Spring WebFlux

Alex Sartain

WebFlux comparison with Spring MVC

- WebFlux is fully synchronous and non-blocking whilst Spring MVC is not
- WebFlux is useful for scalability, but in most instances isn't necessary
  - A system which looks up data in a database and returns it is more suited for Spring MVC
  - A system which makes use of microservices and has many external calls is more suited for WebFlux
- WebFlux introduces 'Mono' and 'Flux' which are 'Publisher' types for asynchronous stream processing
  - A stream is some data e.g. score updates for football matches
  - A system which needs to respond to a stream is more suited to WebFlux

    (https://www.squeed.com/julkalender-2022/spring-mvc-or-spring-webflux/)

Example

Live Score Update Application

- Local Sunday League
  - Results are added to a database after the final score is confirmed
  - Results can be added hours after a game
  - Hundreds of people are interested in the results in the hours or days after
  - For a single game ending 2-0, there is one database call
- Premier League
  - Goals are added to a database as soon as they happen
  - In-depth statistics are added to a database every minute during the 90 minute game
  - Millions of people are interested in the game events as soon as they happen
  - For a single game ending 2-0, there are at least 92 database calls

Example