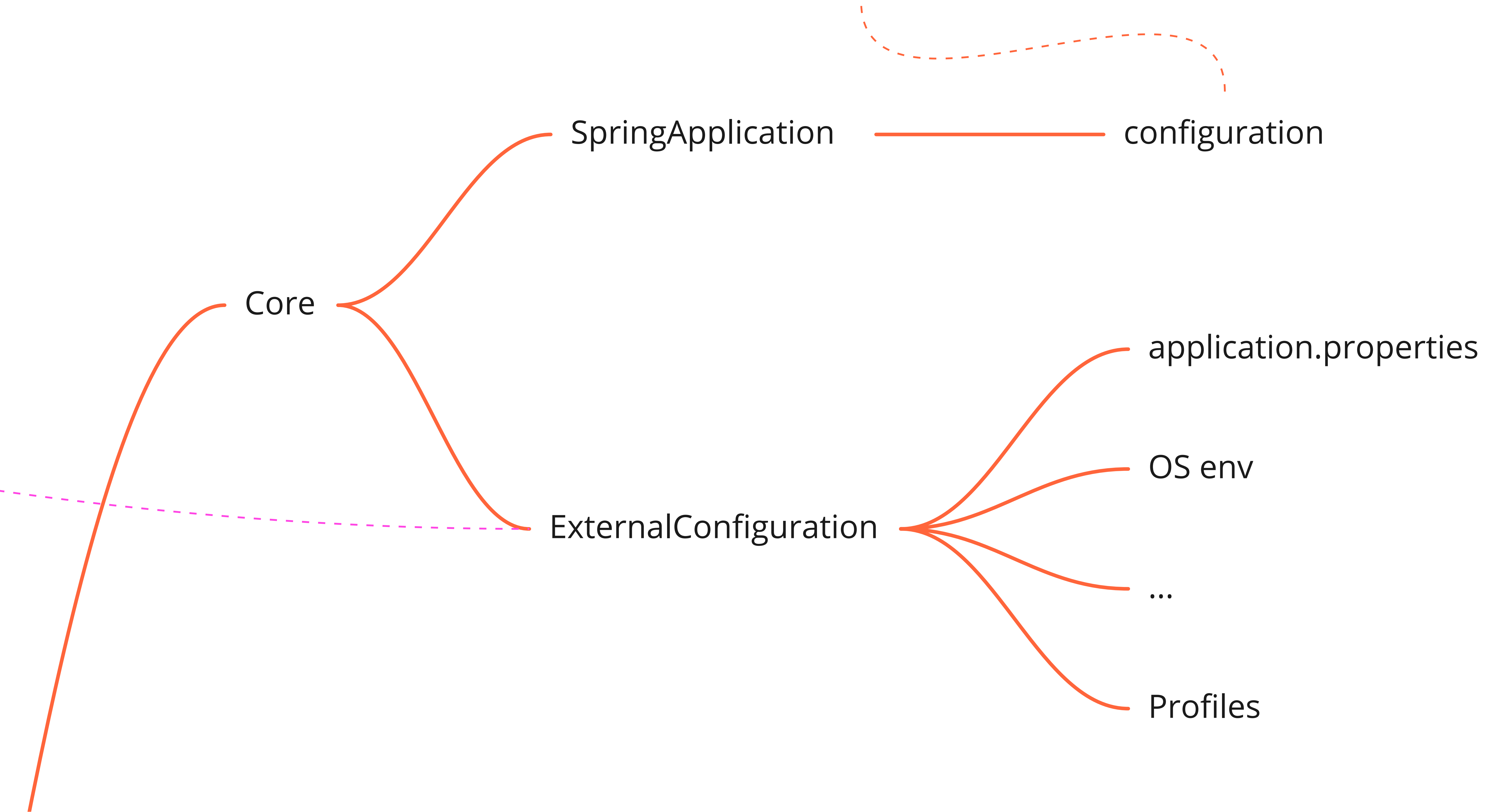
Spring Boot

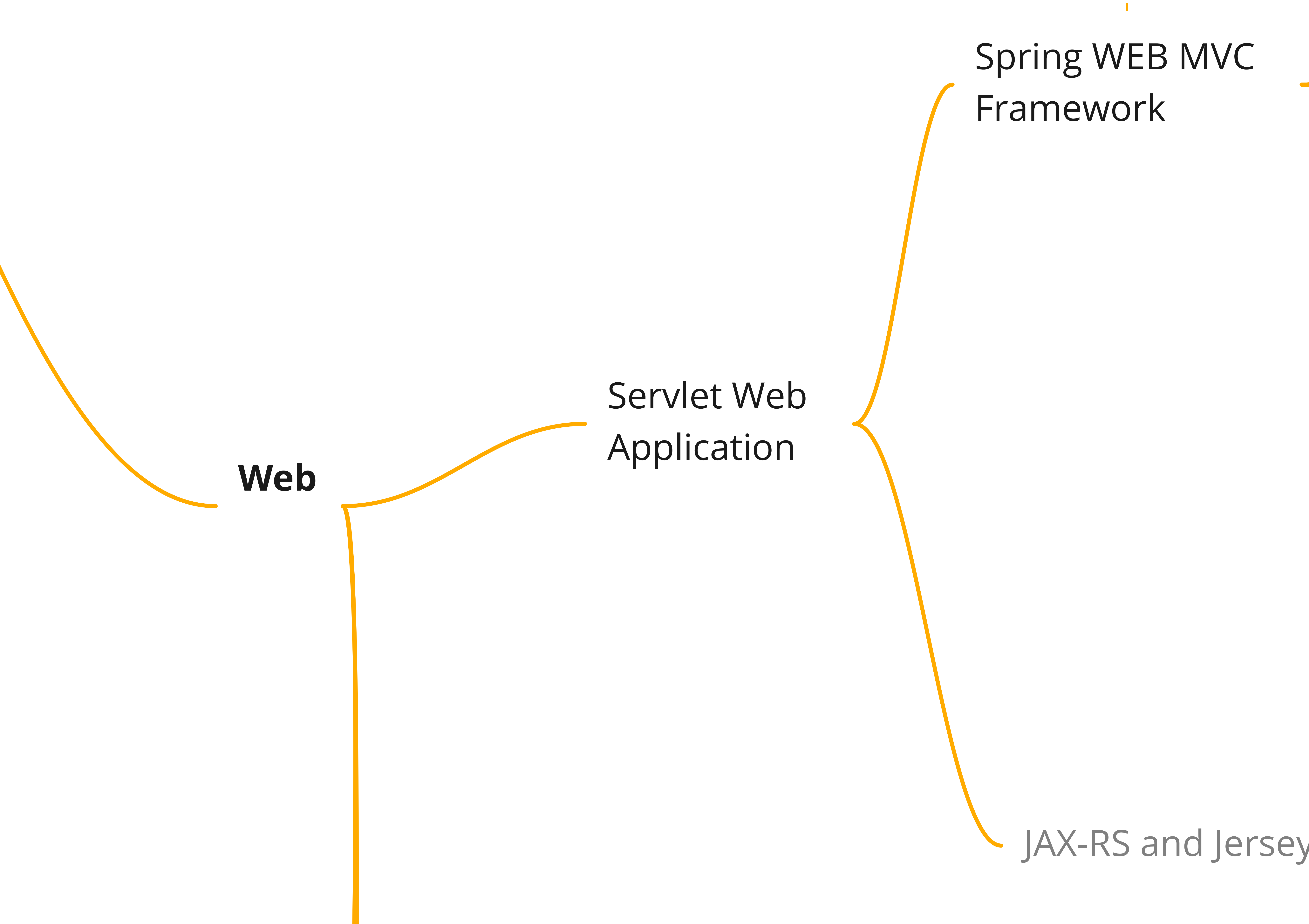
spring.io

Spring Boot

Level up your Java code and explore what Spring can do for you.

```java
import org.springframework.boot.Banner;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class MyApplication {

public static void main(String[] args) {
    SpringApplication application = new SpringApplication(MyApplication.class);
    application.setBannerMode(Banner.Mode.OFF); application.run(args); }
}
```




Spring WEB MVC Framework
Servlet Web Application
Web
JAX-RS and Jersey

```java
@RestController
@RequestMapping("/users")
public class MyRestController {

    private final UserRepository userRepository;

    private final CustomerRepository customerRepository;

    public MyRestController(UserRepository userRepository, CustomerRepository customerRepository) {
        this.userRepository = userRepository;
        this.customerRepository = customerRepository;
    }

    @GetMapping("/{userId}")
    public User getUser(@PathVariable Long userId) {
        return this.userRepository.findById(userId).get();
    }

    @GetMapping("/{userId}/customers")
    public List<Customer> getUserCustomers(@PathVariable Long userId) {
        return this.userRepository.findById(userId).map(this.customerRepository::findByUser).get();
    }

    @DeleteMapping("/{userId}")
    public void deleteUser(@PathVariable Long userId) {
        this.userRepository.deleteById(userId);
    }

}
```

Auto Configuration

HttpMessageConverter

Static Content

Error Handling

CORS

Template Engines

FreeMarker

Groovy

Thymeleaf

spring.io

Handling Form Submission

this guide is designed to get you productive as quickly as possible and using the latest Spring project releases and techniques as recommended by the Spring team

Mustache

Reactive Web Application

Spring Session

Spring Security

Spring Data’s mission is to provide a familiar and consistent, Spring-based programming model for data access while still retaining the special traits of the underlying data store.

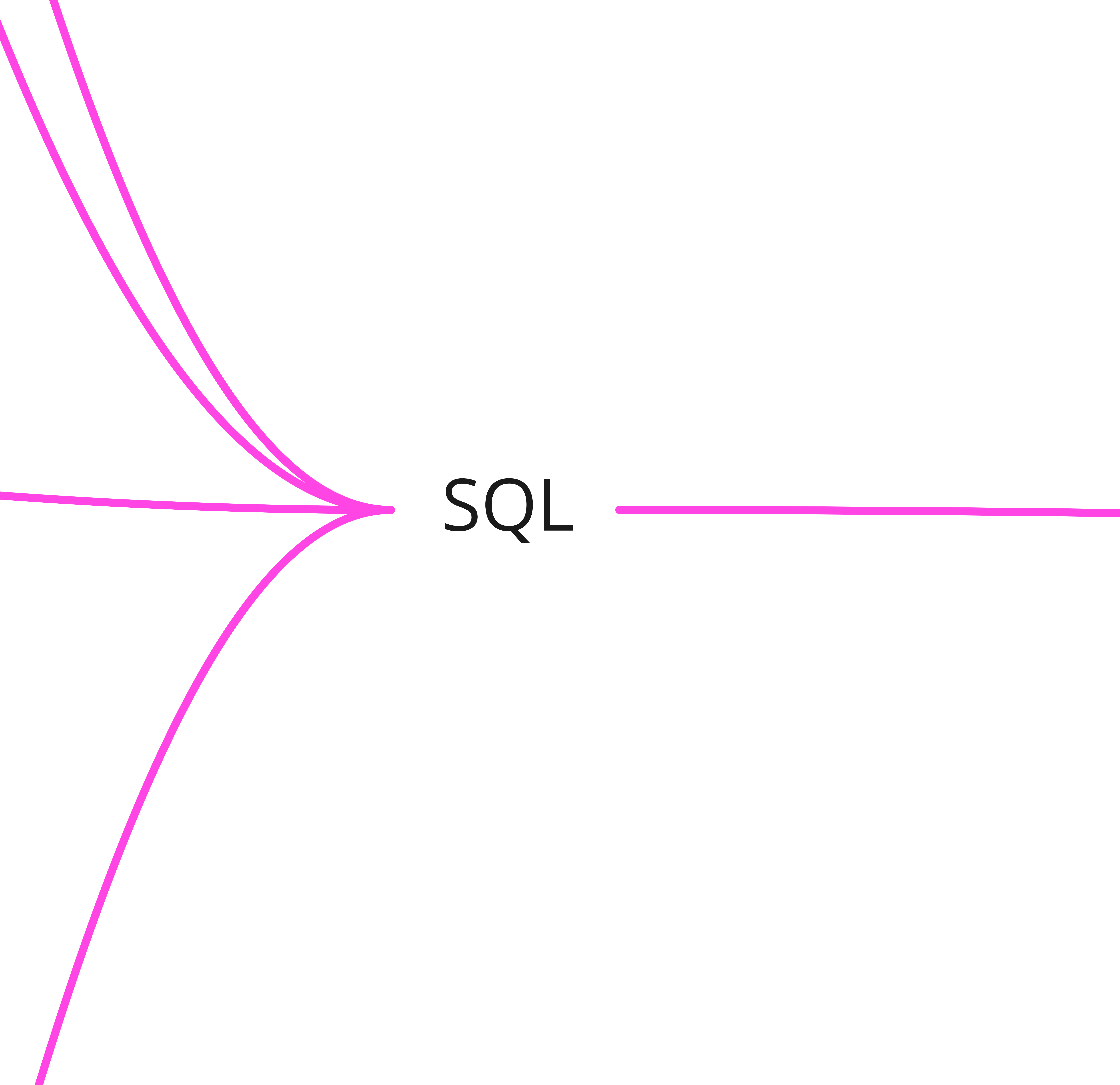
Spring Boot integrates with a number of data technologies, both SQL and NoSQL.

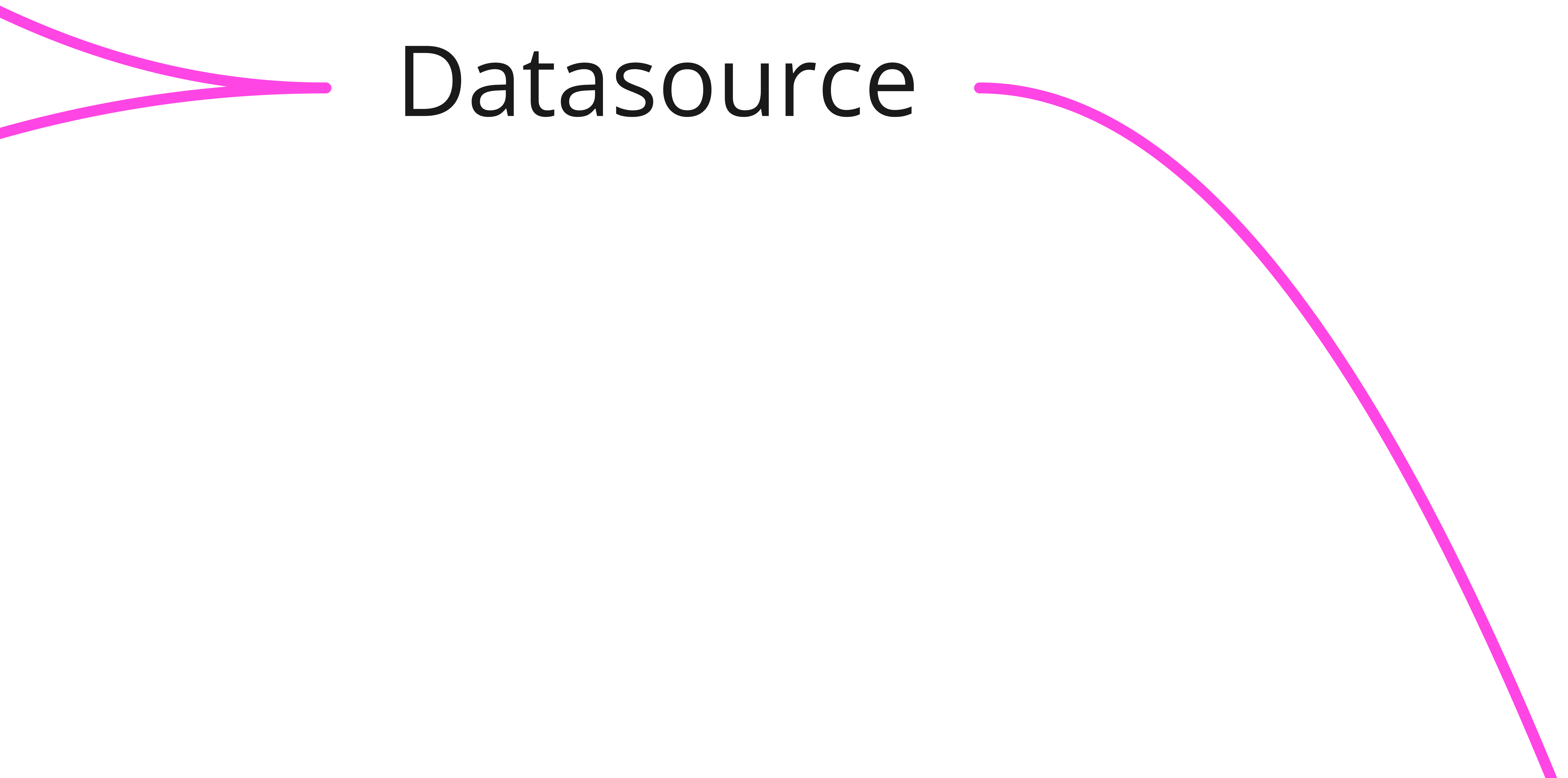
Data

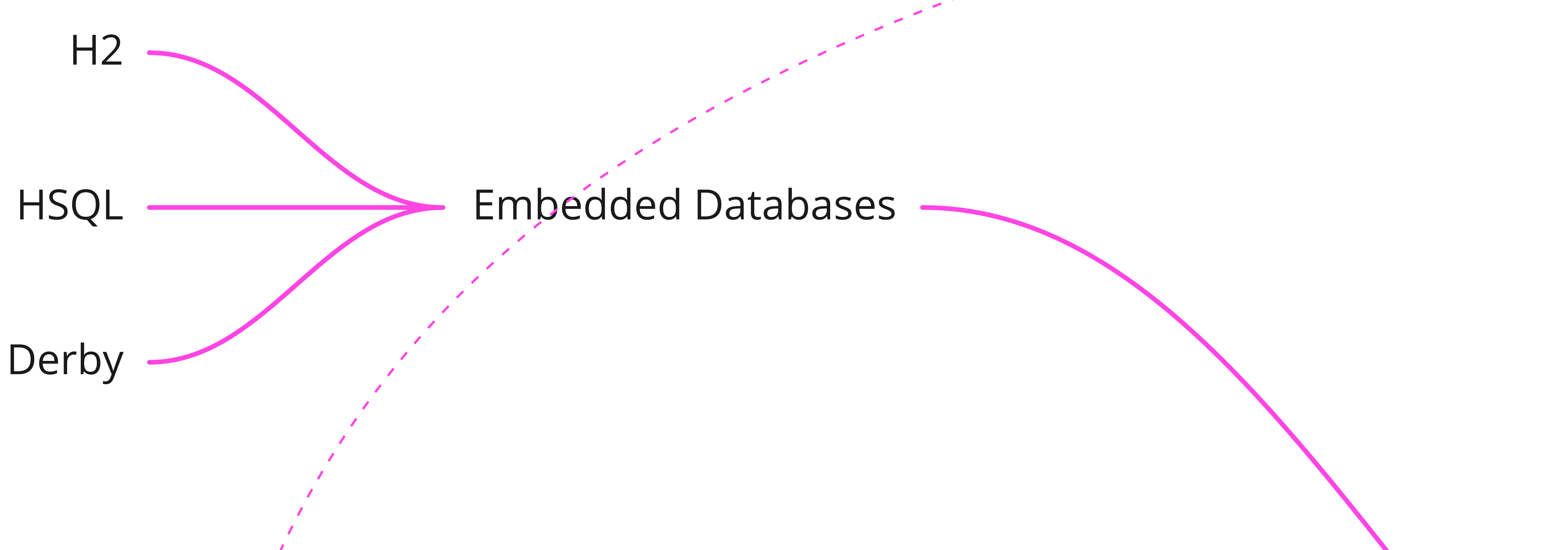
spring.io

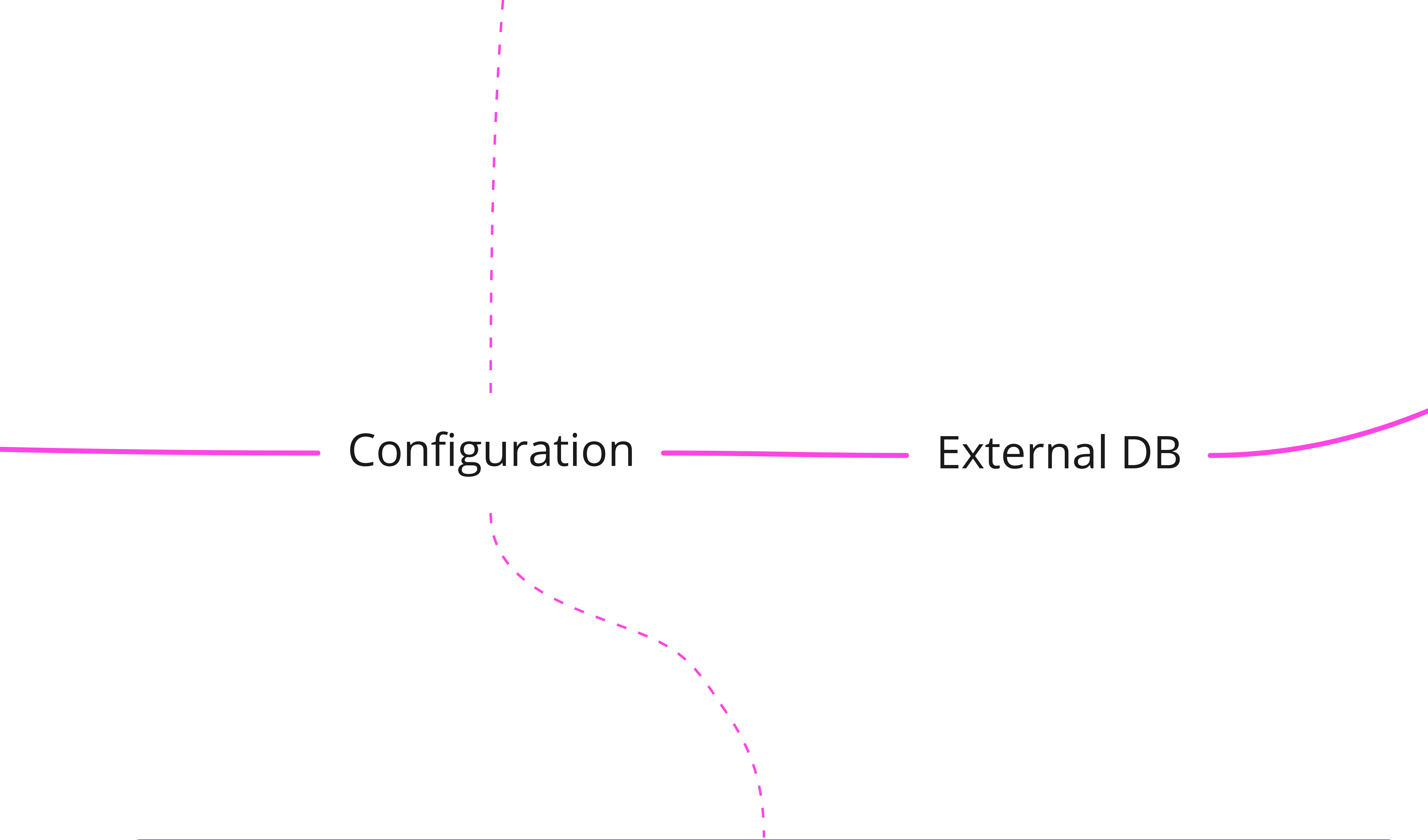
Spring Data

Spring Data is an umbrella project consisting of independent projects with, in principle, different release cadences. To manage the portfolio, a BOM (Bill of Materials - see this example) is published with a curated set of dependencies on the individual...


SQL


Datasource


H2
HSQL
Derby
Embedded Databases



```
spring.datasource.url=jdbc:mysql://localhost/test
spring.datasource.username=dbuser
spring.datasource.password=dbpass
```

spring.io

Accessing data with MySQL

this guide is designed to get you productive as quickly as possible and using the latest Spring project releases and techniques as recommended by the Spring team

MySql Connector

```xml
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>8.0.25</version>
</dependency>
```

JDBCTemplate

```
@Component
public class MyBean {

    private final JdbcTemplate jdbcTemplate;

    public MyBean(JdbcTemplate jdbcTemplate) {
        this.jdbcTemplate = jdbcTemplate;
    }

    public void doSomething() {
        this.jdbcTemplate ...
    }

}
```

```xml
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jdbc</artifactId>
</dependency>
```

Spring Data JPA, part of the larger Spring Data family, makes it easy to easily implement JPA based repositories. This module deals with enhanced support for JPA based data access layers. It makes it easier to build Spring-powered applications that use data access technologies.

JPA and Spring Data JPA

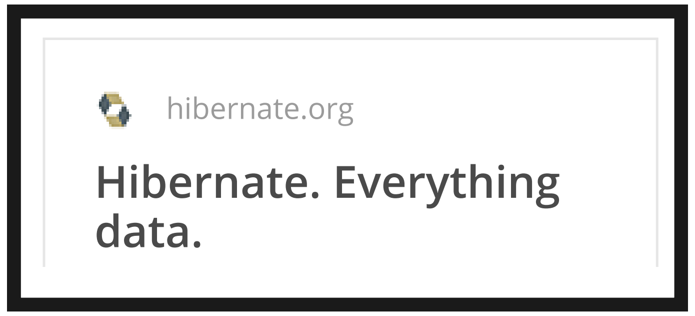
jakarta.ee

JAKARTA EE

Jakarta Persistence 3.0 | The Eclipse Foundation

Release for Jakarta EE 9

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
</dependency>
```


hibernate.org

Hibernate. Everything data.

spring.io

Spring Data JPA

Level up your Java code and explore what Spring can do for you.

docs.spring.io

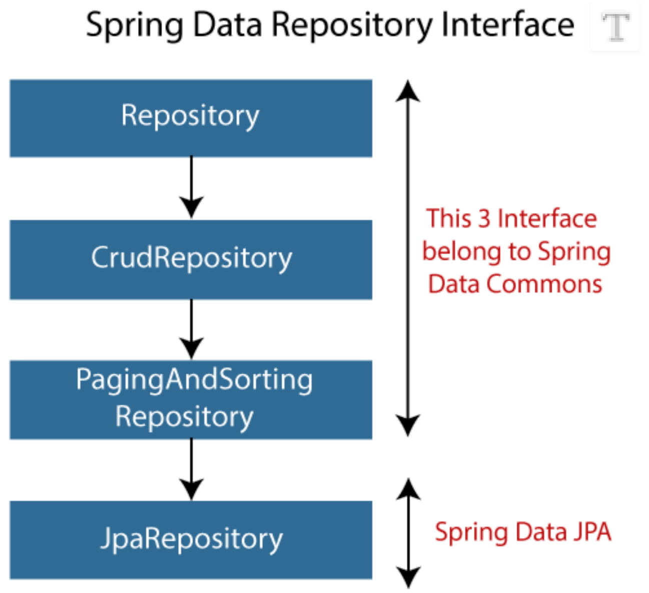
Chapter 12. Object Relational Mapping (ORM) data access

The Spring Framework provides integration with Hibernate, JDO, Oracle TopLink, iBATIS SQL Maps and : in terms of resource management, DAO implementation support, and transaction strategies. For example for Hibernate, there is first-class support with lo...

Entity Classes

```java
import java.io.Serializable;

import jakarta.persistence.Column;
import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.Id;

@Entity
public class City implements Serializable {

@Id @GeneratedValue private Long id;

@Column(nullable = false) private String name;

@Column(nullable = false) private String state;

// ... additional members, often include @OneToMany mappings

protected City() { // no-args constructor required by JPA spec // this one is protected
    since it should not be used directly }

public City(String name, String state) {
    this.name = name;
    this.state = state;

}

public String getName() { return this.name; }
public String getState() { return this.state; }
}
```

Replaces persistence.xml


Spring Data Repository Interface
Repository
CrudRepository
PagingAndSorting Repository
JpaRepository
This 3 Interface belong to Spring Data Commons
Spring Data JPA

The central interface in the Spring Data repository abstraction is Repository. It takes the domain class to manage as well as the ID type of the domain class as type arguments. This interface acts primarily as a marker interface to capture the types to work with and to help you to discover interfaces that extend this one.
The [CrudRepository] and [ListCrudRepository] interfaces provide sophisticated CRUD functionality for the entity class that is being managed.

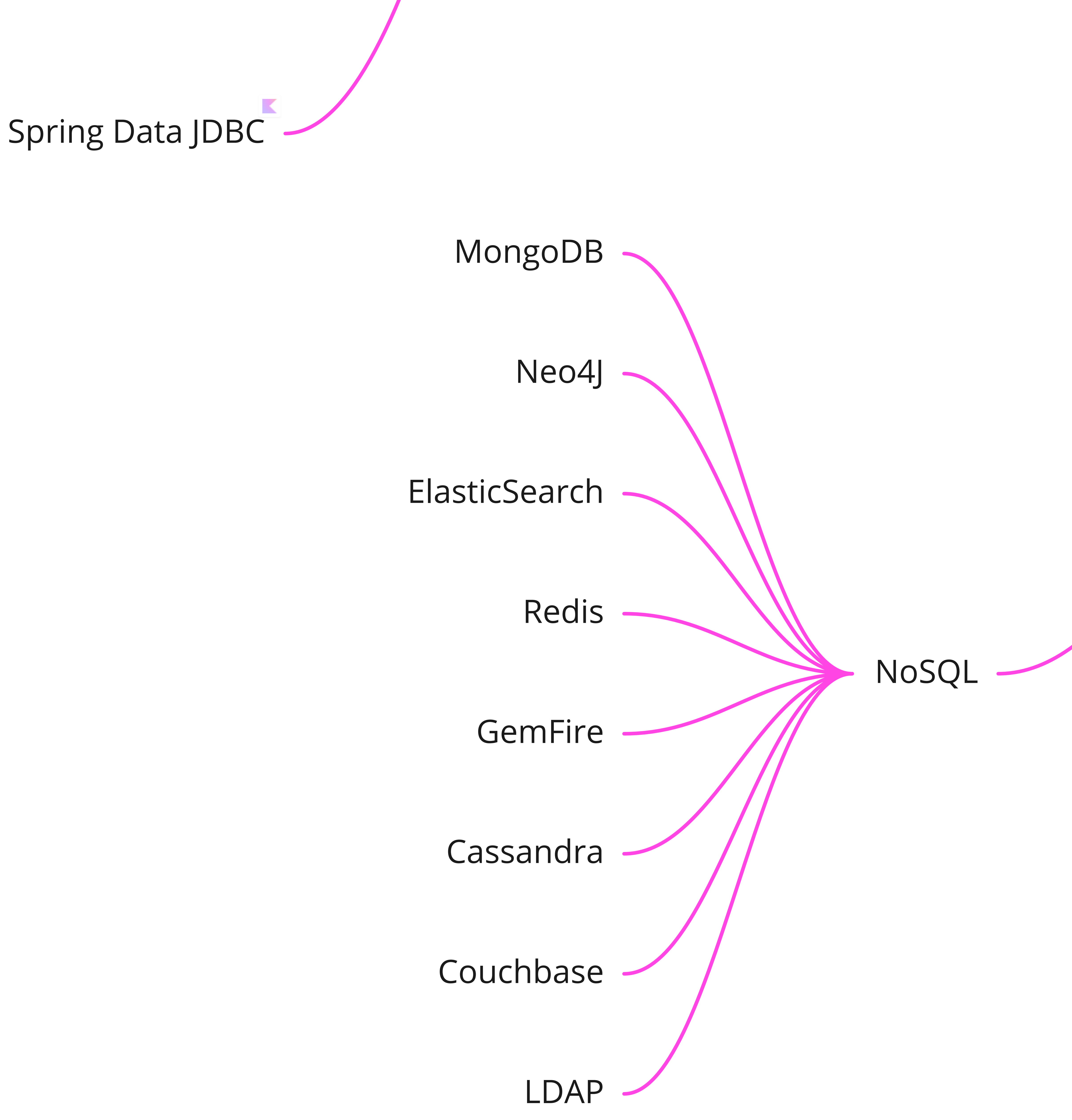
docs.spring.io

JpaRepository (Spring Data JPA Parent 3.0.0 API)

All Methods Instance Methods Abstract Methods Default Methods Deprecated Methods Deletes the entities identified by the given ids using a single query. Deletes all entities in a batch call. Deletes the given entities in a batch which means it will creat...

Repository

```java
import org.springframework.boot.docs.data.sql.jpaandspringdata.entityclasses.City;
import org.springframework.data.domain.Page;
import org.springframework.data.domain.Pageable;
import org.springframework.data.repository.Repository;

public interface CityRepository extends Repository<City, Long> {
    Page<City> findAll(Pageable pageable);
    City findByNameAndStateAllIgnoringCase(String name, String state);
}
```


Spring Data JDBC
MongoDB
Neo4J
ElasticSearch
Redis
GemFire
Cassandra
Couchbase
LDAP
NoSQL