What's New in Spring

2023/2024 edition

Spencer Gibb
@spencerbgibb
social.sdf.org/@spencergibb

Tweet by James Smith from NounProject.com
Mastodon by Icon 54 from iconscout.com

Copyright © 2005-2024 Broadcom, Inc. or its affiliates.

Agenda

- History
- JDK 21
- Performance
- New Clients
- Testing
- New Projects
- Roadmap
- Q+A

Man coding and using three computers by Scopio from Noun Project (CC BY-NC-ND 2.0)

History

History by monkik from NounProject.com

Spring Framework and Friends History

Spring Framework Releases

- 1.0 - 24 March 2004
- 5.0 - 28 September 2017
- 6.0 - 16 November 2022
- 6.1 - 16 November 2023

Spring Cloud Releases

- Angel - 03 March 2015
- Finchley - 18 June 2018
- 2022.0 Dec ‘22 / 2023.0 Dec ‘23

Spring Boot Releases

- 1.0 - 1 April 2014
- 2.0 - 1 March 2018
- 3.0 - 24 November 2022
- 3.1 May 2023 / 3.2 November 2023

What versions are compatible

https://start.spring.io

spring

Previous Release Highlights

Java 17 Baseline

- New Language, Library & JVM Features
- https://advancedweb.hu/new-language-features-since-java-8-to-17/

Jakarta EE 9

- javax.* ⇨ jakarta.*
- Container updates (Tomcat, Jetty, etc...)

Declarative Clients

- OpenFeign ⇨ Spring Framework
- HTTP and RSocket

GraalVM Native / AOT

- Static binary, no JVM needed
- Startup and memory advantages
- Closed world assumptions and other tradeoffs

Observability

- Spring Cloud Sleuth ⇨ Micrometer
- Full stack tracing & metrics
- Observation API

spring

Jonatan Ivanov

Observability for Modern JVM Applications

Fri Feb 23 @ 13:00

ST-Laurent 7 (English)

spring

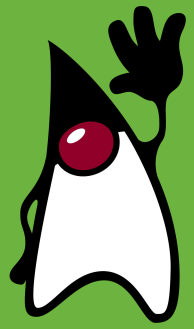
Dan Vega

Spring Boot 3 & Beyond!

Wed Feb 21 @ 14:00

Westmount 6 (English)

spring



JDK 21 and Beyond

JDK 17 & 21

JDK 17 - Sep 2021

- LTS
- Minimum JDK Baseline for:
  - Spring Framework 6.x
  - Spring Boot 3.x
  - Spring Cloud 2021.0+

JDK 21 - Sep 2023

- LTS
- Supported by:
  - Spring Framework 6.1+
  - Boot 3.2+
  - Spring Cloud 2023.0+
- Virtual Threads (Project Loom)

JDK 22 & 23

JDK 22 - March 2024

- Non-LTS
- Boot 3.3 / Spring Cloud 2023.0.x
- JEP 464 Scoped Values (Preview)
- JEP 462 Structured Concurrency (Preview)
- JEP 458, JEP 463 Learning Java Enhancements
- JEP 457 Class-File API (Preview)

JDK 22 Continued

- JEP 447 Statements before super(...) (Preview)
- JEP 459 String Templates (Preview)
- JEP 423 Region Pinning for G1

JDK 23

- Non-LTS
- Boot 3.4 / Spring Cloud 2024.0.x

https://www.infoworld.com/article/3708329/jdk-22-the-new-features-in-java-22.html

Performance

Stopwatch by M. Oki Orlando from NounProject.com

Virtual Threads - Project Loom

https://docs.spring.io/spring-framework/reference/integration/scheduling.html

https://docs.spring.io/spring-boot/docs/current/reference/html/features.html#features.spring-application.virtual-threads

- “A new degree of scalability for Spring MVC applications”
- Scheduler & Executor support in Spring Framework
- Spring Boot `spring.threads.virtual.enabled=true`
    - Auto-config for Framework, Tomcat and Jetty

JMV Checkpoint Restore

https://docs.spring.io/spring-framework/reference/integration/checkpoint-restore.html

https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/#deployment.efficient.checkpoint-restore

- Coordinated Restore at Checkpoint (CRaC)
- Immediate startup for Spring deployments on HotSpot
- Startup a JVM, take a snapshot, & restore the snapshot
- Lifecycle Management updated for consistent resource handling on stop/restart
- More to come

Class Data Sharing (CDS)

https://docs.oracle.com/en/java/javase/21/vm/class-data-sharing.html

https://docs.spring.io/spring-framework/reference/integration/cds.html

- Reduces startup time & memory footprint
- Mature, included in JDK 10 as JEP 310
- Startup a JVM, createst CDS archive on exit
- Use CDS archive and benefit
- More to come

New Clients

Radio by Nick Bluth from NounProject.com

RestClient

https://docs.spring.io/spring-framework/reference/integration/rest-clients.html#rest-restclient

https://docs.spring.io/spring-boot/docs/current/reference/html/io.html#io.rest-client.restclient

- In the beginning was **`RestTemplate`**
- Spring Framework 5 introduced **`WebClient`**: a non-blocking, reactive HTTP client with a fluent API, but…
- **`RestClient`** is the synchronous analog to **`WebClient`**

RestClient

```
@Service
public class MyService {

    private final RestClient restClient;

    public MyService(RestClient.Builder builder) {
        this.restClient = builder.baseUrl("https://example.org").build();
    }

    public Details someRestCall(String name) {
        return this.restClient.get()
                .uri("/{name}/details", name)
                .retrieve()
                .body(Details.class);
    }

}
```

JdbcClient

https://docs.spring.io/spring-framework/reference/data-access/jdbc/core.html#jdbc-JdbcClient

https://docs.spring.io/spring-boot/docs/current/reference/html/data.html#data.sql.jdbc-client

- In the beginning was **`JdbcTemplate`**
- **`JdbcClient`** is unified JDBC client API with a fluent interaction model

JdbcClient

```
@Component
public class MyBean {

    private final JdbcClient jdbcClient;

    public MyBean(JdbcClient jdbcClient) {
        this.jdbcClient = jdbcClient;
    }

    public Optional<Actor> findActorById(long id) {
        String sql = "select first_name, last_name from t_actor where id = ?";
        return this.jdbcClient.sql(sql)
            .param(1212L);
            .query(Actor.class)
            .optional();
    }

}
```

spring

Jetty HttpClient

https://docs.spring.io/spring-framework/reference/integration/rest-clients.html#rest-request-factories

https://docs.spring.io/spring-boot/docs/current/reference/html/io.html#io.rest-client.clienthttprequestfactory

- The Jetty HttpClient is now supported in **`RestTemplate`**, **`RestClient`**, and **`WebClient`**.

Testing

testing by littlestar23 from NounProject.com (CC BY 3.0)

https://docs.spring.io/spring-boot/docs/current/reference/html/features.html#features.testing

Testcontainers

https://docs.spring.io/spring-boot/docs/current/reference/html/features.html#features.testing.testcontainers

https://testcontainers.com/

- The Testcontainers library provides a way to manage services that run inside Docker containers during testing.
- Supports native **`@Testcontainers`** & **`@Container`** annotations.
- The Spring Boot **`@ServiceConnection`** annotation bridges metadata (host, ports, etc.) from Testcontainers to Spring Boot configuration.
- Support for many technologies by default, including RabbitMQ, Kafka, MongoDB, Neo4j, JDBC databases, Flyway and more.

Testcontainers

```
@Testcontainers
@SpringBootTest
class MyIntegrationTests {

    @Container
    @ServiceConnection
    static Neo4jContainer<?> neo4j = new Neo4jContainer<>("neo4j:5");

    @Test
    void myTest() {
        // ...
    }

}
```

spring

Testcontainers DevEx

https://docs.spring.io/spring-boot/docs/current/reference/html/features.html#features.testcontainers

https://testcontainers.com/

- What about just running my app?
- Spring Boot provides SpringApplication.from() to launch your app with the ‘test’ classpath and allows configuring Testcontainers to bootstrap external dependencies.
- `Container` beans lifecycle(start/stop) is managed by Spring boot

Testcontainers DevEx

```
public class TestMyApplication {

    public static void main(String[] args) {
        SpringApplication.from(MyApplication::main).run(args);
    }

}

@TestConfiguration(proxyBeanMethods = false)
public class MyContainersConfiguration {

    @Bean
    @RestartScope // use with DevTools to keep state between restarts
    @ServiceConnection
    public Neo4jContainer<?> neo4jContainer() {
        return new Neo4jContainer<>("neo4j:5");
    }

}
```

Docker Compose

https://docs.spring.io/spring-boot/docs/current/reference/html/features.html#features.docker-compose

https://docs.docker.com/compose/

- Docker Compose is to provide and manage multiple containers for required services.
- Spring Boot provides the `spring-boot-docker-compose` modules to manage the lifecycle of the locally installed `` `docker compose` `` command.
- Spring Boot creates service connection beans for supported containers.

New Projects

new by Ilham Fitrotul Hayat from Noun Project (CC BY 3.0)

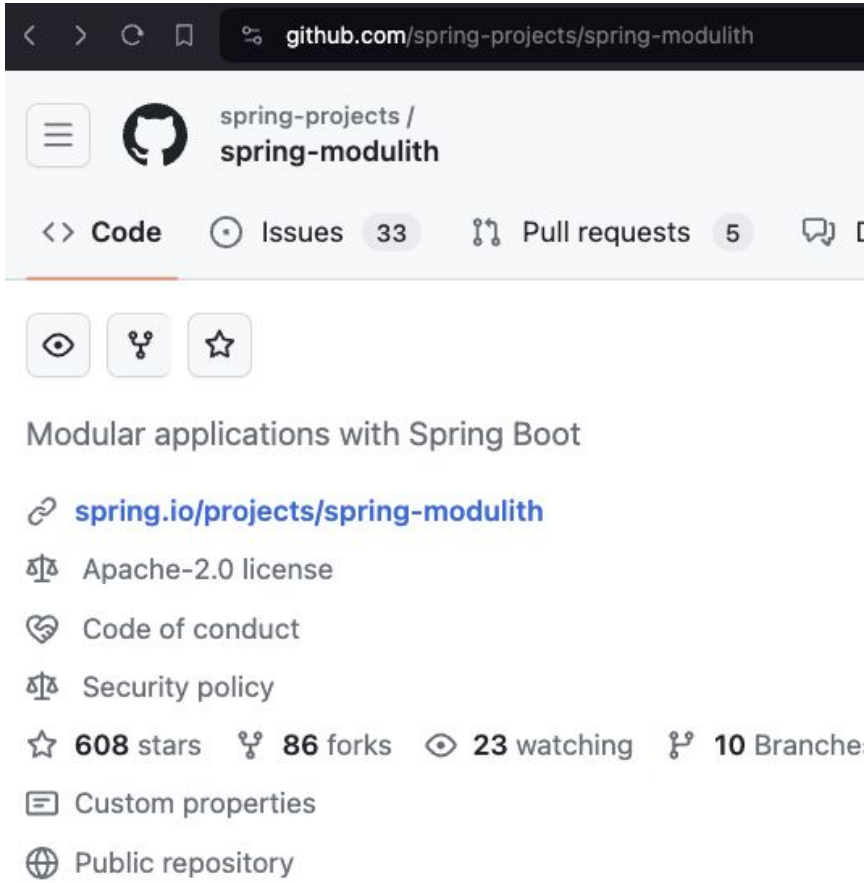
Spring Modulith

Supports Spring Boot developers to implement **logical modules** in a loosely coupled way.

Includes:

- Structural validation
- Documentation
- Testing
- Runtime observation

https://docs.spring.io/spring-modulith/reference/

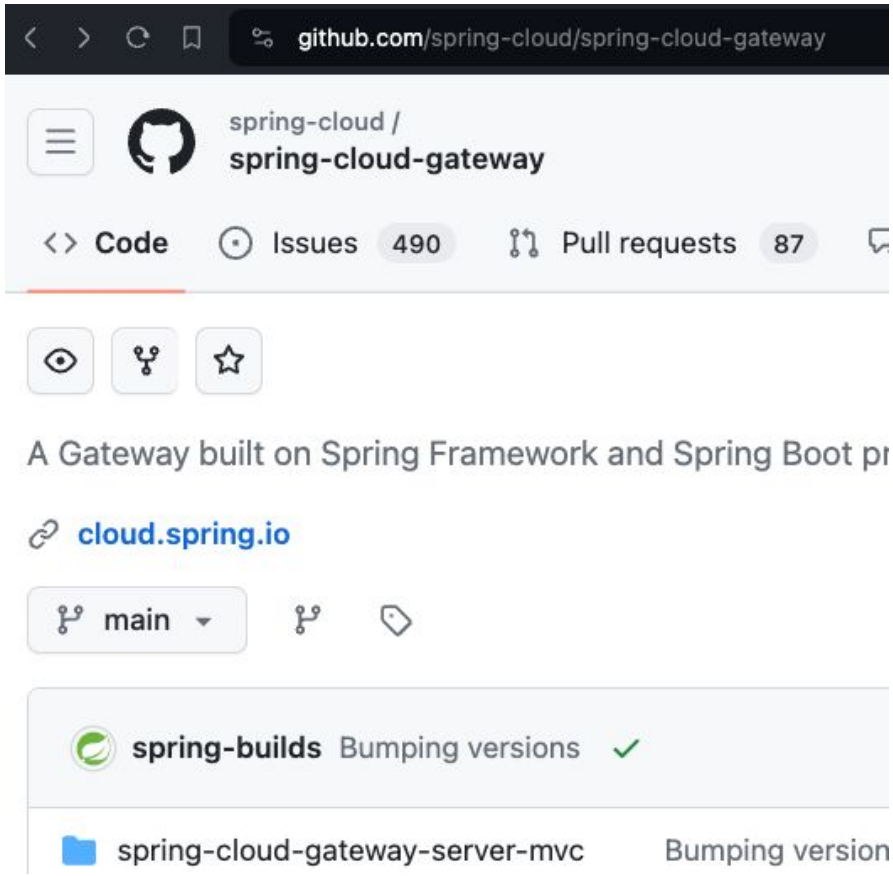


spring

Spring Cloud Gateway MVC

An API Gateway built on Spring WebMvc.fn

- Virtual Threads - Project Loom
- Many Predicates and Filters
- Simplicity
- Explicit vs Implicit
- Org Requires Tomcat or similar
- Embed in Spring MVC Apps
- https://youtu.be/UyxUkAagLFs



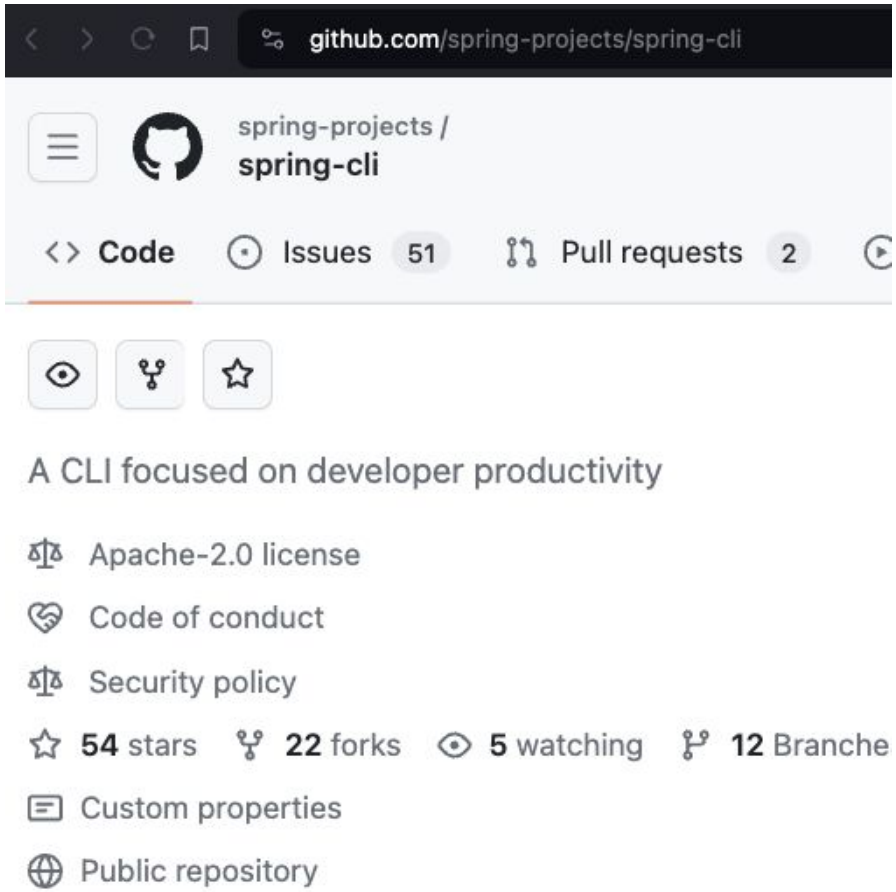
spring https://docs.spring.io/spring-cloud-gateway/reference/spring-cloud-gateway-server-mvc.html

Spring CLI

Built to help create new Spring Boot projects and add functionality to existing projects.

- Built on Spring Shell
- `boot new`
- `boot add`
- AI Commands *(see next slide)*
- Initializr support
- More to come

https://docs.spring.io/spring-cli/reference/



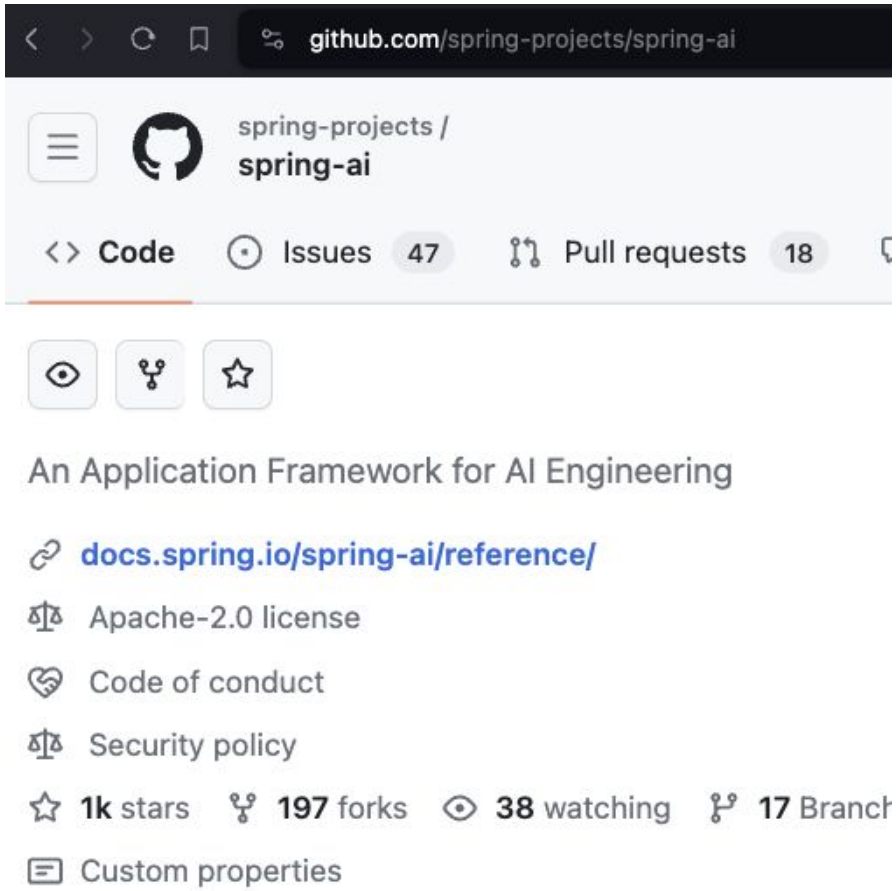
spring

Spring AI

Allows integration of artificial intelligence to Spring Boot applications without unnecessary complexity.

- `ChatClient`, `EmbeddingClient`, and `VectorStore` interfaces
- `PromptTemplate` class
- OpenAI, Azure OpenAI, HuggingFace, Ollama and other implementations
- Spring Boot starters
- Spring CLI commands
  - `spring boot new ai`
  - `spring boot add ai`

https://docs.spring.io/spring-ai/reference/



spring

Roadmap

Map by Endri Widada from NounProject.com

Support for the Previous Generation

Support Timelines (End Dates)

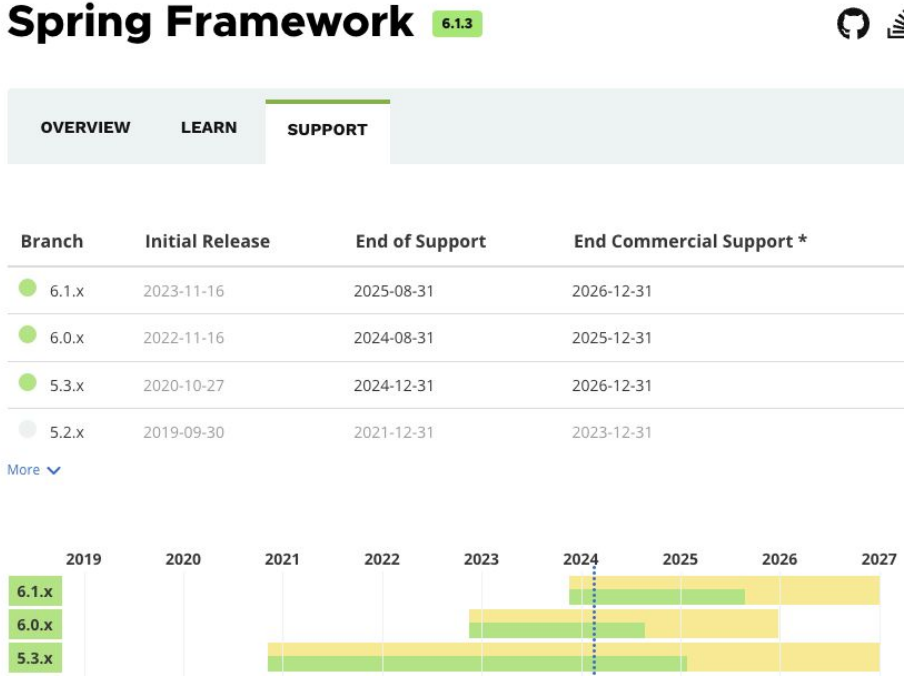
Spring Framework

- 5.3.x OSS Dec 2024, Commercial Dec 2026
- 6.0.x OSS Aug 2024, Commercial Dec 2025
- 6.1.x OSS Aug 2025, Commercial Dec 2026

Spring Boot/Spring Cloud

- 2.7.x/2021.0.x ~~OSS Nov 2023~~, Comm Aug 2024
- 3.0.x/2022.0.x ~~OSS Nov 2023~~, ~~Comm Feb 2024~~
- 3.1.x/2023.0.x OSS May 2024, Comm Aug 2025
- 3.2.x/2023.0.x OSS Nov 2024, Comm Feb 2026

See spring.io for individual projects



New in 2024

More CDS

- Buildpack support
- Exploded Jar Layout
  - Spring Boot build plugins Issue #38276
- Project Leyden (CDS on steroids)
- AOT for HotSpot, CRaC & Leyden

Spring Framework 6.2

- Jakarta EE 11

Commercial Releases

- Quarterly
- CVEs and Customer-requested bug fixes
- Dependency updates, patch level only
- Private maven repository

Roadmap

When will Spring Framework and Friends GA?

- Spring Framework 6.2.0
  - Milestones (Mx) Summer 2024, Release Candidates Early Fall 2024
  - GA - Mid to late Fall 2024
- Spring Boot
  - 3.3.0 - M2 Feb 2024, M3 Mar 2024, RC1 Apr 2024, GA May 2024
    - Corresponding Spring Cloud 2023.0.x release compatible with 3.3.0
  - 3.4.0 - Milestones after Framework 6.2 milestone, RC1 Oct 2024, GA Nov 2024
- Spring Cloud 2024.0.0
  - M1 (Jun), M2 (Jul), M3 (Aug), M4 (Sep), RC1 (Oct)
  - GA - Late Nov, arly Dec following Spring Boot 3.4.0 GA

Questions

Thank you

https://gibb.tech/preso/confoo-2024-whats-new-in-spring/

@spencerbgibb

social.sdf.org/@spencergibb

Question by Mani Cheng from NounProject.com
Tweet by James Smith from NounProject.com
Mastodon by Icon 54 from iconscout.com

Copyright © 2005-2024 Broadcom, Inc. or its affiliates.