Introduction Spring Framework

What is the Spring Framework?

- Spring is the most popular application development framework for enterprise Java
- Open source since 2003
- Supports all major application servers and Java EE standards
- Handles the infrastructure so developer can focus on the business logic
- Promote good programming practice

Dependency Injection

Introduction to the Concept

- The mechanism that is the heart of the Spring Framework
- Dependency Injection helps to keep classes as independent as possible.
  - Increase reuse by applying low coupling
  - Makes testing easier
  - Makes code better readable

Dependency Injection

Introduction to the Concept

An injection is the passing of a dependency (a service) to a dependent object (a client). Passing the service to the client, rather than allowing a client to build or find the service, is the fundamental requirement of the pattern.

Dependency injection is a pattern where the container passes objects by name to other objects, via either constructors, properties, or factory methods.

Dependency Injection

Relationship Between DI and Inversion of Control

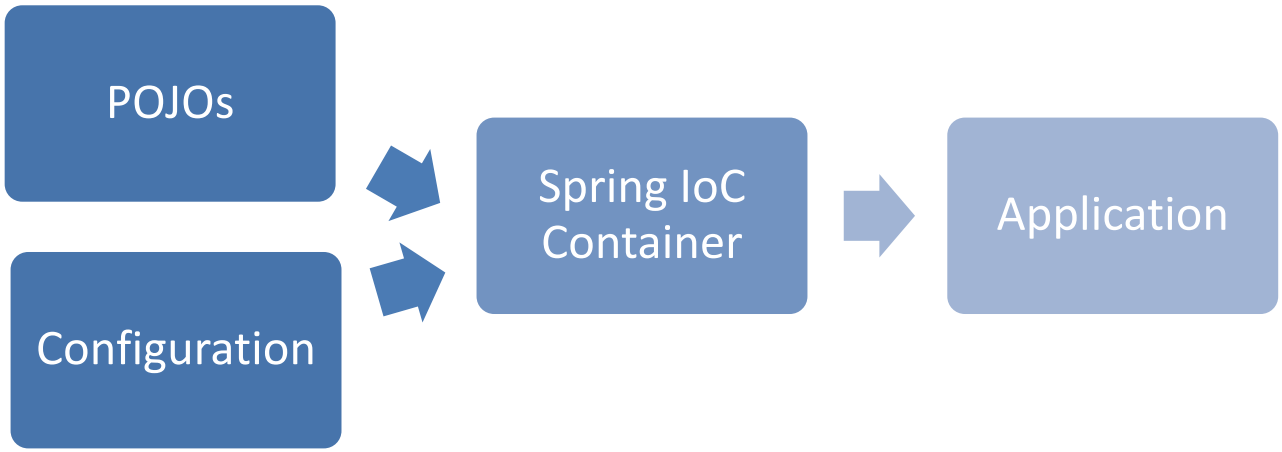
In software engineering, inversion of control (IoC) describes a design in which custom written parts of a computer program receive the flow of control from a generic, reusable library.

Inversion of Control (IoC) is a general concept, and it can be expressed in many different ways. Dependency Injection is merely one concrete example of Inversion of Control.

IoC Container

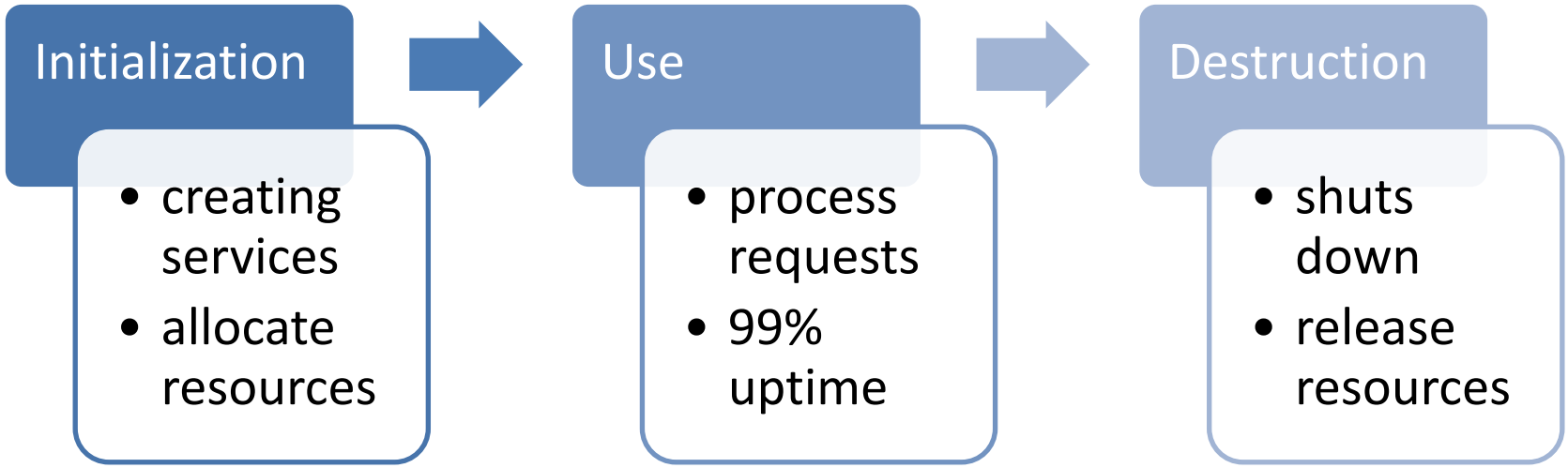
- The Spring container (IoC Container) is at the core of the Spring Framework.

- The container will create the objects, wire them together, configure them, and manage their complete lifecycle from creation till destruction.



Application Lifecycle

Application Lifecycle



Spring Ecosystem

Spring Framework

Core	IoC container, Events, Resources, i18n, Validation, Data Binding, Type Conversion, SpEL, AOP
Testing	Mock objects, TestContext framework, Spring MVC Test, WebTestClient
Data Access	Transactions, DAO support, JDBC,ORM, Marshalling XML
Web Servlet	Spring MVC, WebSocket, SockJS, STOMP messaging
Web Reactive	Spring WebFlux, WebClient, WebSocket
Integration	Remoting, JMS, JCA, JMX, Email, Tasks, Scheduling, Cache
Languages	Kotlin, Groovy, Dynamic languages

Spring Projects

- Spring IO platform
- Spring Boot
- Spring Cloud DataFlow
- Spring Cloud
- Spring Data
- Spring Integration
- Spring Batch
- Spring Security
- Spring Hateoas
- Spring REST Docs
- Spring Social
- Spring AMQP
- Spring Mobile
- Spring Android
- Spring Webflow
- Spring Webservices
- Spring LDAP
- Spring Session
- Spring Shell
- Spring Flo
- Spring Kafka
- Spring Statemachine
- Spring Roo
- Spring Scala

Spring Data

- Spring Data Commons
- Spring Data Gemfire
- Spring Data JPA
- Spring Data KeyValue
- Spring Data LDAP
- Spring Data MongoDB
- Spring Data REST
- Spring Data Redis
- Spring Data for Apache Cassandra
- Spring Data for Apache Solr
- Spring Data Aerospike
- Spring Data Couchbase
- Spring Data DynamoDB
- Spring Data Elasticsearch
- Spring Data Hazelcast
- Spring Data Jest
- Spring Data Neo4j
- Spring Data Vault

Spring Cloud

- Spring Cloud Config
- Spring Cloud Netflix
- Spring Cloud Bus
- Spring Cloud for Cloud Foundry
- Spring Cloud Cluster
- Spring Cloud Consul
- Spring Cloud Security
- Spring Cloud Sleuth
- Spring Cloud Data Flow
- Spring Cloud Stream
- Spring Cloud App Starters
- Spring Cloud Task
- Spring Cloud Zookeeper
- Spring Cloud Amazon Web Services
- Spring Cloud Connectors
- …

Advantages of Using Spring Framework

Advantages of Using Spring Framework

- Open source
- Lightweight and fast
- Modular structure
- Non-invasive (POJO based)
- Low coupling because of Dependency Injection
- Reuseable software
- AOP support
- Stable and lots of resources
- Enormous community & ecosystem
- Extensible for other frameworks