

Research Spring Boot 3

Spring Boot 3 Migration

In terms of most useful features which AFN could benefit.

Problem Details

For error messaging no longer a need for Zalando problem library, Spring has added ProblemDetail to springMVC and spring webflux. However it needs custom code to integrate the new problem classes.

Observability

Big changes happened around observability, sleuth is no longer available as of spring boot 3. Micrometer will be used. Minimum impact for imperative tomcat application, however **webflux will need custom code to make tracing propagation work**. See availability-service as an example.

Hibernate

As of spring boot 3 hibernate version was bumped to 6 and better support is offered for postgres database, out of the box array support and better enum support.

Validation

Jakarta validation is no supported and so it means a refactoring of all packages to include classes from jakarta and not javax.validation. Other than that should work.

HttpInterfaces

With the spring boot 3, spring web version 6 with new http interfaces api is included. This enables working with the latest webclient from netty seeming-less and have non-blocking clients even in imperative tomcat applications.

The performance boost is impressive using the new http interfaces versus other clients in tomcat application.

Normal recommended pool of connections for apache http client is 20, but for this test also 200 was used (much more resources needed if 200 blocking connections are used).

Normal recommended pool of connections for netty client is 200. Netty can have bigger pool of connections as it is non blocking client and netty connections are lighter on memory.

Please consider following example a :

- tomcat 10
- spring boot 3 web application
- java 17
- Netty version used is 4 (version 5 is still in Alpha release)
- local machine M1 apple macbook
- server is a mockserver running in docker.
- tool used jmeter. Ramp up period 30 seconds.
- The logic simulates a bases of availability service logic, container being tomcat + real availability service for comparison

Notes	Server	Client Technology	Max Connections	Delay configured for mock response from product api	Delay configured for mock response from inventory	Throughput with 50 parallel users	Throughput with 100 parallel users	Throughput with 150 parallel users	Throughput with 500 parallel users
delay 0 millis is not a real scenario, neither 200 concurrent blocking connections can be easily managed by one pod.	tomcat 10	Apache Http Client version 4	200	0	0	1520	1780	1780	
delay 0 millis is not a real scenario	tomcat 10	Apache Http Client version 4	20	0	0	1350	N/A	N/A	
looks more of a real scenario with prod configurations, since configs for max connection per route are recommended 20-50 for apache client	tomcat 10	Apache Http Client version 4	20	400	400	21	N/A	N/A	

looks more of a real scenario with prod configurations, since configs for max connection per route are recommended 20-50 for apache client	tomcat 10	Apache Http Client version 4	50	400	400	N/A	N/A	55	
looks more of a real scenario with prod configurations, since configs for max connection per route are recommended 20-50 for apache client	tomcat 10	Apache Http Client version 4	50	0	0	N/A	N/A		720
Number of connections for blocking is very high for a pod, unlikely to have this in production, could be tested	tomcat 10	Apache Http Client version 4	200	400	400	49	96	160	
	tomcat 10	Http Interfaces with Netty Client behind	200	0	0	1620	1730	1780	1700
for non blocking clients, real scenario with possible prod connections can be handled	tomcat 10	Http Interfaces with Netty Client behind	200	400	400	45	108	155	
Loom is in preview, not available for a long time in the future	tomcat 10	Apache Http Client version 4 + loom virtual threads activated	20	400	400	15	Not completed due to errors	Not completed due to errors	
Loom is in preview, not available for a long time in the future	tomcat 10	Http Interfaces with Netty Client behind + LOOM activated	200	400	400	45	Test not done since no difference is observed from normal scenario	Test not done since no difference is observed from normal scenario	
availability service real server has additional mappings and checks	netty webflux	Webclient netty	200	0	0	890	1010	1170	1000
availability service real server has additional mappings and checks, the delay simulate real time scenario NO CACHE	netty webflux	Webclient netty	200	400	400	48	94	130	
Loom enabled on availability service	netty webflux	Webclient netty	200	0	0	890	Test not done since no difference is observed from normal scenario	Test not done since no difference is observed from normal scenario	
	tomcat 10	okhttp client	50	0	0	1250	1450	1630	1640