

SpringOne Platform by Pivotal.

Servlet or Reactive stack: You have choices...

By Rossen Stoyanchev
@rstoya05



About This Talk

Servlet and Reactive stacks
(Spring Framework 5)

Big shift towards async...
Understand and make choices

About the Speaker

Spring Framework committer

Spring MVC *for a while...*

Spring WebFlux *from inception*

Motivation for change

REACTIVE

REACTIVE, EVERYWHERE ??



Asynchronicity

Thread pools



Asynchronicity

Actors

Fibers / Project Loom

Coroutines

Event loop

Thread pools

Servlet



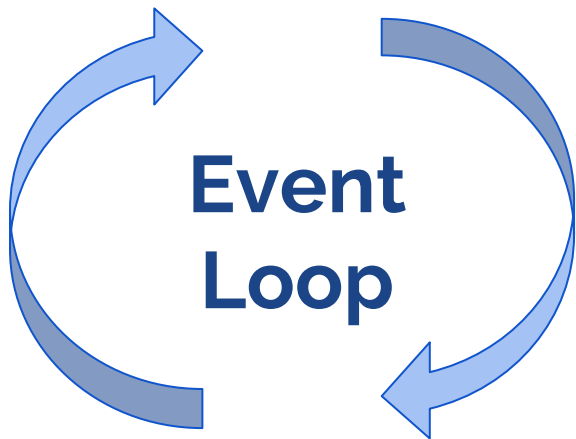
100s, 1000s threads
(blocked)

Reactive



Small, fixed # of threads
(running)

Non-blocking concurrency



Declarative composition of asynchronous logic

CompletableFuture, ReactiveX, Reactor, ...

Java 8 lambdas

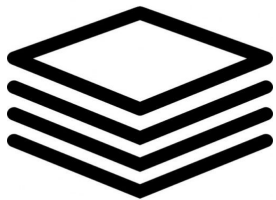
Functional programming models

An aerial, high-angle photograph of a dense urban skyline, likely New York City, featuring numerous skyscrapers and buildings. A semi-transparent teal rectangular box is centered over the middle of the image, containing the title text. The text is in a bold, sans-serif font for the first line and a script font for the second line.

Stack choices *and why choice matters*

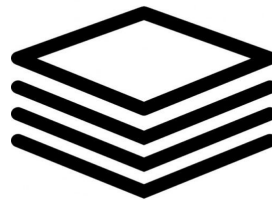
@Controller

Servlet



@Controller

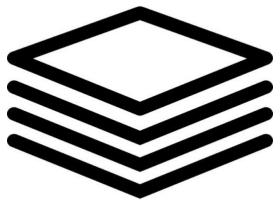
Reactive



Reactive client

@Controller

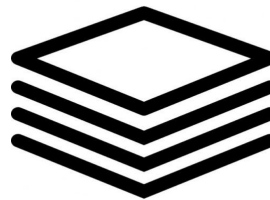
Servlet



Reactive client

@Controller

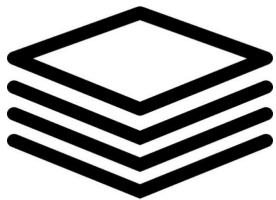
Reactive



Reactive client

@Controller

Servlet

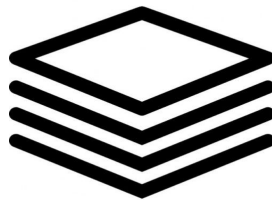


Reactive client

@Controller

Functional
endpoint

Reactive



Reactive client

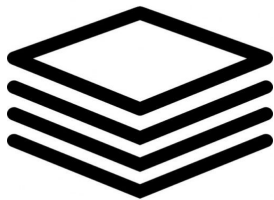
@Controller

Reactive client

@Controller

Functional
endpoint

Servlet

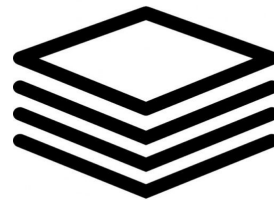


Tomcat

Jetty

Servlet container

Reactive



Tomcat

Jetty

Servlet 3.1 container

Reactive client

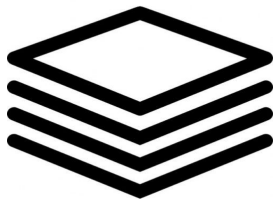
@Controller

Reactive client

@Controller

Functional
endpoint

Servlet

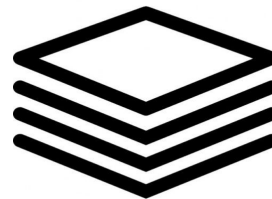


Tomcat

Jetty

Servlet container

Reactive



Netty

Tomcat

Jetty

Undertow

Servlet 3.1 container

Demo

An architectural rendering of a modern, multi-story building with a central courtyard. The building features a complex, stepped design with numerous balconies and terraces. The central courtyard is a large, open space with a paved ground, surrounded by the building's structure. People are depicted walking on the balconies, terraces, and in the courtyard, suggesting a vibrant, community-oriented environment. A prominent teal rectangular box with a black border is overlaid on the image, containing the text "Stack architecture" in a bold, black, sans-serif font.

Stack architecture

Spring MVC

Servlet API

Blocking I/O

Tomcat, Jetty, ...



History of Servlet API

1997	1.0	
...	...	
2009	3.0	Async Servlet
2013	3.1	Servlet non-blocking I/O
...	...	

```
graph TD; A([Controller +  
Reactive client]) --- B[Servlet API]; B --- C[Async Servlet]; C --- D[Tomcat, Jetty, ...]; D -- "Blocking I/O" --> B;
```

**Controller +
Reactive client**

Async Servlet

Servlet API

Blocking I/O

Tomcat, Jetty, ...

History of Servlet API

1997	1.0	
...	...	
2009	3.0	Async Servlet
2013	3.1	Servlet non-blocking I/O
...	...	

**Controller +
Reactive client**

Async Servlet

Servlet API



Blocking I/O

Tomcat, Jetty, ...

Controller +
Reactive client

Async Servlet

Servlet API

Blocking I/O

Tomcat, Jetty, ...

Spring Web API

Reactor, Reactive Streams

Reactor
Netty

Servlet 3.1
Non-blocking I/O

Netty

Tomcat, Jetty, ...

Spring MVC

Spring WebFlux

Servlet API

Spring Web API

Reactor, Reactive Streams

Blocking I/O

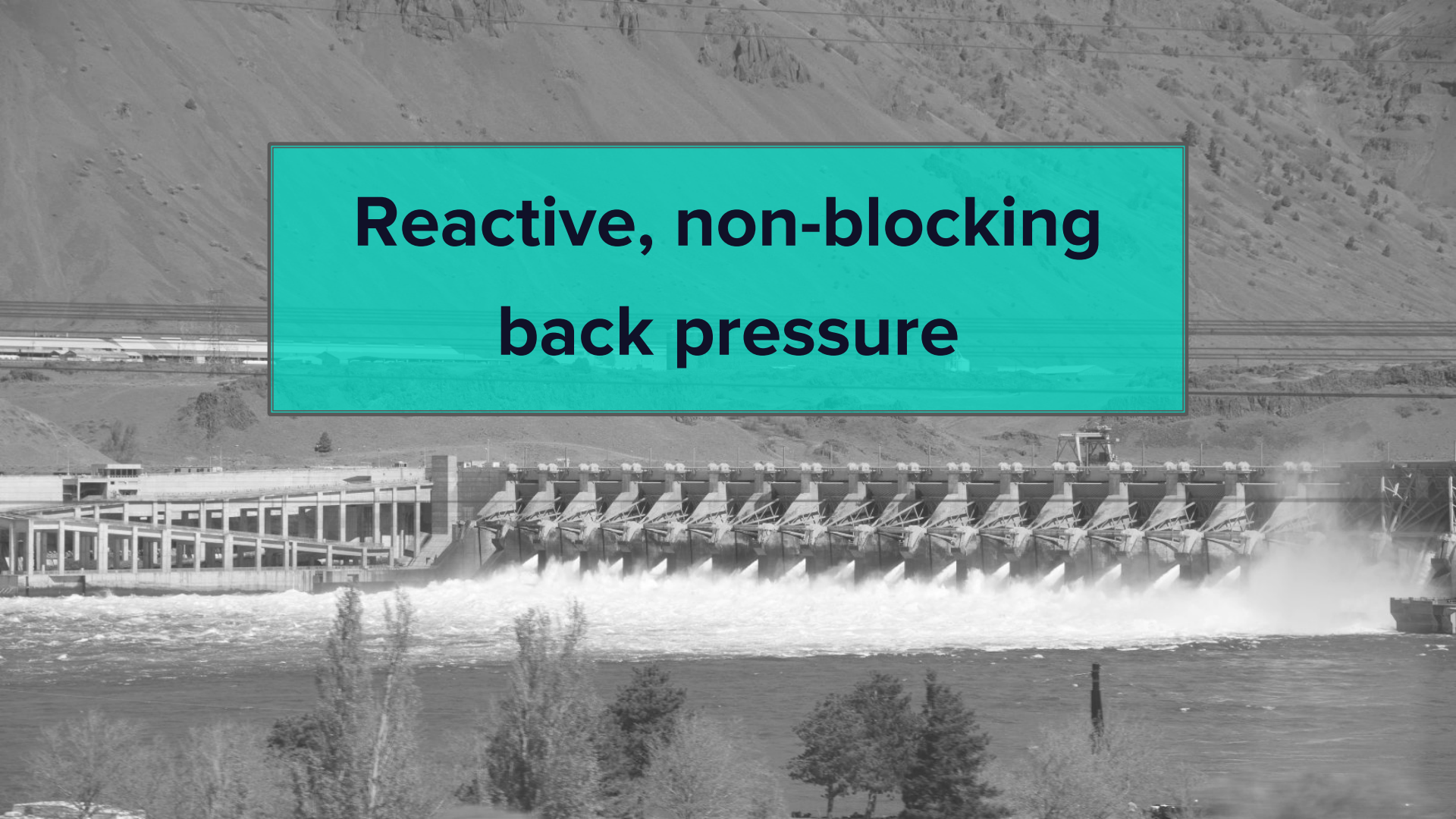
Non-blocking I/O

Tomcat, Jetty, ...

Netty

Tomcat, Jetty, ...

**Reactive, non-blocking
back pressure**

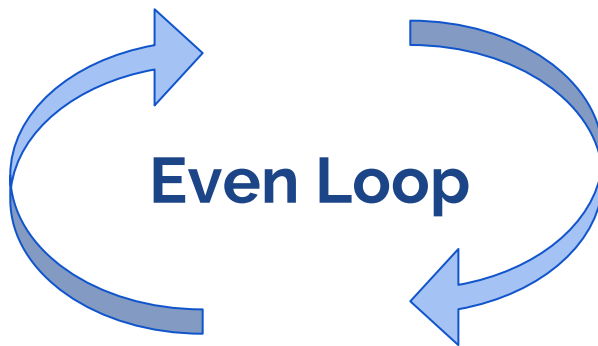


← **Blocking expected** →



Tomcat, Jetty, ...

← **Blocking not expected** →



WebFilter, WebHandler

Mono<Void>

Request

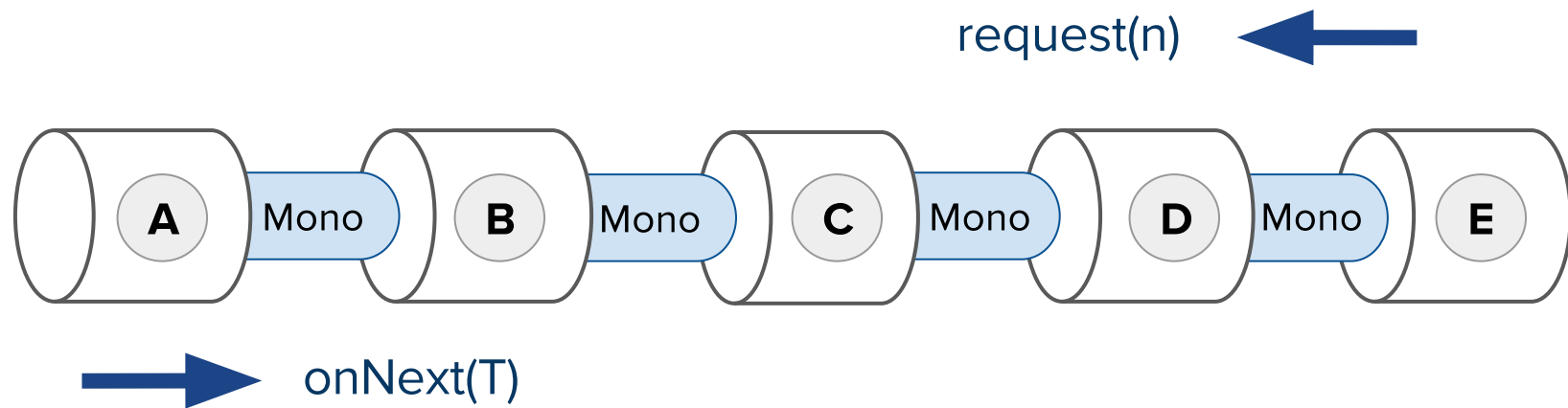
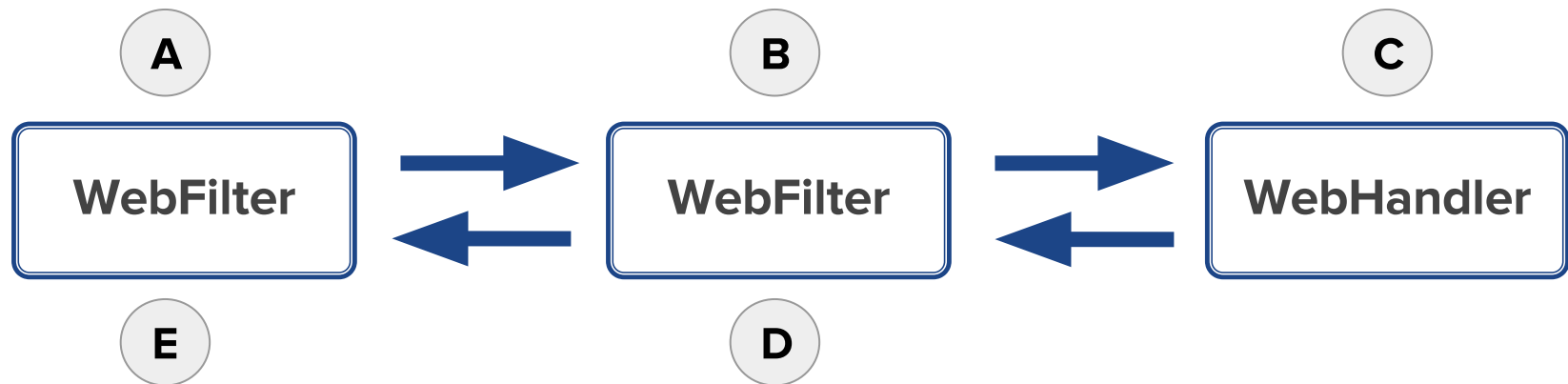
Flux<DataBuffer> `getBody()`

Response

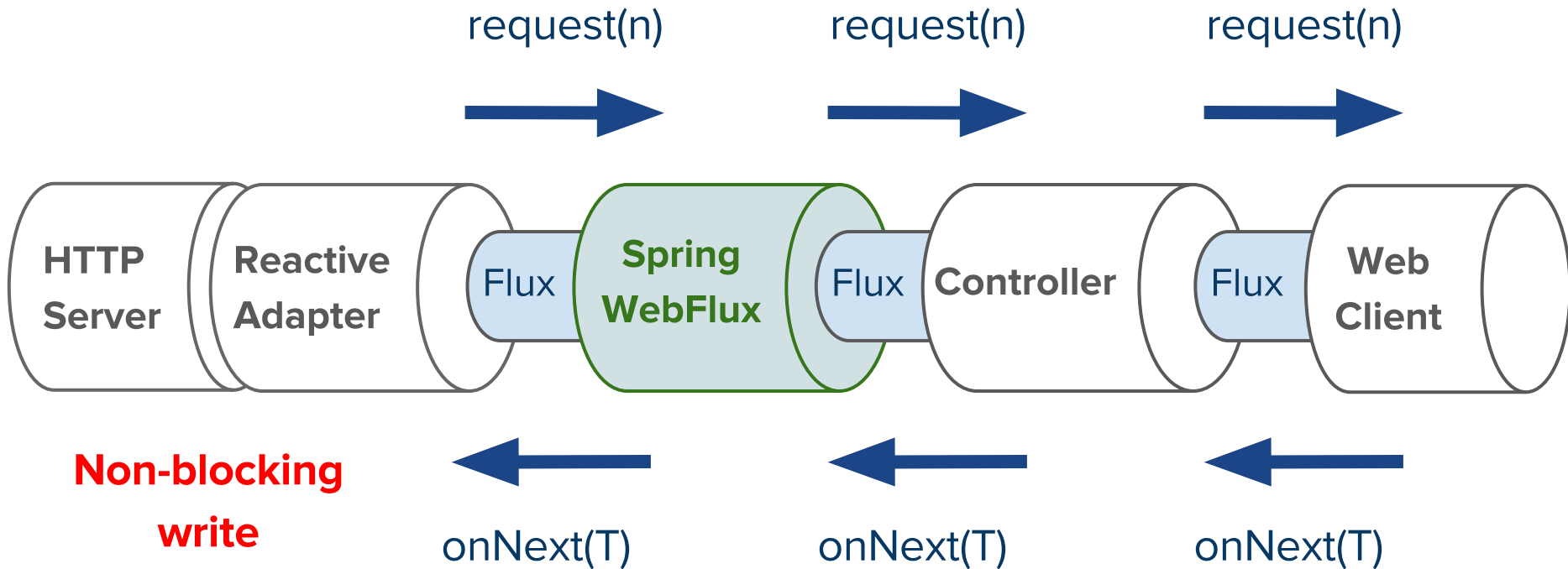
`writeWith (Flux<DataBuffer>)`

Codecs

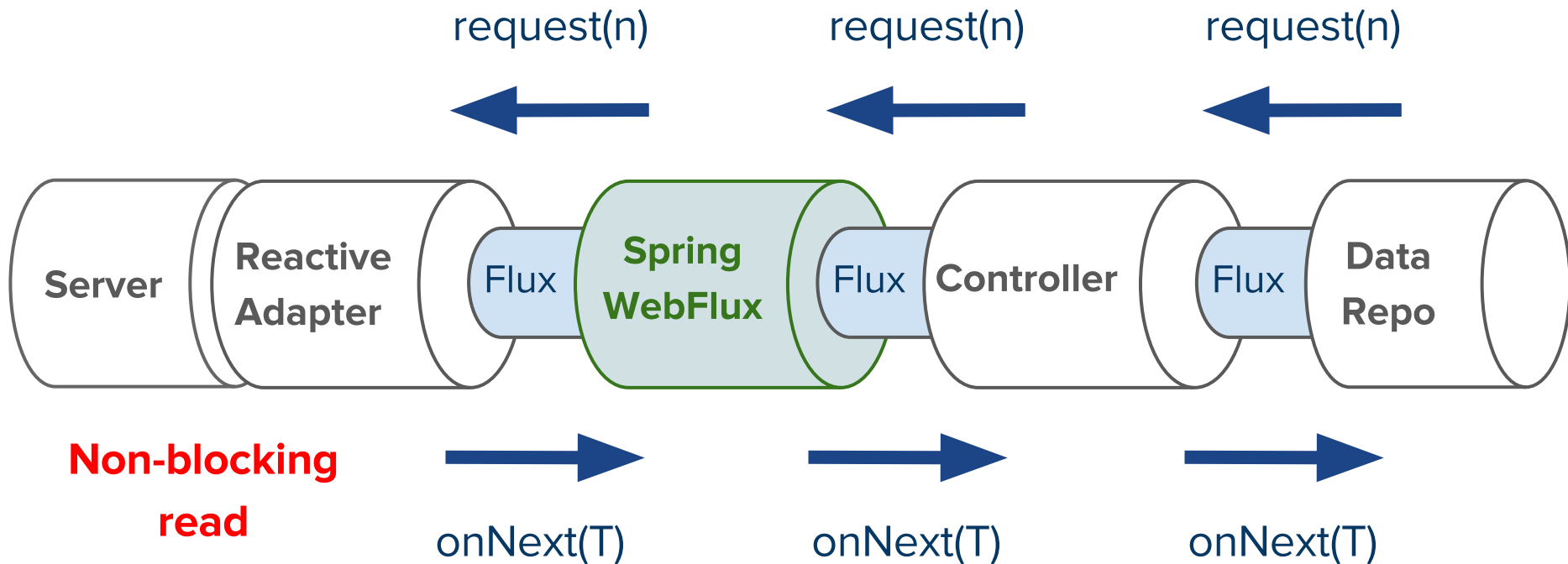
Flux<DataBuffer>  **Flux<T>**



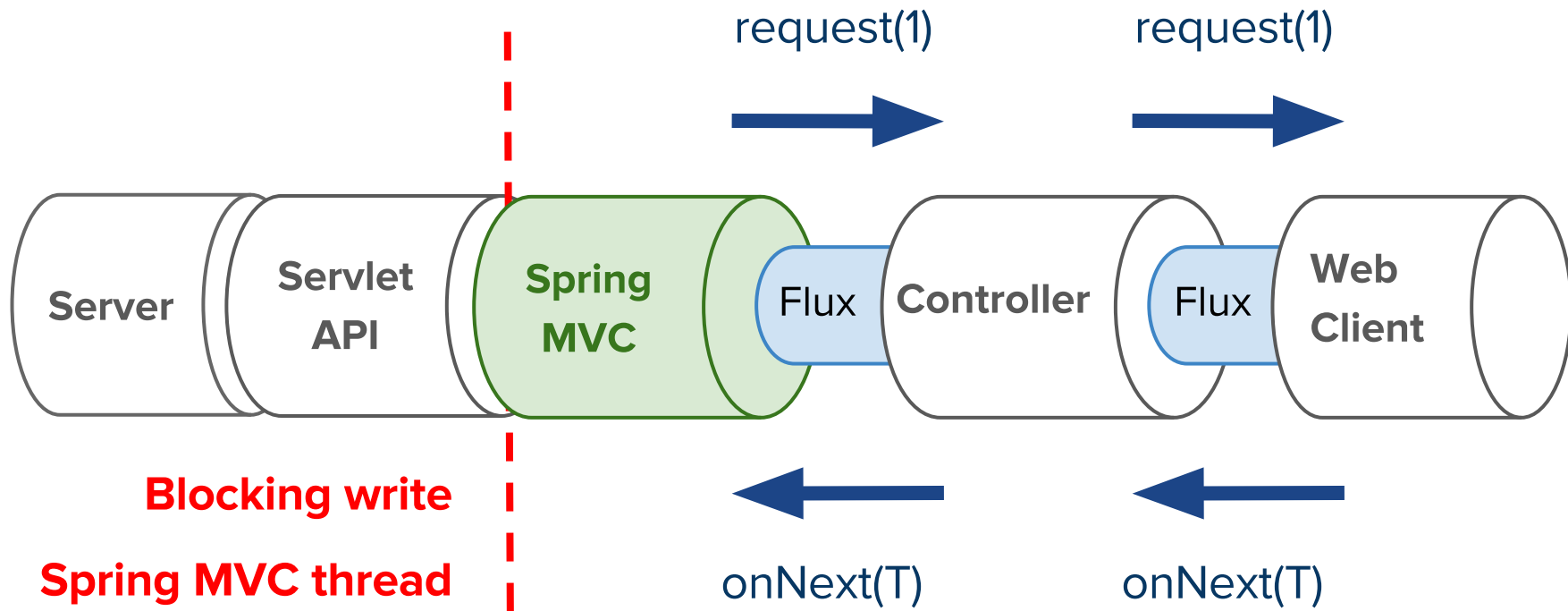
WebFlux Response



WebFlux Request



Spring MVC Response



Demo

Making Choices

Spring MVC | | WebFlux ?

&&

Spring MVC  WebFlux

Both available side by side
in Spring Framework 5

You've a wider range of options
in your tool chest

Spring MVC

Imperative logic,
simplest to write
and debug

JDBC, JPA,
blocking deps

@Controller

Reactive clients

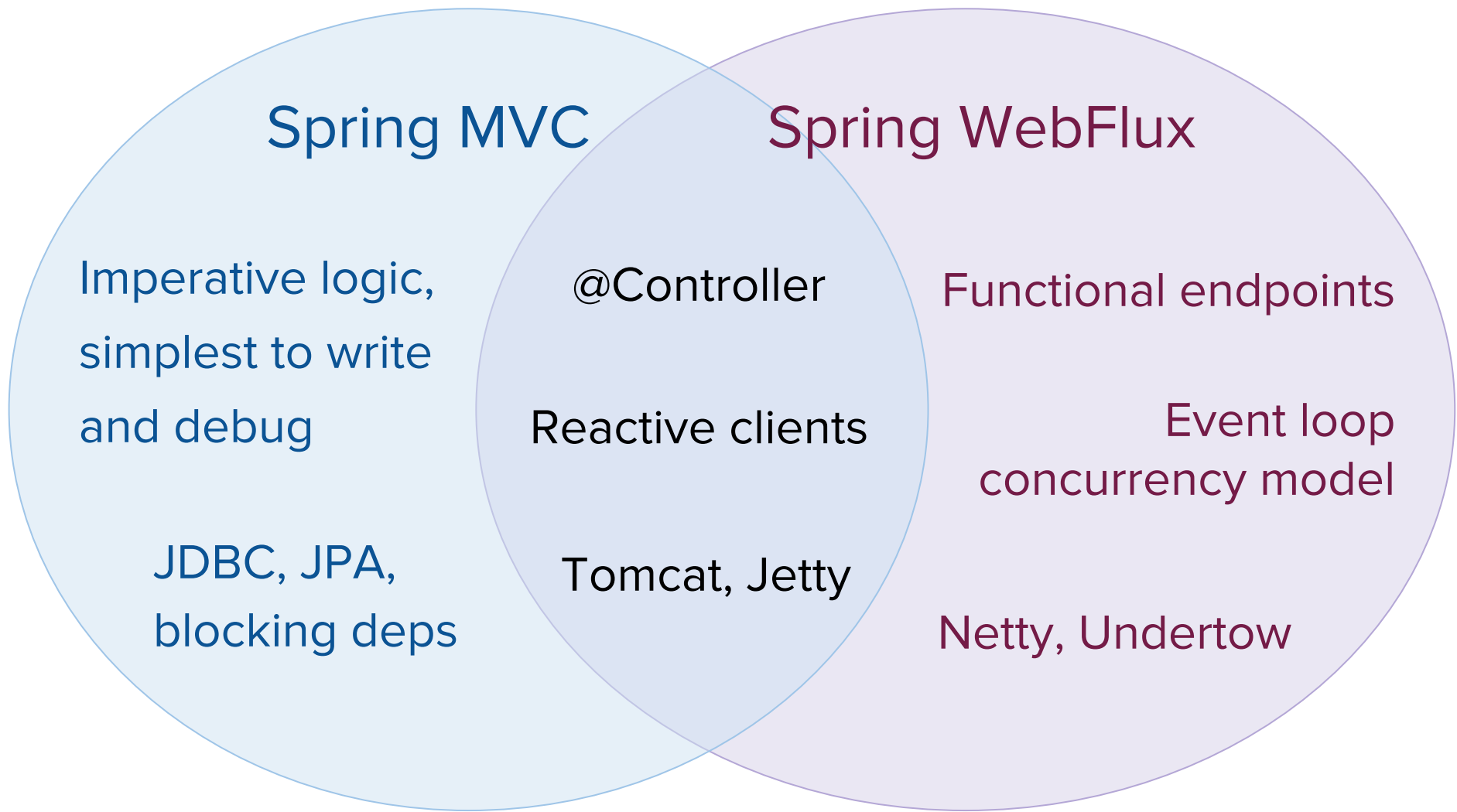
Tomcat, Jetty

Spring WebFlux

Functional endpoints

Event loop
concurrency model

Netty, Undertow



Spring
MVC

Reactive
clients

Spring
MVC

Spring
WebFlux

Tomcat,
Jetty

@Controller

Spring
WebFlux

Netty,
Undertow

Functional
endpoints

Spring
WebFlux

If it ain't broken, don't fix it

**Spring
MVC**



Reactive
clients

Spring
MVC

Spring
WebFlux

Tomcat,
Jetty

@Controller

Spring
WebFlux

Netty,
Undertow

Functional
endpoints

Spring
WebFlux

Imperative is simple until it's not

Spring
MVC

**Reactive
clients**

**Spring
MVC**



Spring
WebFlux

Tomcat,
Jetty

@Controller

Spring
WebFlux

Netty,
Undertow

Functional
endpoints

Spring
WebFlux

Reasons for WebFlux

Gateways, edge services, high traffic

Latency, streaming scenarios

High concurrency with less hardware resources

Reasons for WebFlux (continued)

Functional programming model

Lightweight and transparent (less magic, more control)

Immutability

Testing performance

Look Out For...

Learning curve

Team size, background

3rd party dependencies that block

Thank You

[https://github.com/
rstoyanchev/demo-reactive-spring](https://github.com/rstoyanchev/demo-reactive-spring)

SpringOne Platform by Pivotal.



#springone