Spring Framework Servlet-API

MVC Architecture : Manual

Architecture is implemented strictly, disciplined way

remove lot of Boiler-plate code

abstract the low level complexity

Focus more on business logic

Most popular frameworks to develop java application

J2EE : Java 2 Enterprise Edition : Framework to develop web app using java

Complex in nature

lots of deployment descriptor

lots of interface, abstract classes needs to be created to expose a single service

productivity reduces, reduces efficiency

Rod Johnson

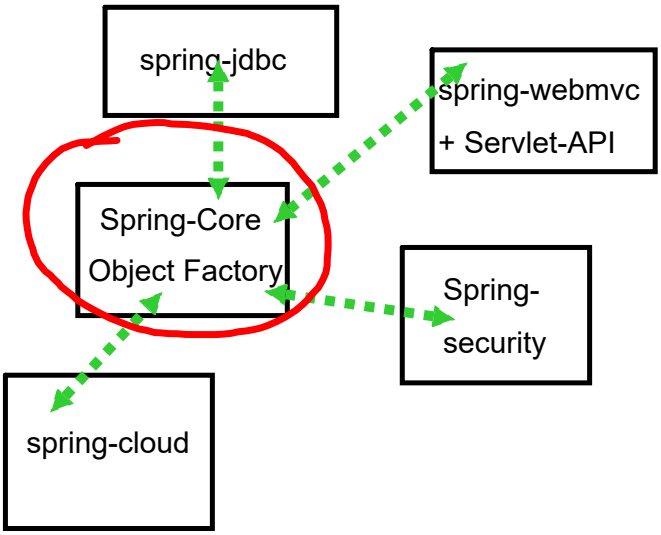
=> Object Factory : Responsible for creating and managing object

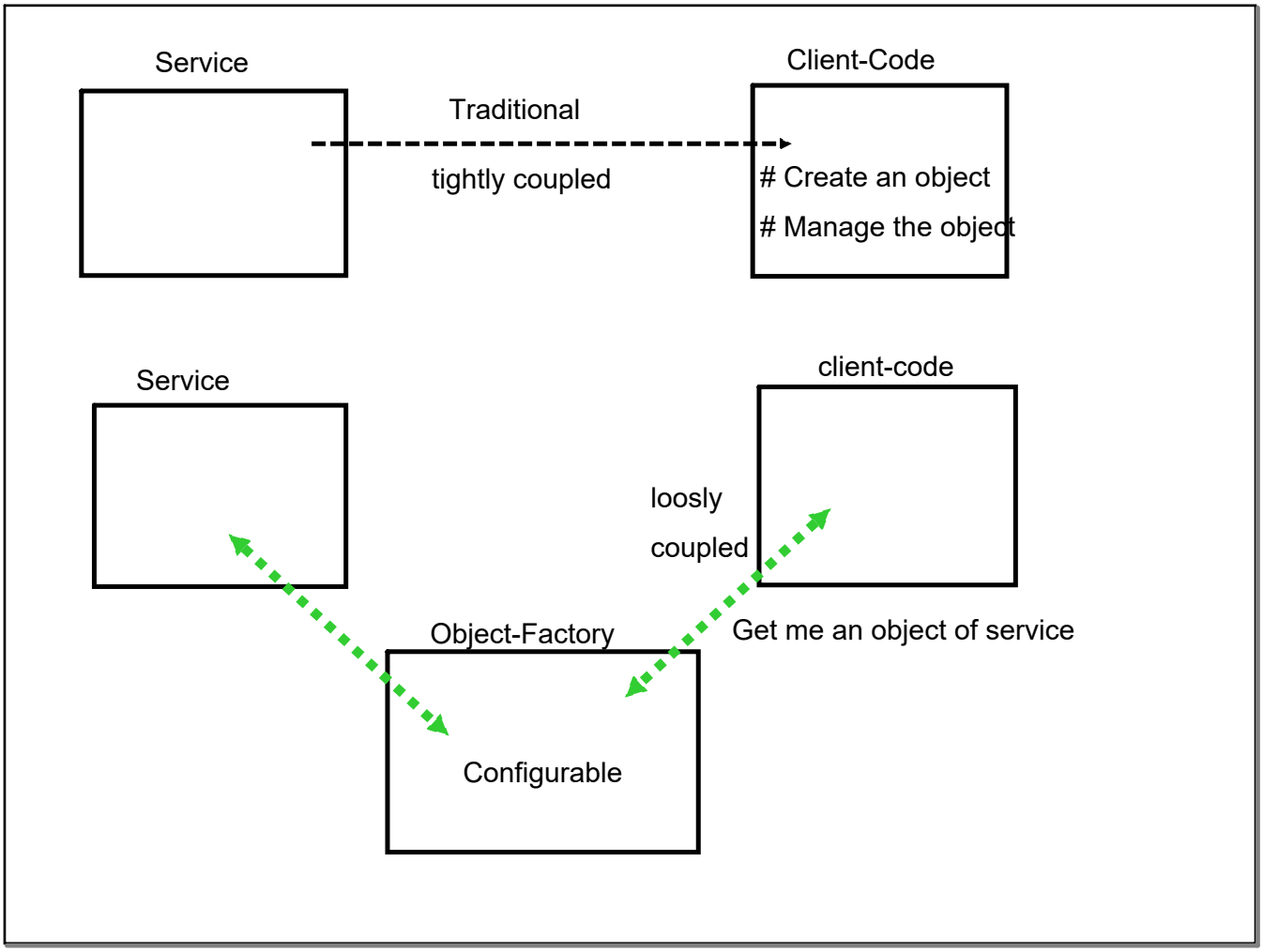
Increased the Productivity

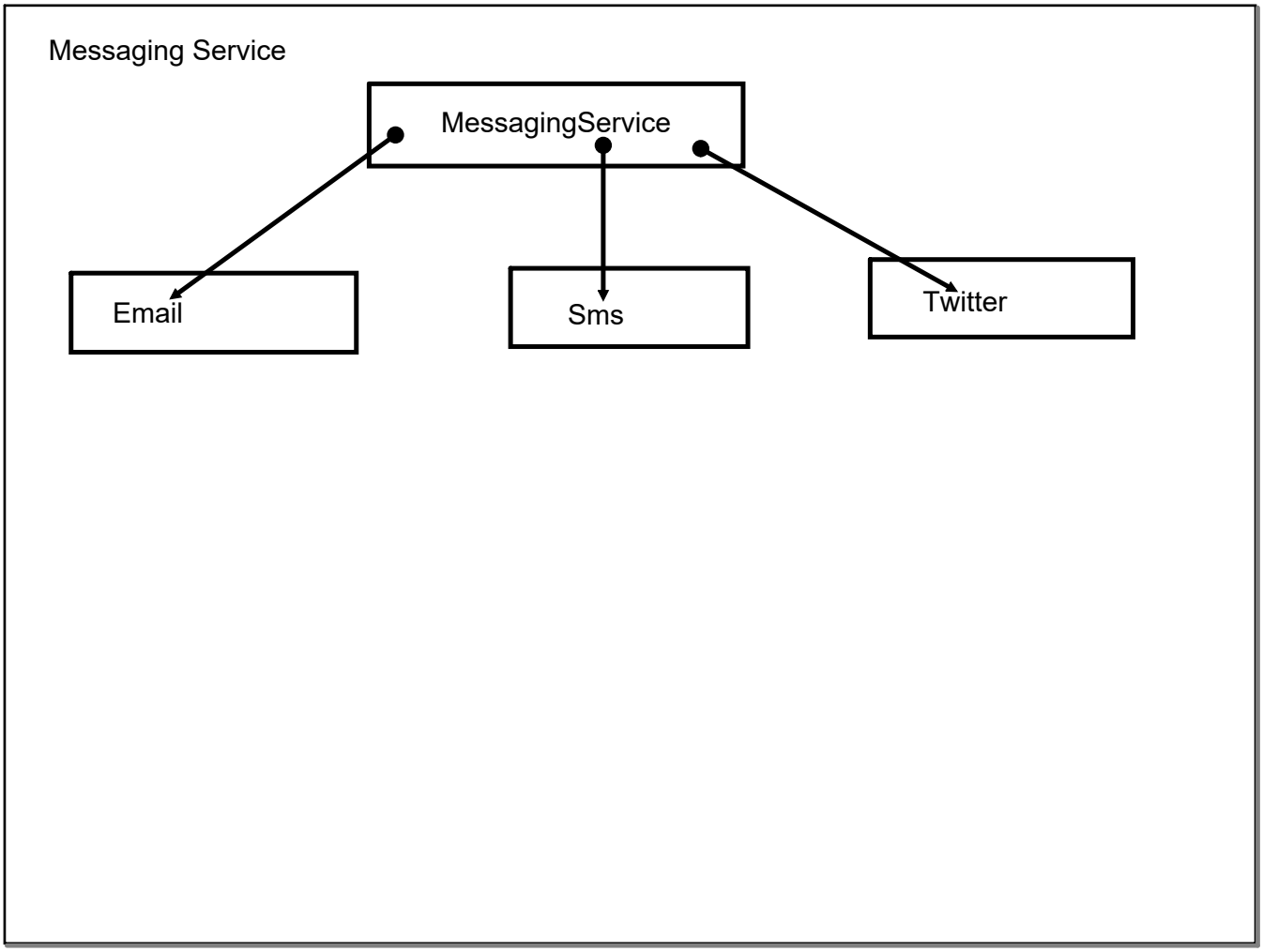
Increased the efficiency

1. Object Factory
2. Highly Modular
3. POJOs

Spring Framework




Service
Client-Code
Traditional
tightly coupled
Create an object
Manage the object
Service
client-code
loosly
coupled
Object-Factory
Get me an object of service
Configurable


Messaging Service
MessagingService
Email
Sms
Twitter

Object Factory | Bean Factory | Application Context

Provided by Spring - Core Module

A Custom Configuration needs to be provided to define the behavior of Object Factory

XML Based Configuration (Legacy)
Annotation Based Configuration (Modern)
Pure Java Based Configuration (Modern)

Std Spring Framework :

bundle of few Modules

=> Core

=> Spring-web-mvc

=> Spring AOP (proxy)

Bean Factory works on two key principals

1. IoC : Inversion of Control
2. DI : Dependency Injection

IoC : Outsourcing the (control of) creation and management of Object

XML Based Config :

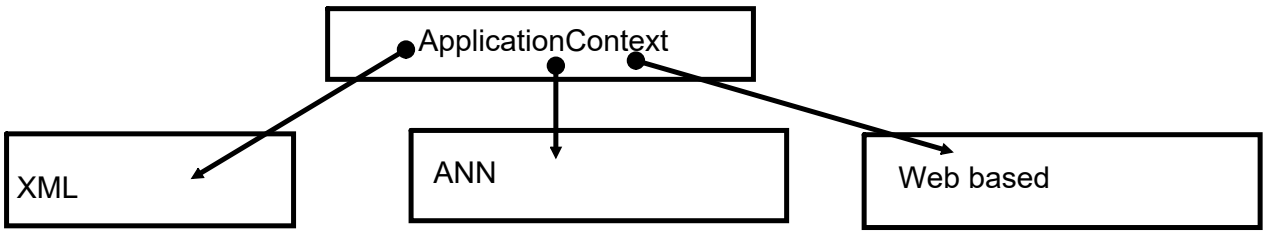
XML file + certain dependencies for support of additional spring tags

BEAN : Container(Object Factory) managed Object

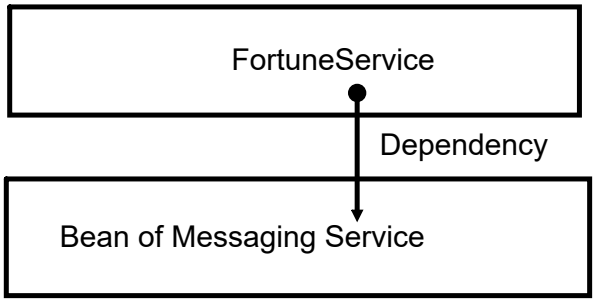
Multiple classes provided for Bean Factory

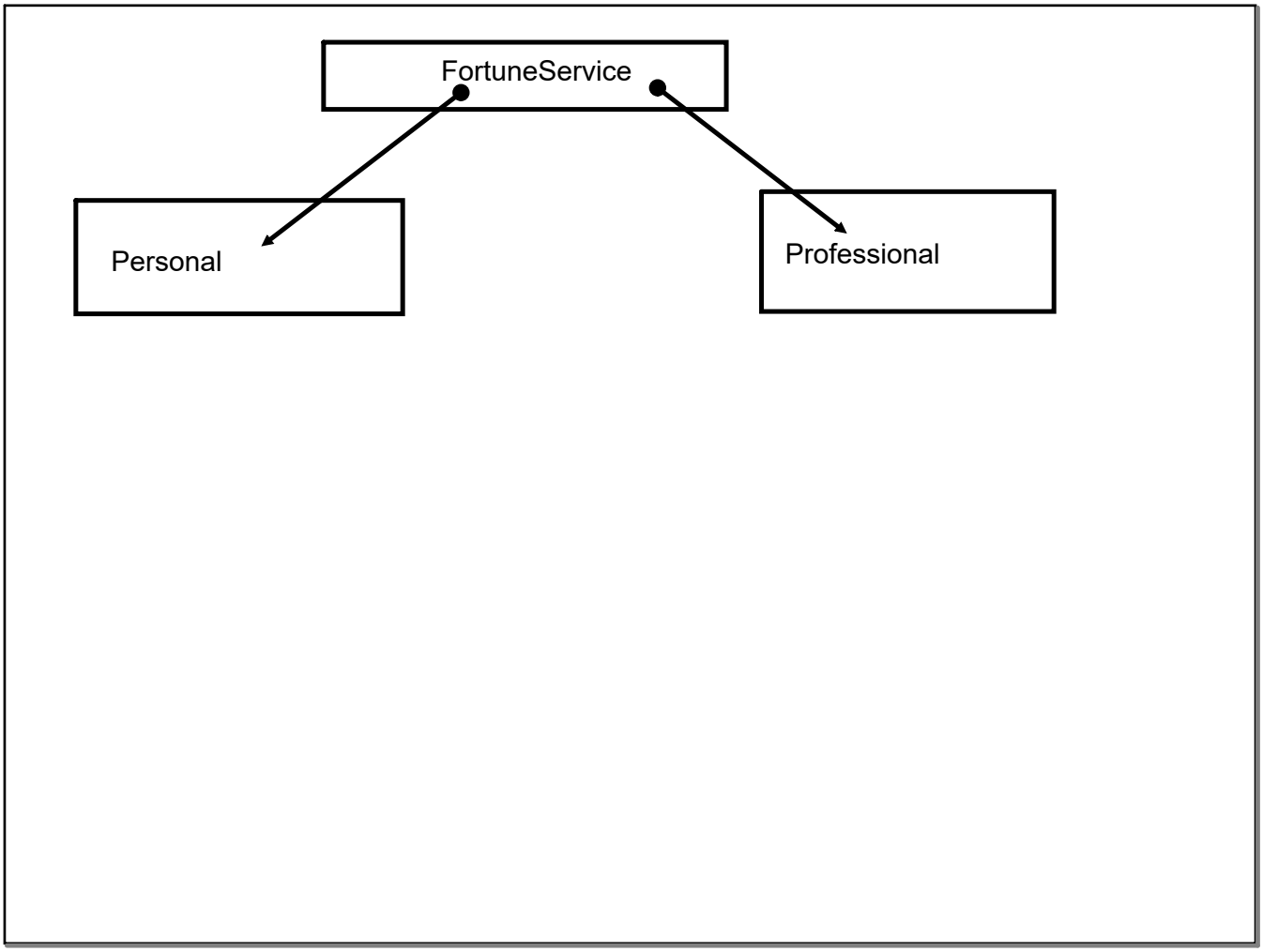
way of config (XML or java)

env for which bean factory (simple java, web app)



Each bean must have a unique id




FortuneService
Personal
Professional