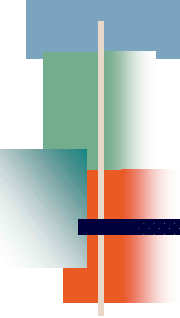


# 웹프로그래밍 개발 기술 발전과 프레임워크



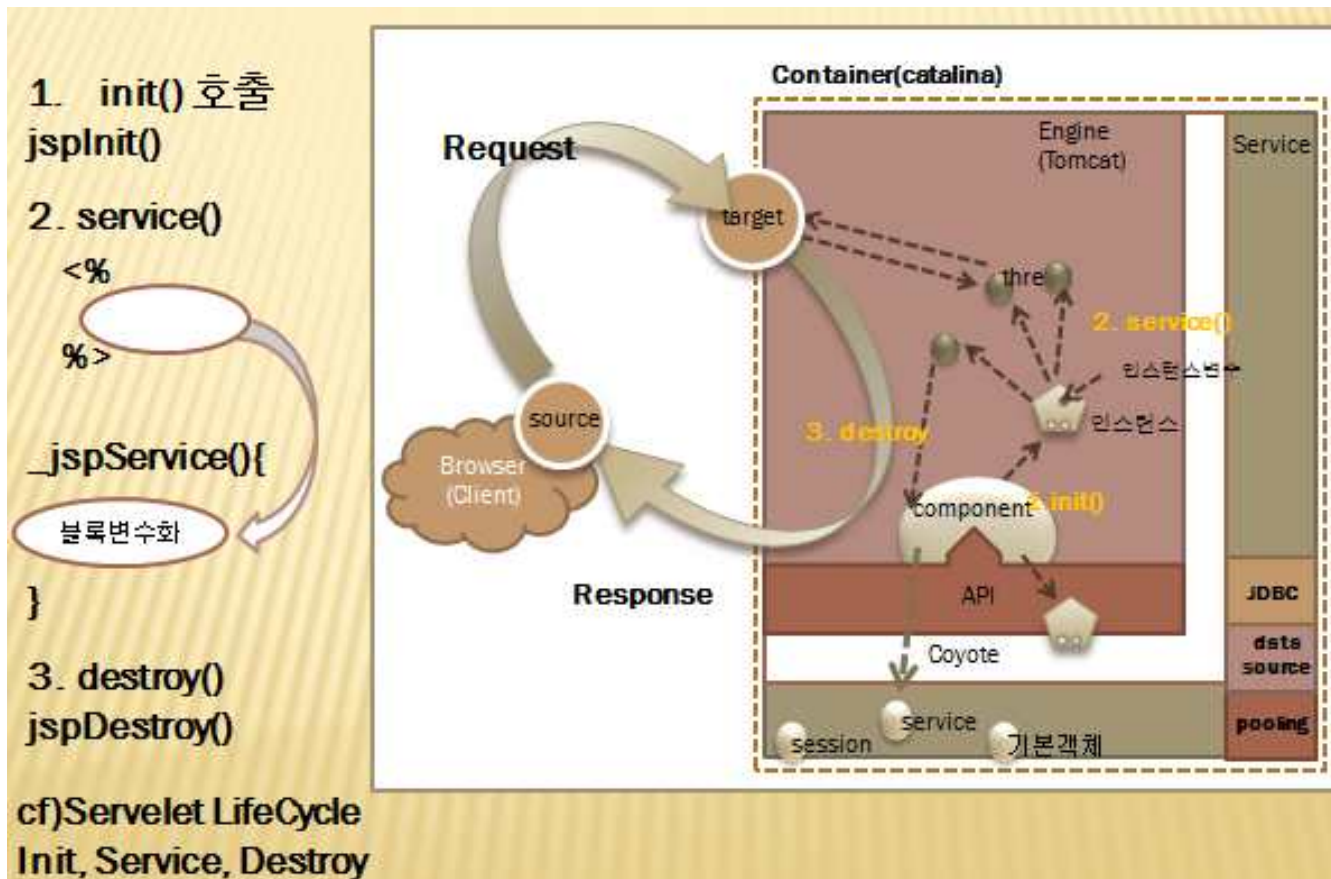
# 목 차

1. 웹 프로그래밍 기본
2. 웹 프로그래밍 기술발전
3. 프레임워크 기본
4. JDBC와 DL 프레임워크
5. View(PL)과 AJAX

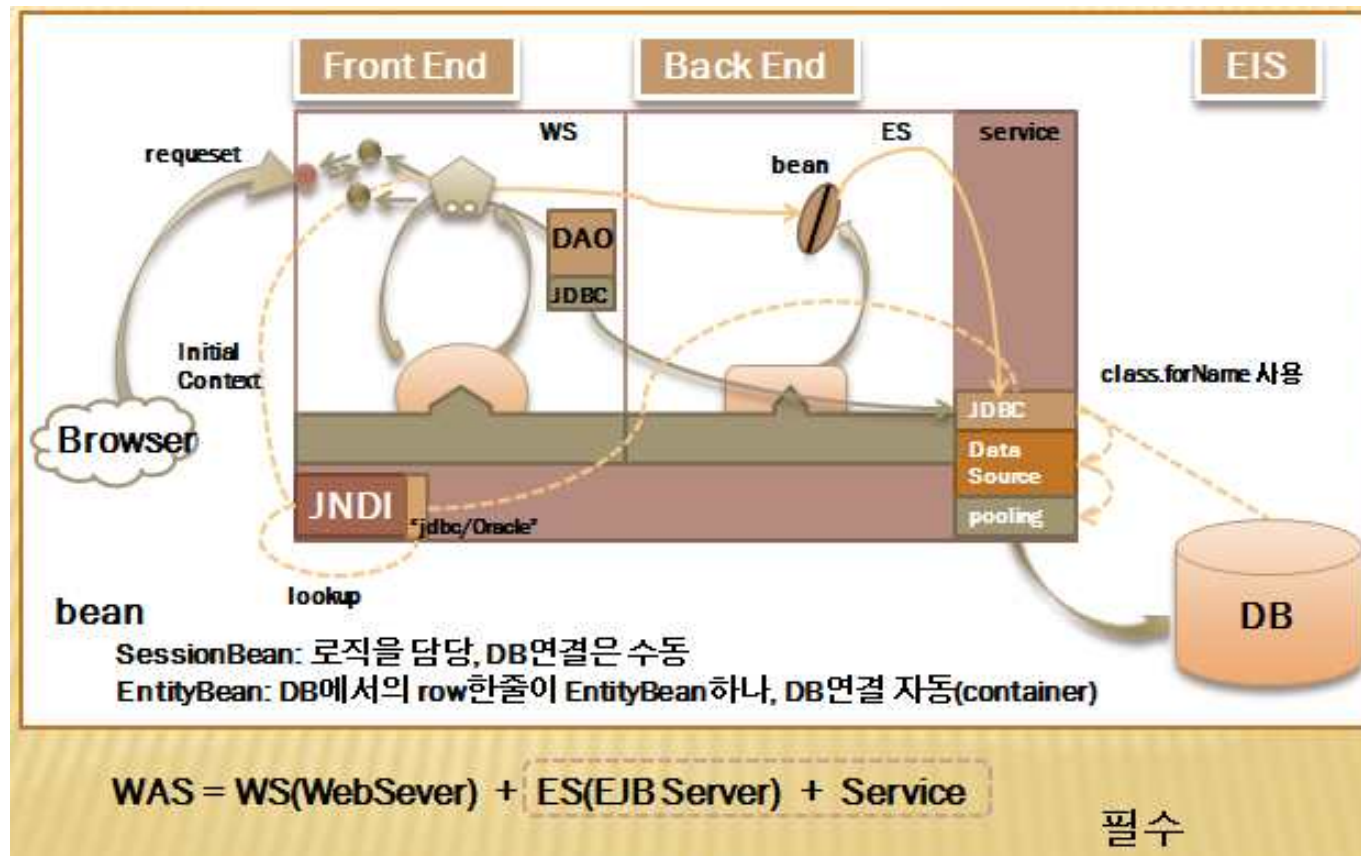


# 1. 웹 프로그래밍 기본

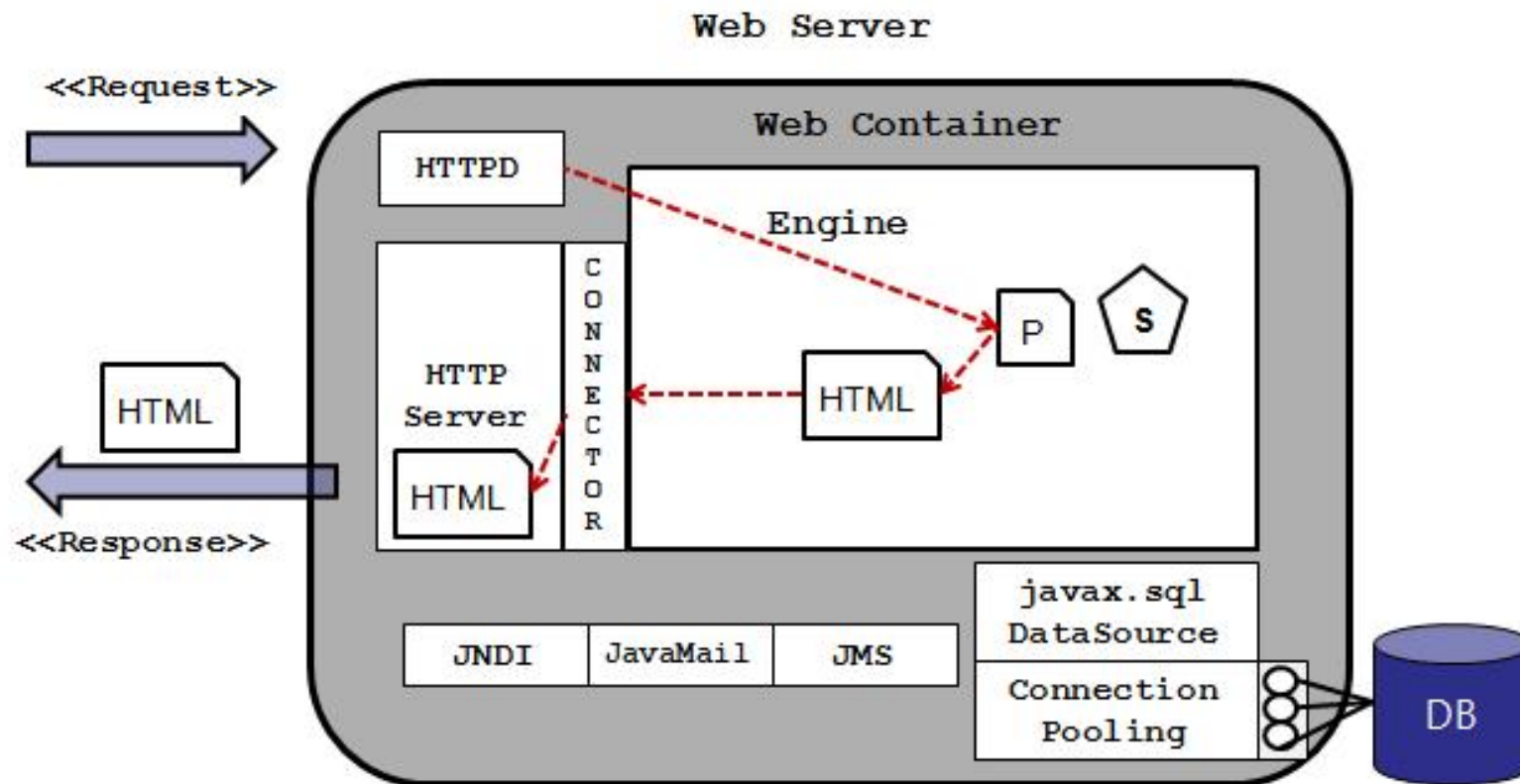
# @Web Server



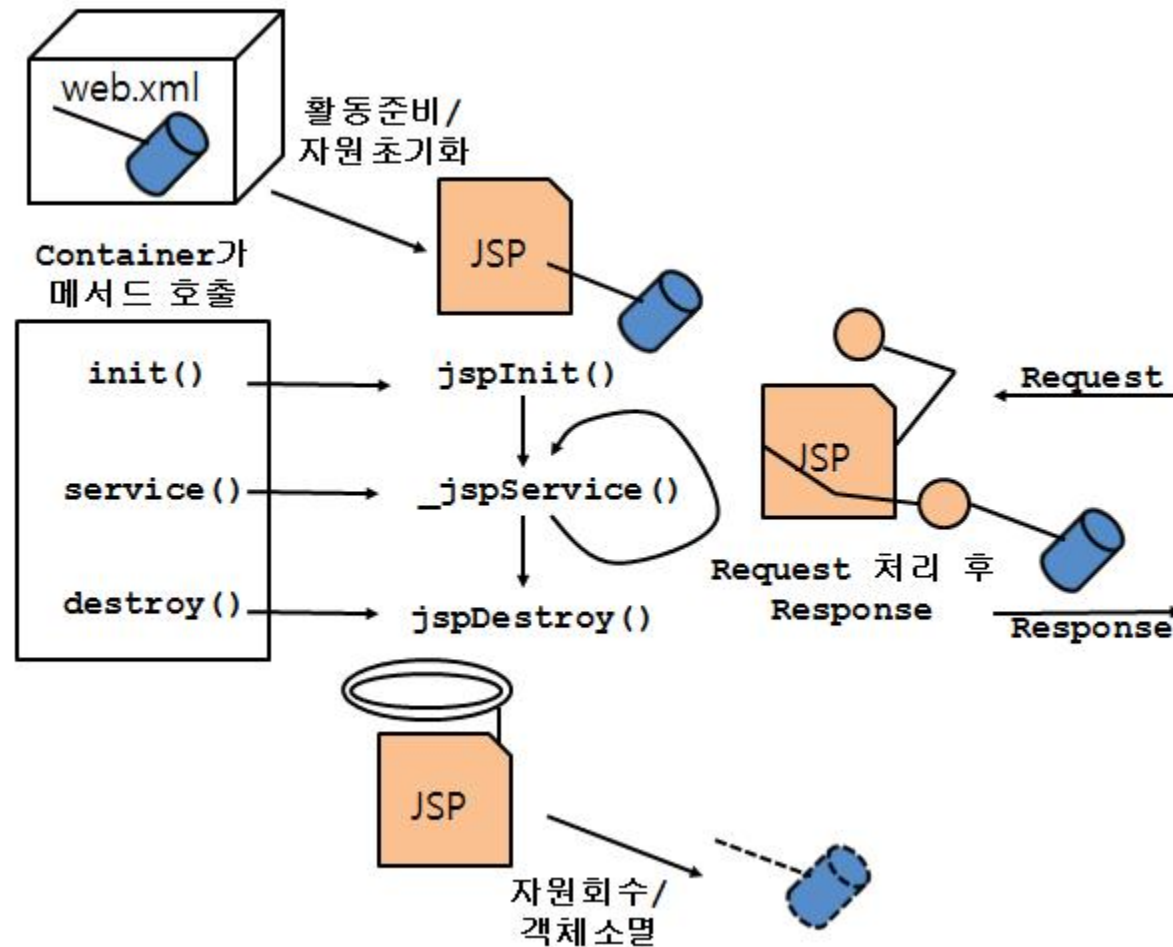
# @WAS



# @Tomcat

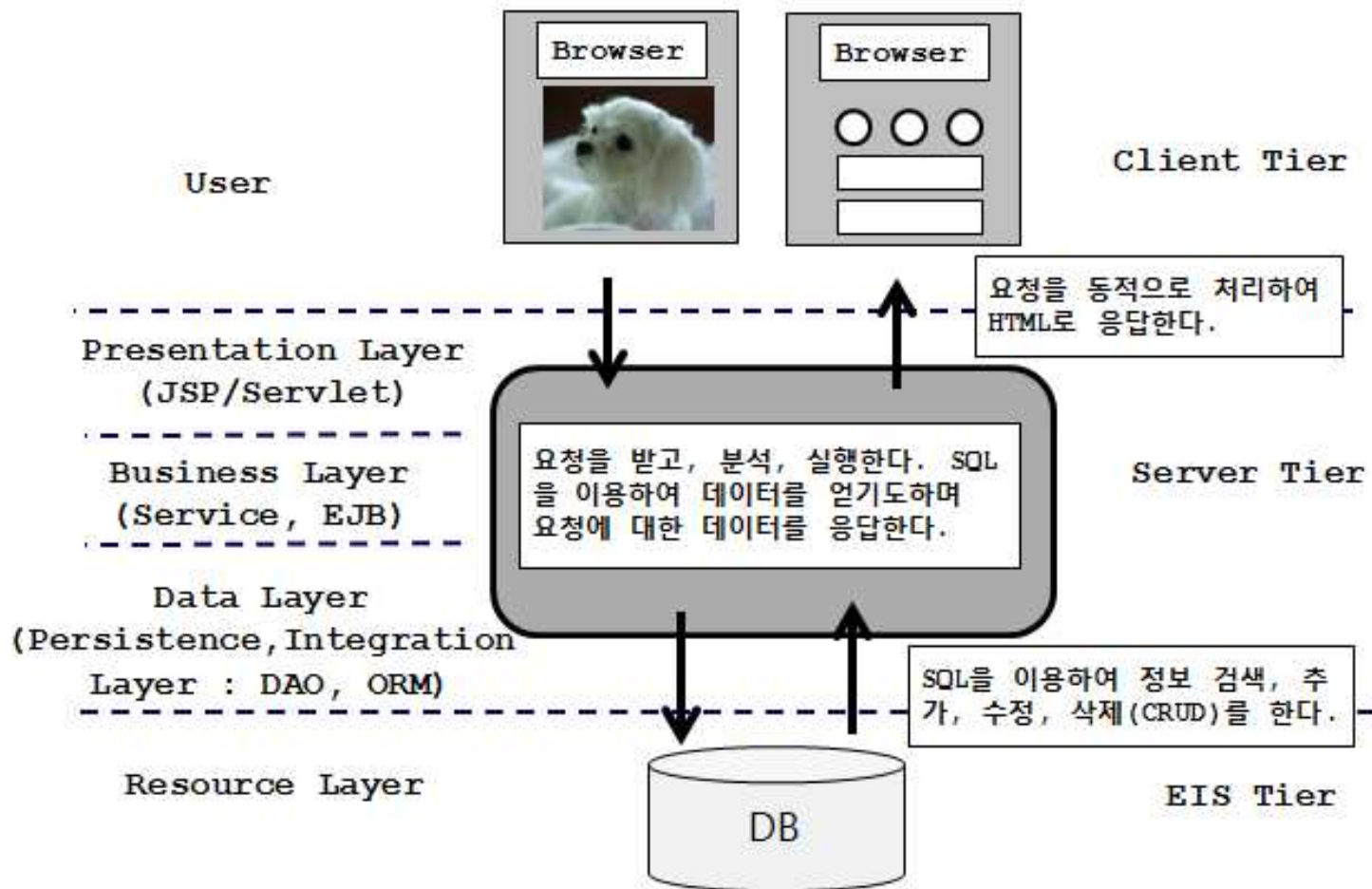


# @Life Cycle



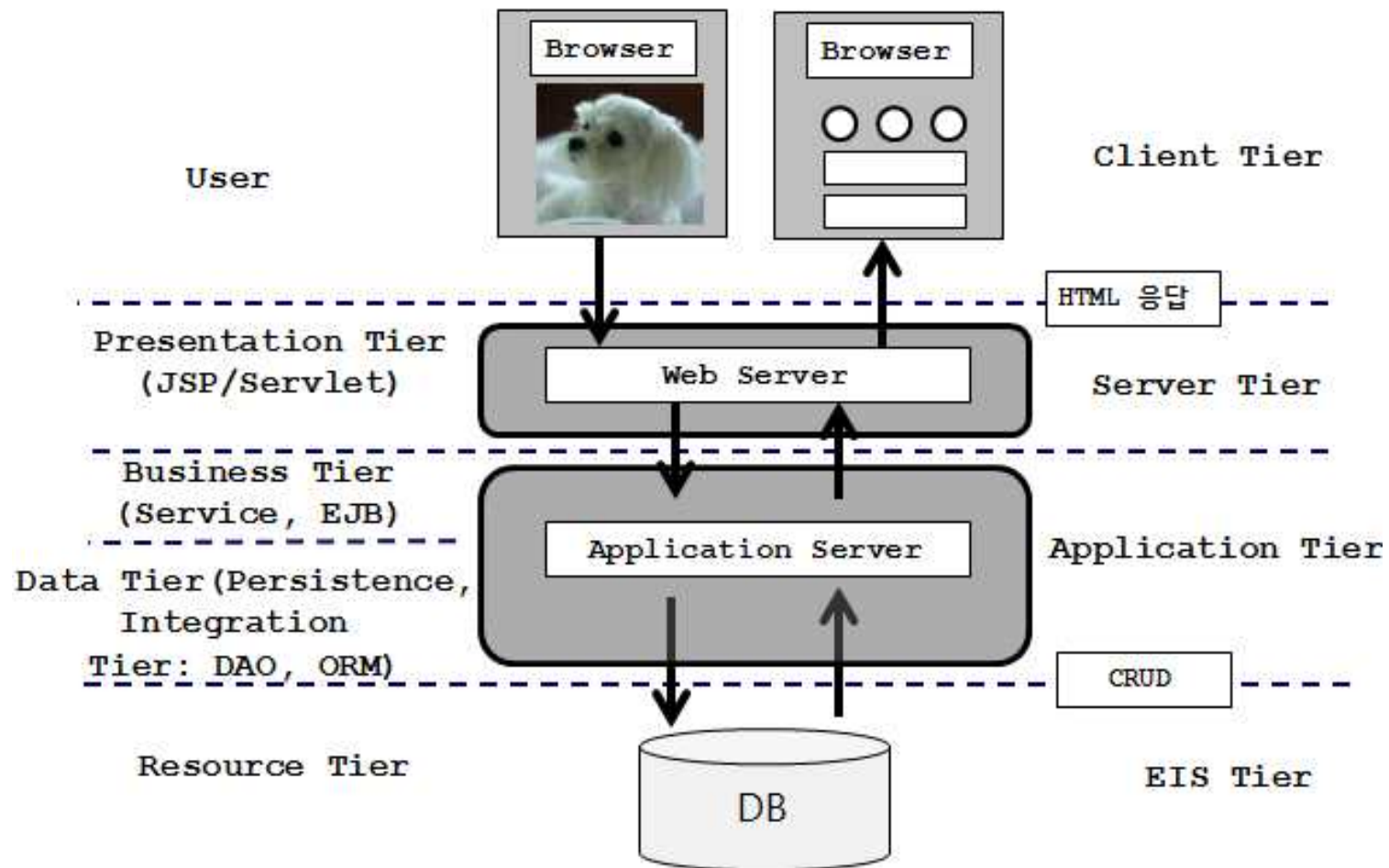


# @3Tier

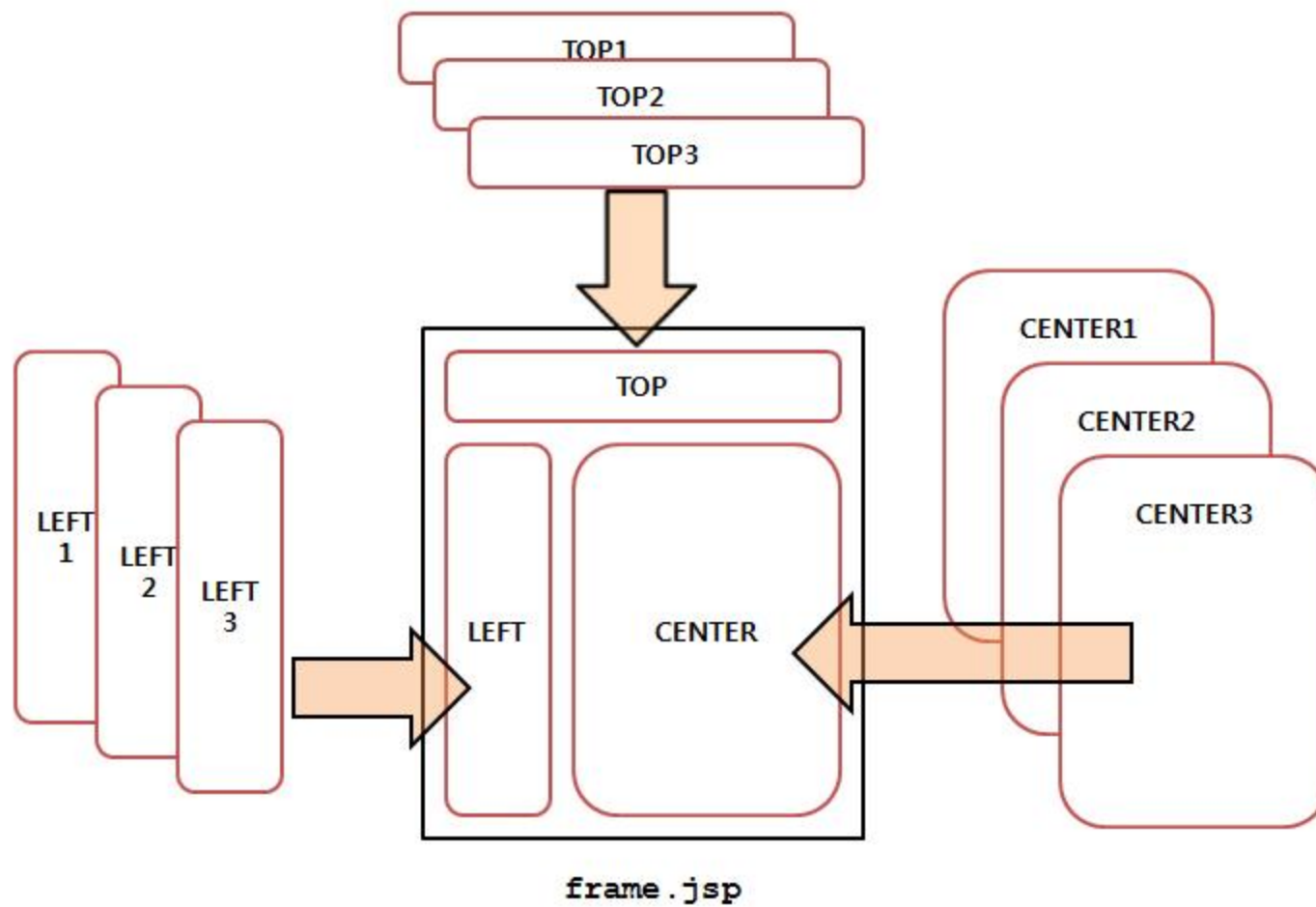




# @4Tier



# @Module화





# @Listener

---

이벤트 감지

```
<listener>
```

```
<listener-class>hyonny.goods.event.BookContextListener</listener-  
class>
```

```
</listener>
```

```
<listener>
```

```
<listener-class>hyonny.goods.event.ShoppingListener</listener-class>  
</listener>
```

```
<listener>
```

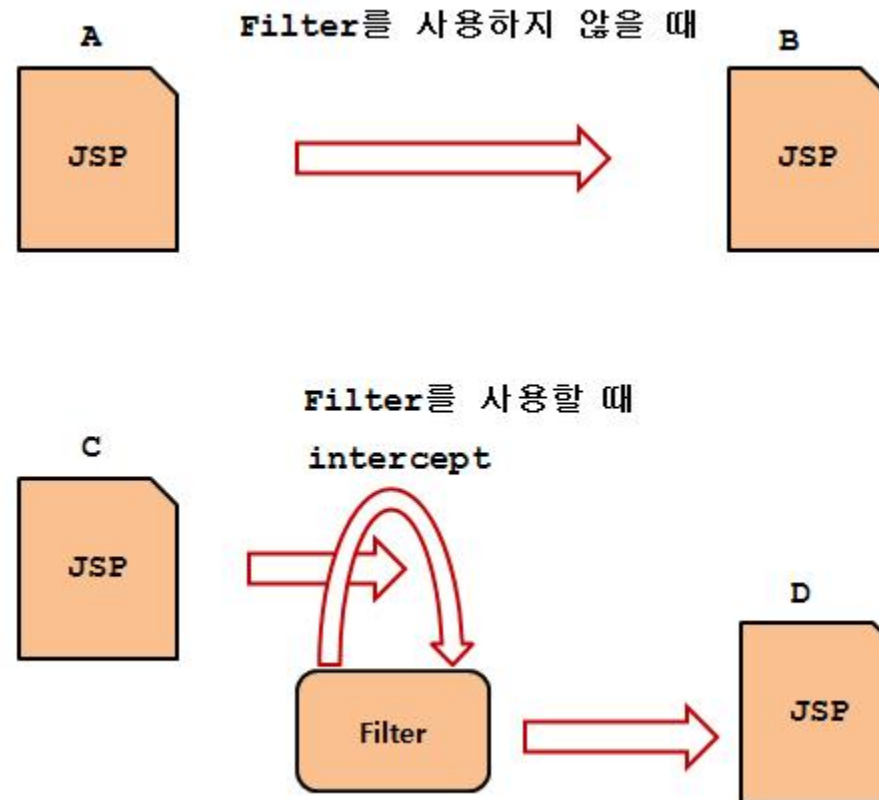
```
  <listener-class>
```

```
    hyonny.goods.event.ShoppingAttributeListener
```

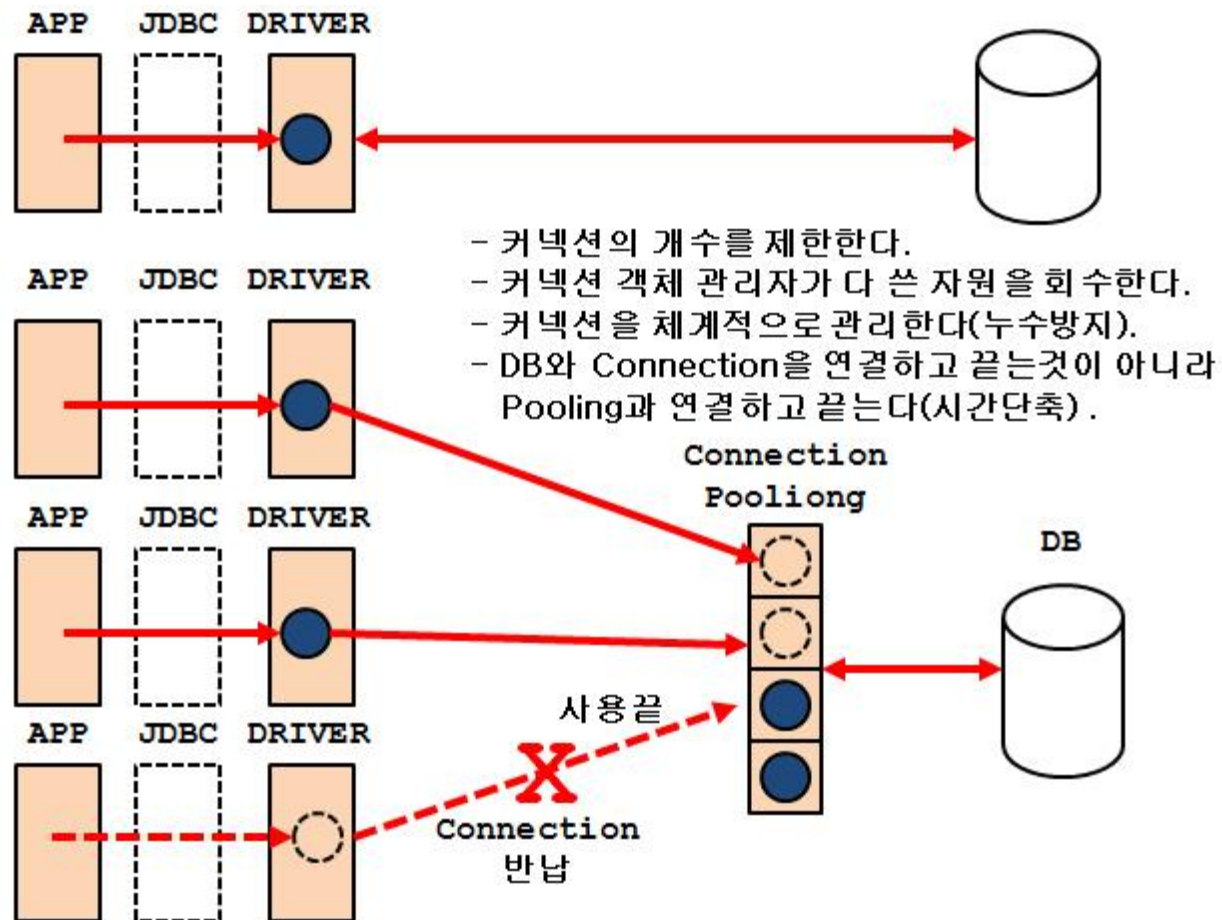
```
  </listener-class>
```

```
</listener>
```

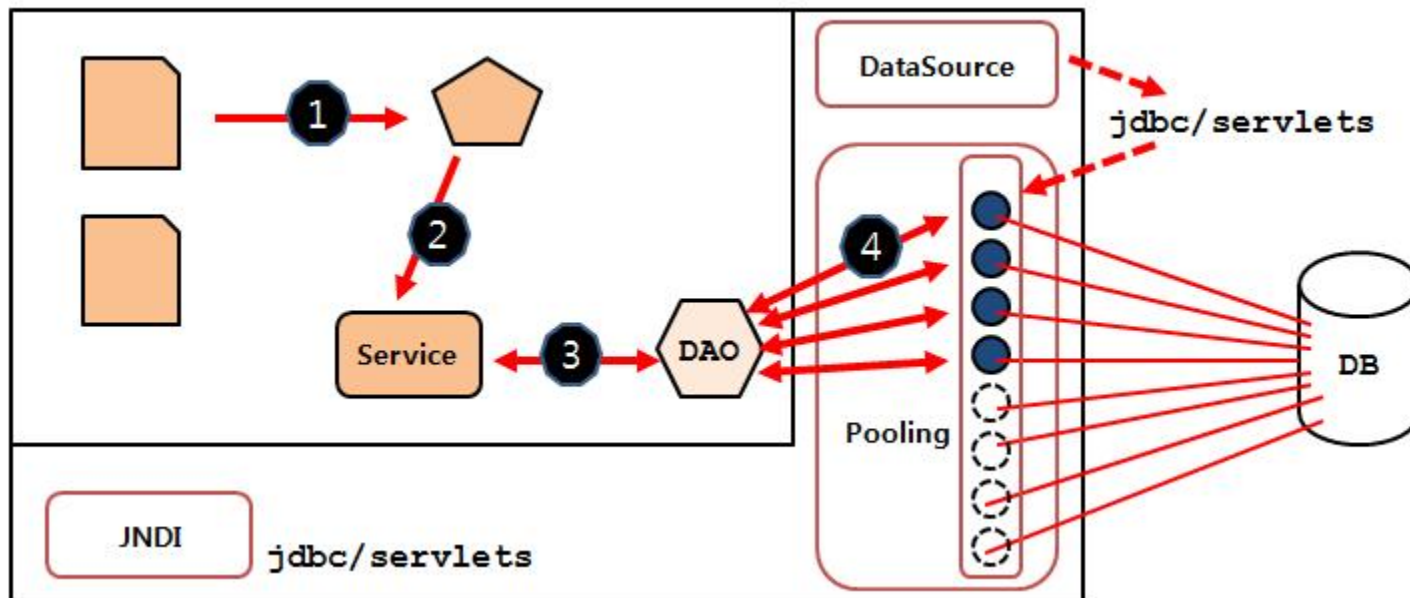
# @Filter



# @Pooling



# @Pooling

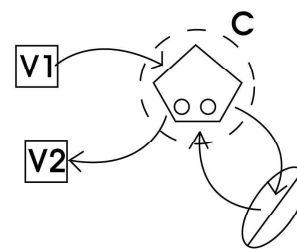

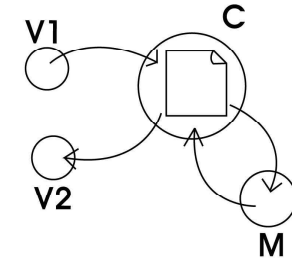
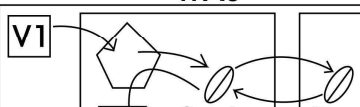




## 2. 웹 프로그래밍 기술발전

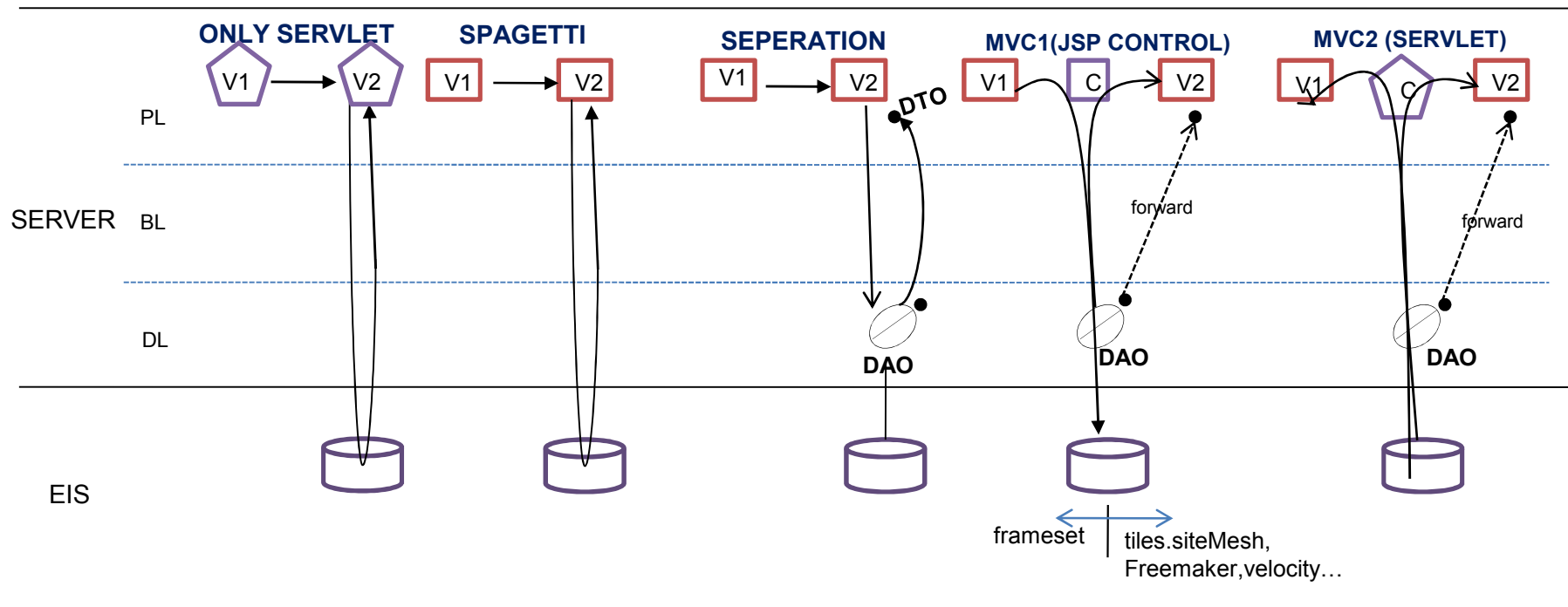


# @History

<ul style="list-style-type: none"><li>* 계층구조</li><li>* life cycle</li><li>* url mapping</li></ul>	<ul style="list-style-type: none"><li>* scriptElement view + viewlogic</li></ul>	<ul style="list-style-type: none"><li>* scriptlet 줄이자 view + viewlogic</li></ul> <div>DL<ul style="list-style-type: none"><li>WorkBean</li><li>DAO =&gt; DTO</li></ul></div>	EL JSTL    \${cust.id} OGNL controller 7가지		<div>FrontControl<div><div></div><div>View</div><div>Model</div></div></div>
	<div>HelperView 패턴</div> <div>XML&lt;div&gt;UseBean -&gt; Action</div> <div>CustomTag -&gt; 사용자 정의</div> <div><div>① Class xxx extends TagSupport</div><div>② xxx.tld</div><div>③ &lt;%@taglib prefix="xxx" url="/xxx.tld"%&gt;</div><div>④ &lt;yyy:show/&gt;</div></div>		<div>Multi Controller</div> <div>↓</div> <div>Action</div> <div>↓</div> <div>Struts =&gt; Spring</div>	<div>WAS<div><div><div>V1</div><div>V2</div></div><div></div><div>Service</div><div>Bean</div></div></div> <div>MVC EJB Model2</div>	

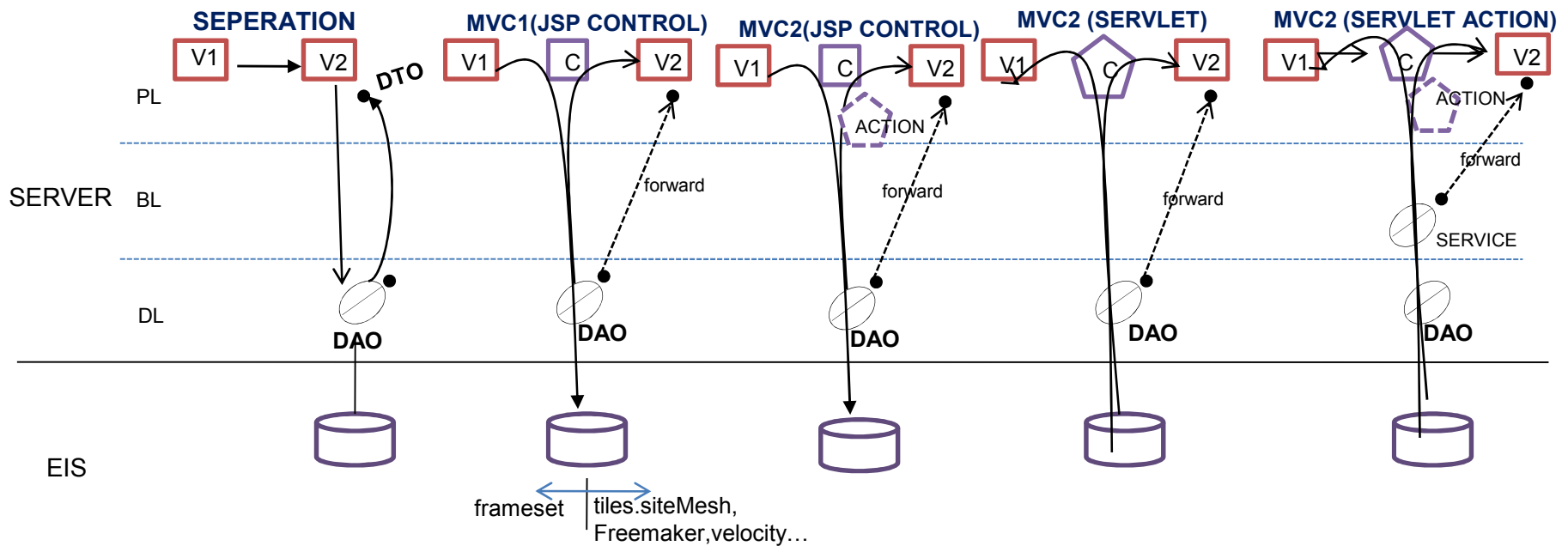
# @기술발전 1

Client

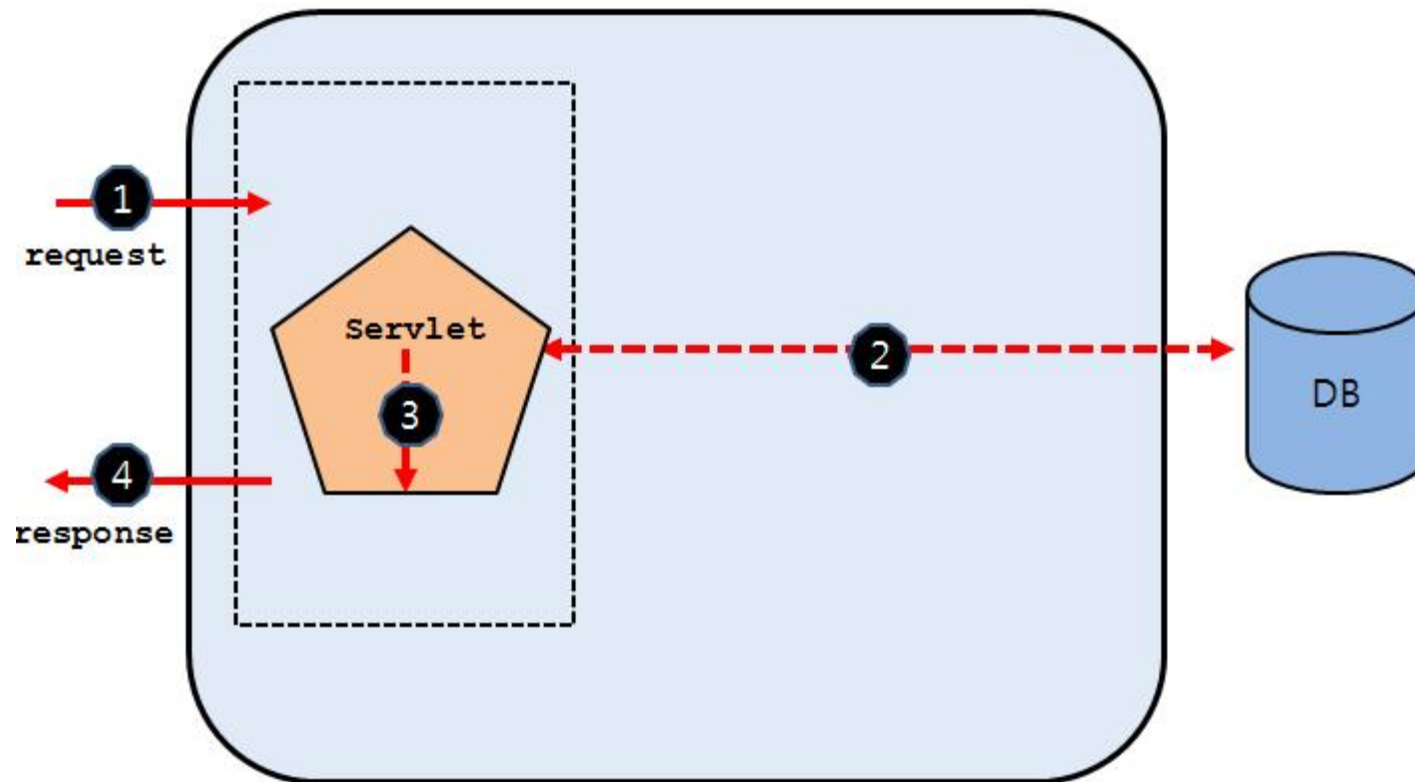


# @기술발전 2

종류

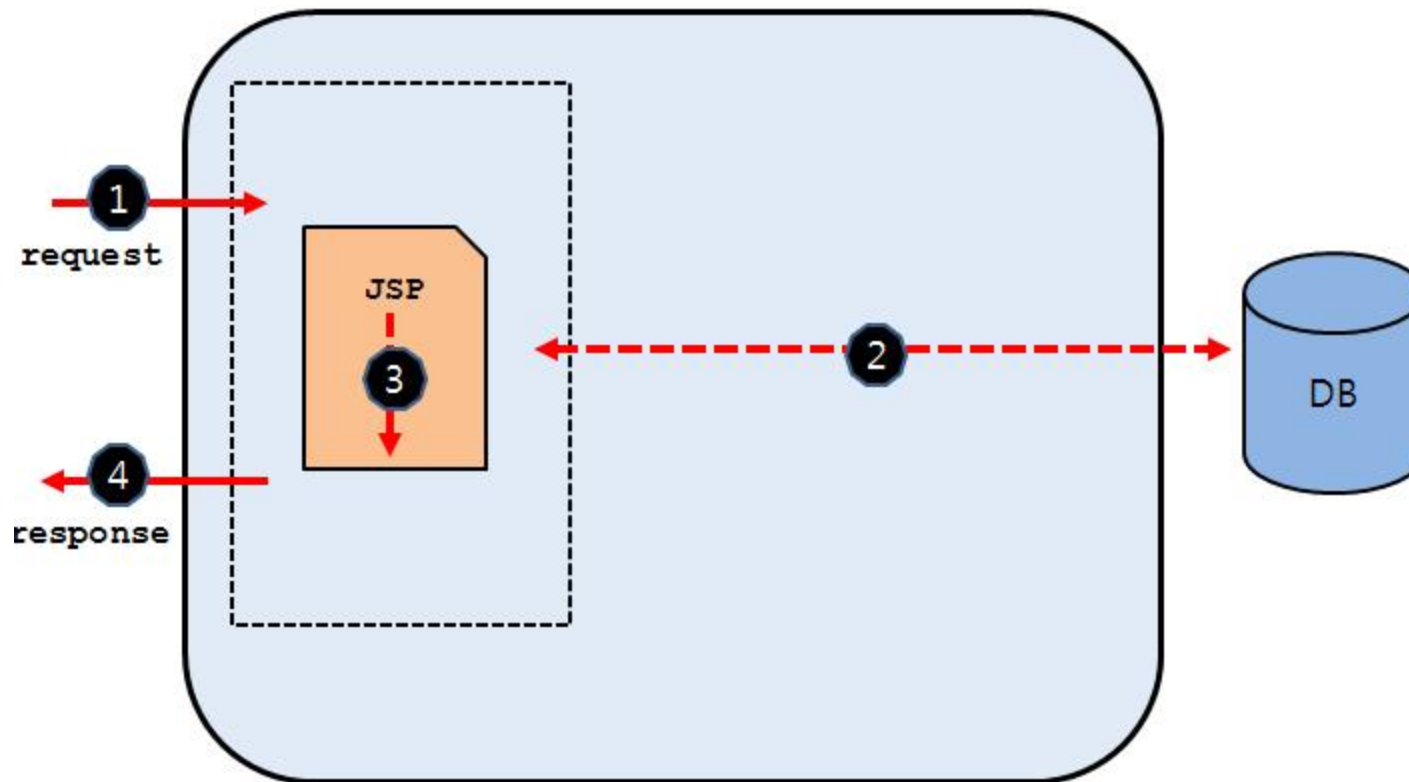


# @1997 Only Servlet



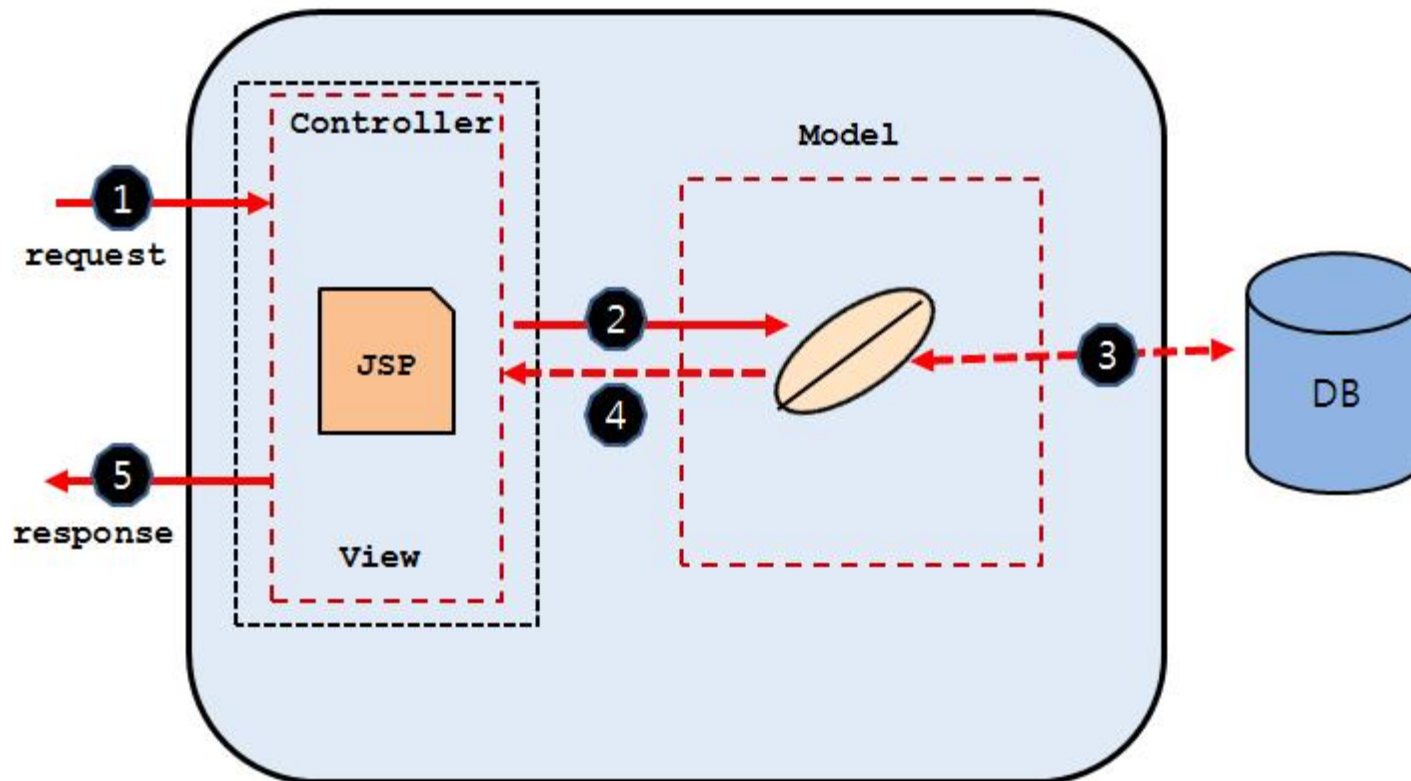
- ① 요청    ④ 응답    ② 데이터를 처리하거나 얻는다.  
③ 처리결과를 화면으로 만든다.

# @1999 Spagetti



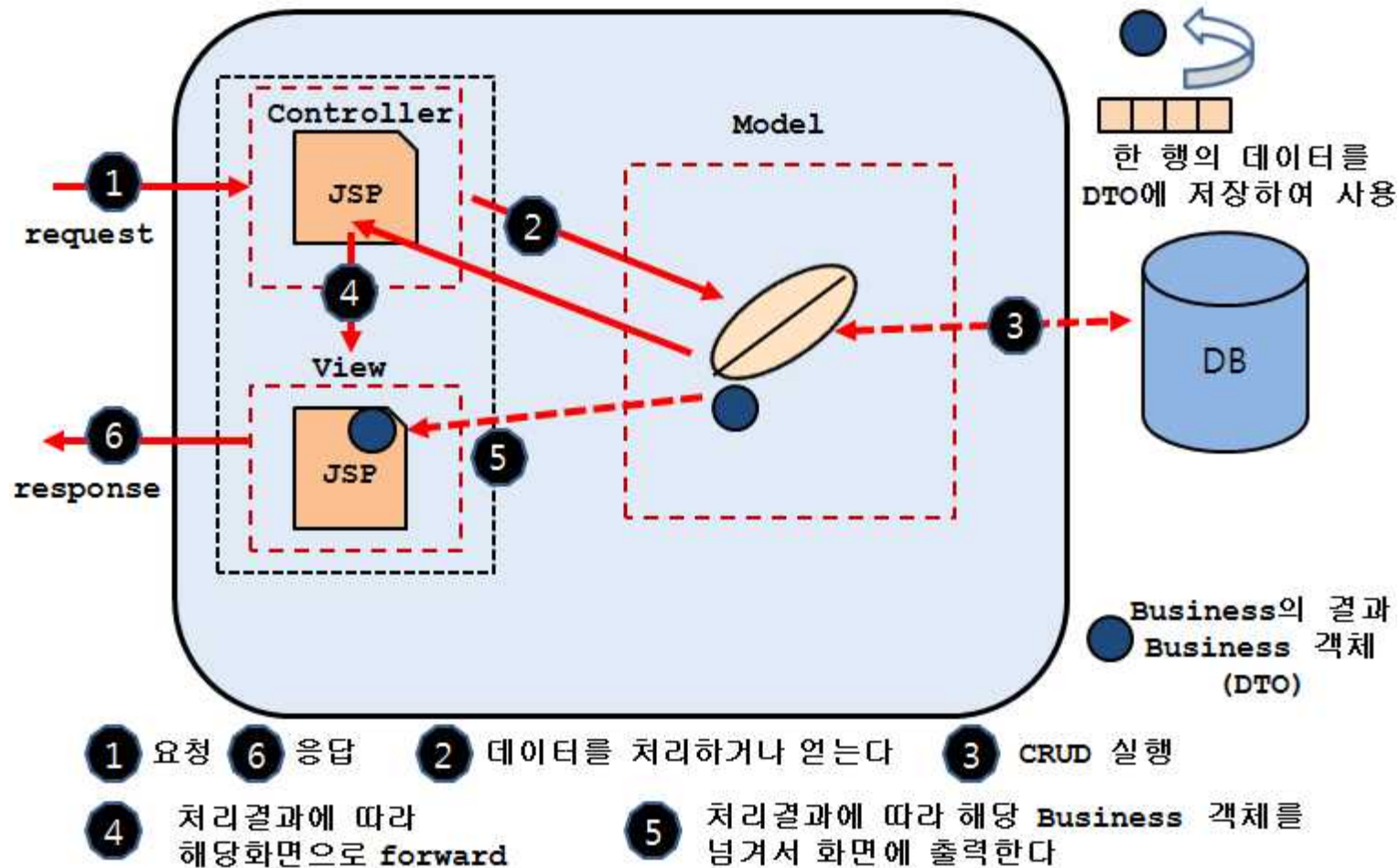
- ① 요청 ④ 응답 ② 데이터를 처리하거나 얻는다.
- ③ 처리결과를 화면으로 만든다.

# @2000 Separation



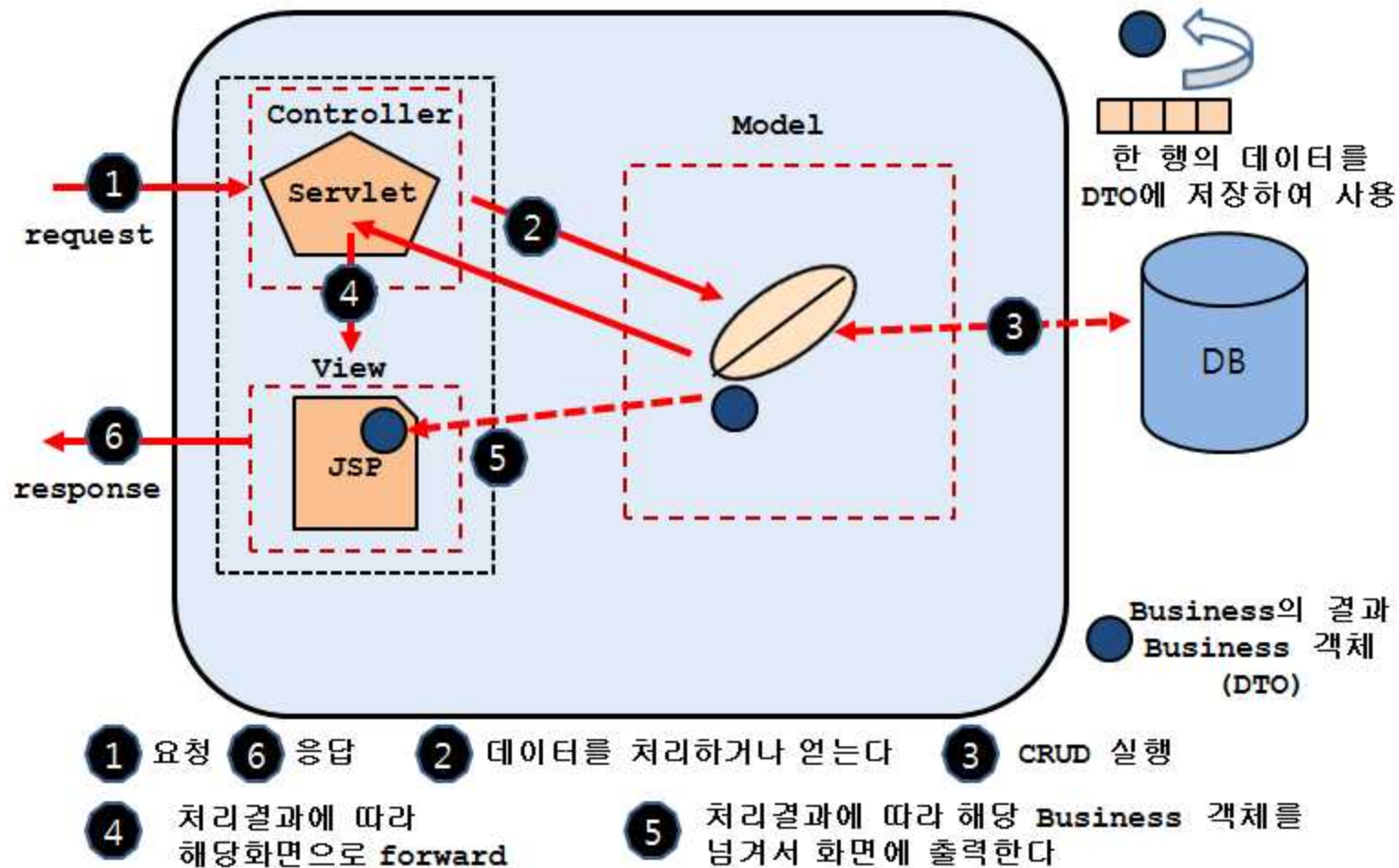
- ① 요청   ⑤ 응답   ② 데이터를 처리하거나 얻는다   ③ CRUD 실행  
④ 처리결과를 화면에 출력한다

# @2002JSP Centric

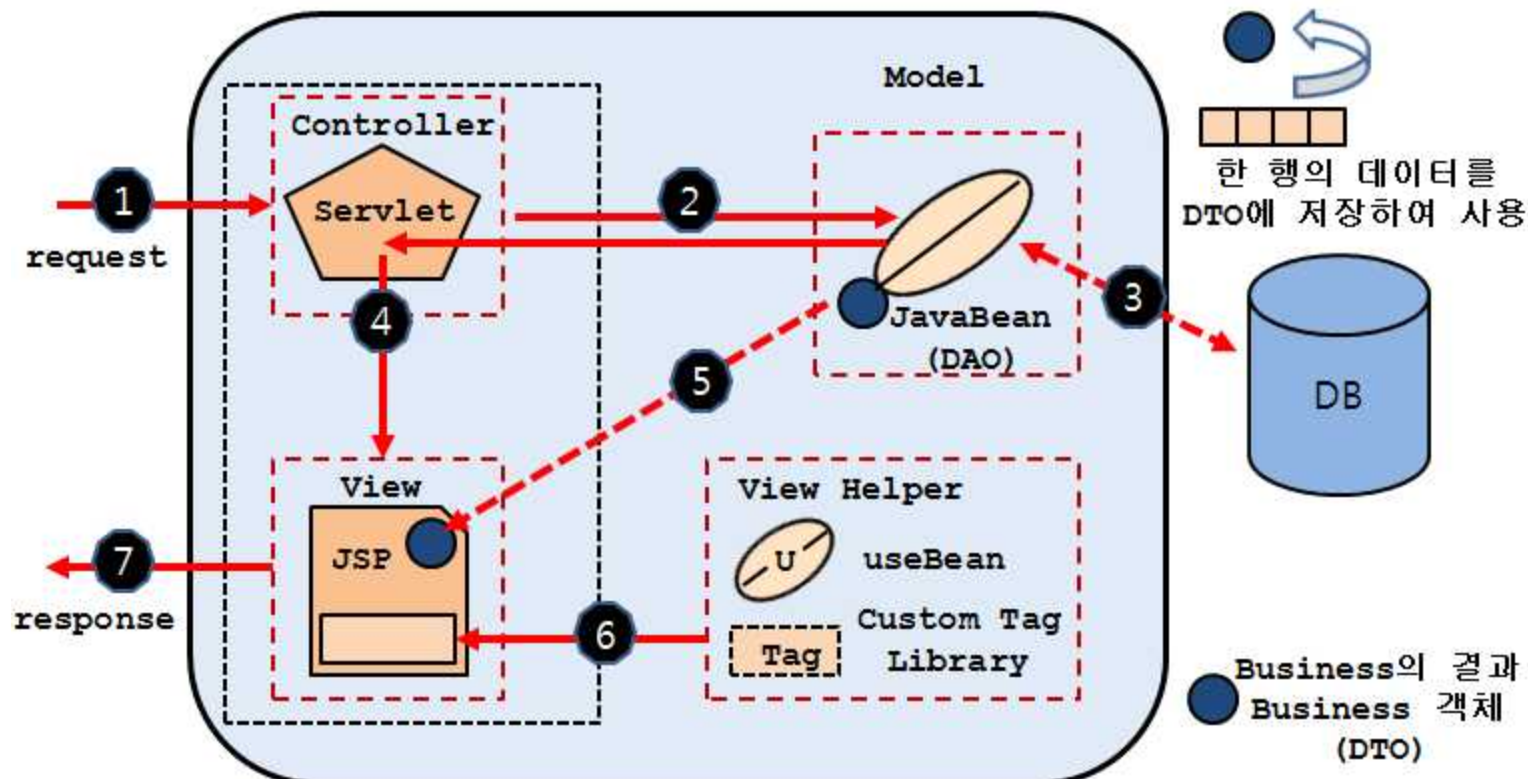




# @2004 Servlet Cetric

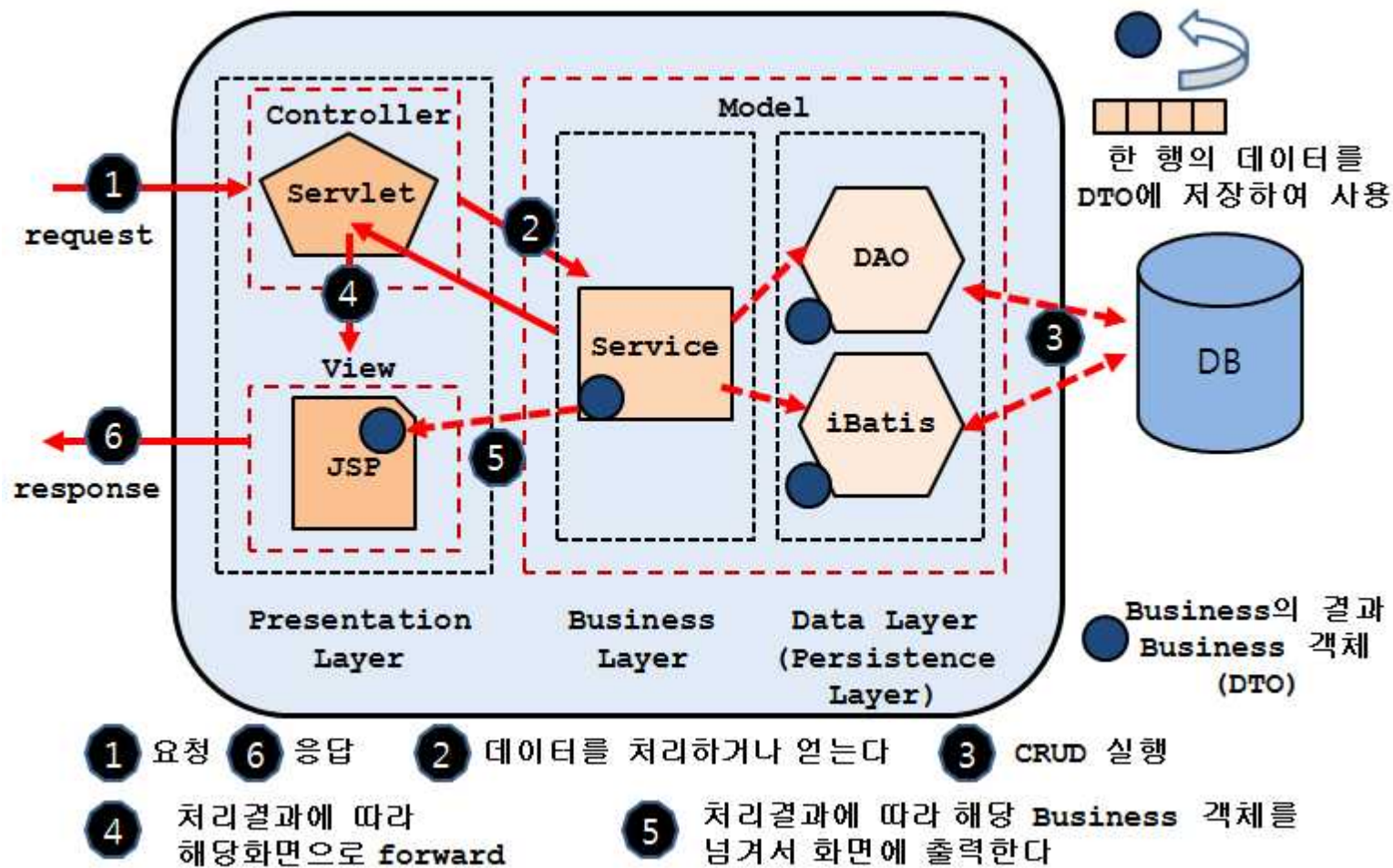


# @Helper View



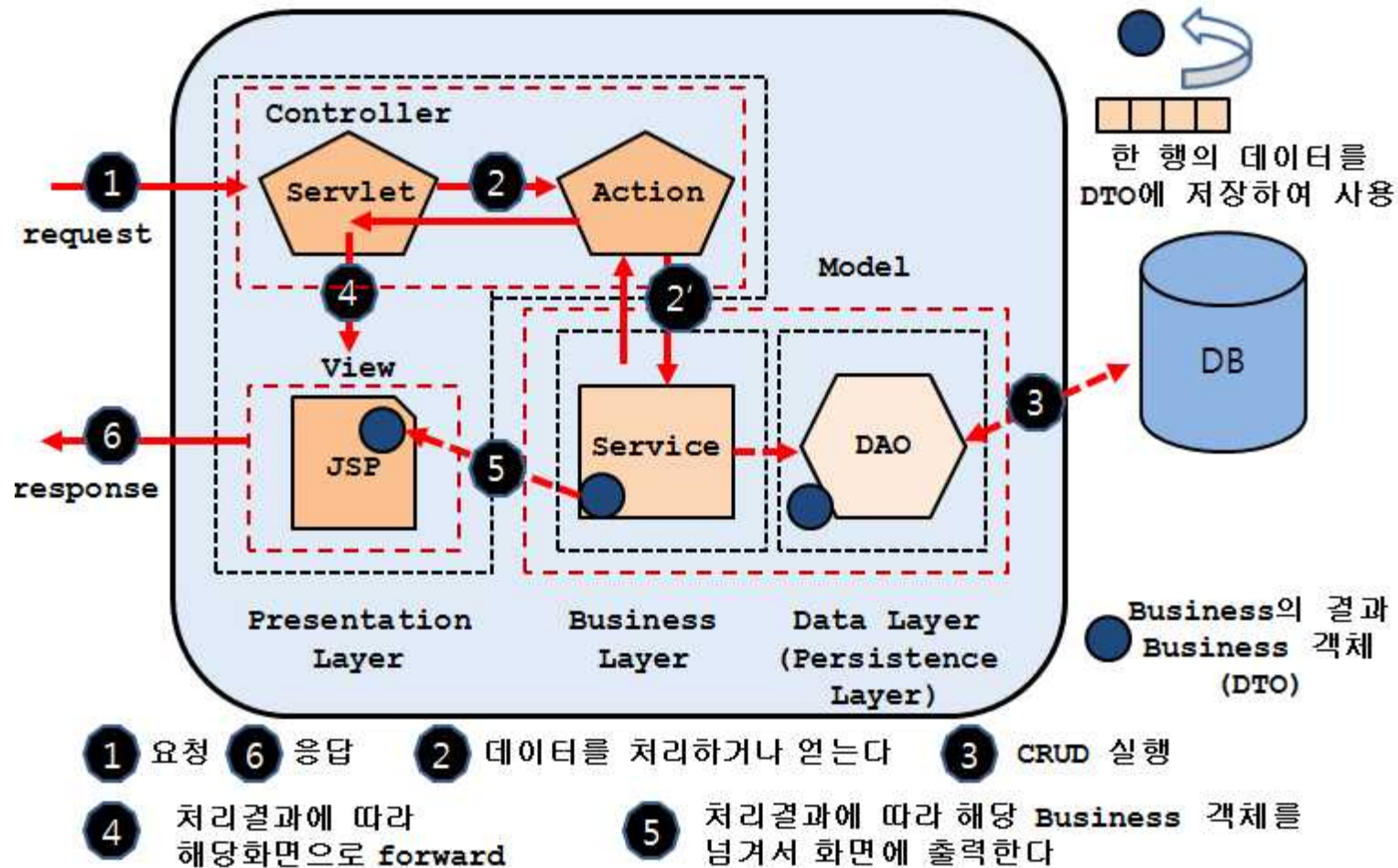
- ① 요청 ⑦ 응답 ② 데이터를 처리하거나 얻는다 ③ CRUD 실행  
 ④ 처리결과에 따라 해당화면으로 forward ⑤ 처리결과에 따라 해당 Business 객체를 넘긴다. ⑥ 부분화면생성 Business 객체 출력

# @Front/Facade

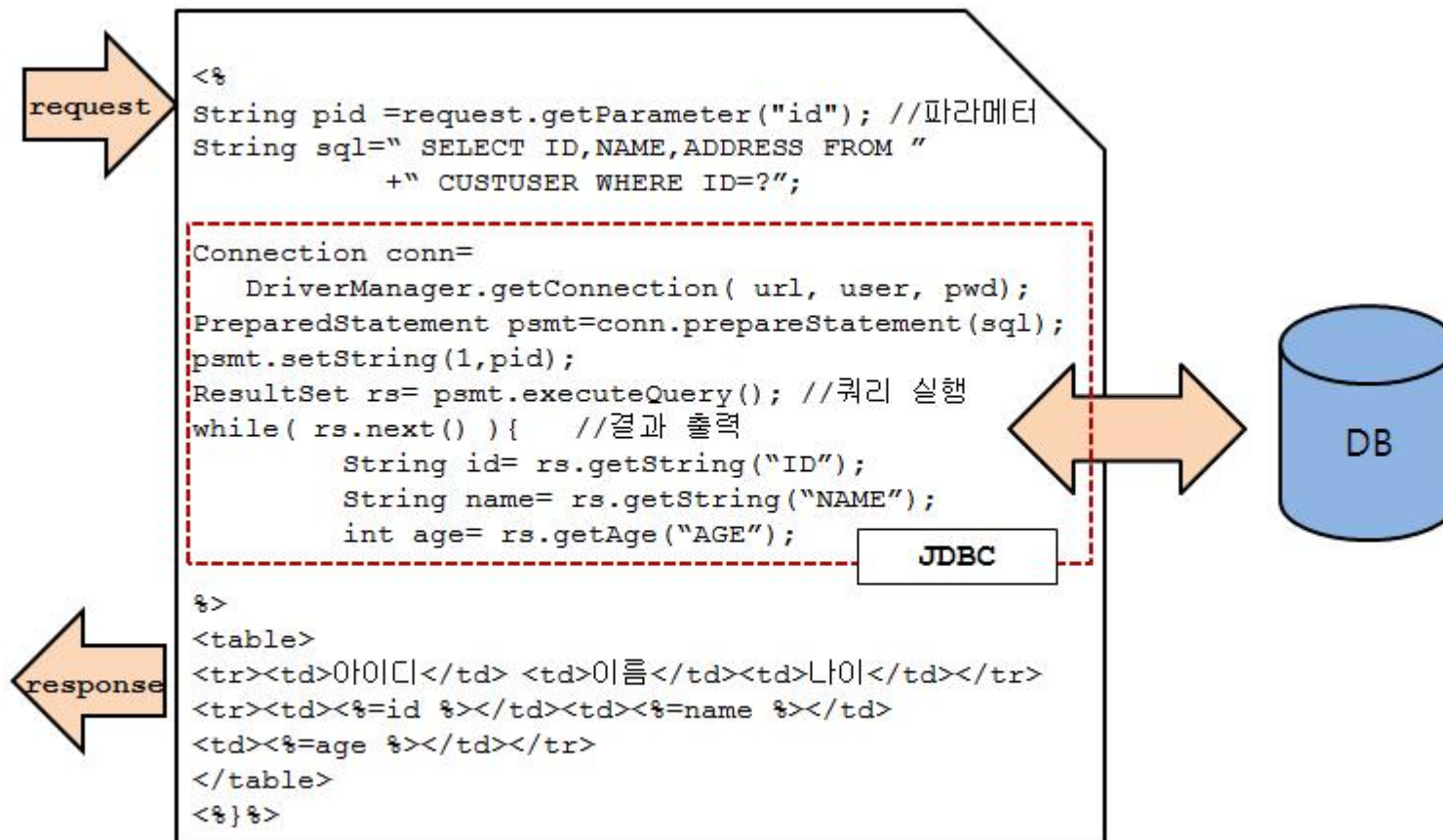




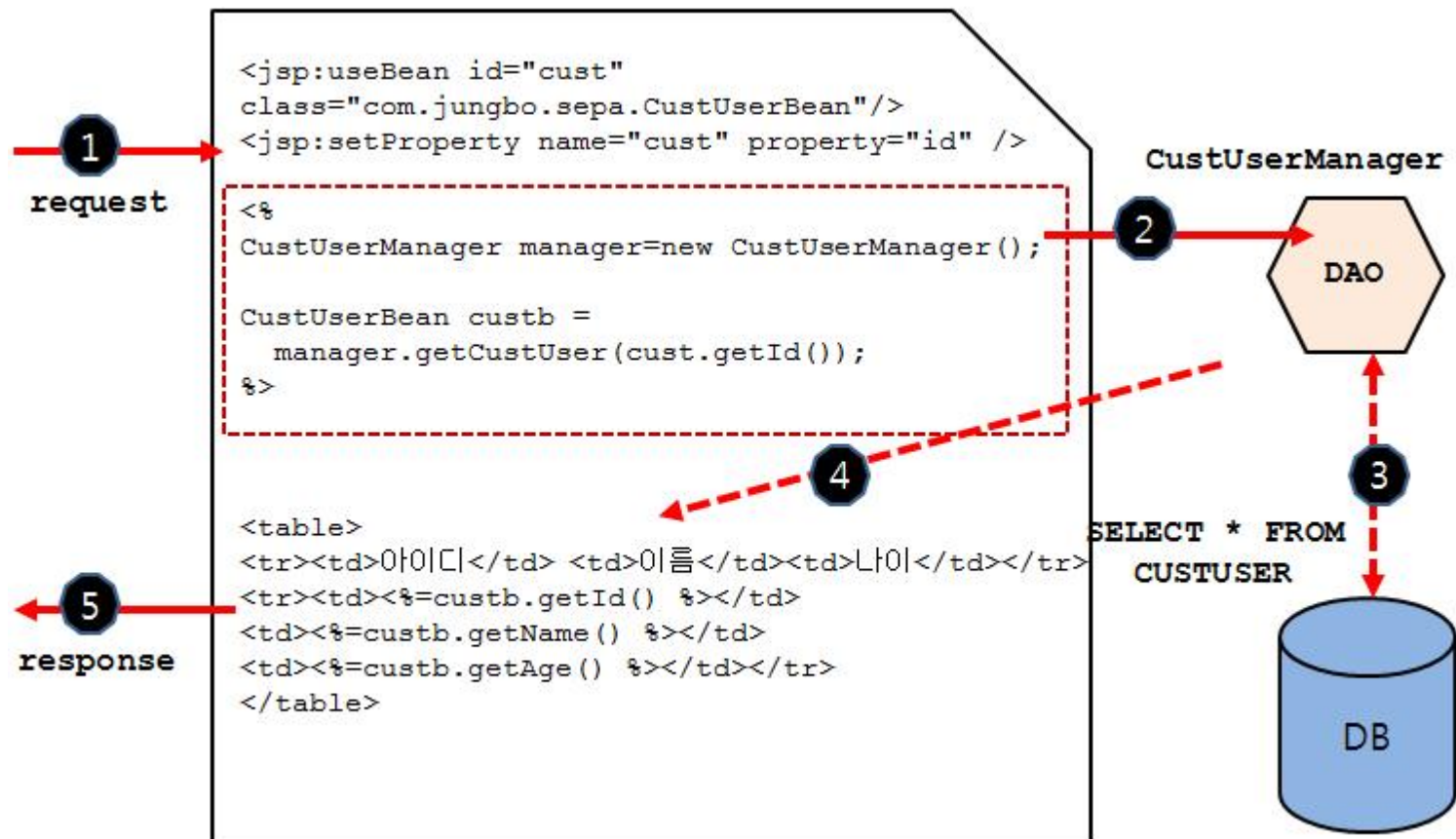
# @2004Action Servlet



# @Spagetti

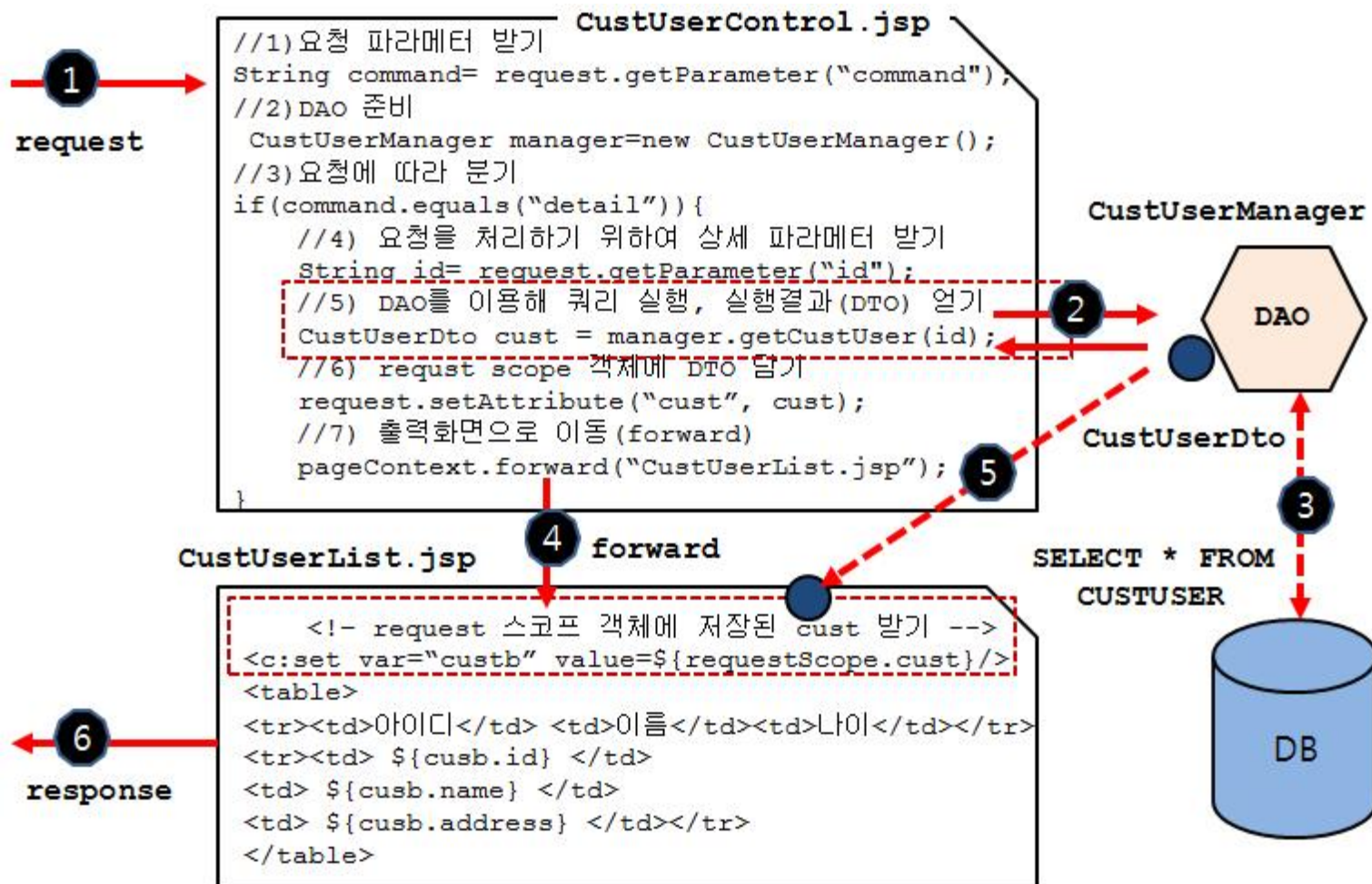


# @Separation



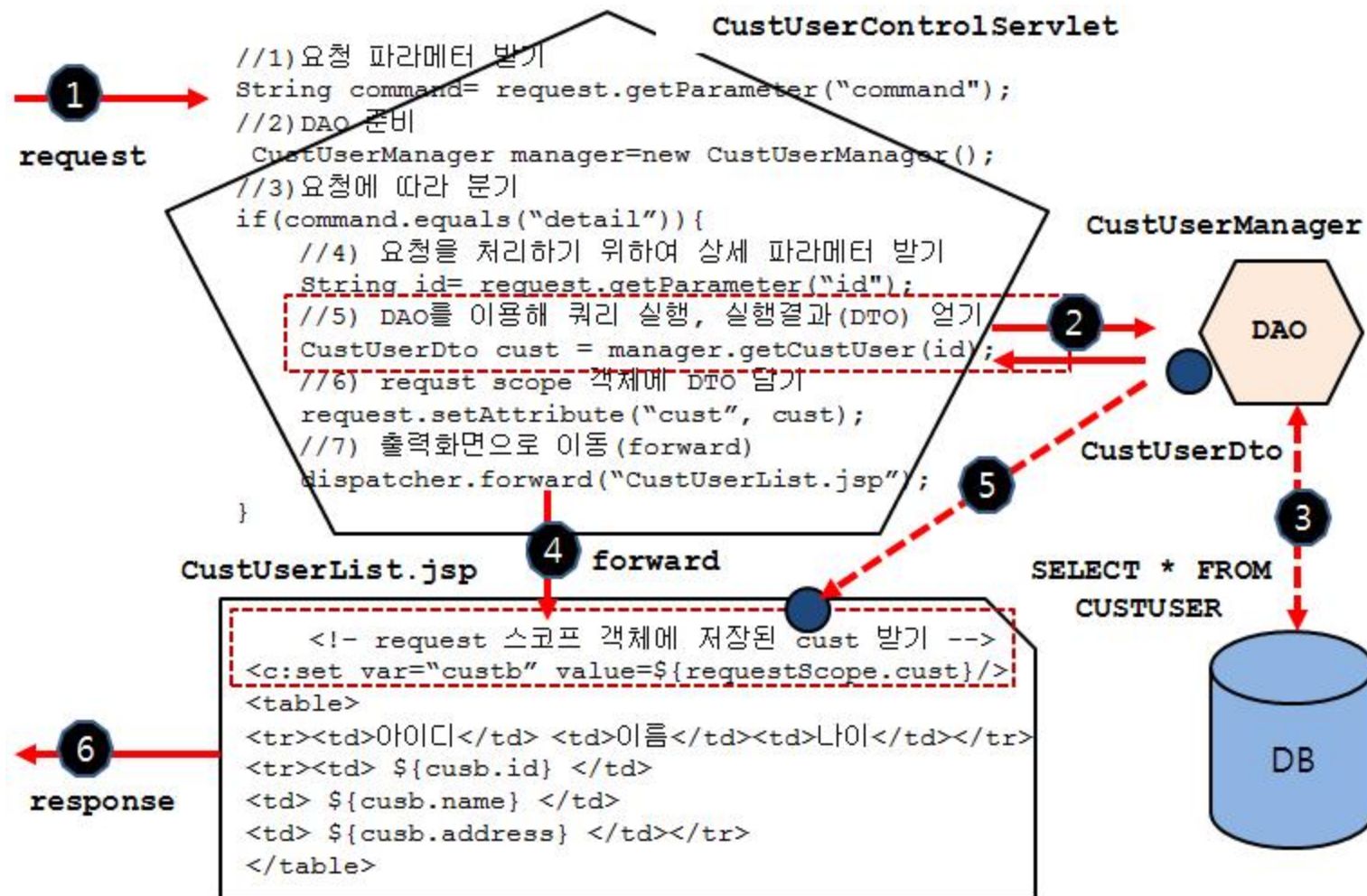


# @JSP Centric

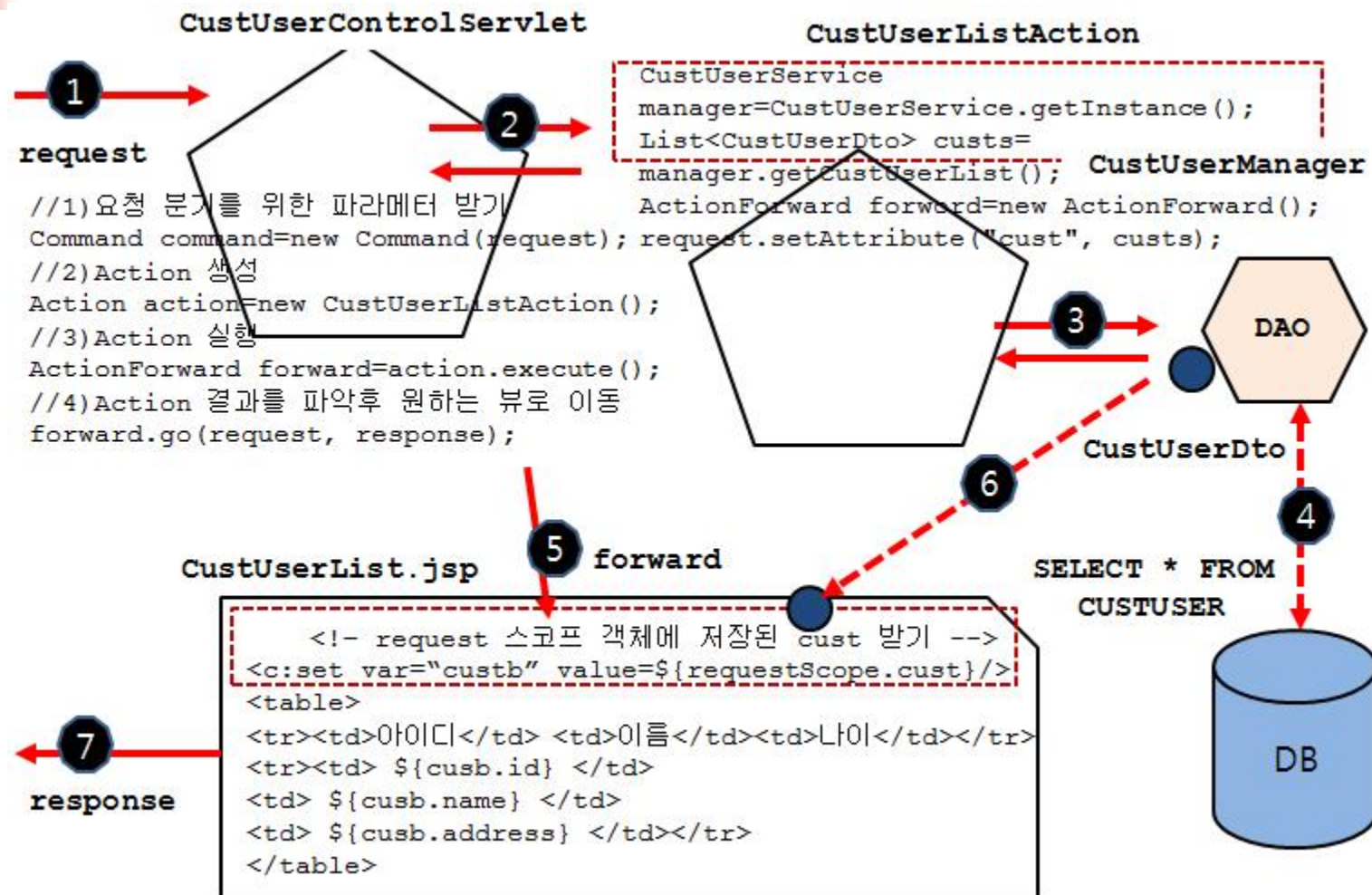




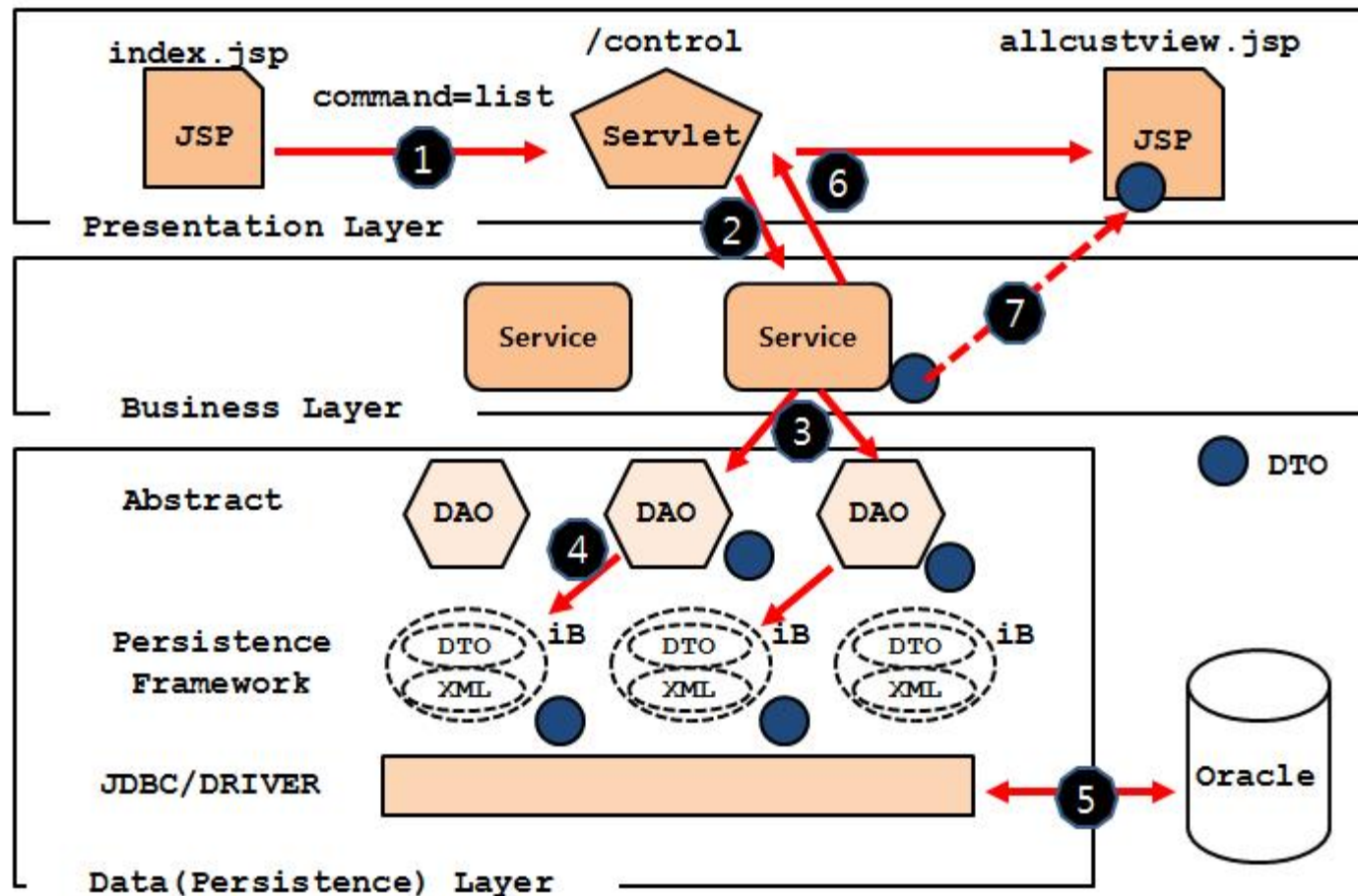
# @Multi Servlet, Facade



# @Action Servlet

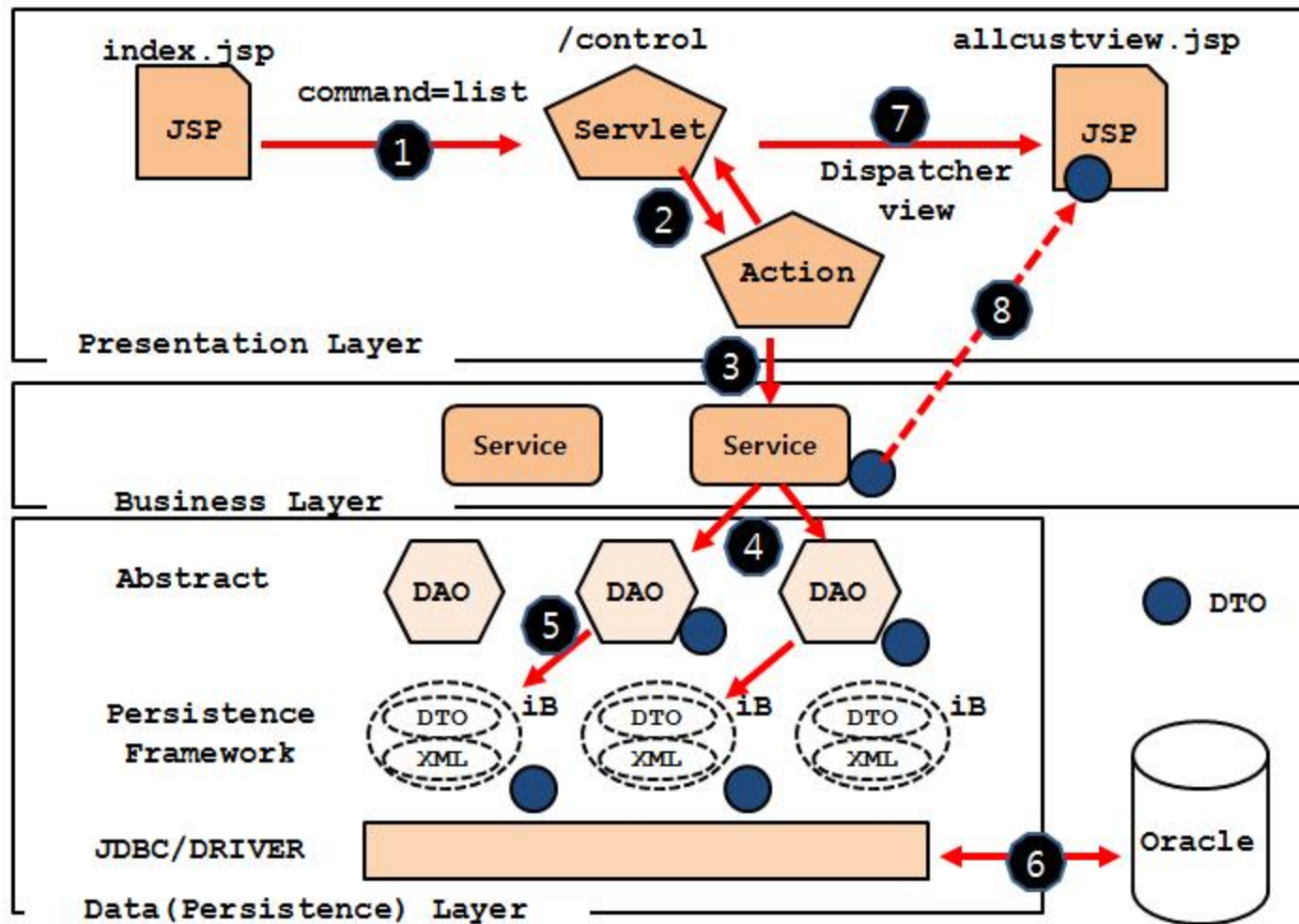


# @Front Controller

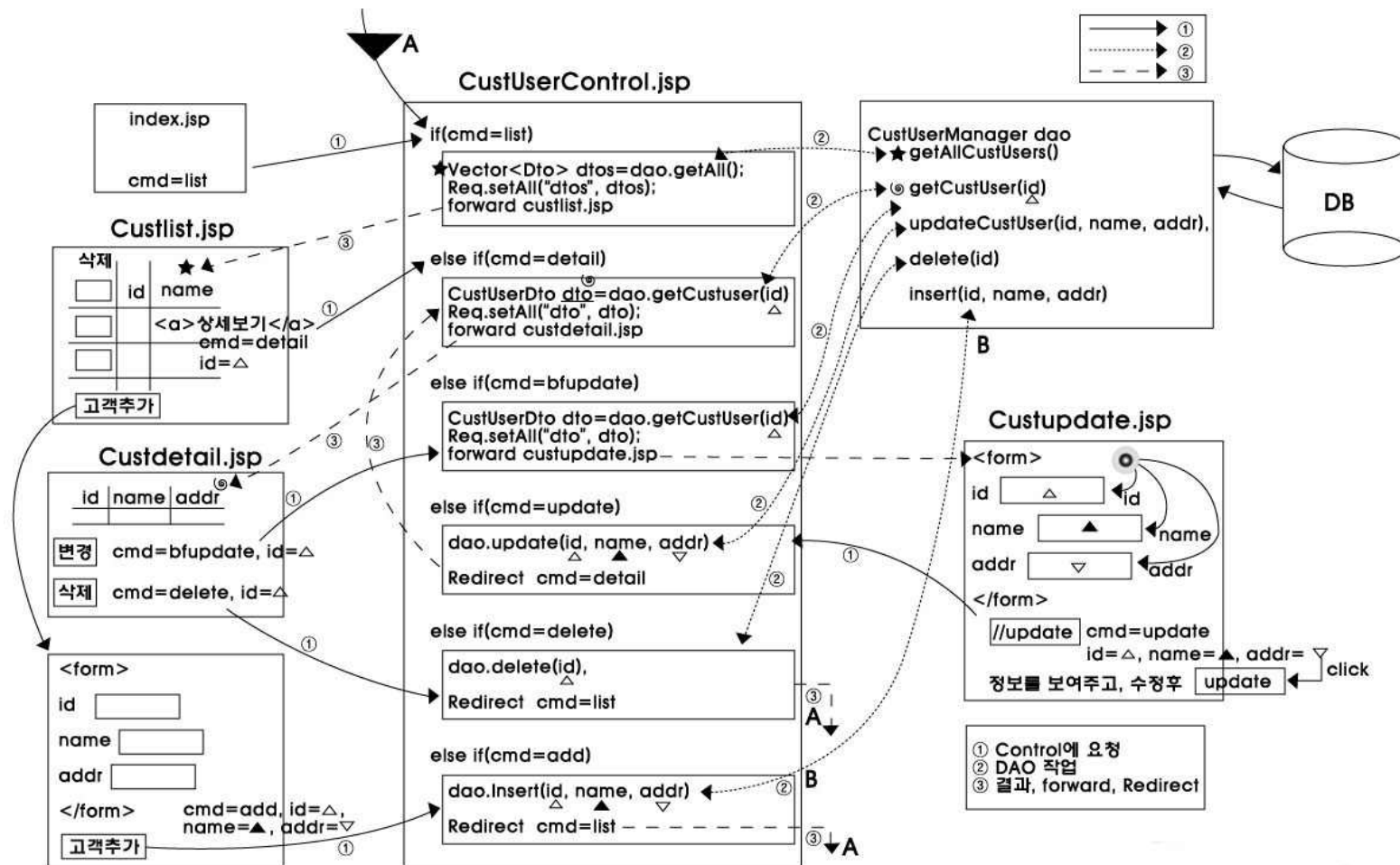




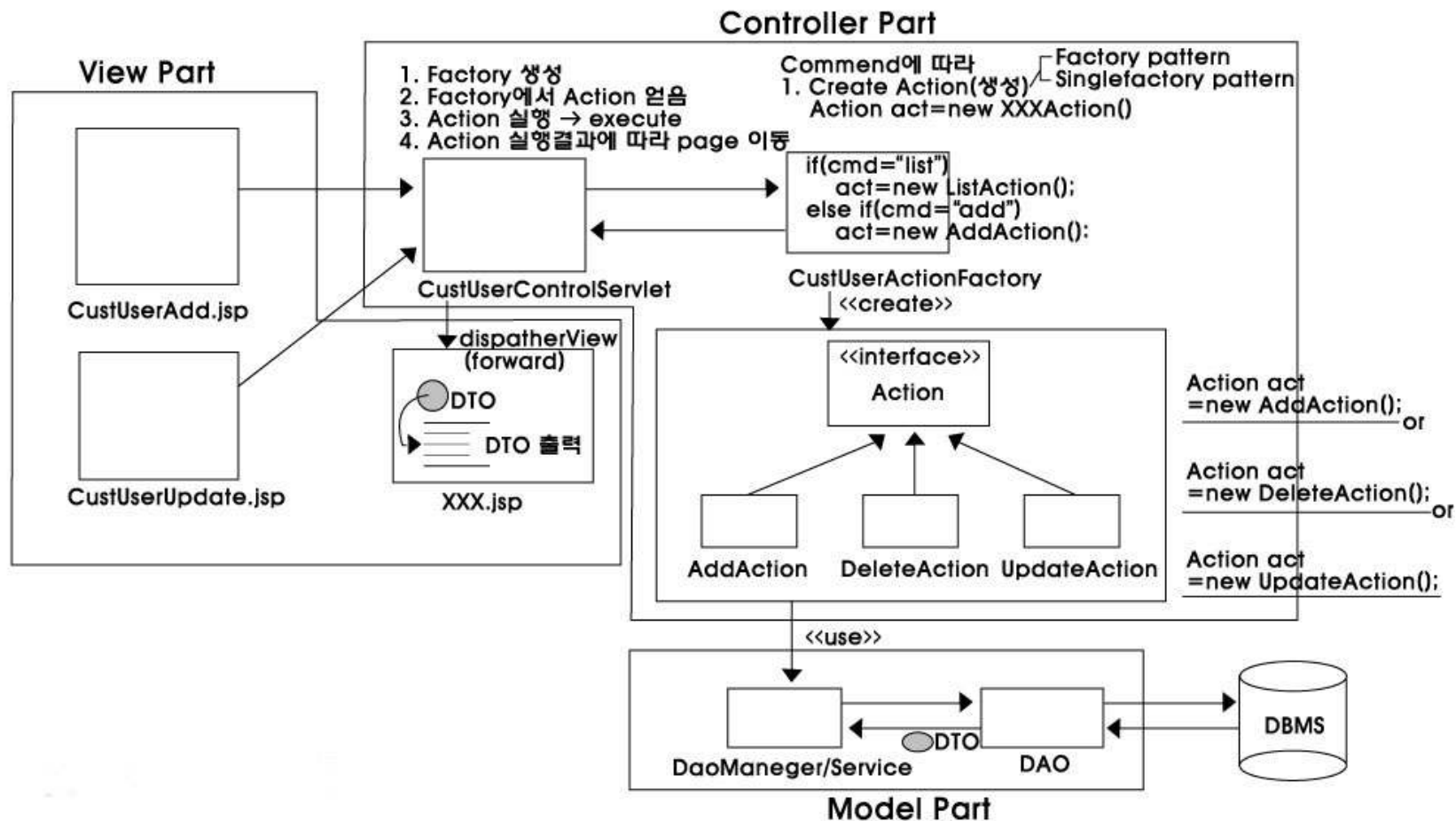
# @Action Servlet



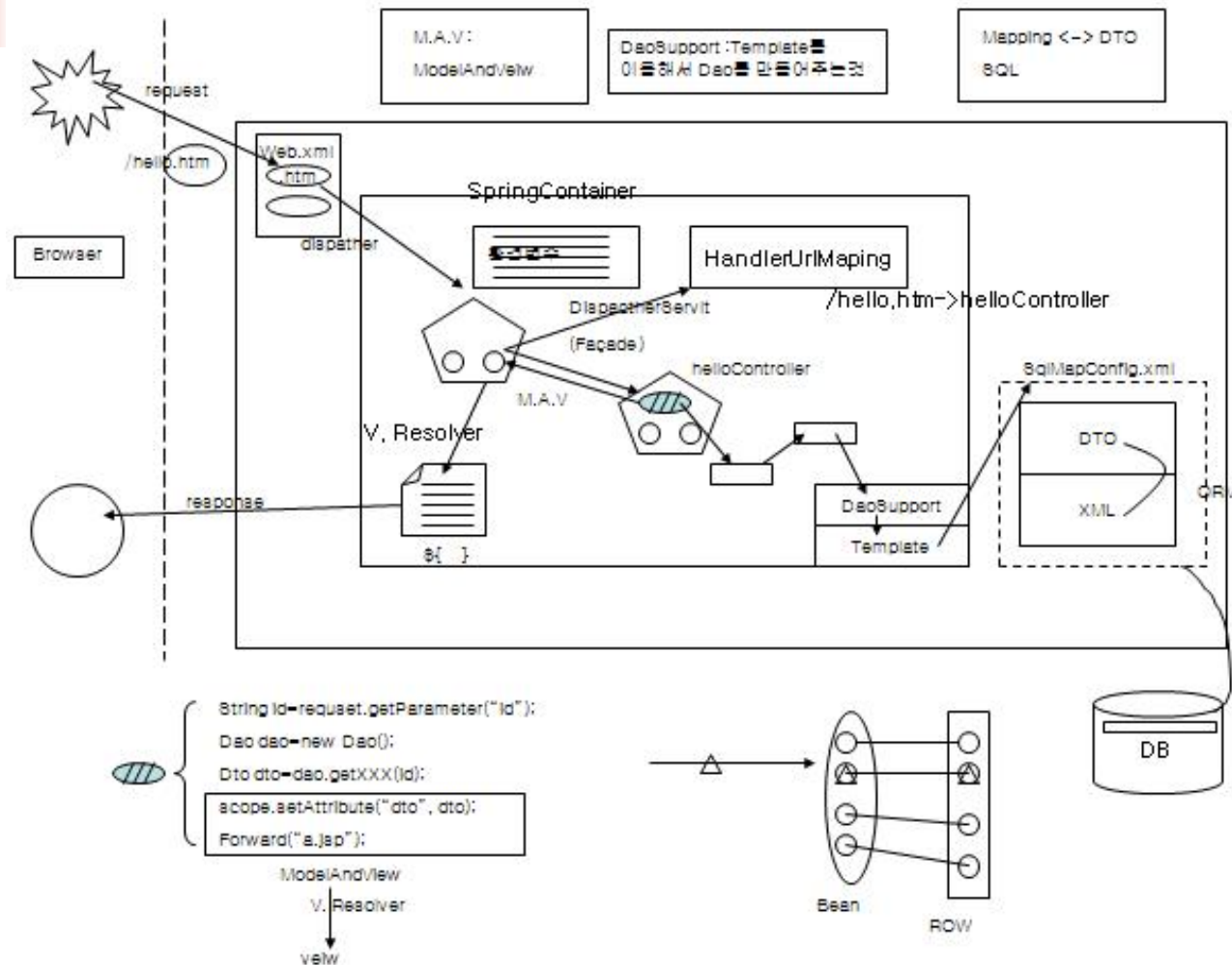
# @JSP Centric



# @Servlet Centric/Action



# @Spring Framework







# @Spring Framework

---

**1. DI**

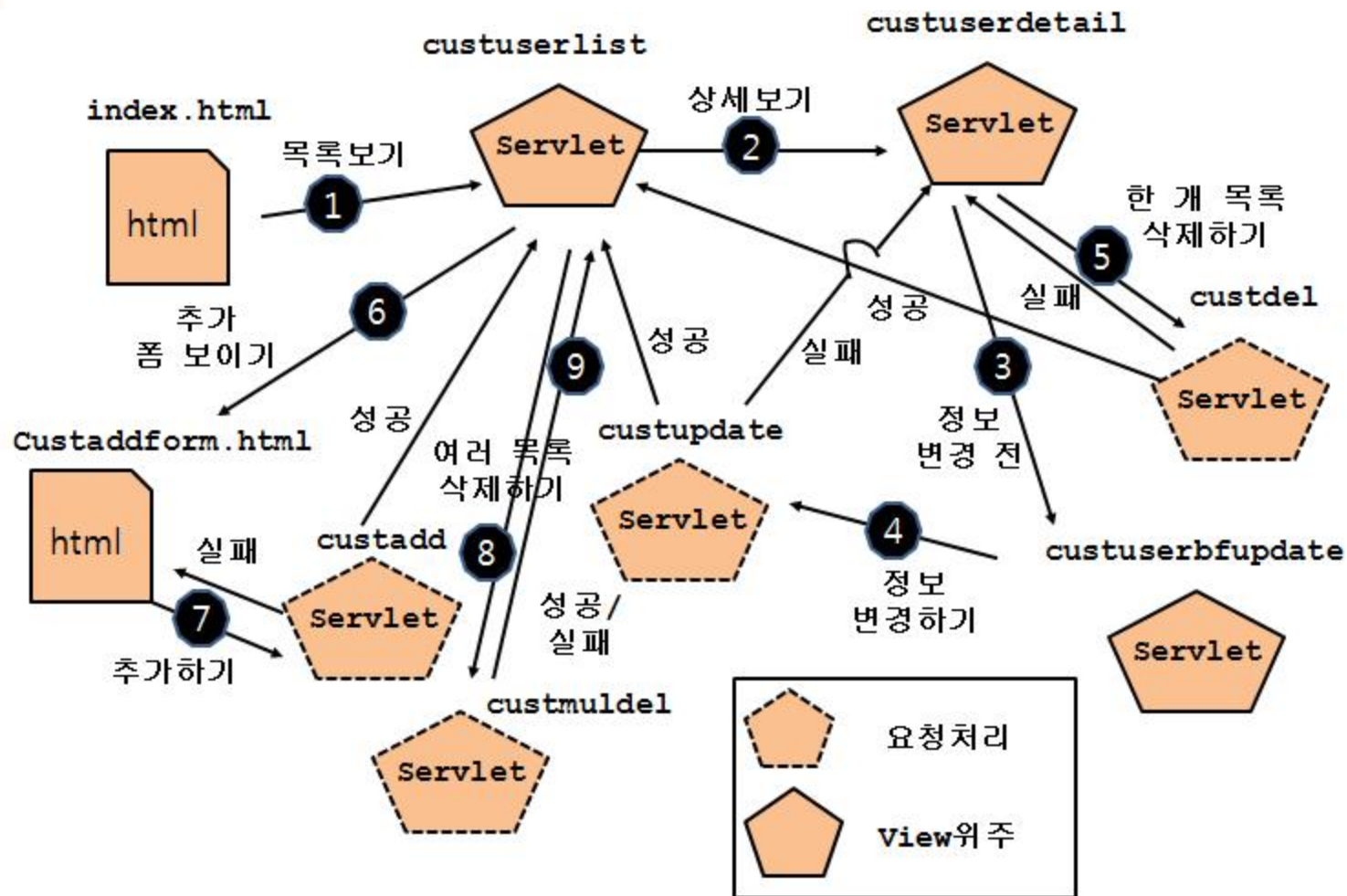
**2. IoC**

**3. AOP = CC + CCC**

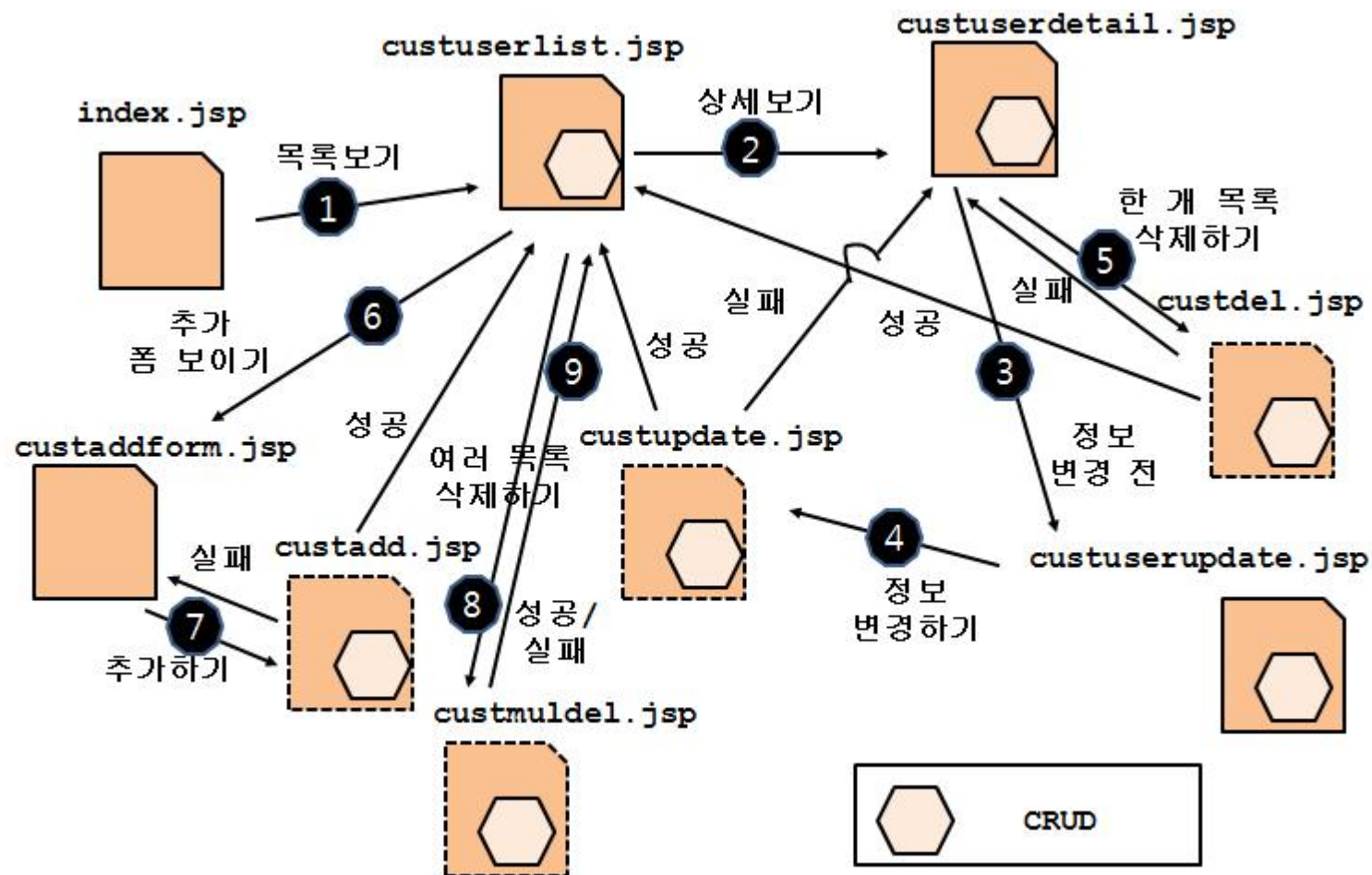


### 3. 프레임워크 기본

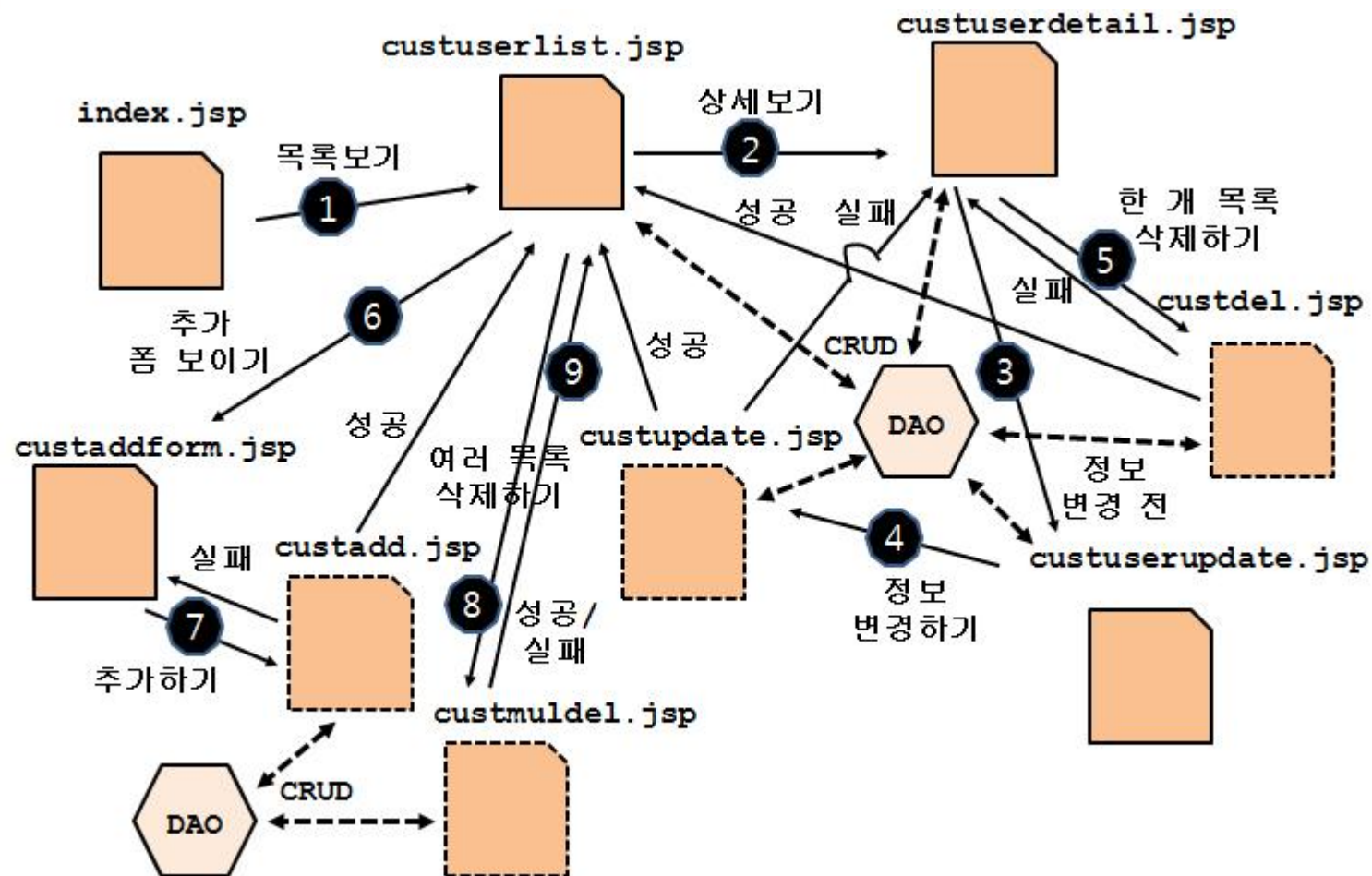
# @Only Servlet



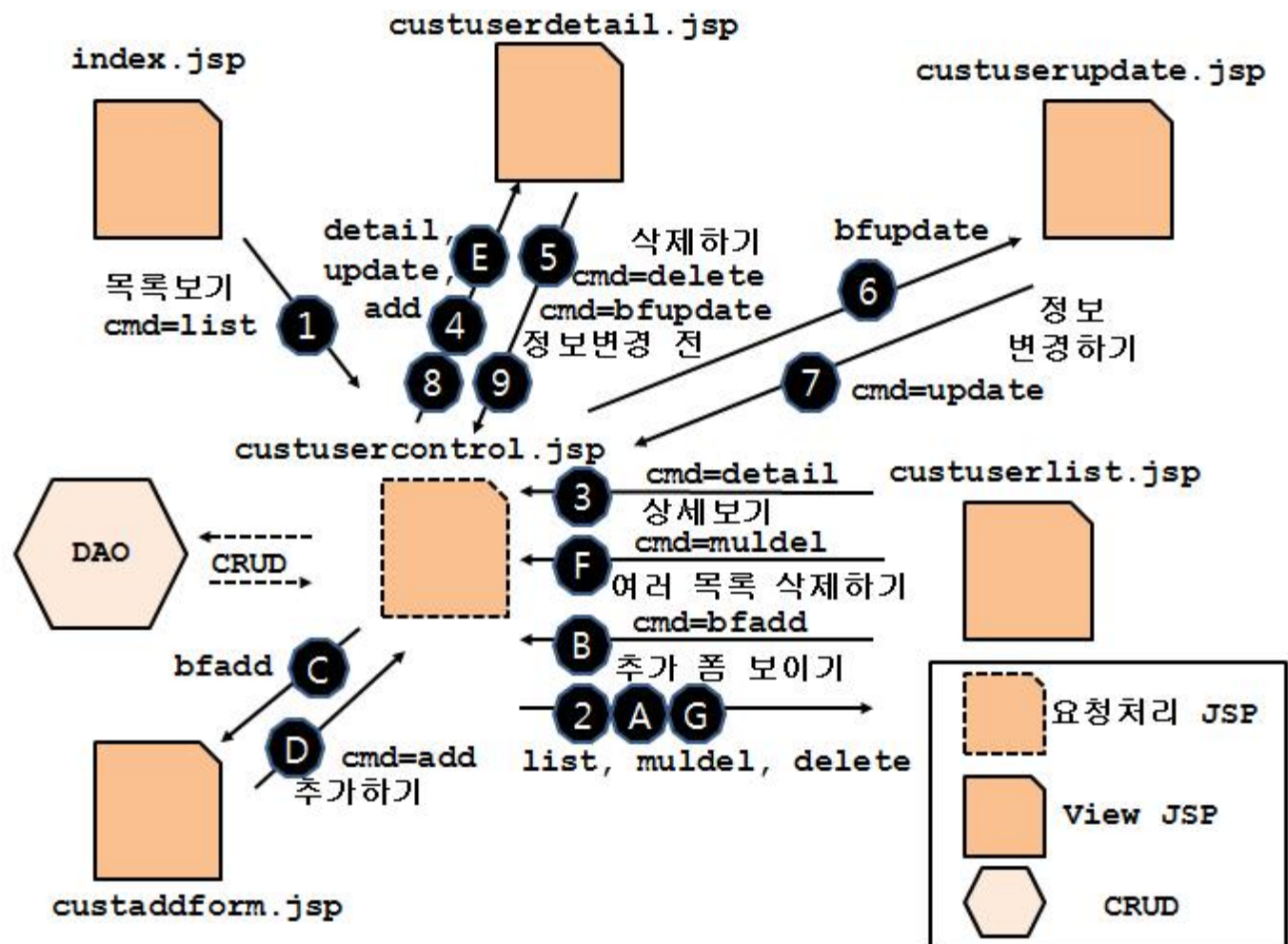
# @Spagetti



# @Separation



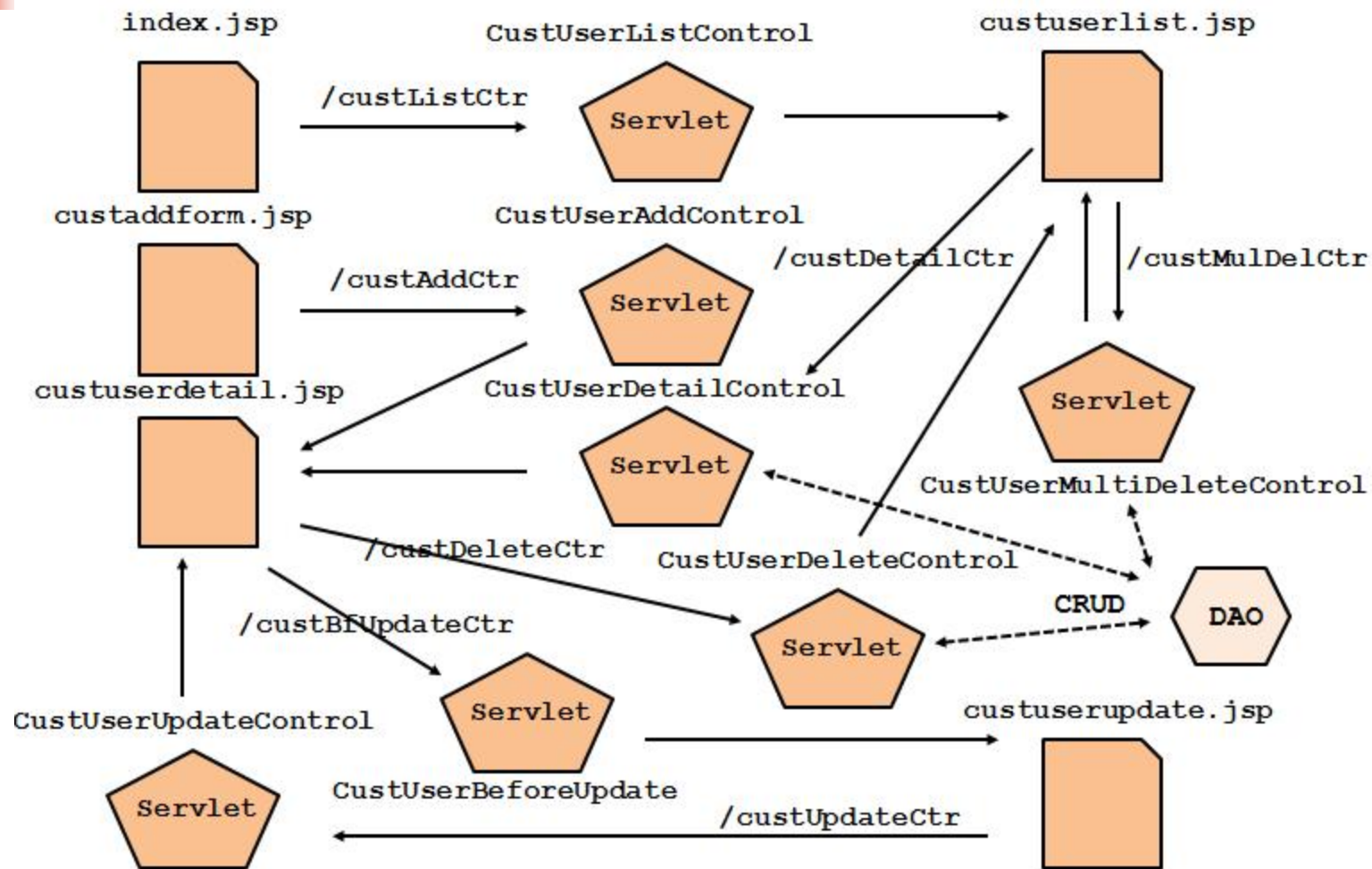
# @JSP Centric





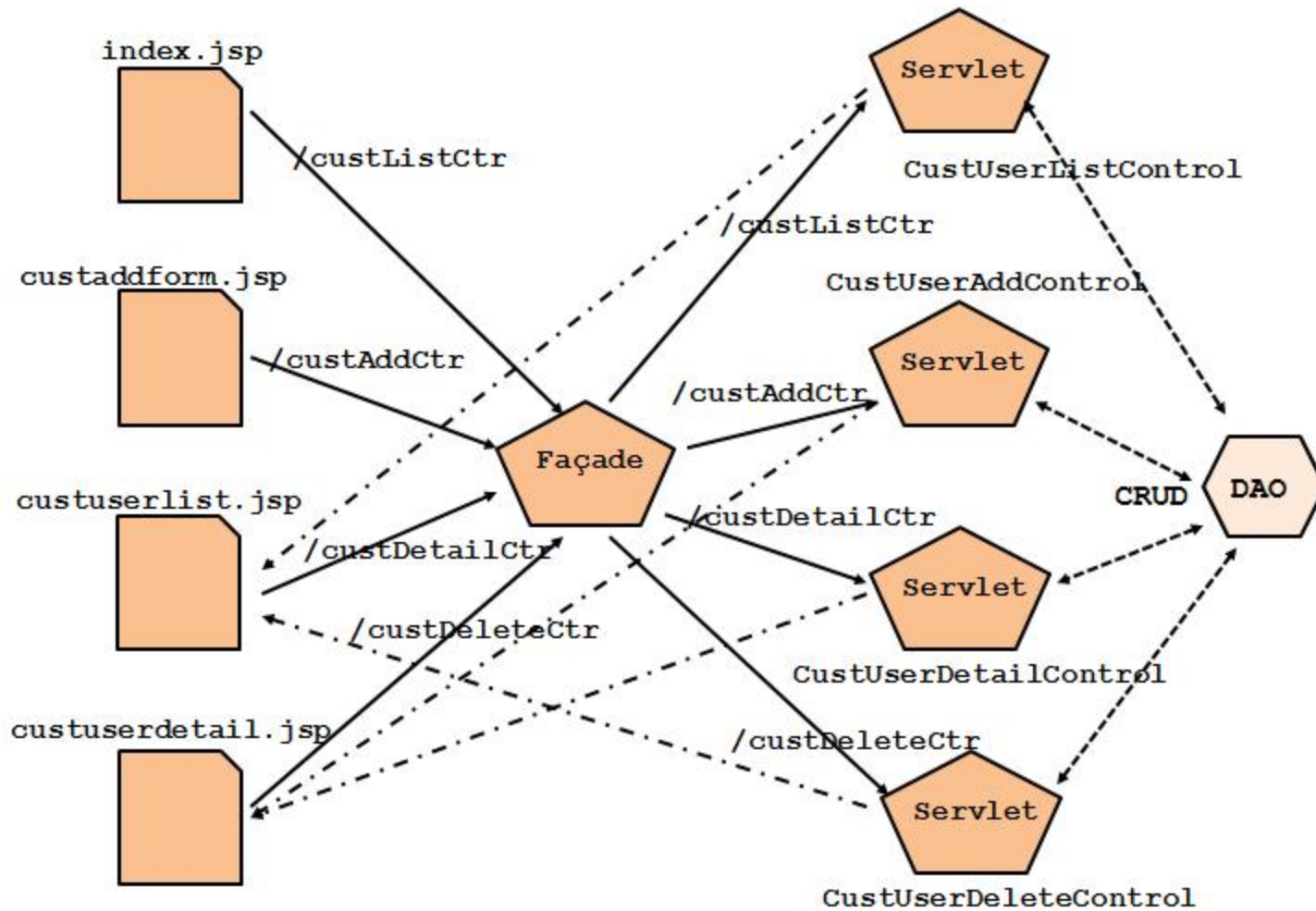


# @Multi Front Controller

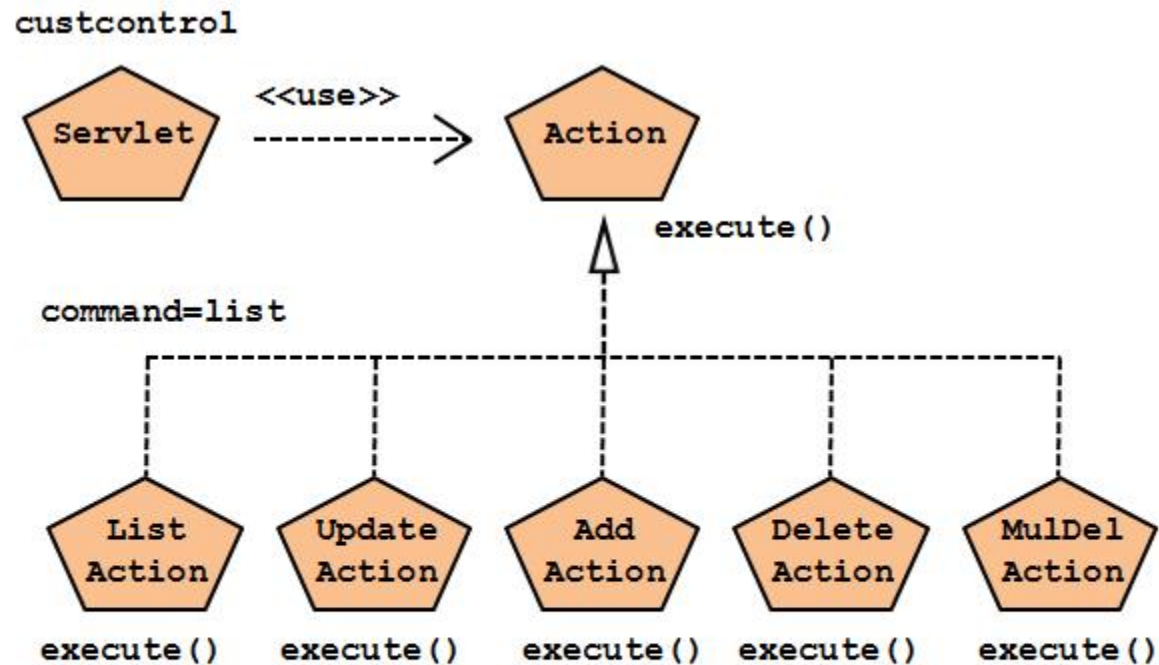




# @Facade

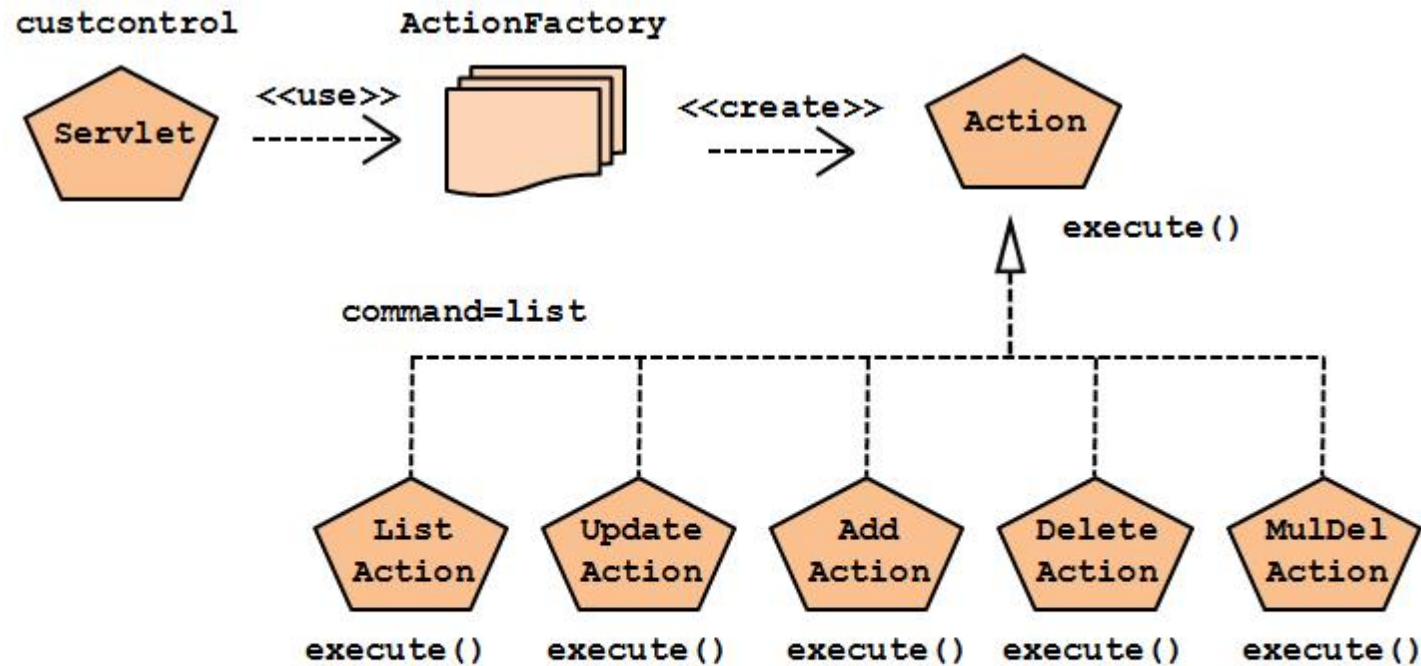


# @Action 계층구조



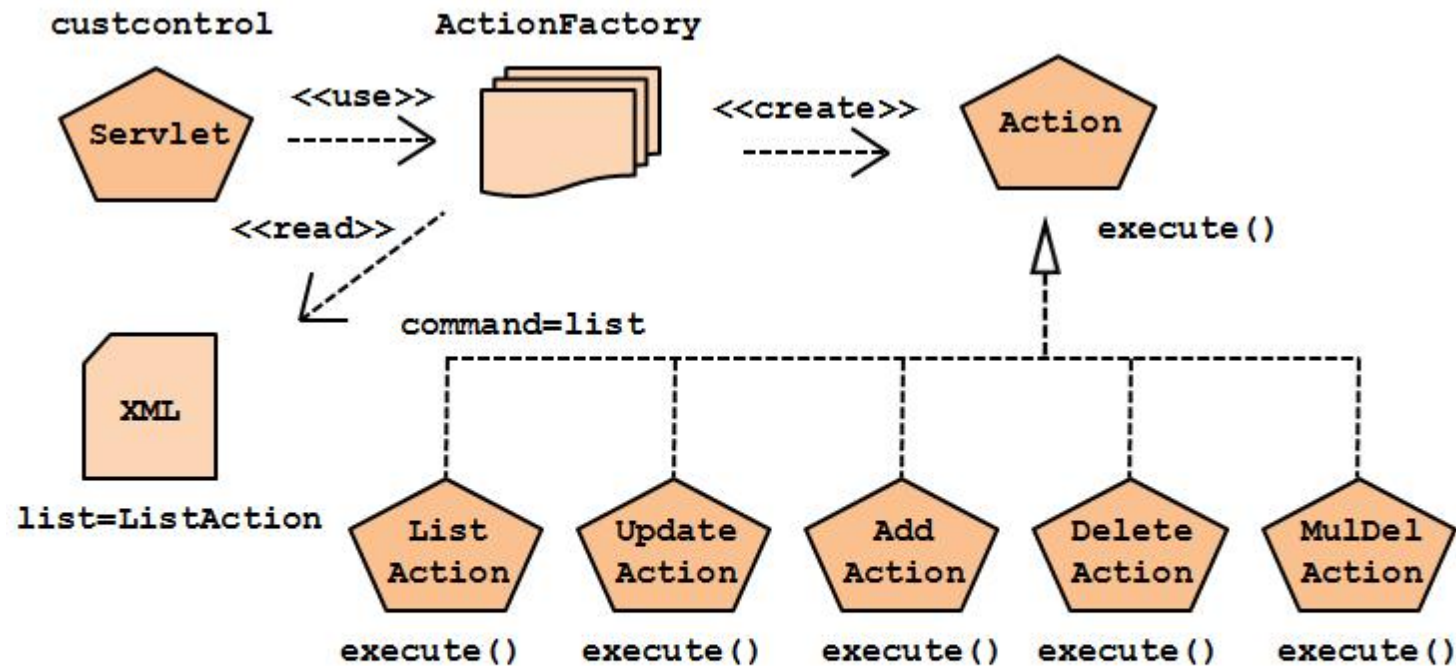
```
Action action = new ListAction();  
//ListAction 실행  
action.execuete();
```

# @Action Factory



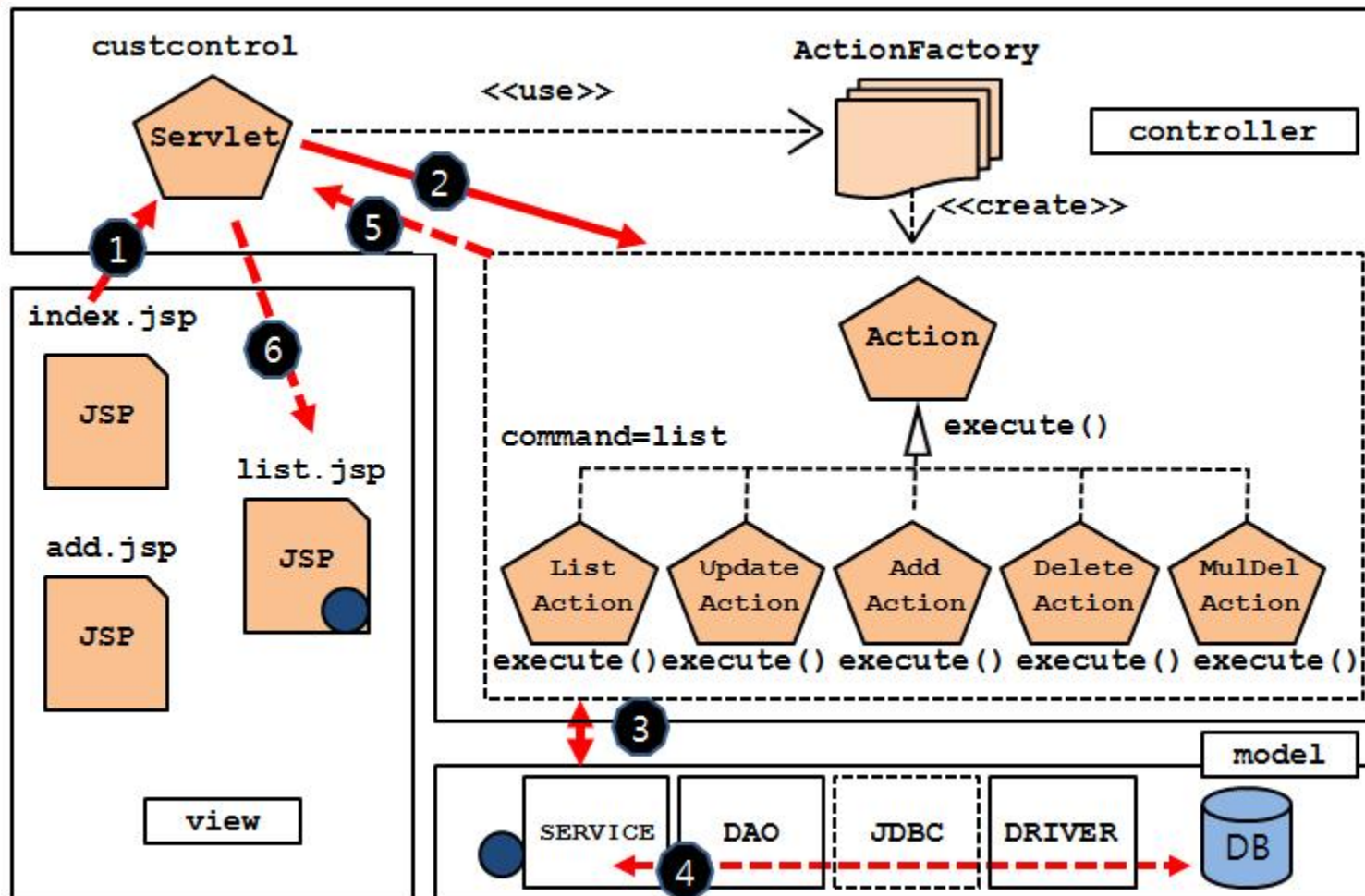
```
Action action = ActionFactory.getAction(command);
//ListAction 실행
action.execute();
```

# @Action Factory



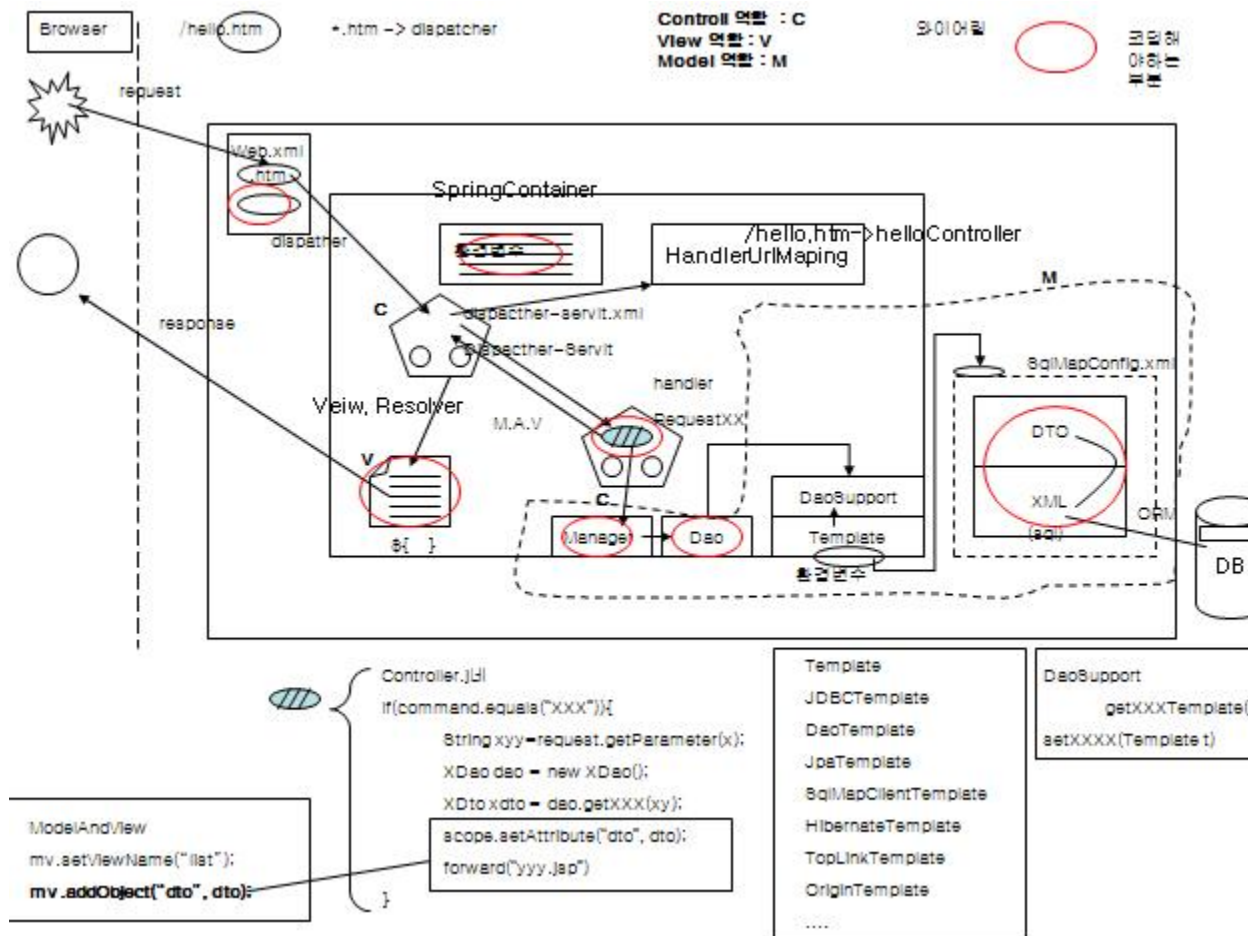
```
Action action = ActionFactory.getAction(command);
//ListAction 실행
action.execute();
```

# @Action Servlet





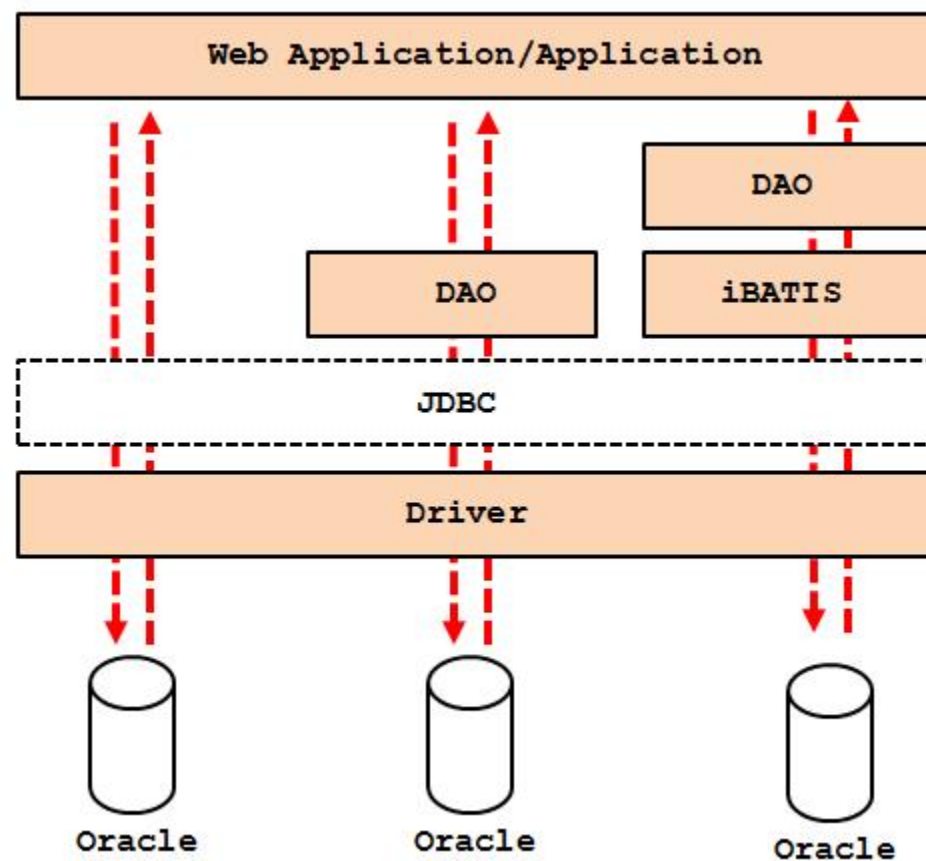
# @Spring Framework



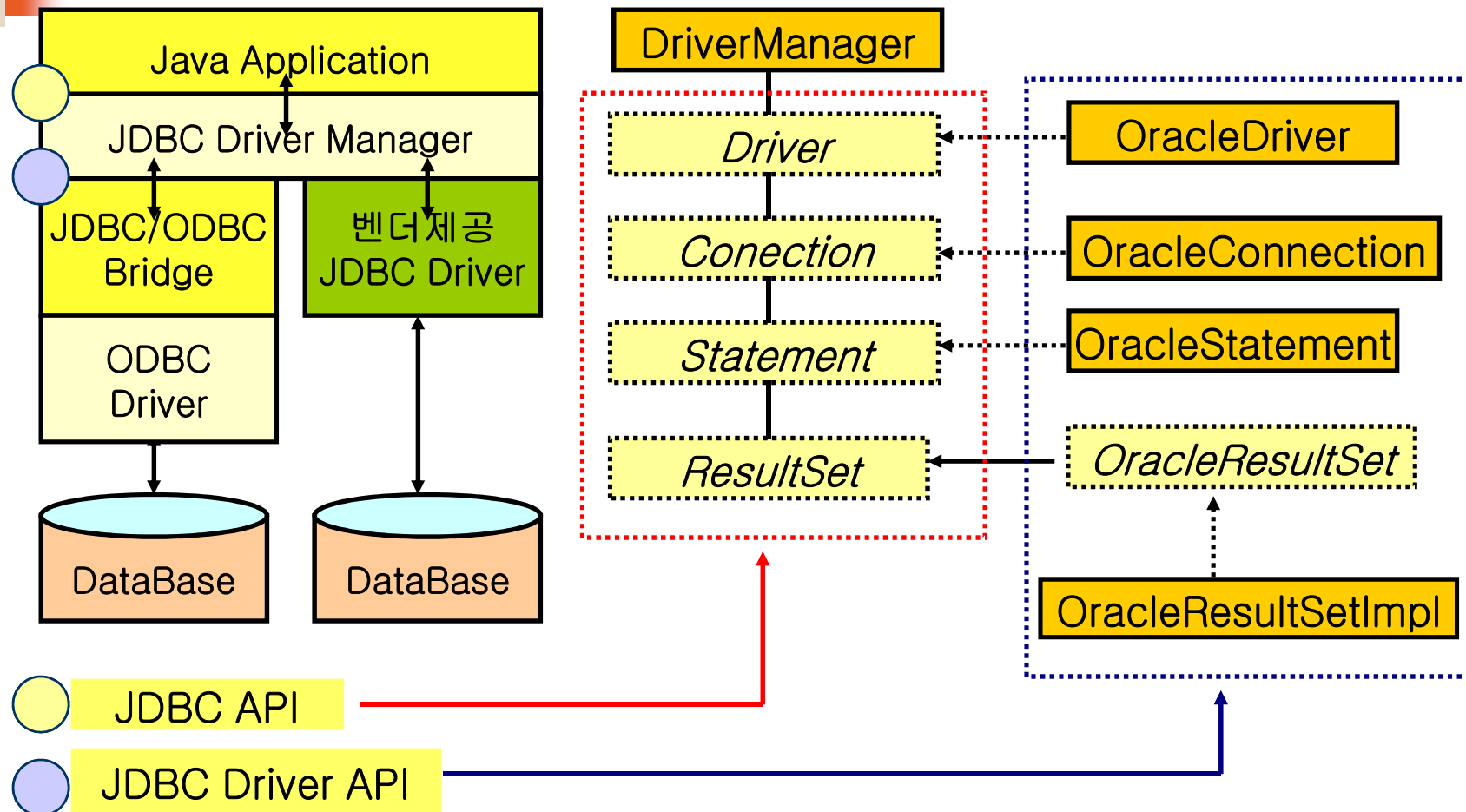


## 4. JDBC와 DL 프레임워크

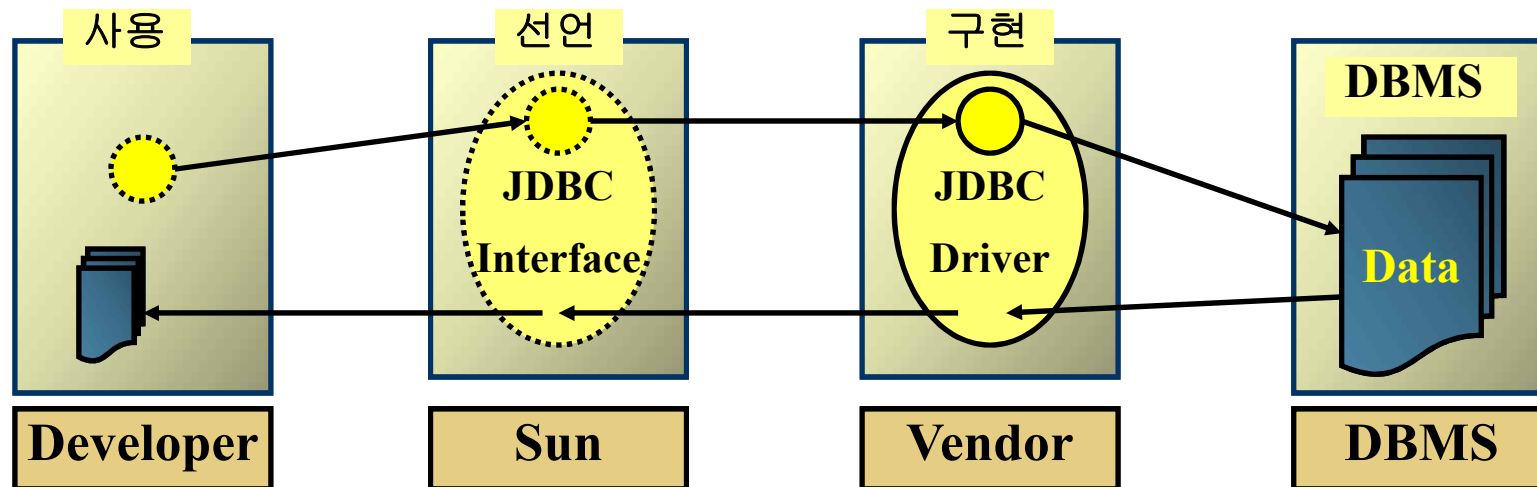
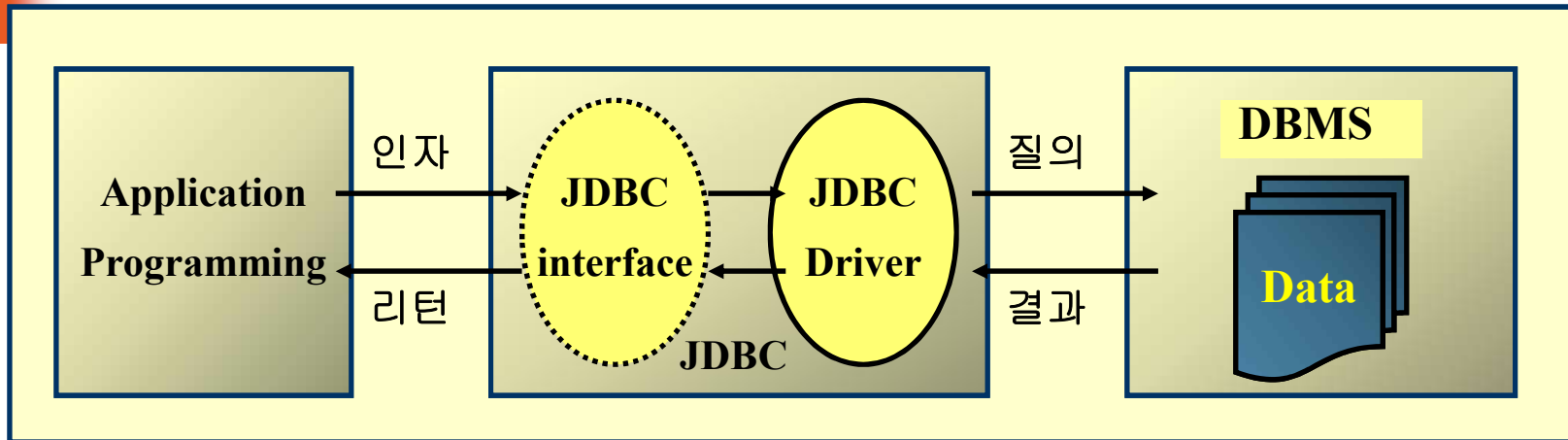
@DL



# @JDBC & Driver

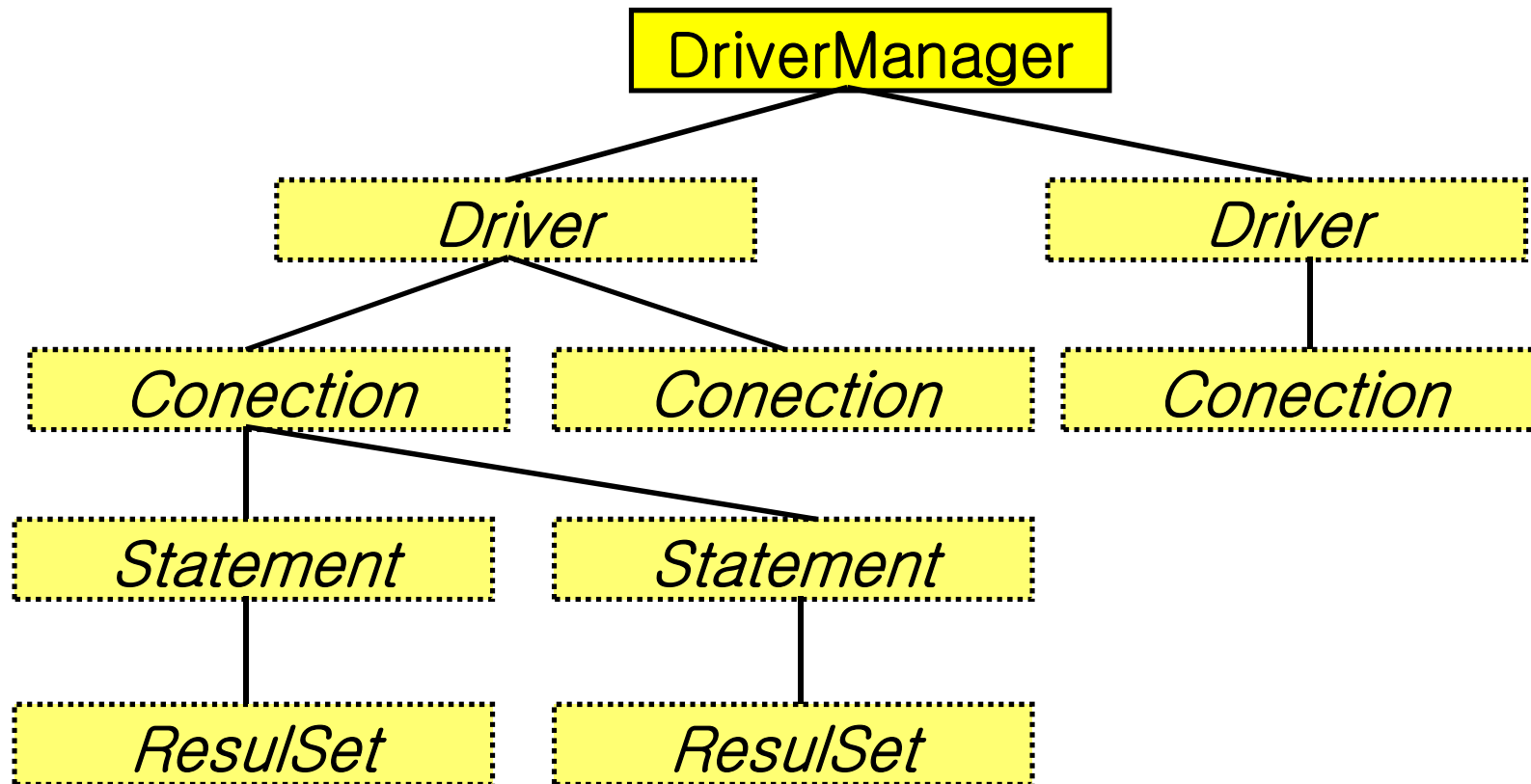


# @ JDBC 역할-사용, 선언, 구현 분리

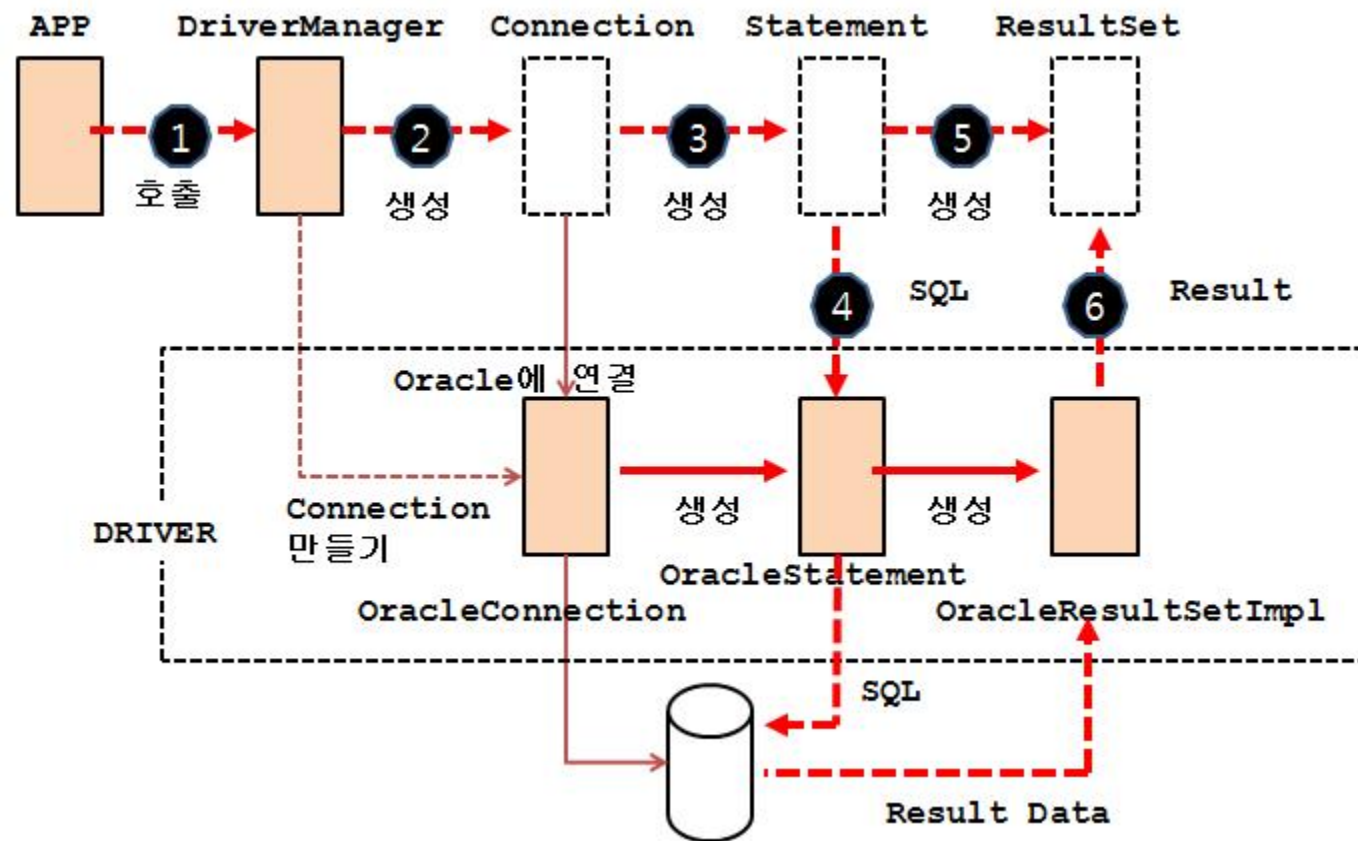




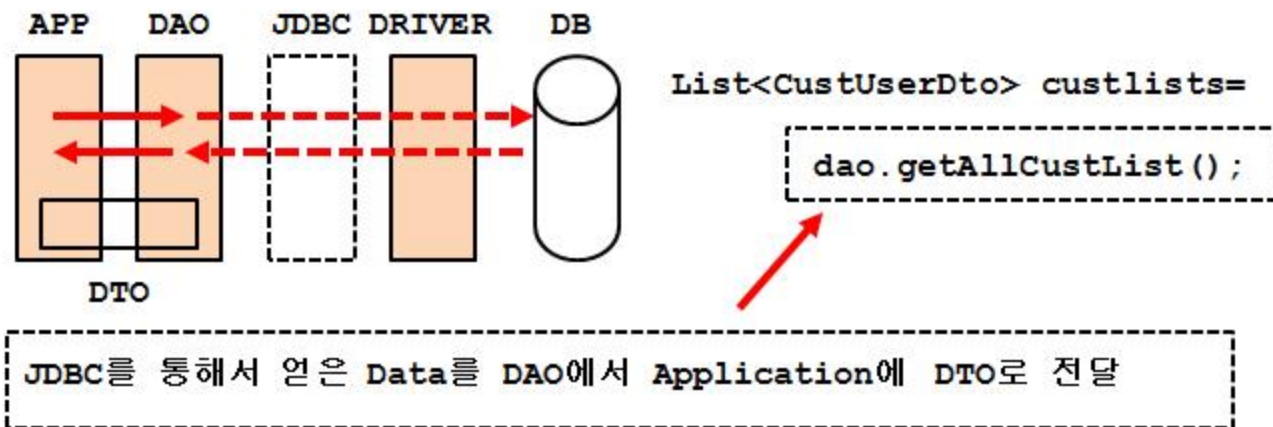
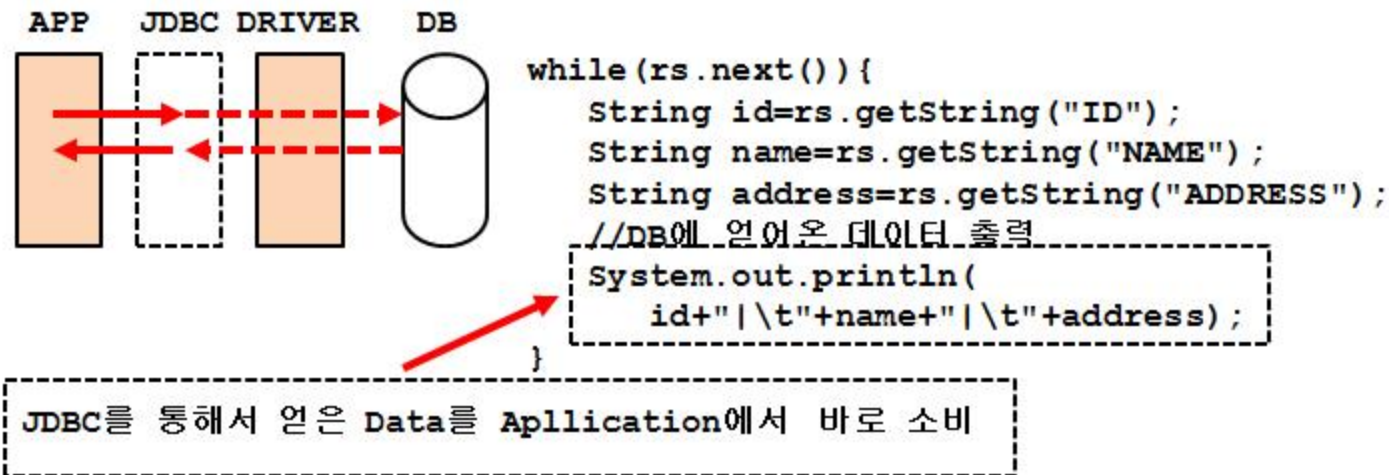
# @DriverManager



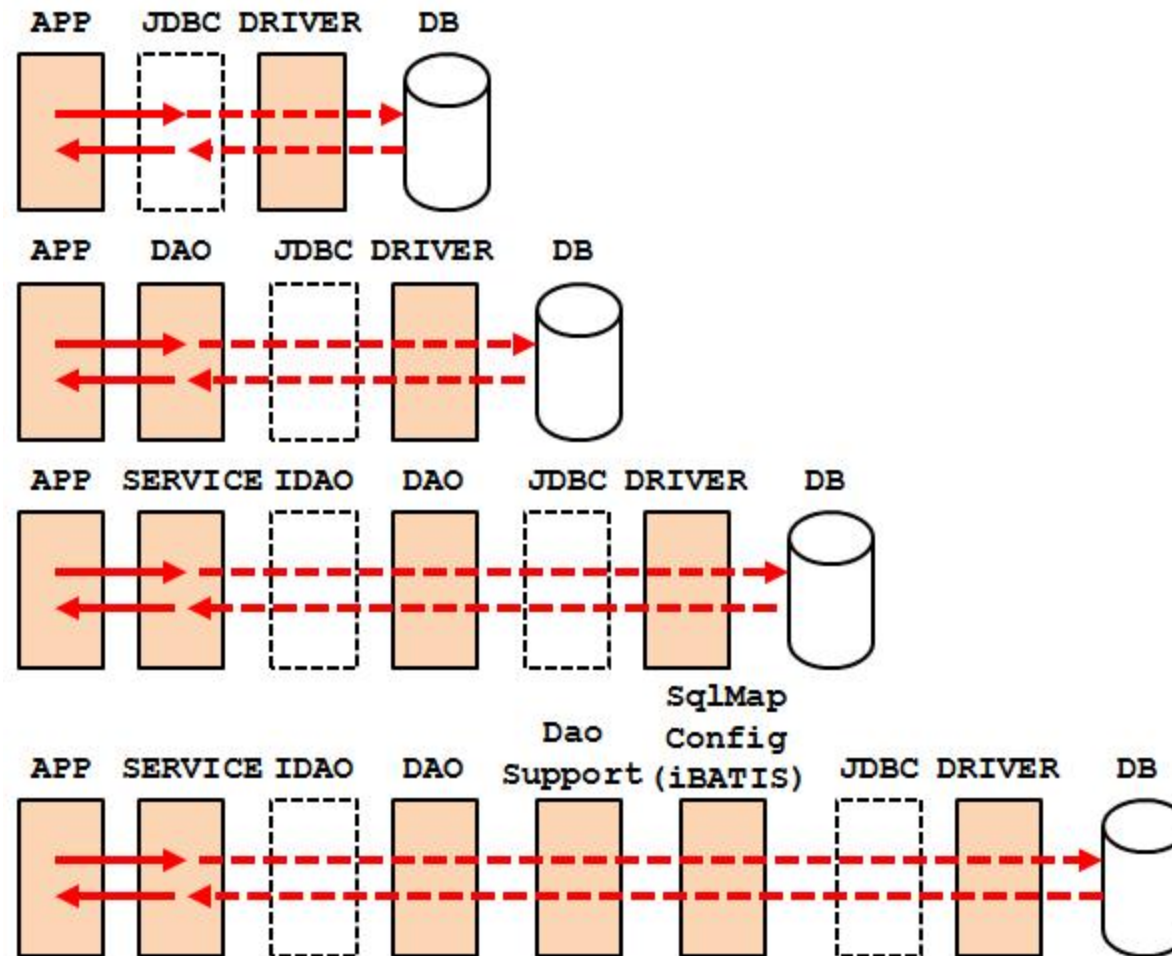
# @Oracle Driver



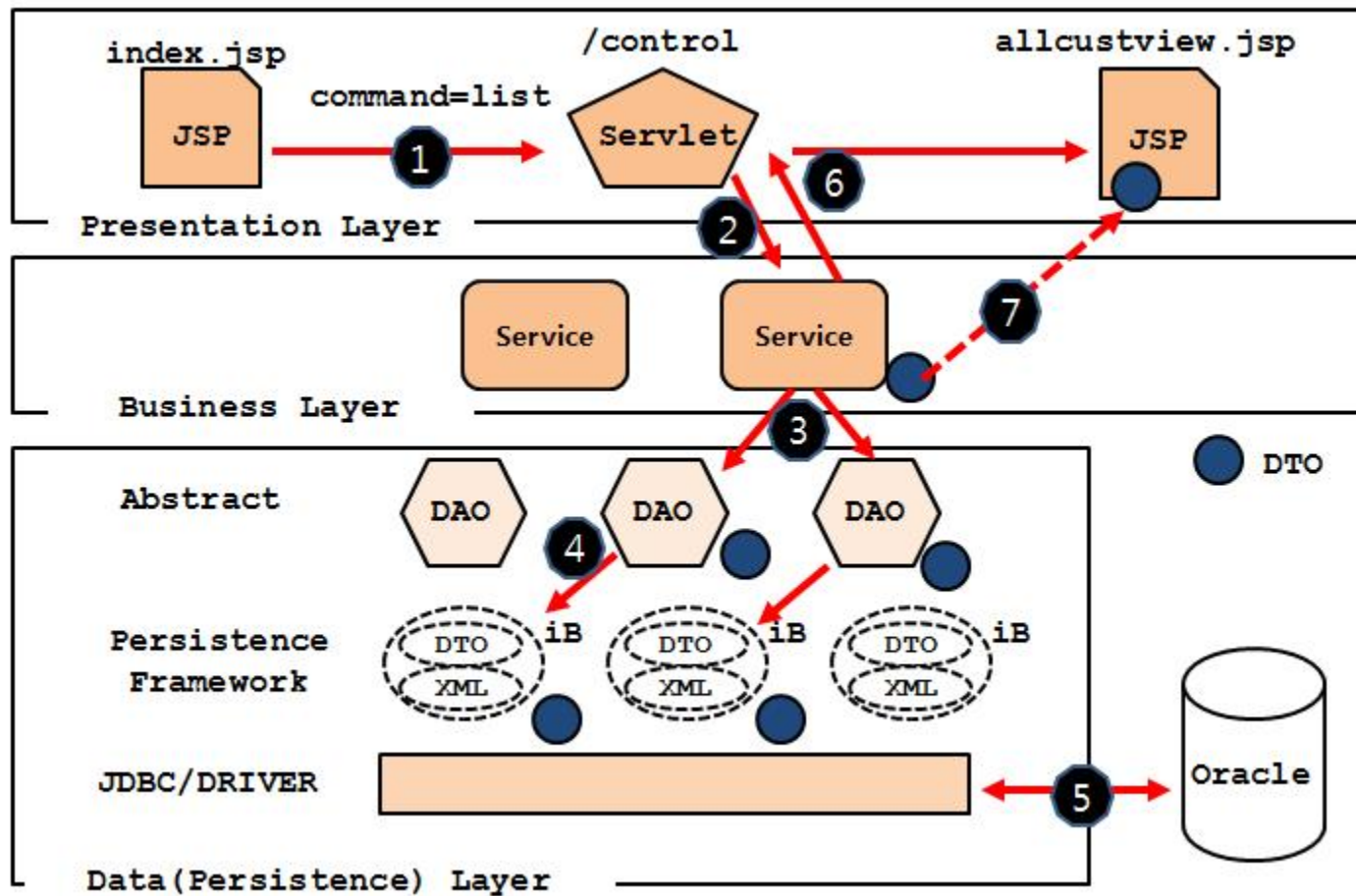
# @interface와 Driver



# @추상화 프로그래밍

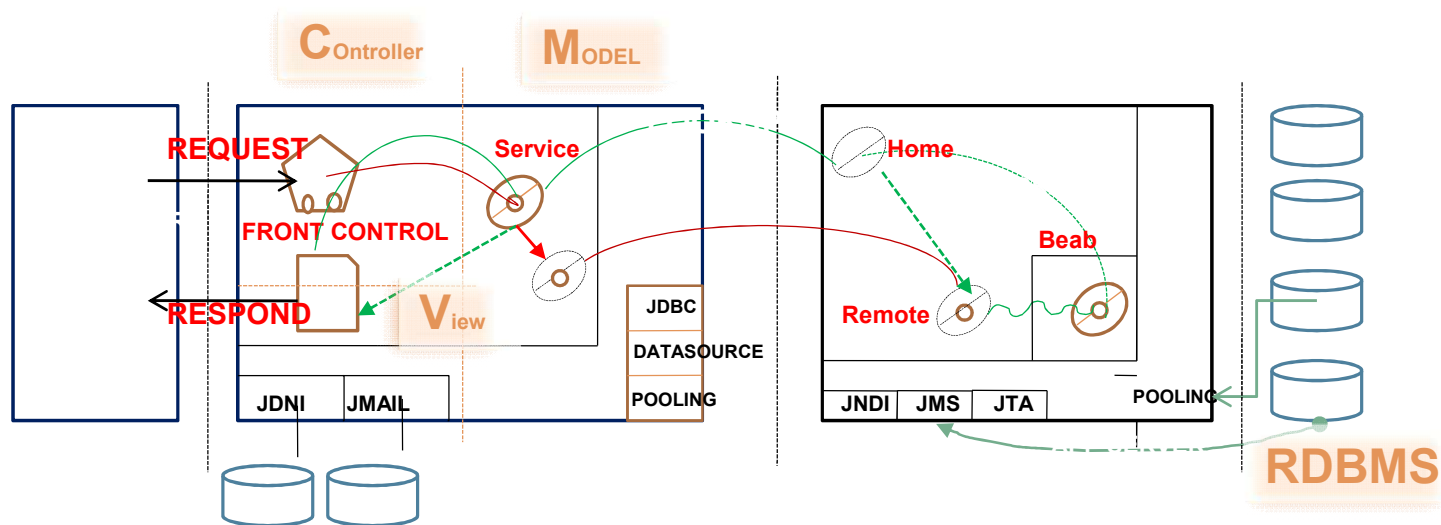


# @추상화 프로그래밍

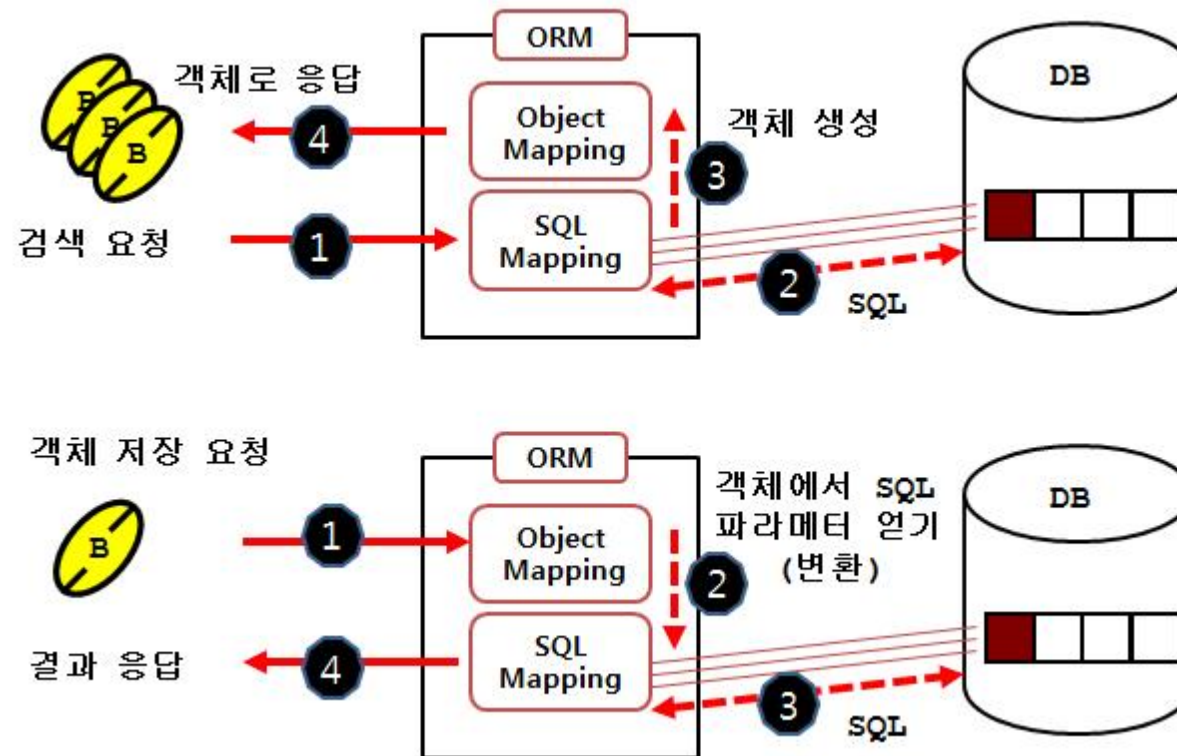




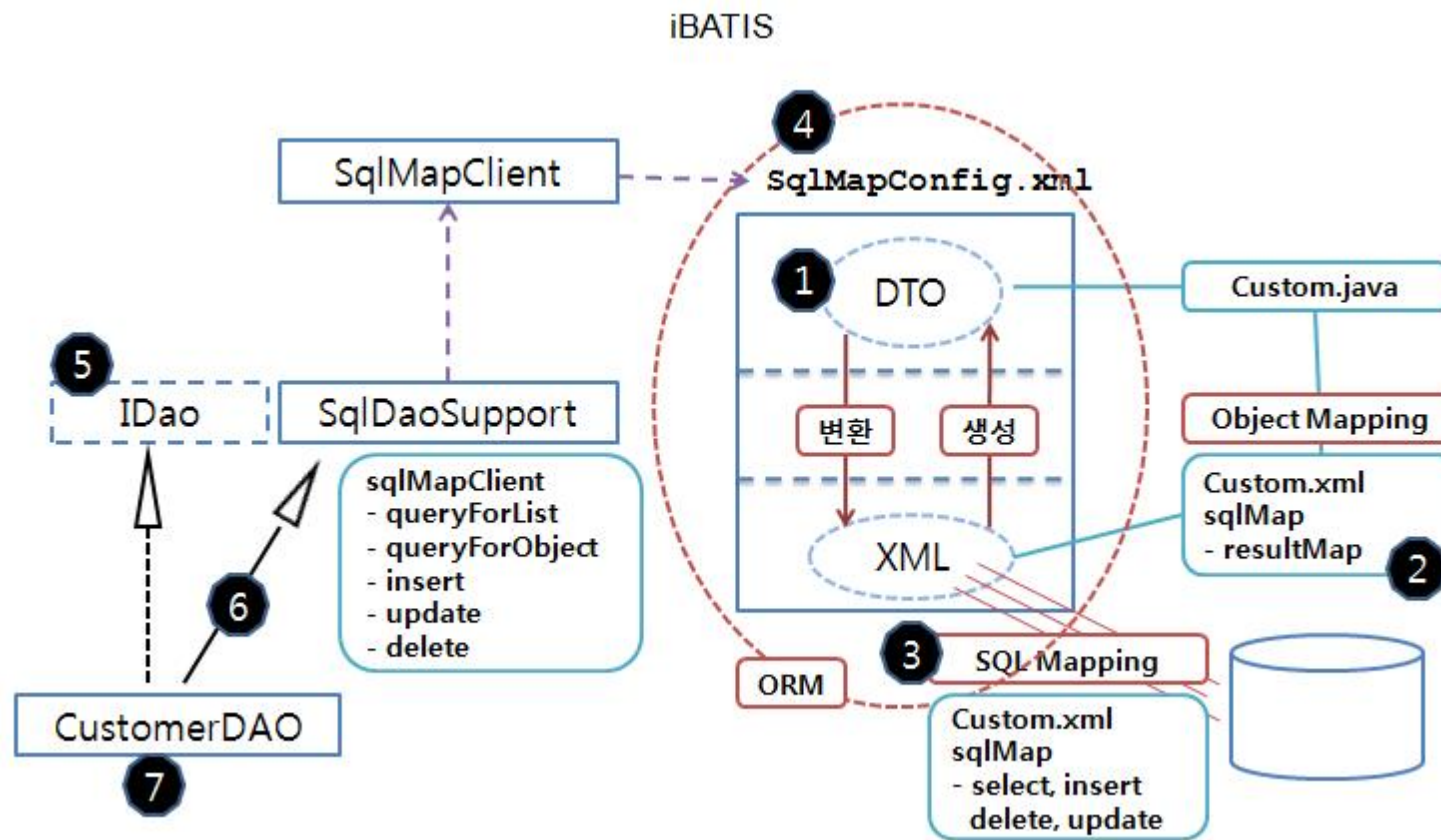
# @EJB MVC Model2



# @ORM



# @iBatis(MyBatis)





## 5. View(PL)과 AJAX



# @CSV, XML

---

## CSV

@@@Jungbo,15400,1100@@Hankyung,26200,1000@@@

## XML

```
<kosdaq>
  <stock>
    <stockname>Jungbo</stockname>
    <stockprice>15400</stockprice>
    <prevstockprice>1100</prevstockprice>
  </stock>
  <stock>
    <stockname>Hankyung</stockname>
    <stockprice>26200</stockprice>
    <prevstockprice>200</prevstockprice>
  </stock>
</kosdaq>
```





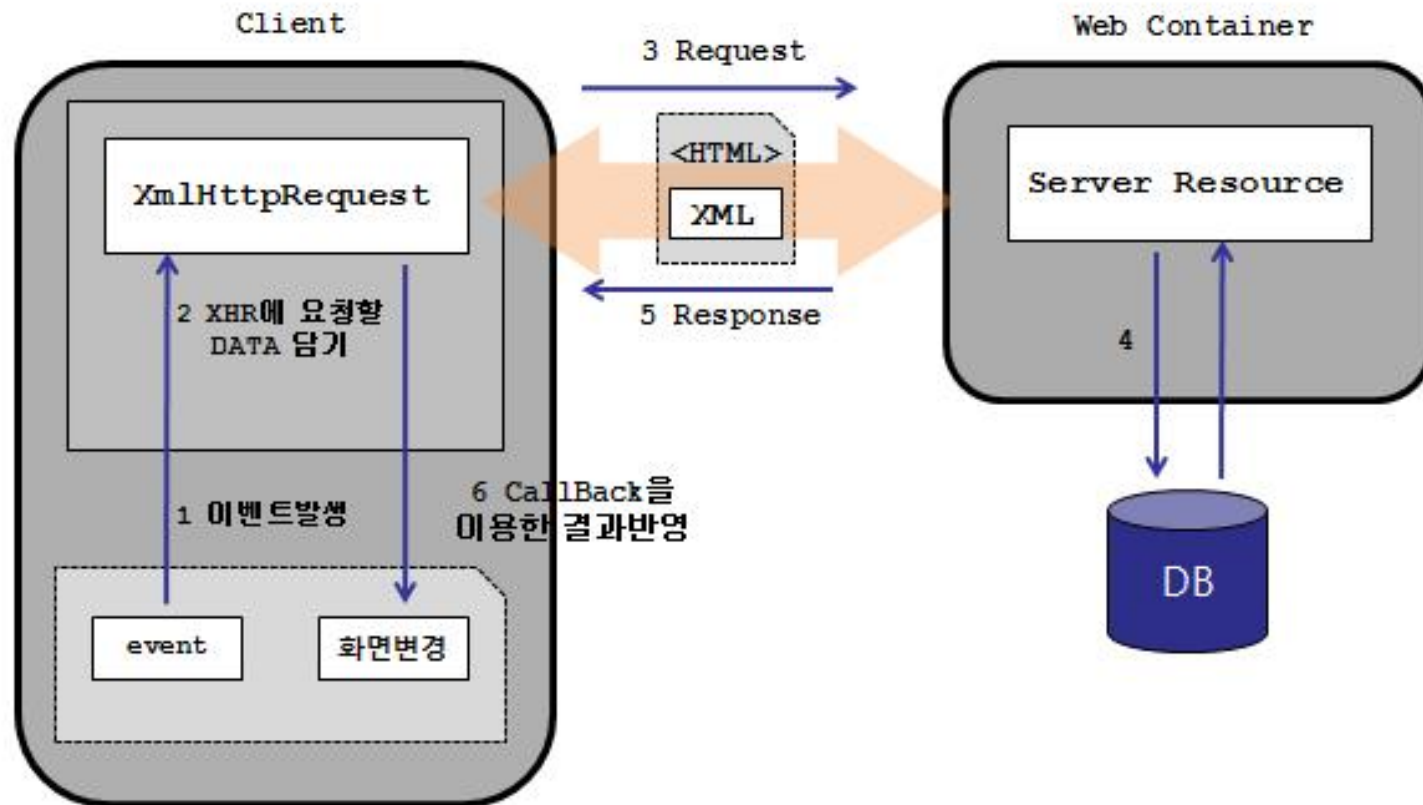
# @JSON

---

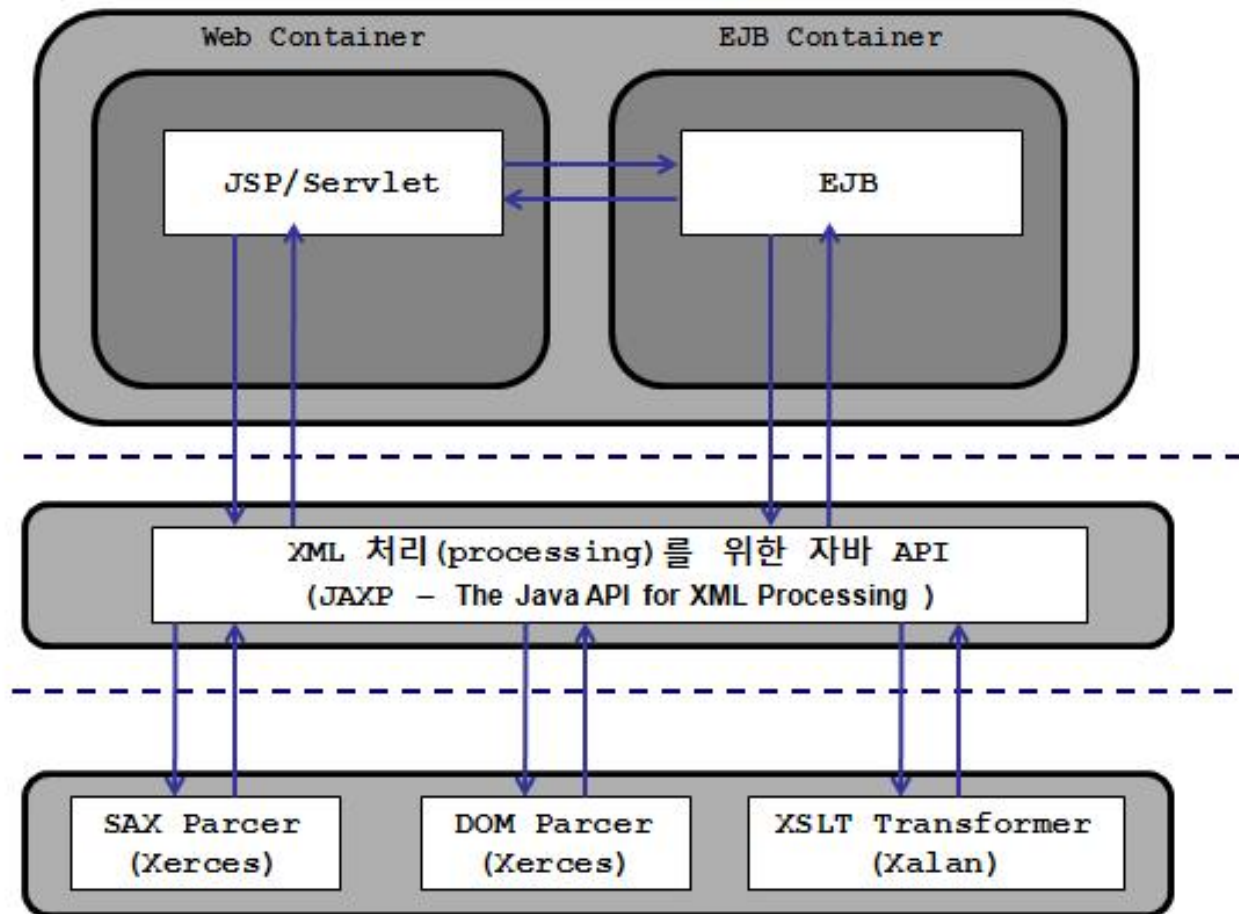
JSON

```
{“kosdaq”: {“stock” : [{“stockname”: “Jungbo”,  
                        “stockprice”: “15400”,  
                        “prevstockprice”: “1100”},  
                {“stockname”: “Hankyung”,  
                “stockprice”: “26200”,  
                “prevstockprice”: “1000”}  
            ]  
        }  
    }
```

# @AJAX 라운드트립



# @AJAX & XML 처리



# @AJAX

