reactor

Reactive Streams

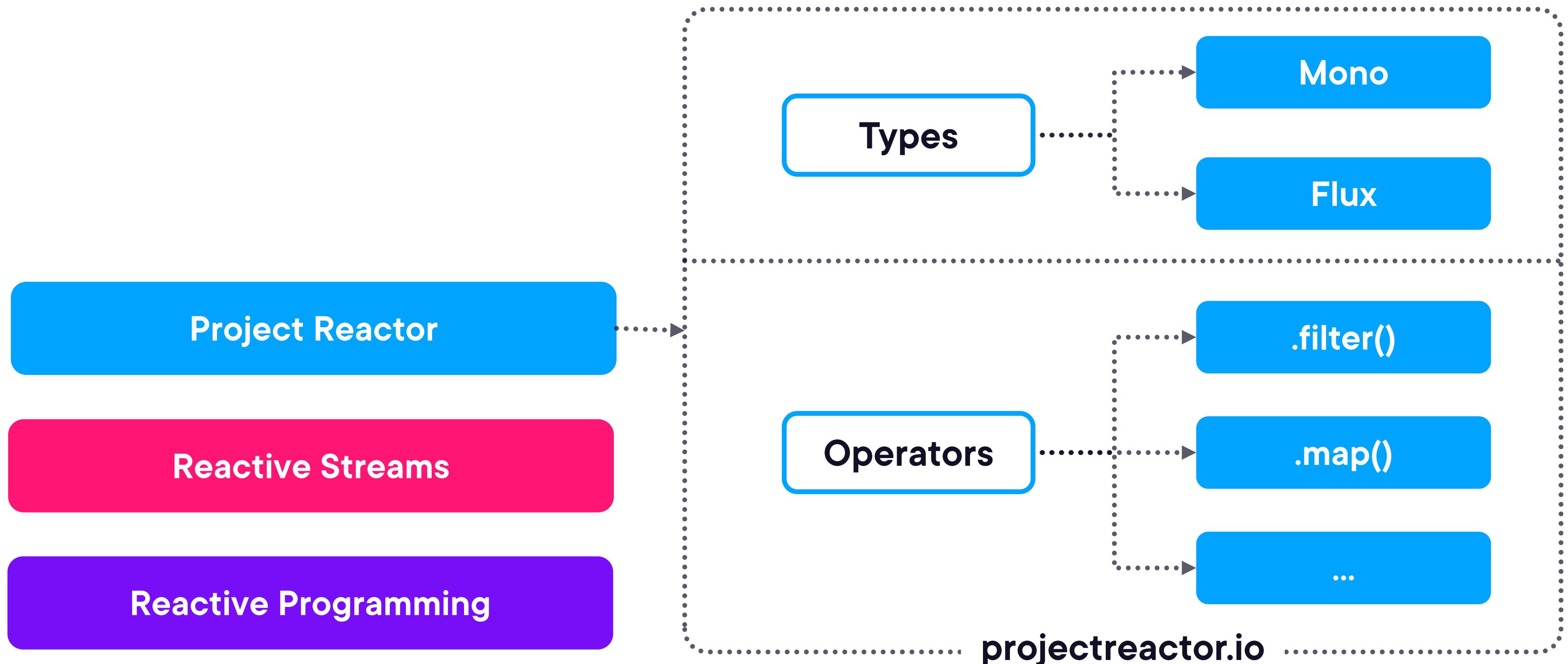
Reactive Programming



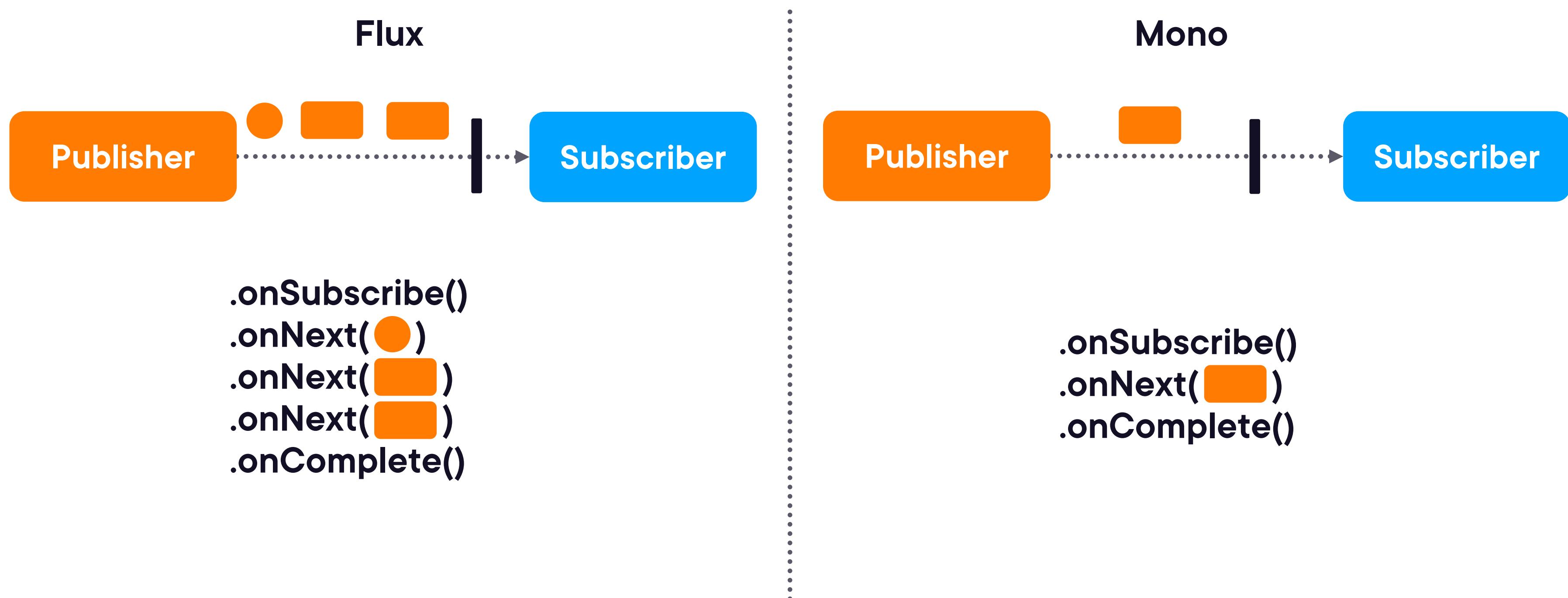
Project Reactor and Spring WebFlux



Project Reactor



Project Reactor Types



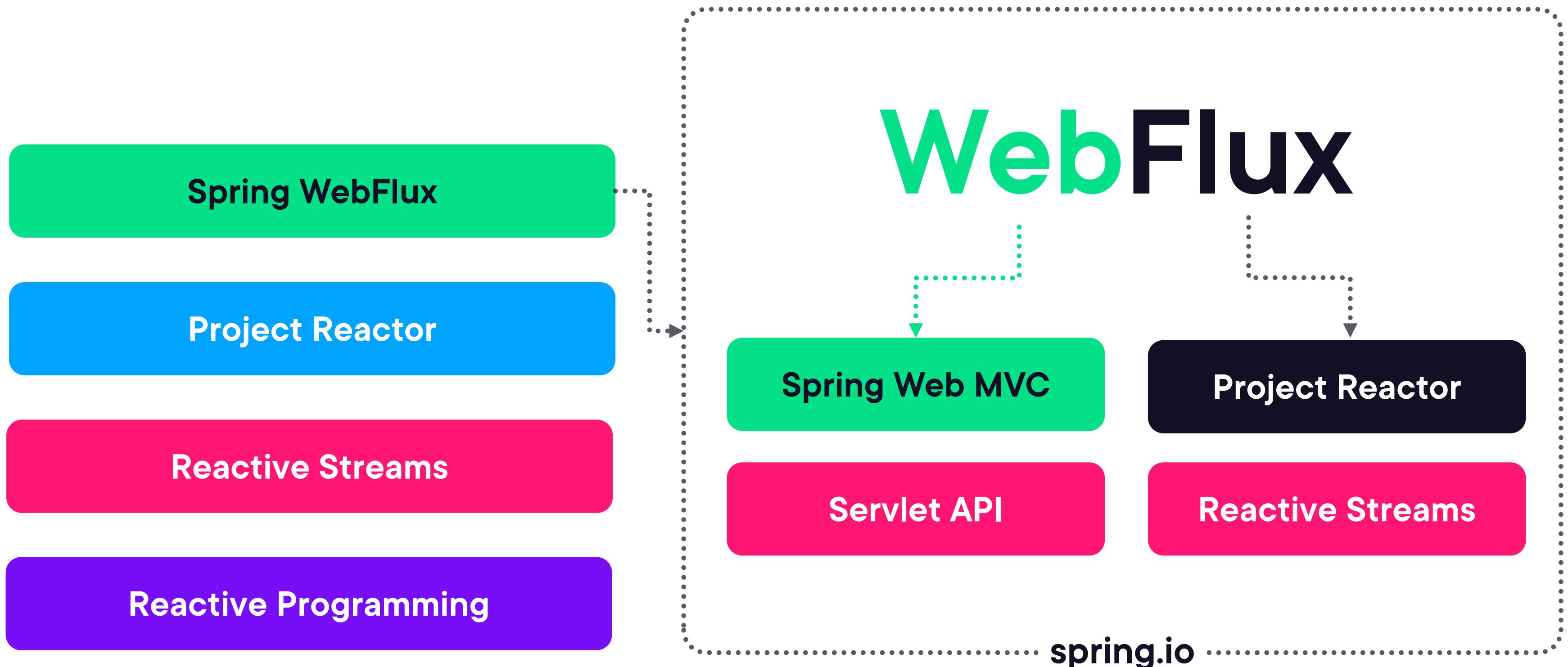
Project Reactor Types



- `.onSubscribe()`
- `.onNext(○)`
- `.onNext(■)`
- `.onError()`



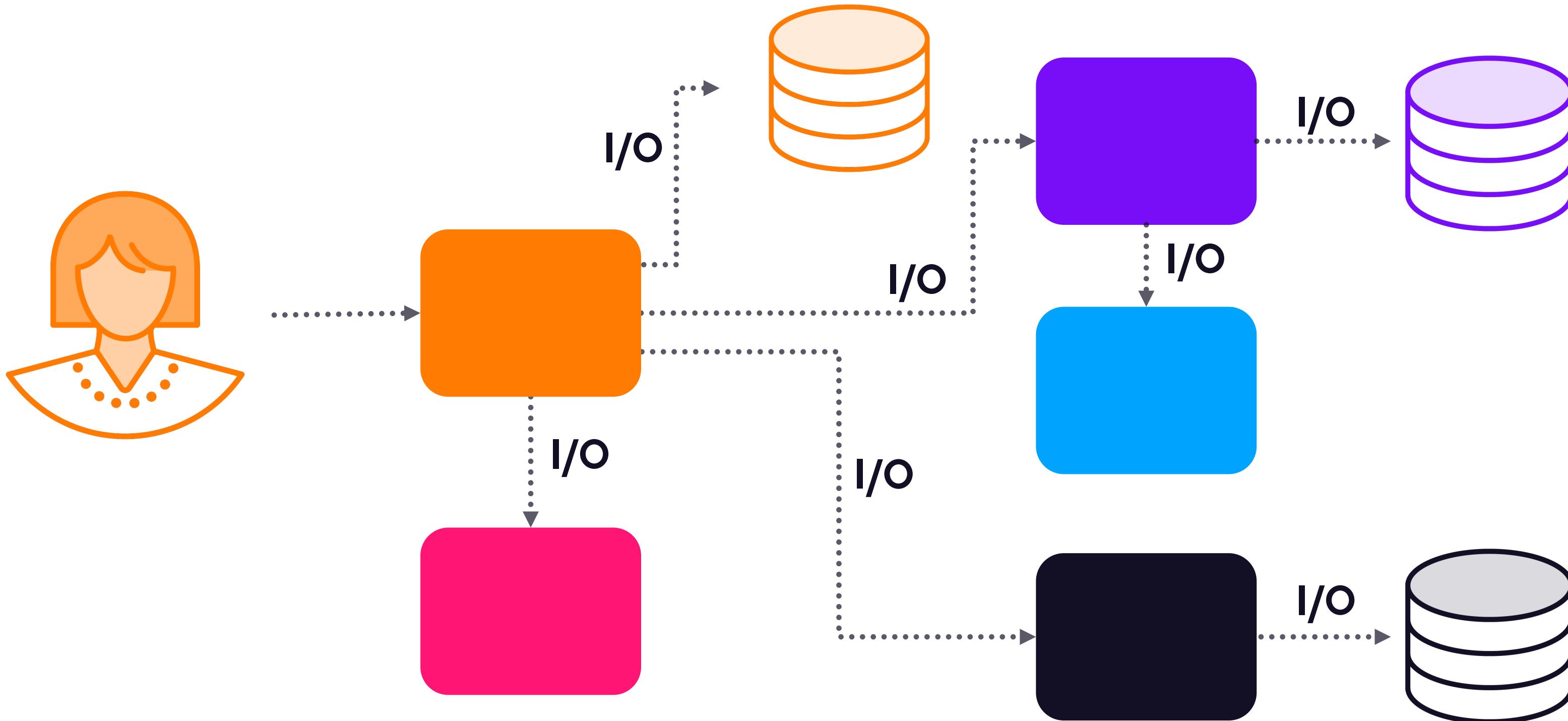
Spring WebFlux



Spring WebFlux



Spring WebFlux in Microservices Architecture



Project Loom

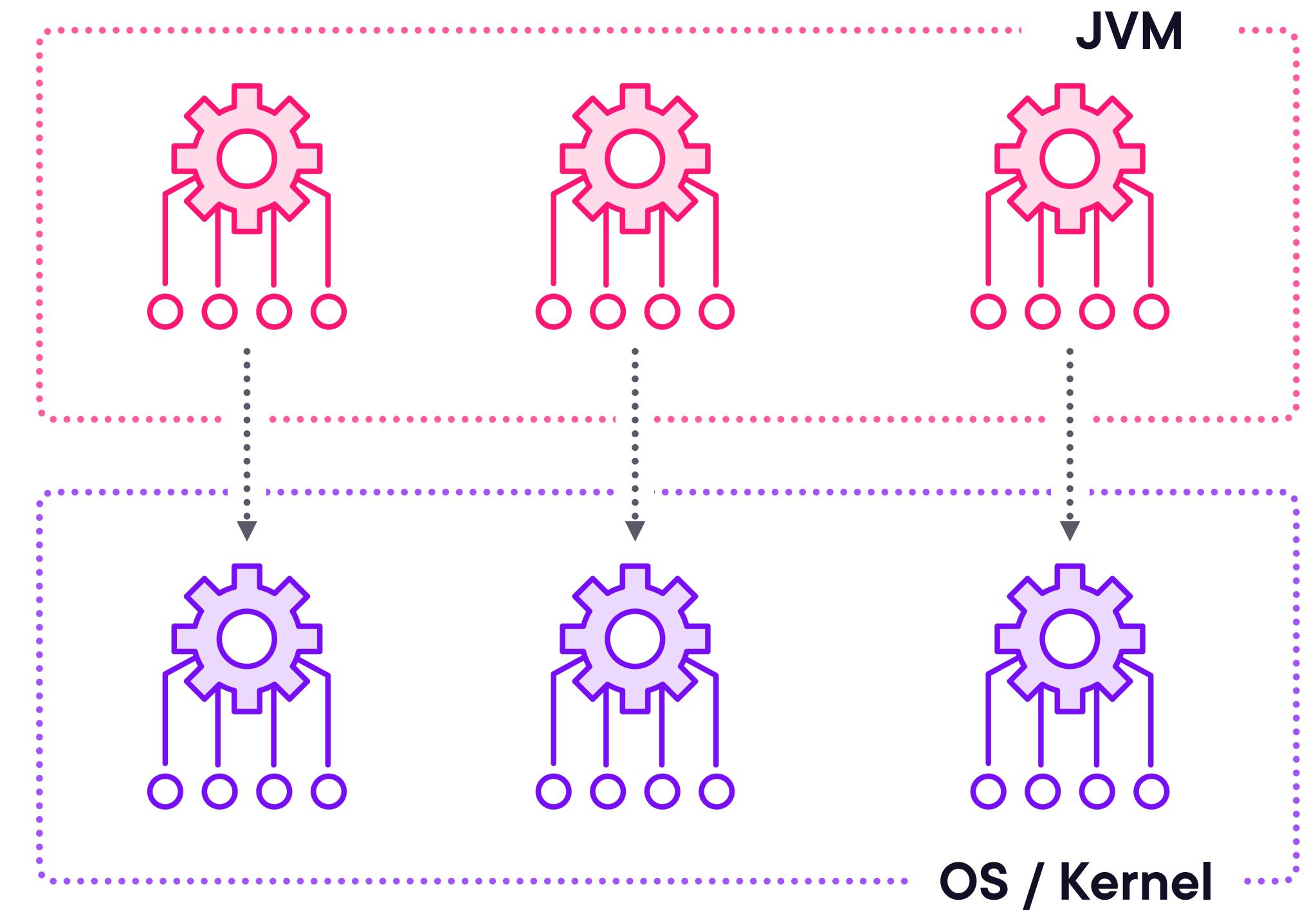


Java Threads

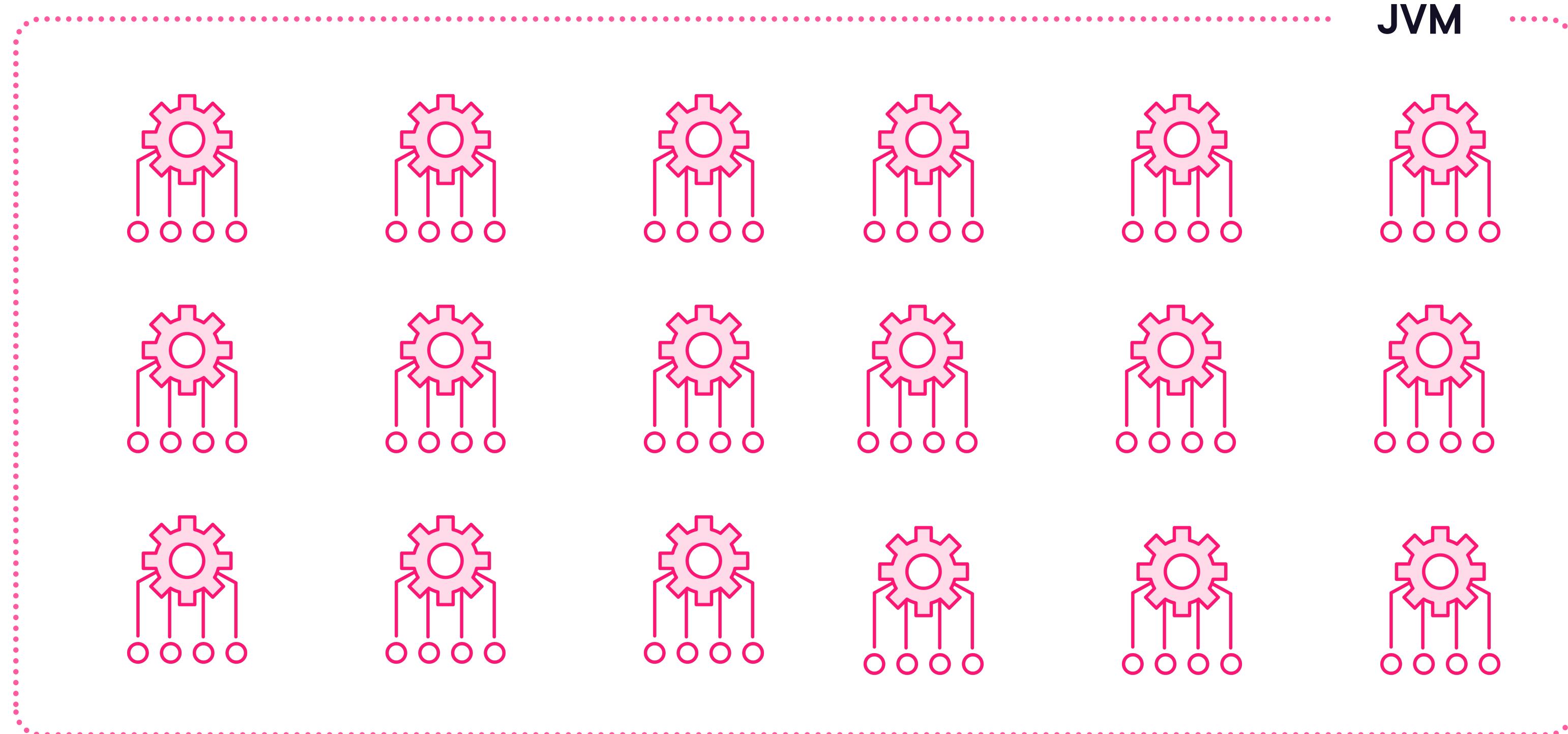
Context Switching

Blocking

Expensive thread creation /
deletion



Virtual Threads



Reactive Programming Use Cases

Async signals

**Server-client
bidirectional
communication**

Stream processing



Demo



Set up Spring WebFlux 6 project

Build reactive endpoints

Prerequisites:

- Java 17
- Gradle 8
- Docker



Summary



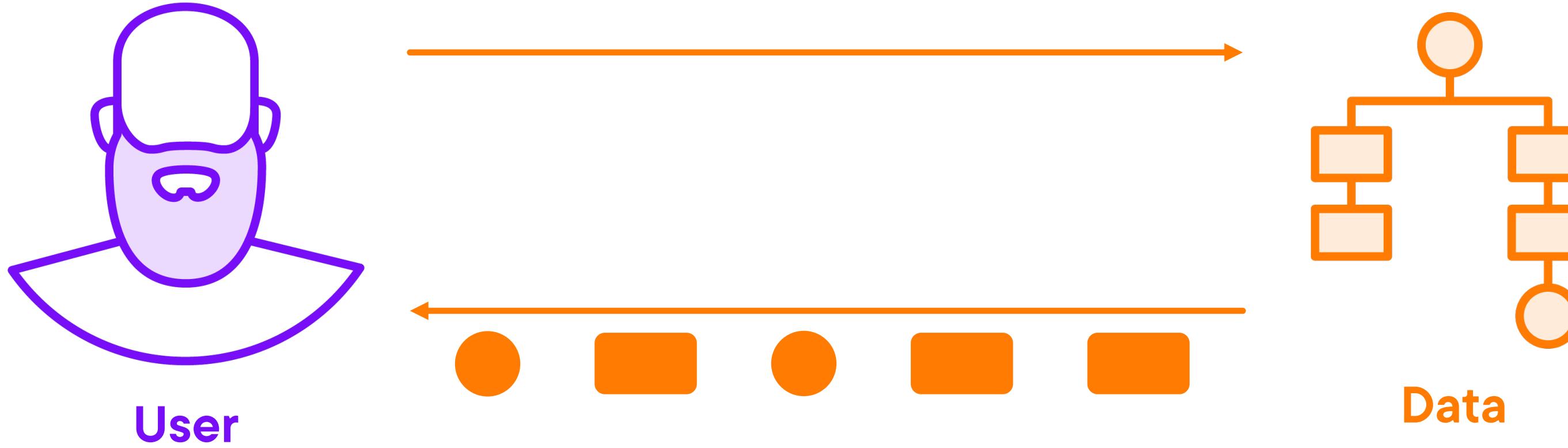
Reactive Programming

Project Reactor

Spring WebFlux



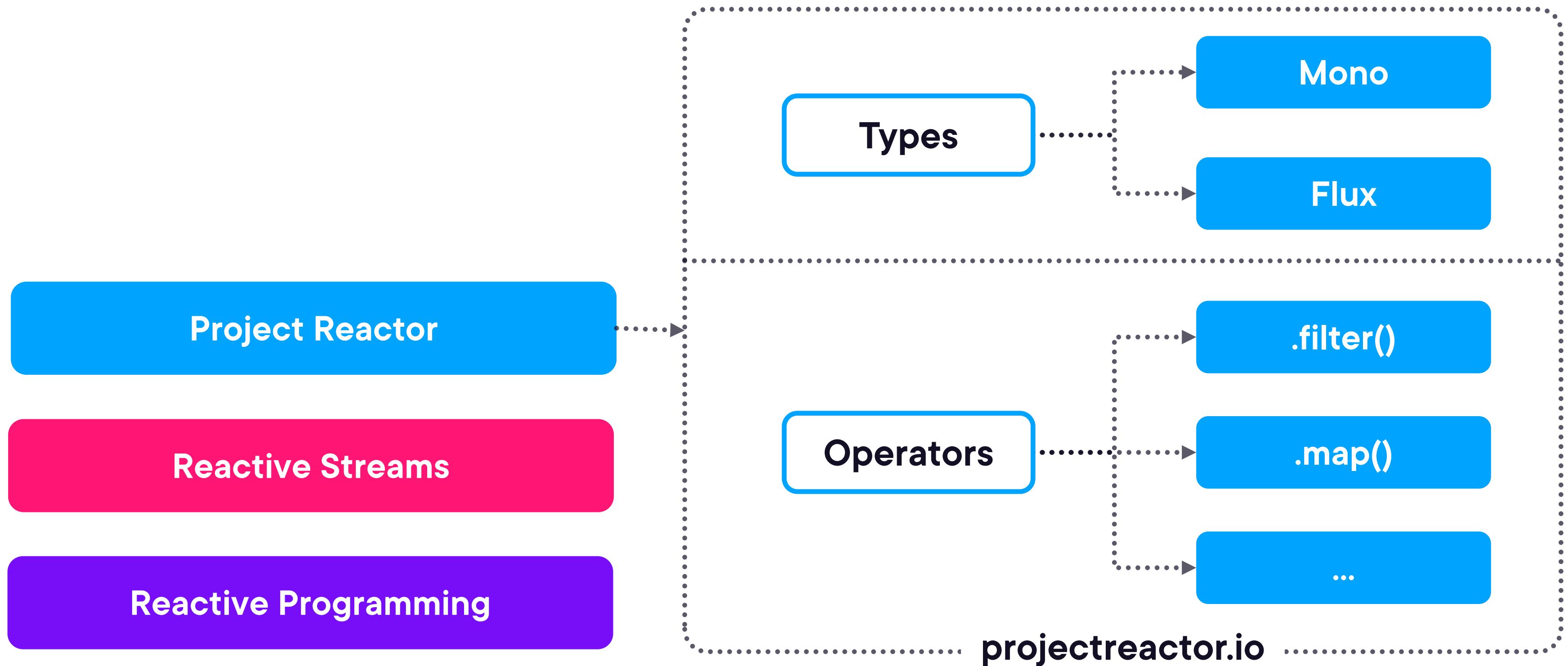
Reactive Programming



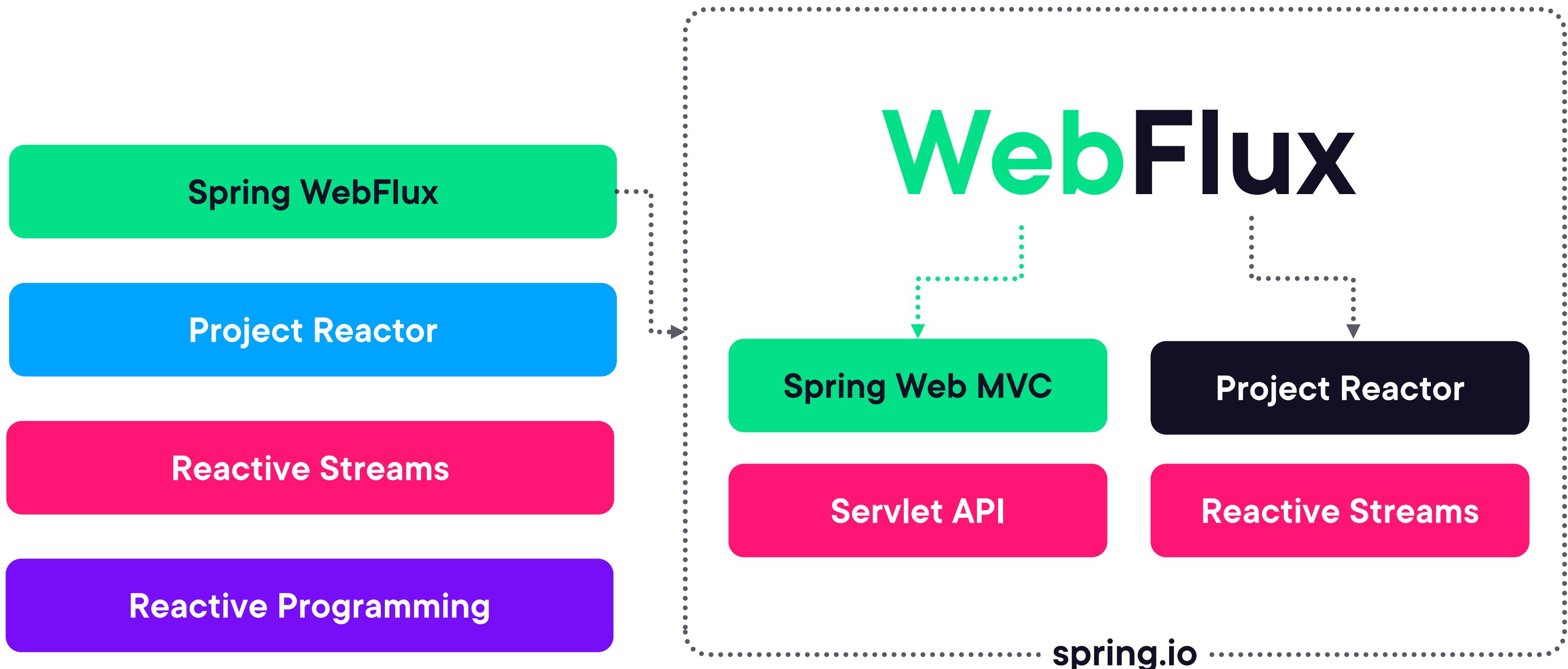
```
getData() {  
    //  
    .filter( )  
    .process( )  
    .return( )  
}
```



Project Reactor



Spring WebFlux



Up Next:

Building Reactive Applications

