

JCF 정복하기

2011. 09.04

기술개발팀

김민아 대리

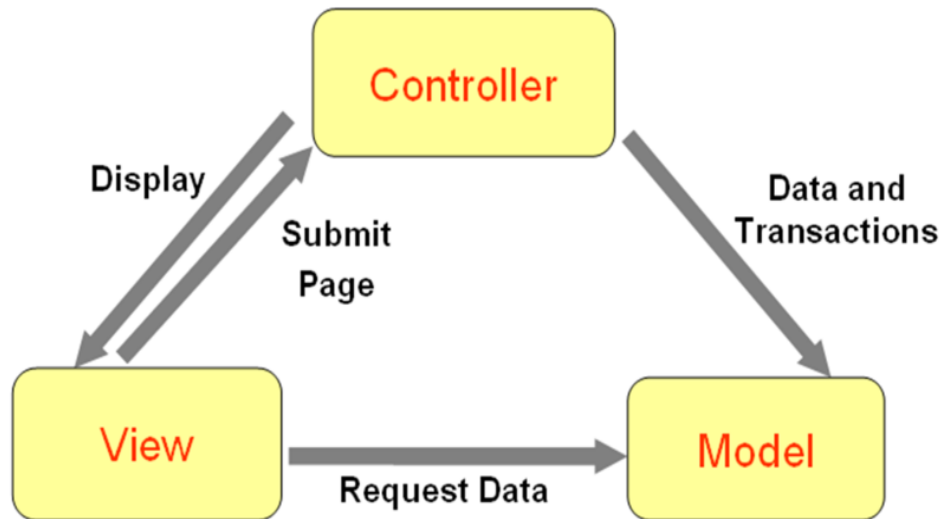
목차

1. Spring MVC
2. JCF SUA
3. Spring IOC
4. Spring AOP
5. JCF Query 는?

#1. Spring MVC란?

MVC 패턴이란?

- MVC(Model-View-Controller) 패턴은 코드를 기능(역할)에 따라 Model, View, Controller 3가지 요소로 분리한다.

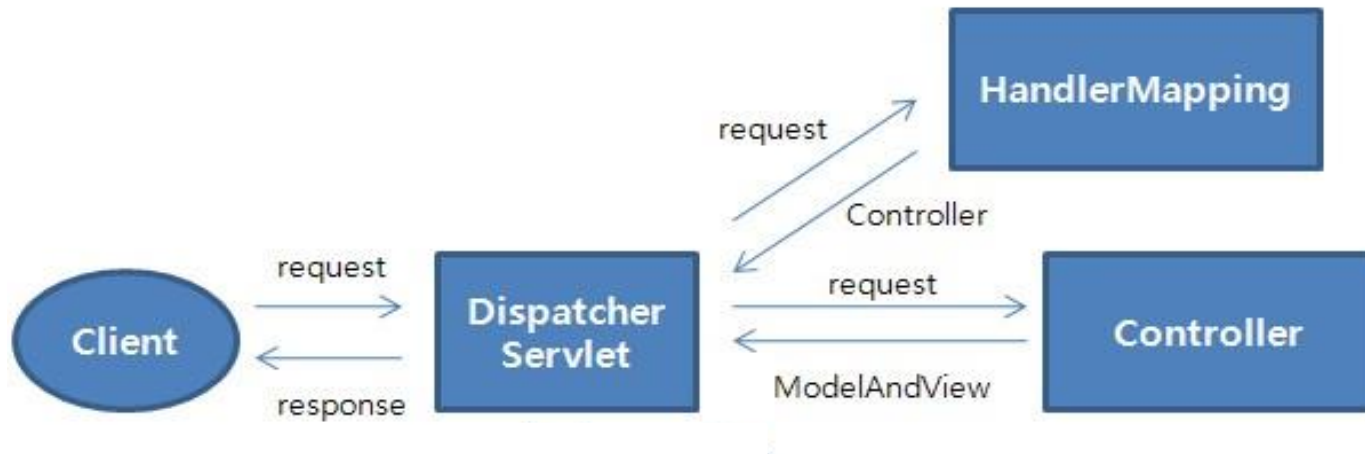


- Model : 어플리케이션의 데이터
- View : Model의 정보를 사용자에게 표시한다. 하나의 Model을 다양한 View에서 사용
- Controller : 사용자의 요청을 받아 Model에 변경된 상태를 반영하고, 다시 View에 넘기는 역할을 한다.

MVC 패턴은 UI 코드와 비즈니스 코드를 분리함으로써 종속성을 줄이고, 재사용성을 높이고, 보다 쉬운 변경이 가능

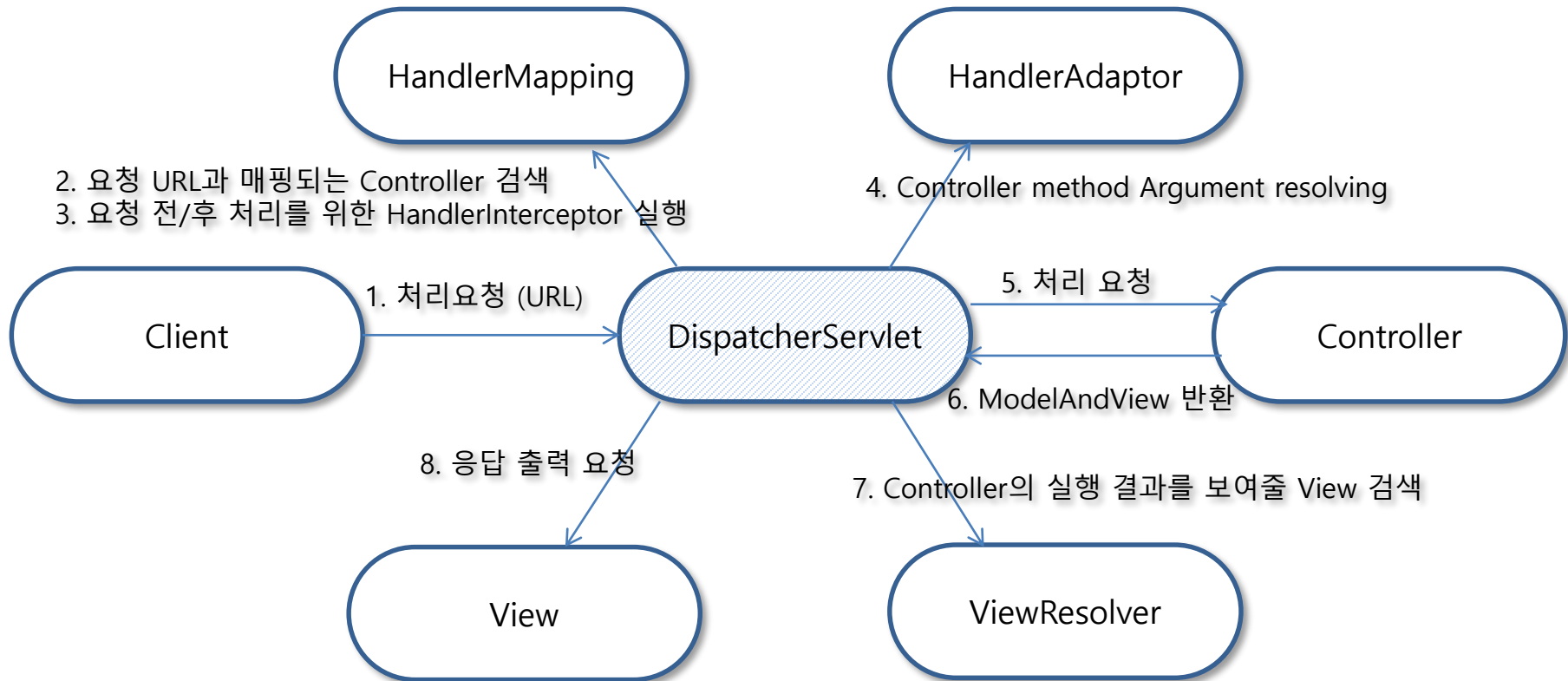
Spring MVC 구조

component	개요
DispatcherServlet	spring MVC Framework의 Front Controller, 웹요청과 응답의 Life Cycle을 주관한다.
HandlerMapping	웹요청시 해당 URL을 어떤 Controller가 처리할지 결정한다.
Controller	비지니스 로직을 수행하고 결과 데이터를 ModelAndView에 반영한다.



1. Client의 요청이 들어오면 DispatcherServlet이 가장 먼저 요청을 받는다.
2. HandlerMapping이 요청에 해당하는 Controller를 return 한다.
3. Controller는 비즈니스 로직을 수행(호출)하고 결과 데이터를 Client 에 다시 보낸다.

Spring MVC 흐름

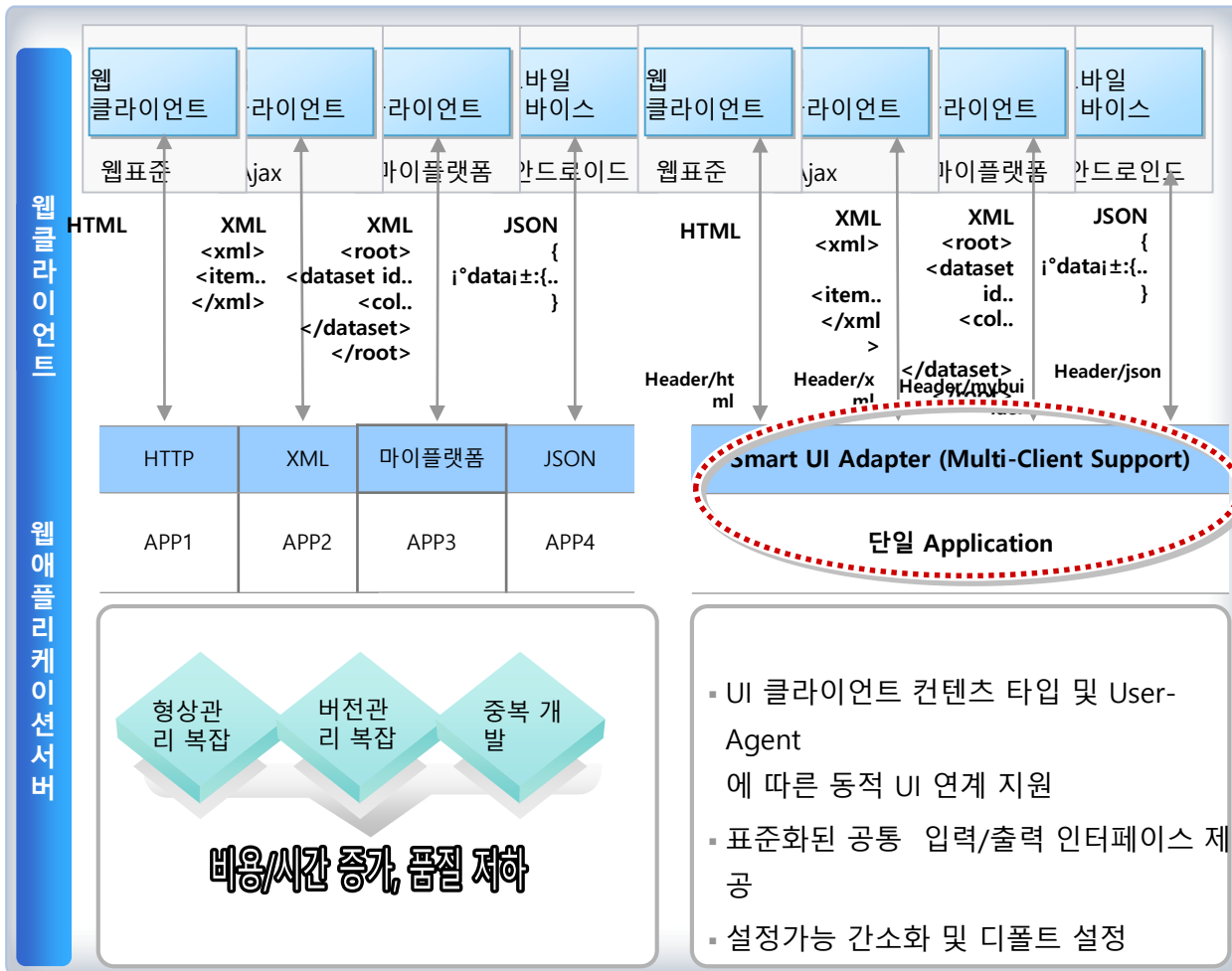


#2. JCF SUA란?

구현목표

- 웹 요청과 MCI 처리 흐름의 일원화
- 일관된 채널 식별 방안 제시
- UI 기술과 MCI 로직의 분리
- 채널설정 간소화

JCF SUA란?



적용효과

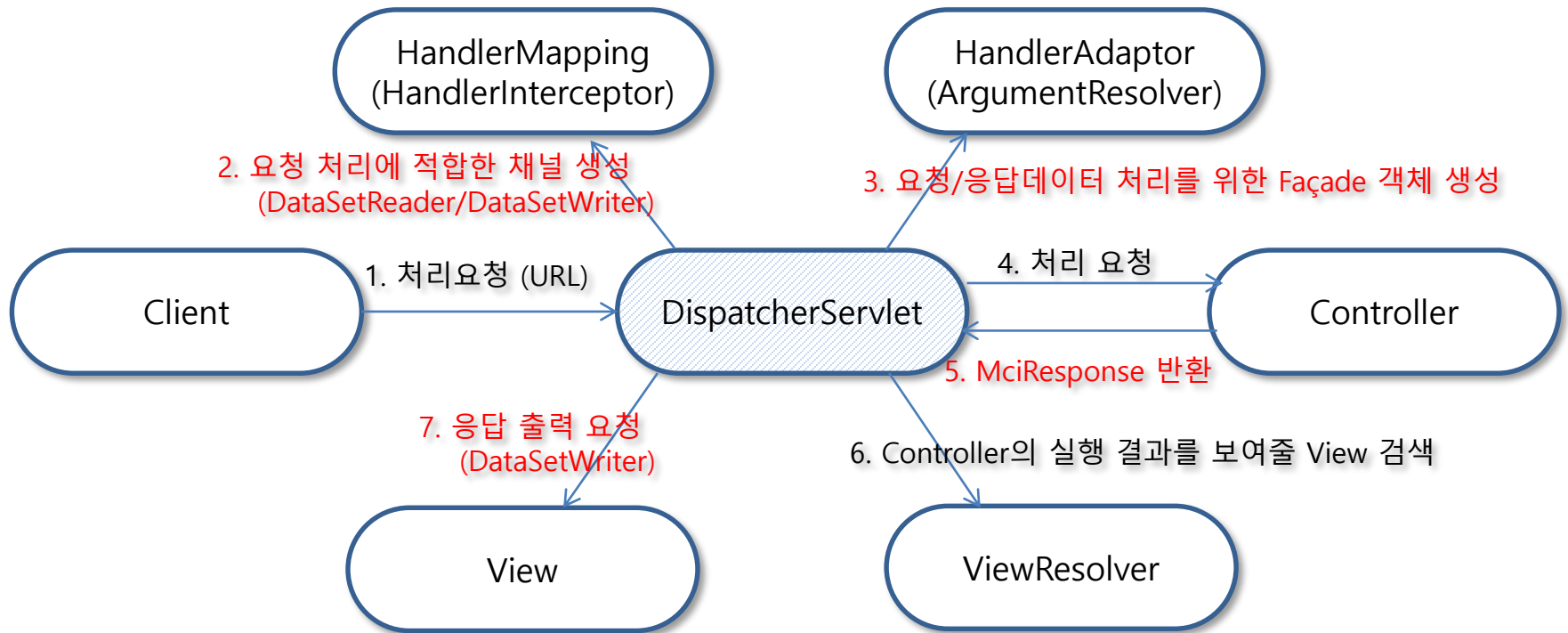
1 개발효과

- UI 기술에 독립적인 표준 인터페이스 사용을 통한 단순하고 표준화된 개발이 가능

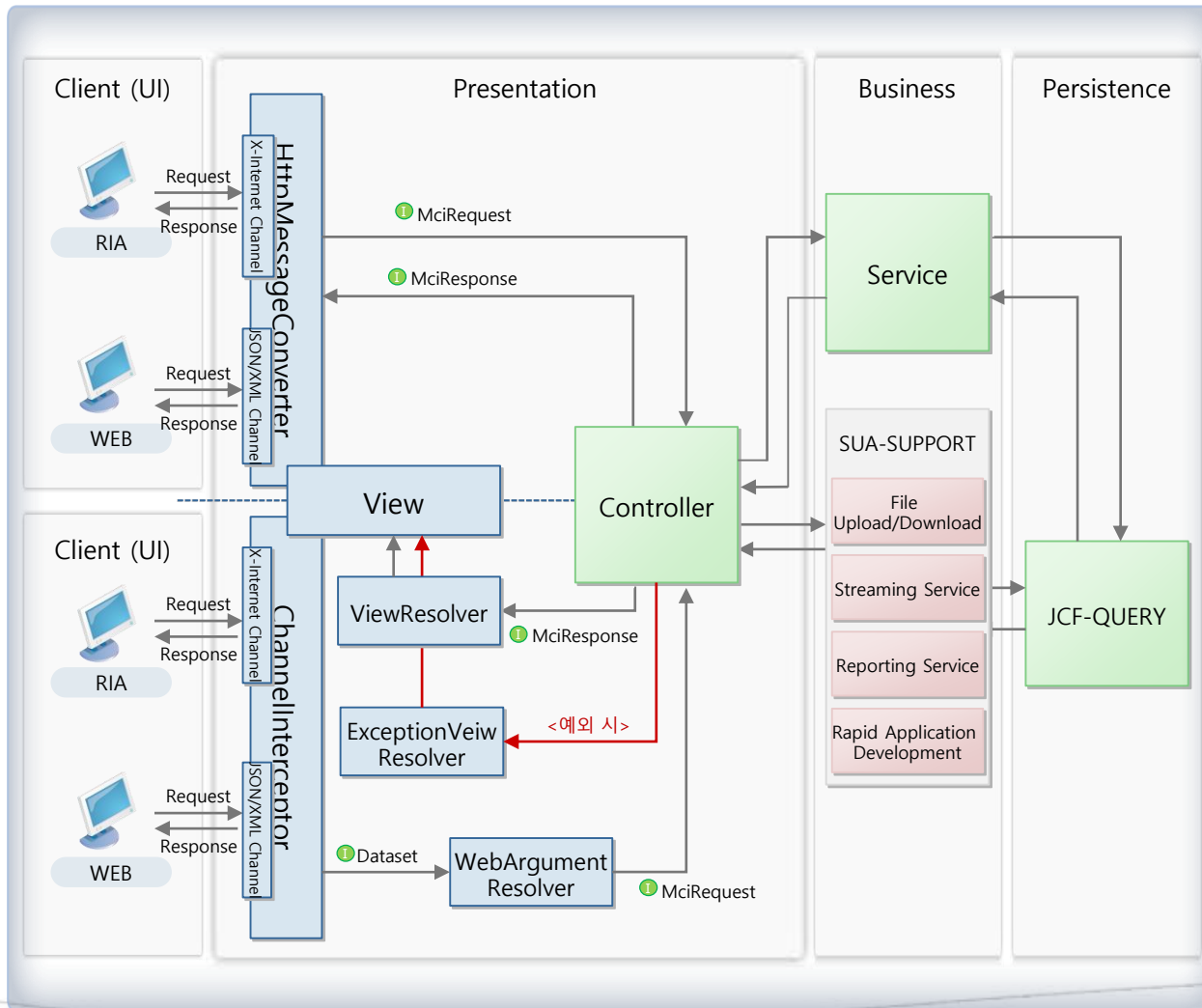
2 운영효과

- UI 변경에 따른 서버 소스 변경이 없으므로 향후 UI 확장 및 변경에 즉시 대응이 가능

Spring MVC 기반의 흐름



JCF SUA 구조



특장점

1 다양한 UI 동시 연계

- RIA, 웹표준, X인터넷 등 다양한 UI에 동시 연계가 가능

2 UI 독립적 개발

- UI 기술에 관계없이 단일모듈로 표준웹, 모바일 등 개발 가능

3 공통기능

- 표준화된 파일처리(Upload/Download) 서비스 제공
- 안정적 대용량 데이터 처리를 위한 데이터 분할 전송 기능 제공
- 조회된 데이터를 Excel 파일로 전송
- COC기반의 패턴 설정을 이용한 개발 간소화 기능 제공

JCF SUA 맛보기 !!

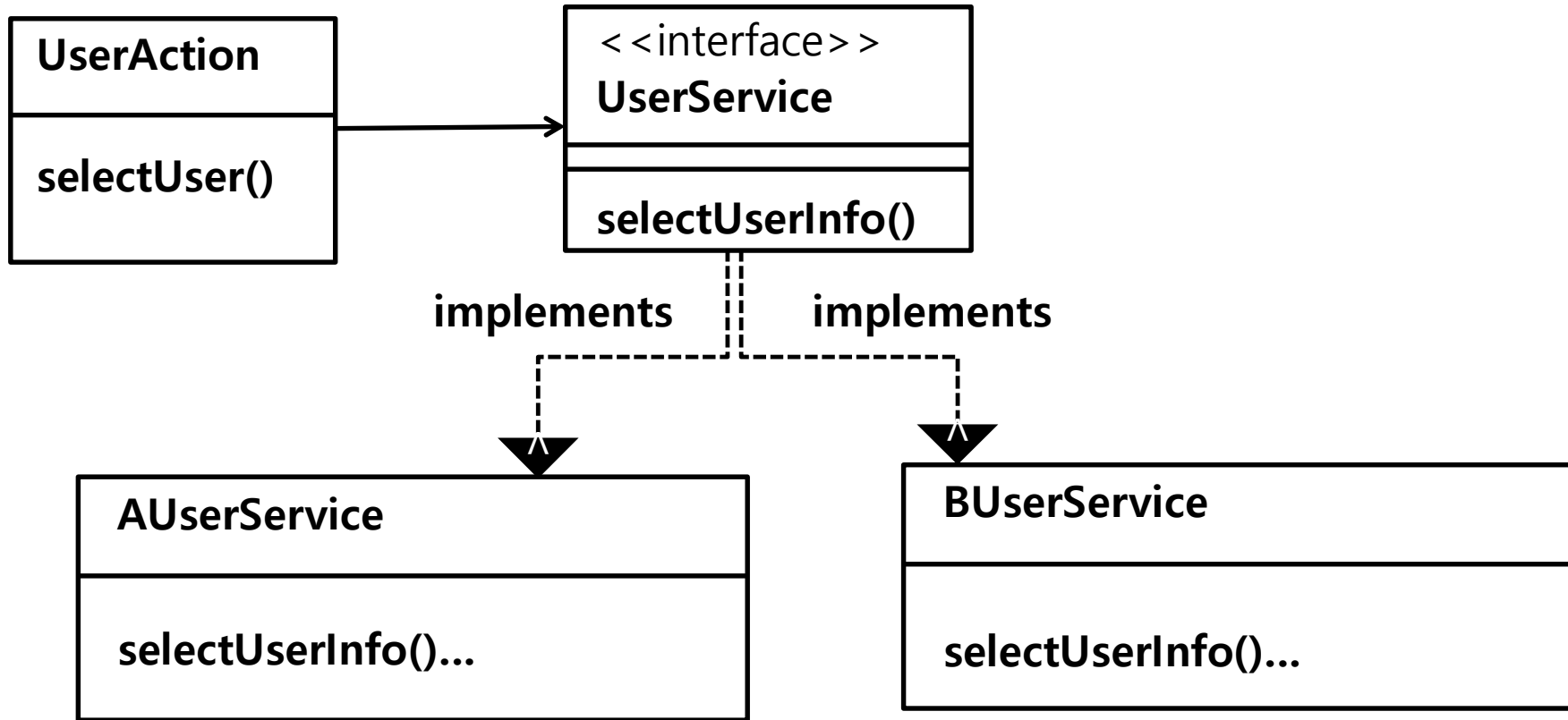
#3. Spring IOC (Inversion of Control)

Spring IOC

가정해보자 !!

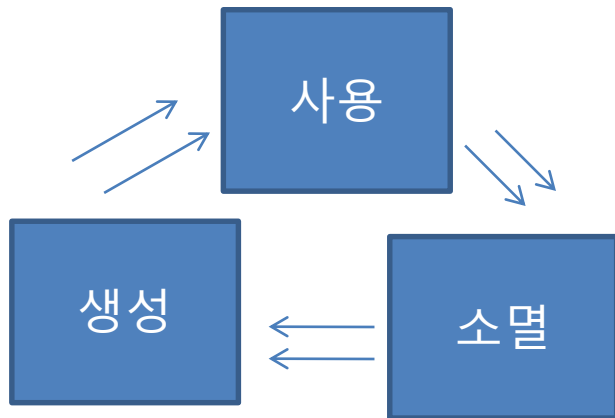
1. 우리가 만든 어플리케이션을 솔루션으로 판다면..
2. UserService에서 사용자 정보를 가져오는 로직이 회사마다 틀리다면.
3. Java 파일에 if문을 넣을 것인가 Java 파일은 건드리지 말고 방법이 없을까?

Spring IOC



Spring IOC – 객체의 생성과 소멸

《일반 객체》



호출 시 생성, 사용, 소멸 반복

《Spring bean》



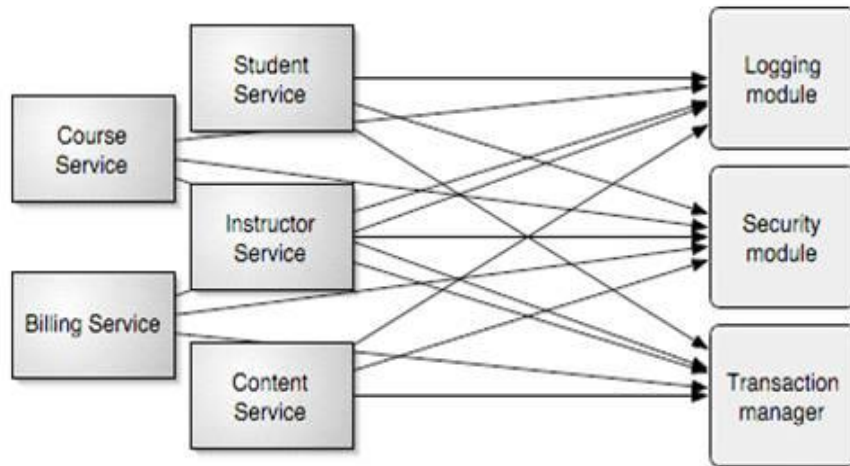
사용을 위해 대기 중

#4. Spring AOP

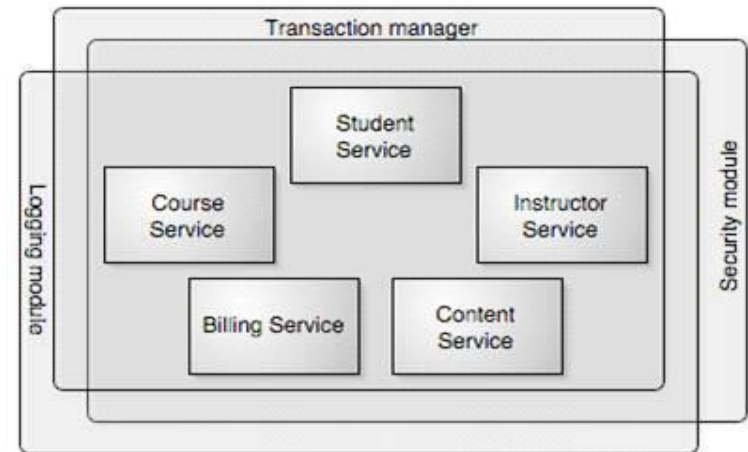
(Aspect Oriented Programming)

Spring AOP

- 객체 지향 개발을 보완
- 다양한 객체들에 분산되어 중복적으로 존재하는 공통 관심 사항(Concern)들을 묶어서 모듈화하여 Aspect들을 분리

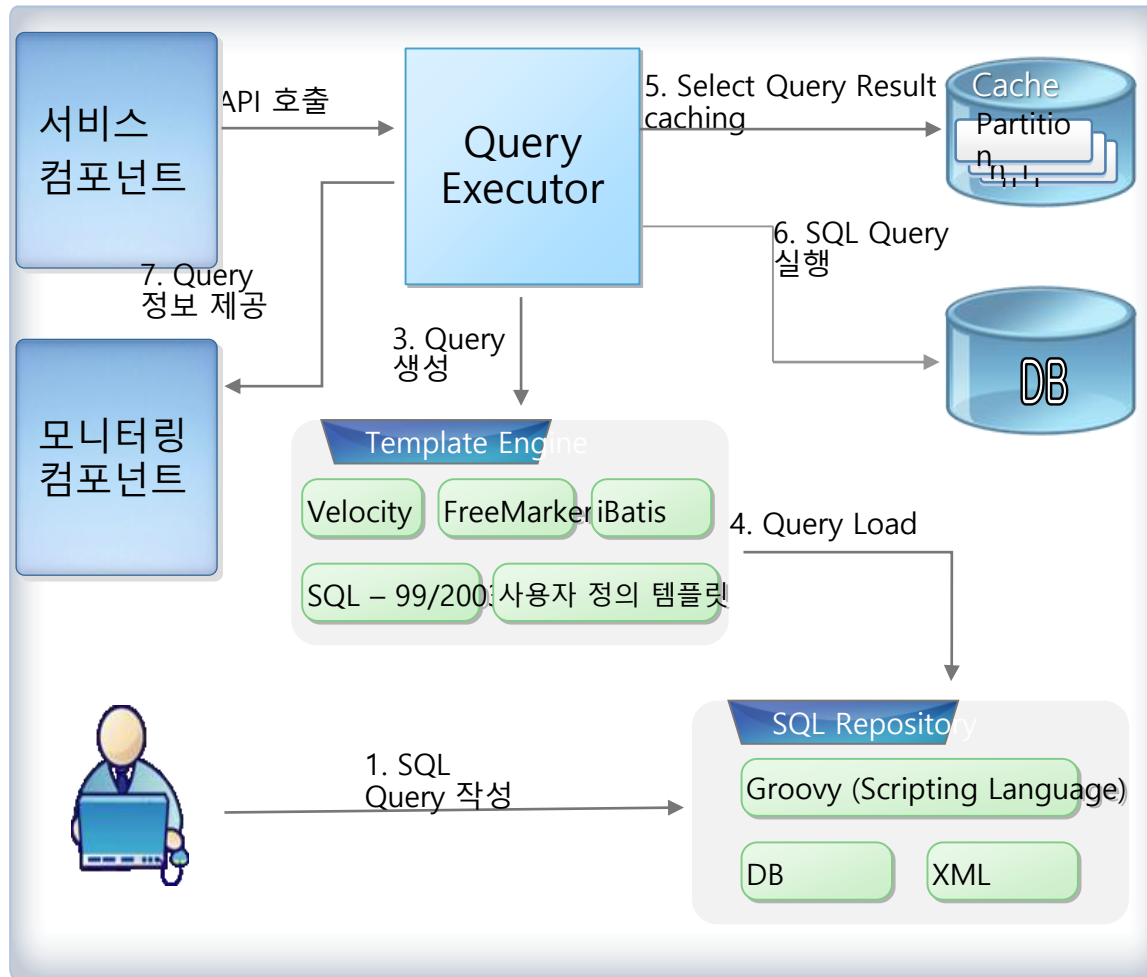


VS



#5. JCF Query

JCF-Query



특장점

1 기술 종속성 제거

- iBATIS, Spring JDBC 등 다양한 Data 처리 계층 지원

2 개발 및 유지보수 편리

- 쿼리 에러로 인한 로딩 실패 보완
- 쿼리 에디터 스위칭 지원

3 유연한 쿼리 작성

- 다양한 작성방법 지원 (Groovy 등 스크립팅 언어, DB 저장, XML 등)
- 템플릿 엔진 이용 동적 쿼리 생성 (커스텀 매크로 사용)

JCF Query 상세

