Introduction to Microservices, Spring Boot & Spring Data

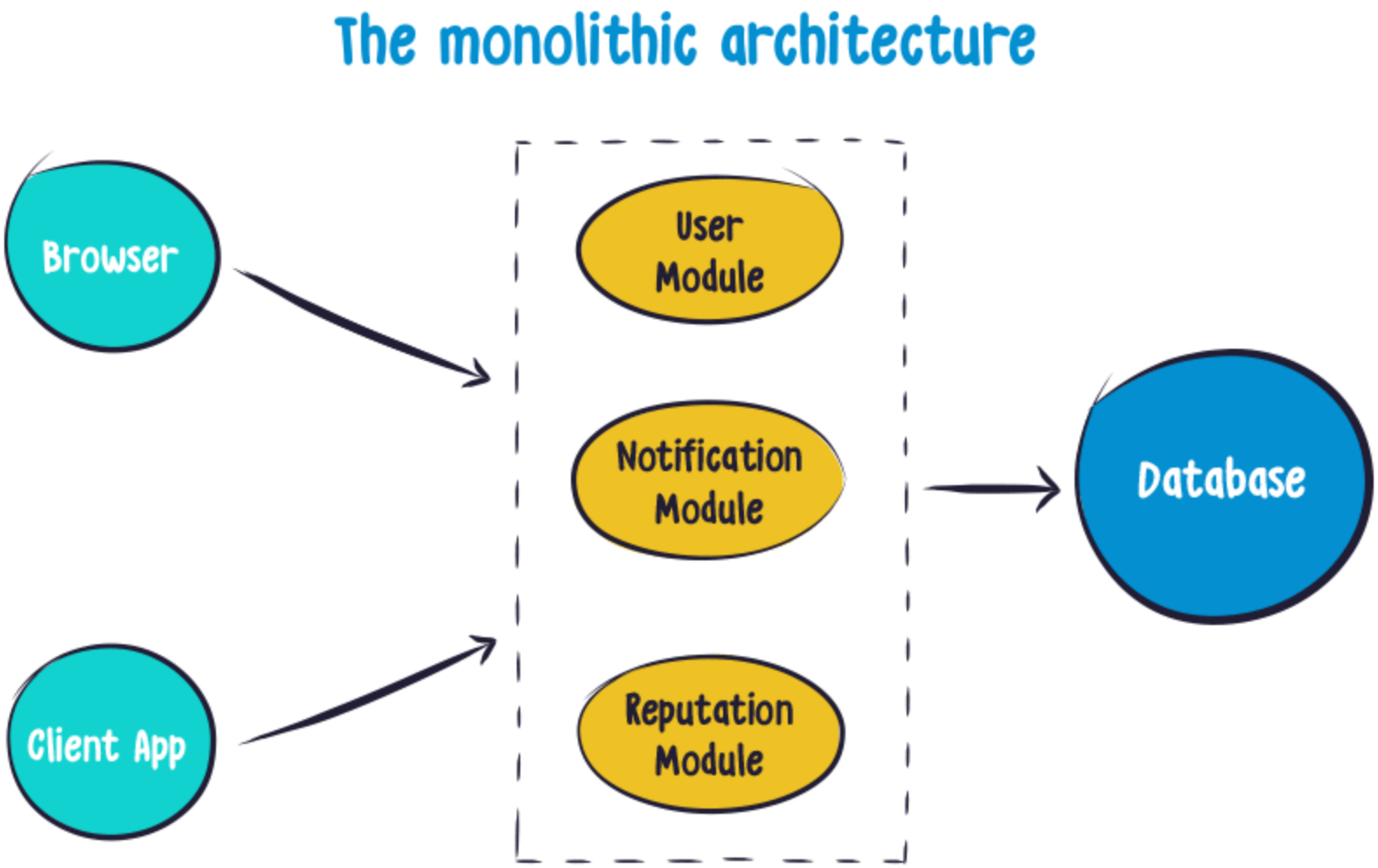
By

Anand Kulkarni

anand.pune38@gmail.com

Microservices

Monolithic Architecture



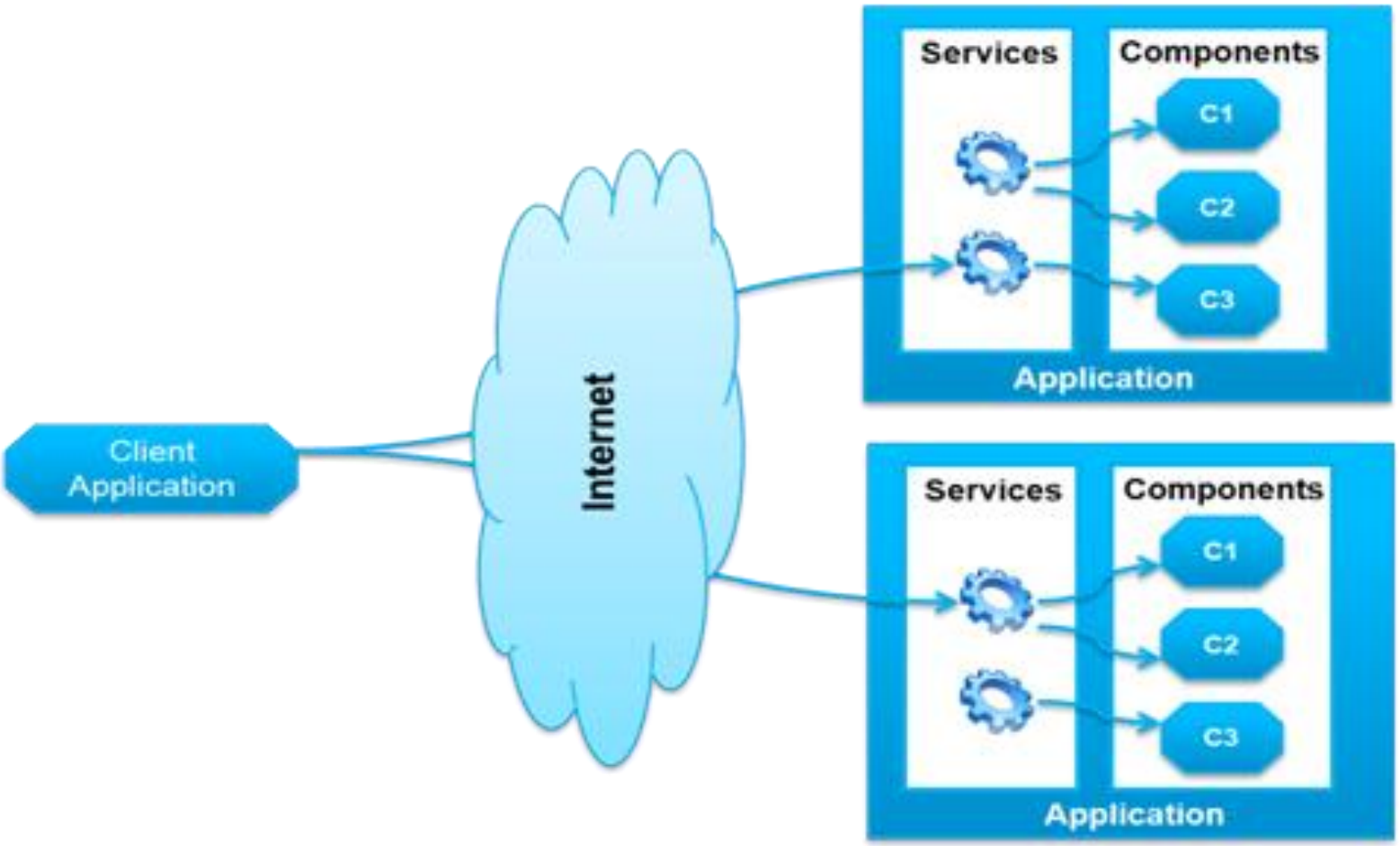
Monolithic Architecture

Monolithic architecture is built from single piece of material.

Monolithic application has single code base with multiple modules. Modules are divided as either for business features or technical features.

It has single build system which build entire application and/or dependency. It also has single executable or deployable binary.

Service Oriented Architecture (SOA)



Service Oriented Architecture (SOA)

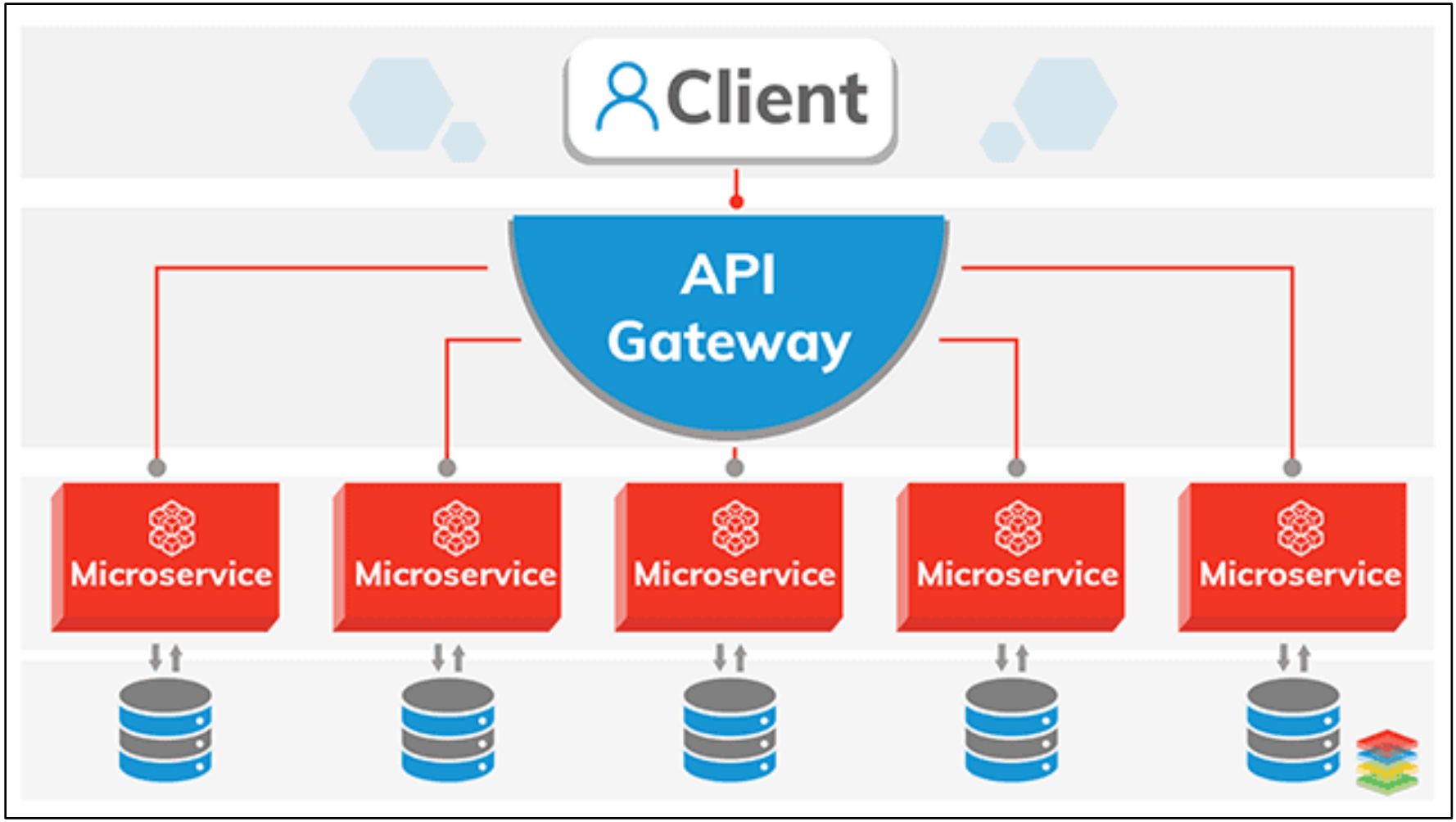
- SOA is a style of software design where services are provided to the other components by application components, through a communication protocol over a network.
- A SOA service is a discrete unit of functionality that can be accessed remotely and acted upon and updated independently.

Service Oriented Architecture (SOA)

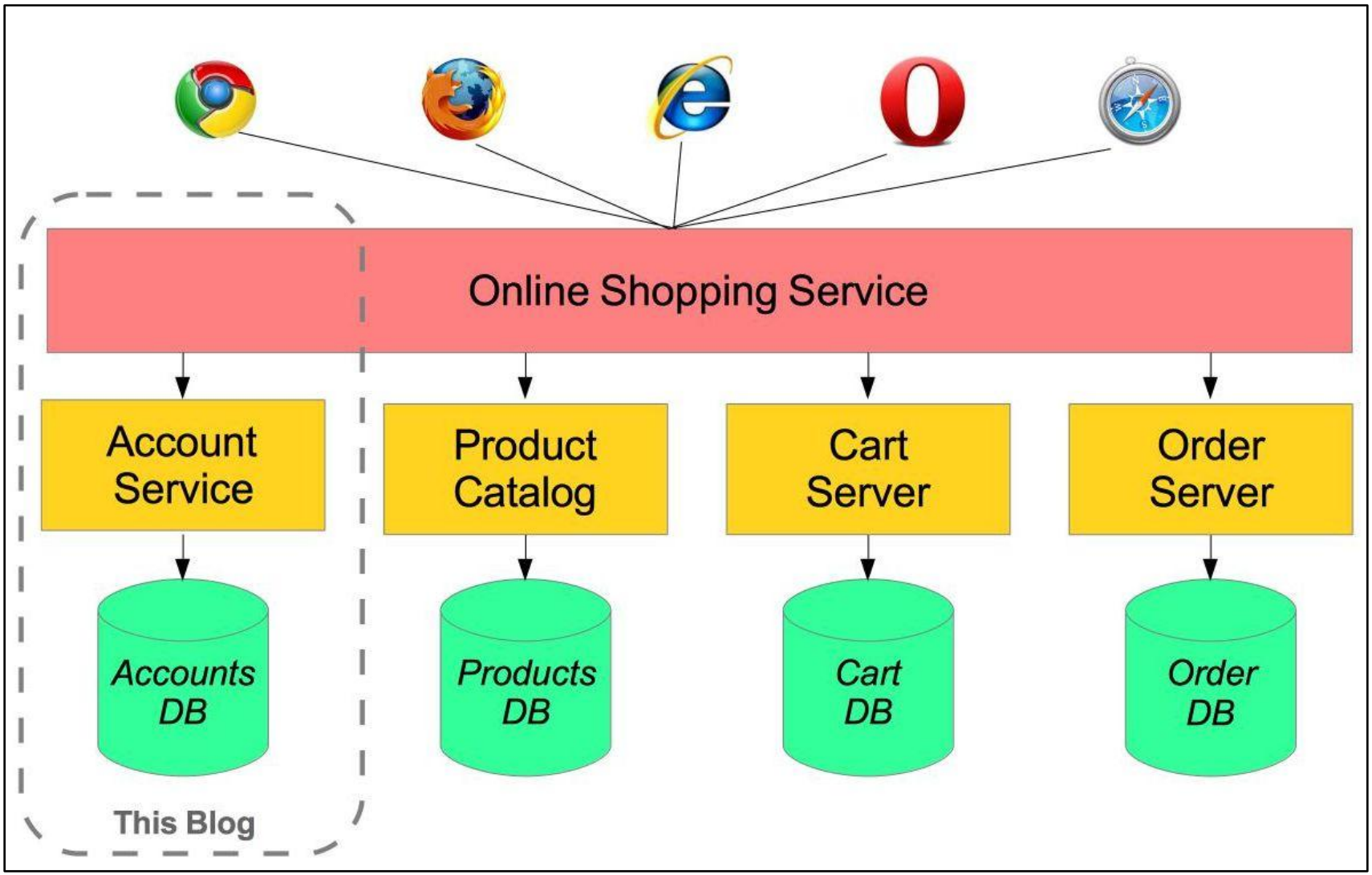
Services provided by SOA has following properties:

1. It logically represents a business activity with a specified outcome.
2. It is self-contained.
3. It is a black box for its consumers, meaning the consumer may no be aware of the service's inner workings.
4. It may consist of other underlying services.

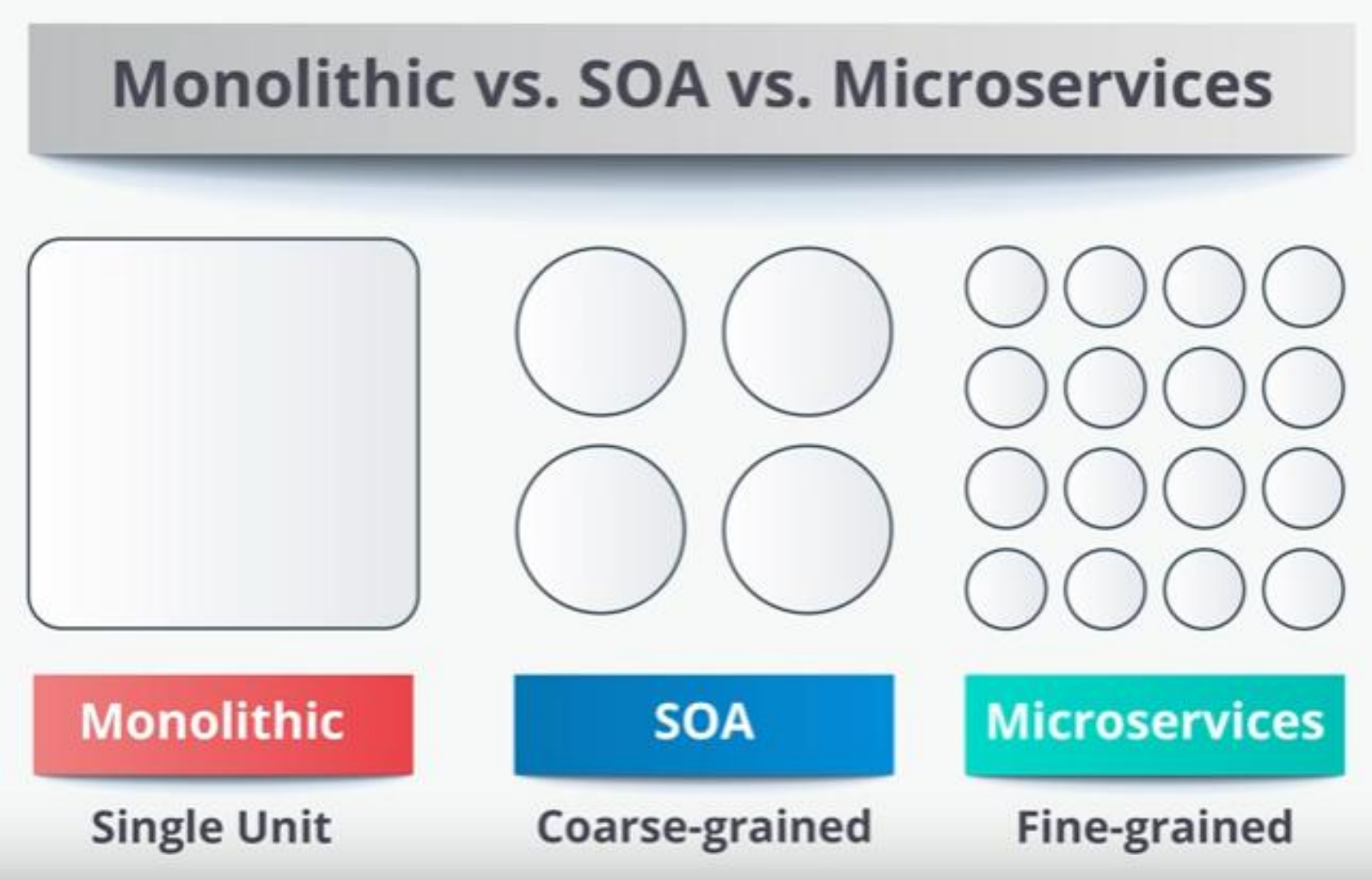
Microservices Architecture



Microservices in Shopping Cart App



Monolithic vs SOA vs Microservices



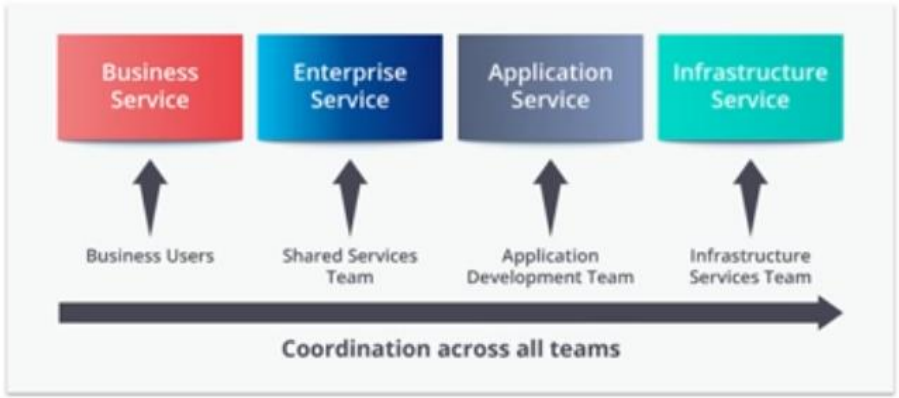
SOA vs Microservices

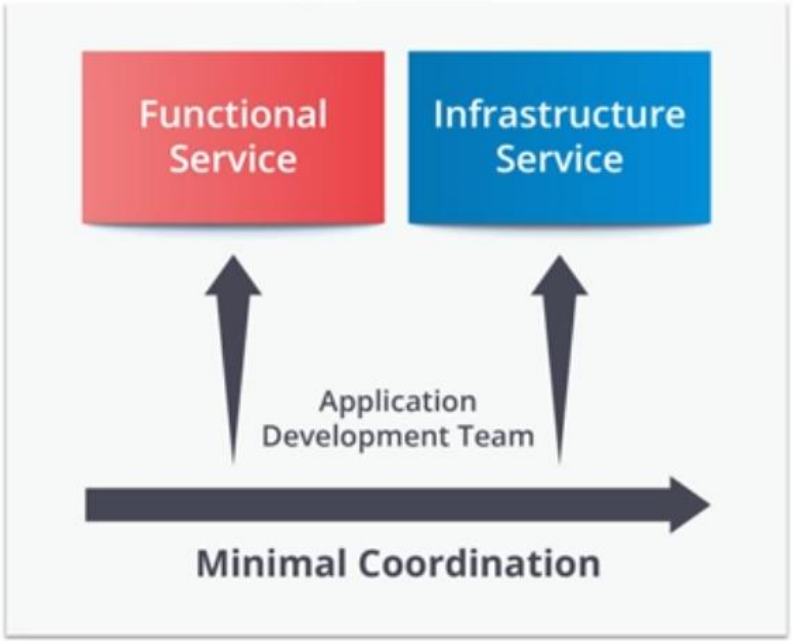
SOA Vs Microservice

SOA is like an orchestra where each artist is performing with his/her instrument while the music director guides them all.

With Microservices each dancer is independent and know what they need to do. If they miss some steps they know how to get back on the sequence.

SOA vs Microservices





SOA

Microservices

SOA vs Microservice

Business service – It is going to perform your core business operations. It can be represented by WSDL.

Enterprise service – It defines the functionality defined by Business service. Enterprise service does it with the help of Application service & Infrastructure service.

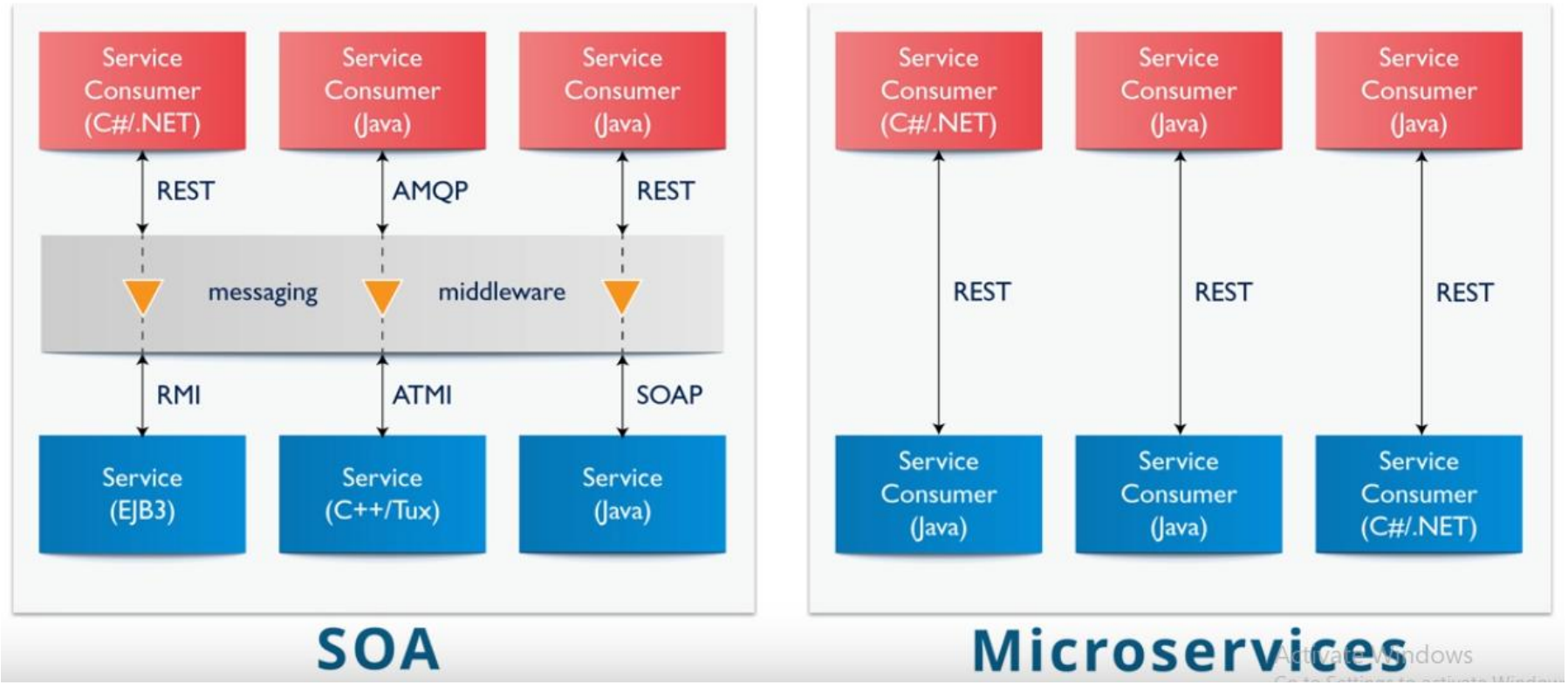
Application service – It is an actual implementation of core functionality of that business feature.

SOA vs Microservice

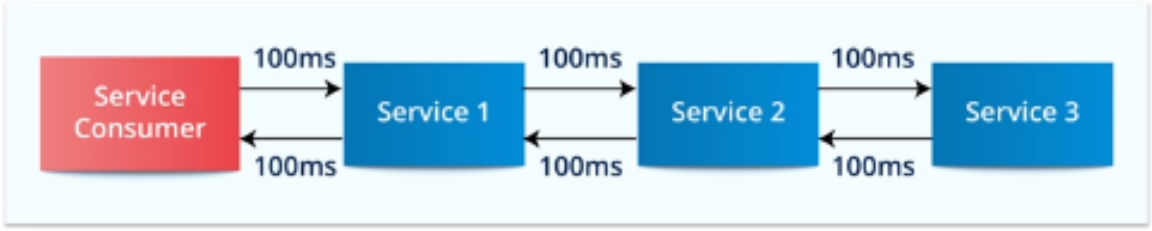
Infrastructure service – It represents operations those are non-technical such as logging, security etc.

Functional service – It is a combination of Business service, Enterprise service & Application service.

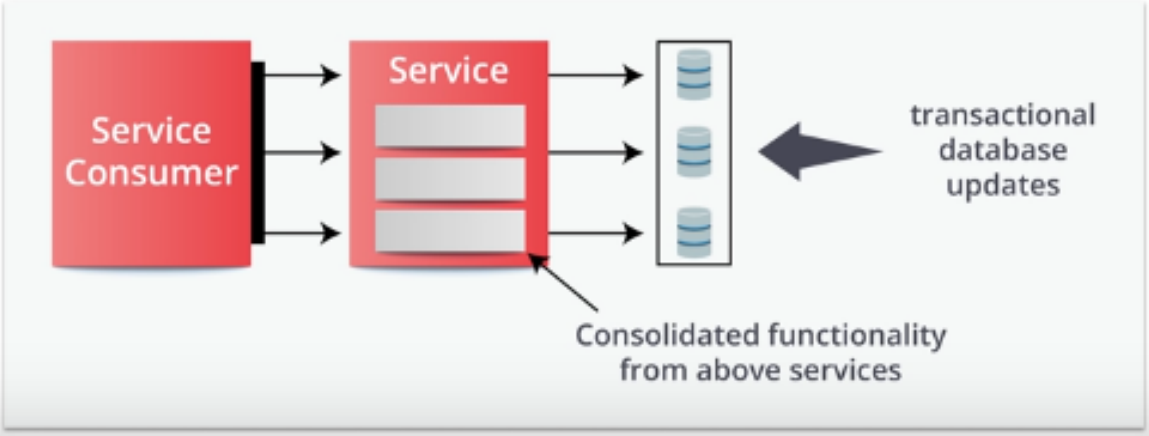
SOA vs Microservices



Service Granularity - SOA vs Microservices

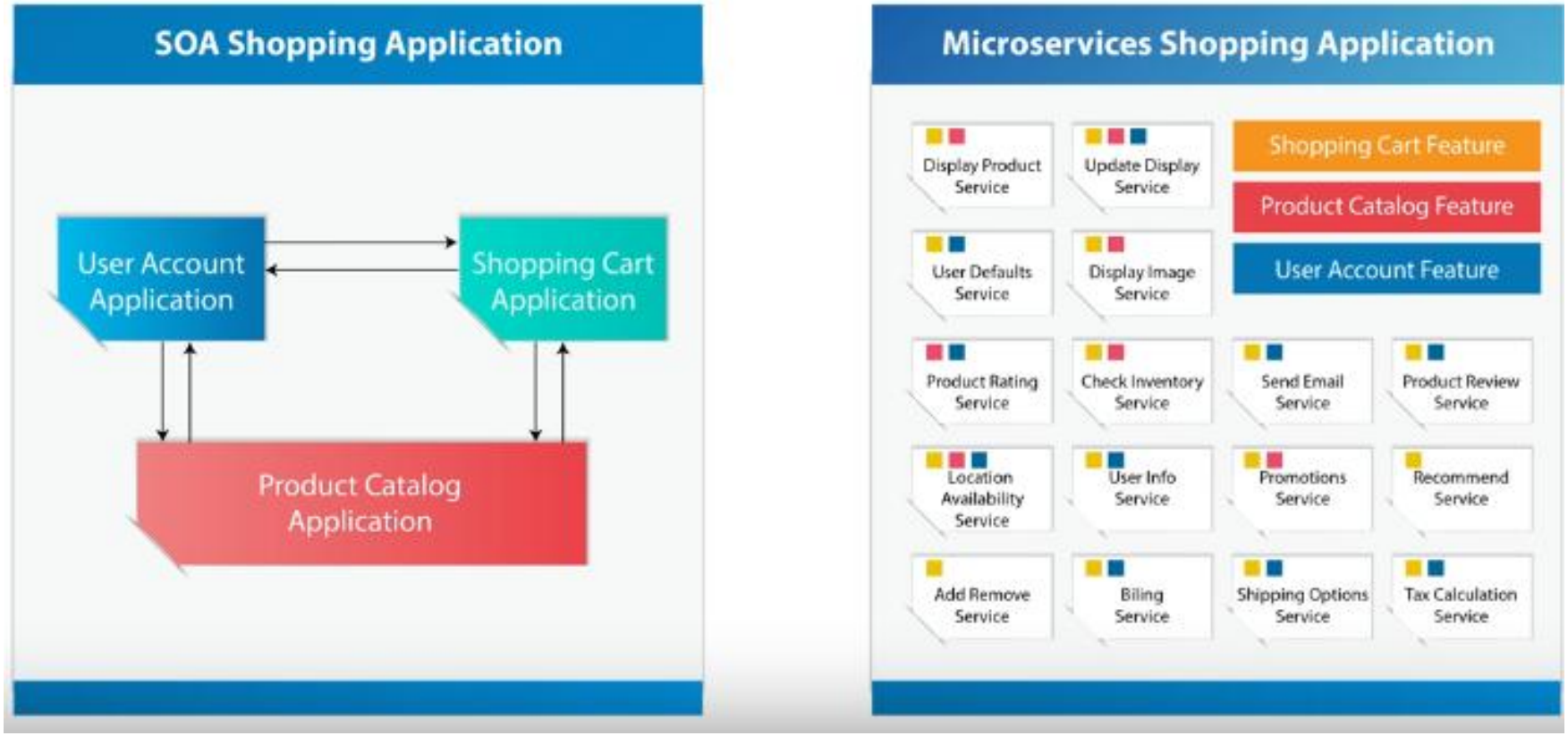


Microservices



SOA

Shopping Cart App - SOA vs Microservices



What is Microservice?

Microservice is an architectural style that structures an application as a collection of services that are:

1) Highly maintainable and testable
2) Loosely coupled
3) Independently deployable
4) Organized around business capabilities
5) Owned by a small team

The microservice architecture enables the rapid, frequent and reliable delivery of large, complex applications.

Advantages of Microservices

- **Simple To Deploy -** Deploy in pieces without affecting other services.
- **Simpler To Understand -** Follow code easier since the function is isolated and less dependent.
- **Reusability Across Business -** Share small services like payment or login systems across the business.
- **Faster Defect Isolation -** When a test fails or service goes down, isolate it quickly with microservices.
- **Minimized Risk Of Change -** Avoid locking in technologies or languages - change on the fly without risk.

Disadvantages of Microservice

- **Refactoring –** If proper relationship among components have been wrongly implemented initially then it is very difficult to refactor an existing microservice based application. Because there are several calls made among different microservices.
- **Complexity -** Splitting an application into multiple independent services generates more artifacts to manage with potentially diverse deployment processes. This can increase the complexity of deploying the entire application at once.

Disadvantages of Microservice

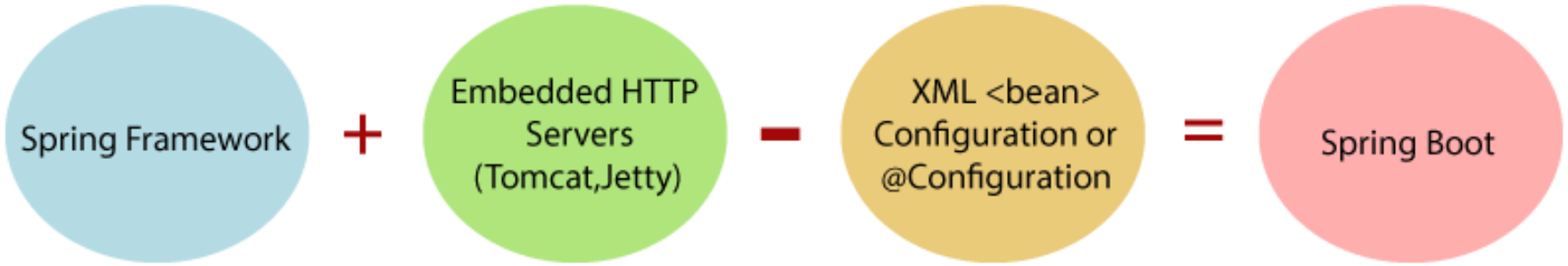
- **Testing -** Spinning up test environments is more involved with microservices due to the increased number of nodes required.
- **Performance -** Communication over a network is considerably slower than in memory. Microservice architecture needs more network communication than Monolithic architecture & hence it may face performance challenges.

Spring Boot

What is Spring Boot?

1) Spring Boot provides an easier and faster way to set up, configure, and run both simple and web-based applications.
2) Spring Boot is a Spring module that provides the RAD (Rapid Application Development) feature to the Spring framework.
3) Spring Boot is the combination of Spring Framework and Embedded Servers.
4) Spring Boot uses convention over configuration that means it decreases the effort of the developer. Hence, there is no need of writing XML deployment descriptor.

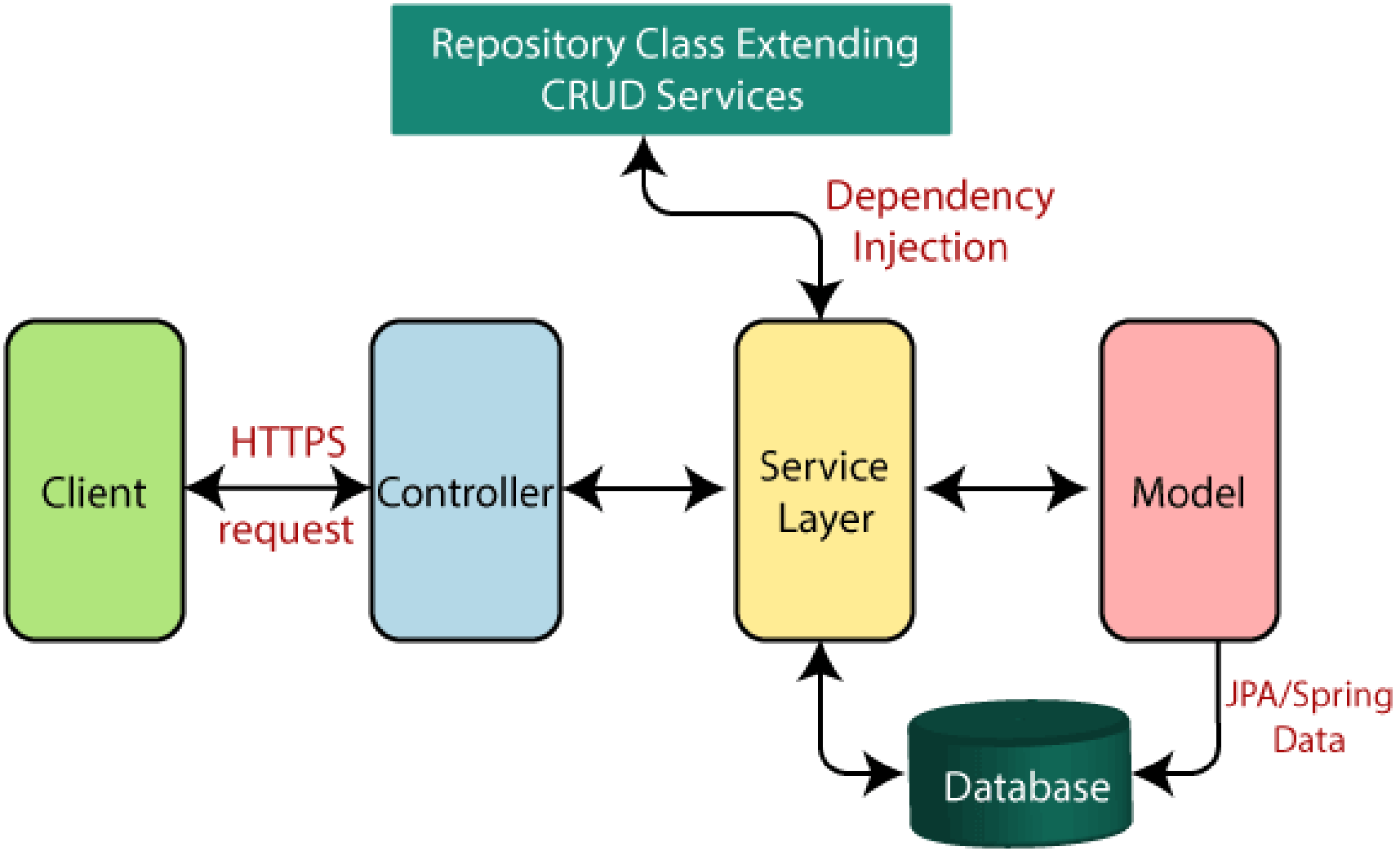
What is Spring Boot?



Advantages of Spring Boot

1) It creates stand-alone Spring applications that can be started using Java -jar.
2) It tests web applications easily with the help of different Embedded HTTP servers such as Tomcat, Jetty, etc. We don't need to deploy WAR files.
3) There is no requirement for XML configuration.
4) It offers the number of plug-ins.
5) It increases productivity and reduces development time.

Spring Boot flow architecture

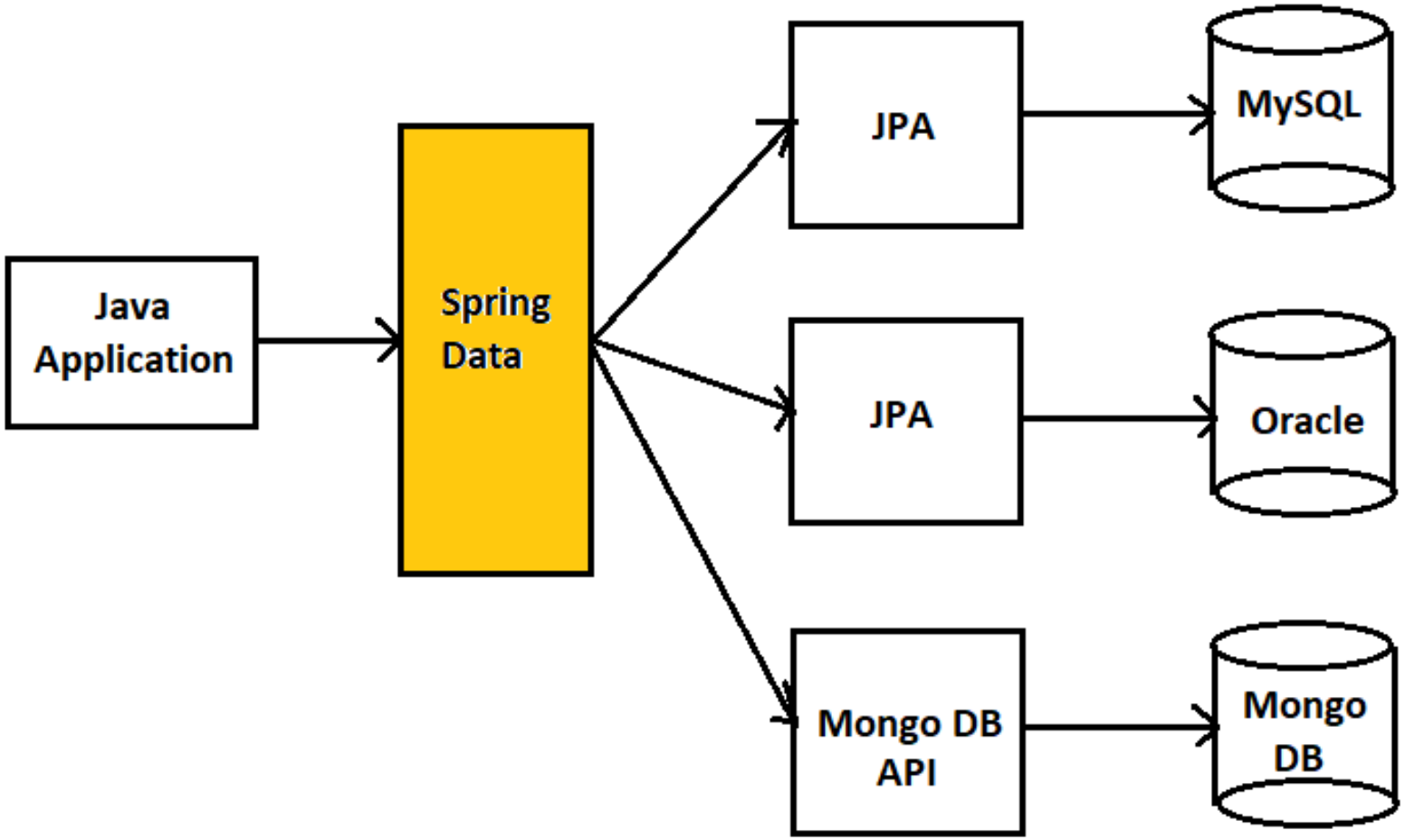


Spring Boot vs Spring MVC

Spring MVC	Spring Boot
Spring MVC is a model view controller-based web framework under the Spring framework.	**Spring Boot** is a module of Spring for packaging the Spring-based application with sensible defaults.
It provides **ready to use** features for building a web application.	It provides default configurations to build **Spring-powered** framework.
It requires build configuration manually.	There is no need to build configuration manually.
A Deployment descriptor is **required**.	There is **no requirement** for a deployment descriptor.
It specifies each dependency separately.	It avoids boilerplate code and wraps dependencies together in a single unit.
It takes **more** time to achieve the same.	It **reduces** development time and increases productivity.

Spring Data

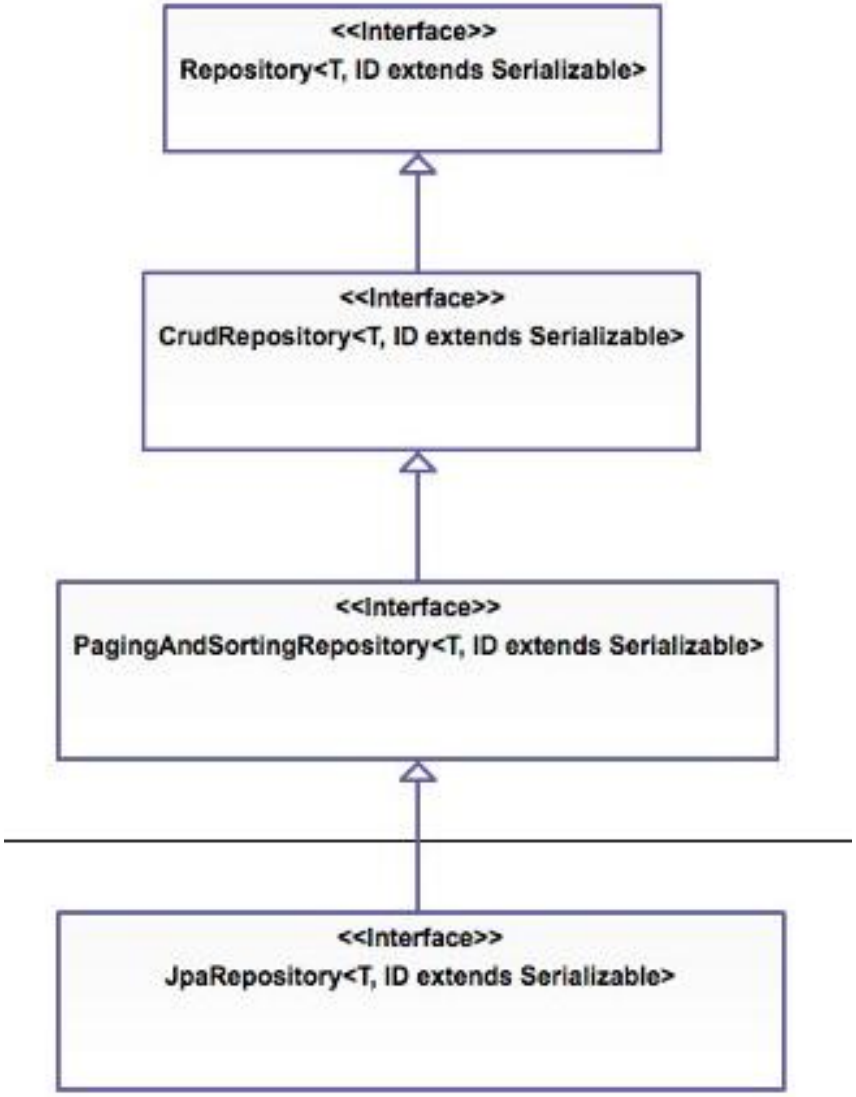
What is Spring Data?



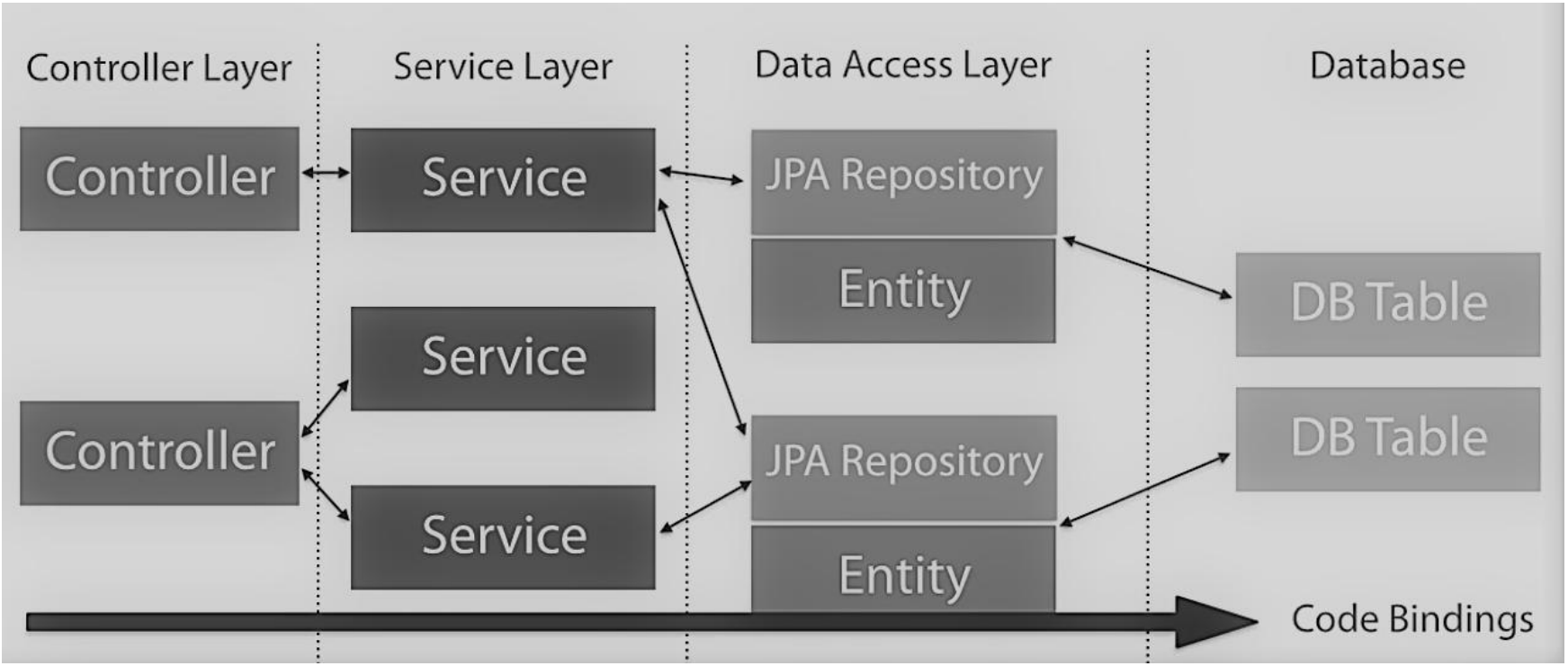
What is Spring Data?

1) Spring Data is a part of the Spring Framework.
2) Spring Data provides abstraction layer over SQL as well as NoSQL databases.
3) The goal of Spring Data abstraction is to significantly reduce the amount of boilerplate code required to implement data access layers for various persistence stores.
4) Spring Data provides abstraction is in the form of Repository API.

Spring Data JPA Repository API



Spring Data JPA Application Flow



Thank you!!!