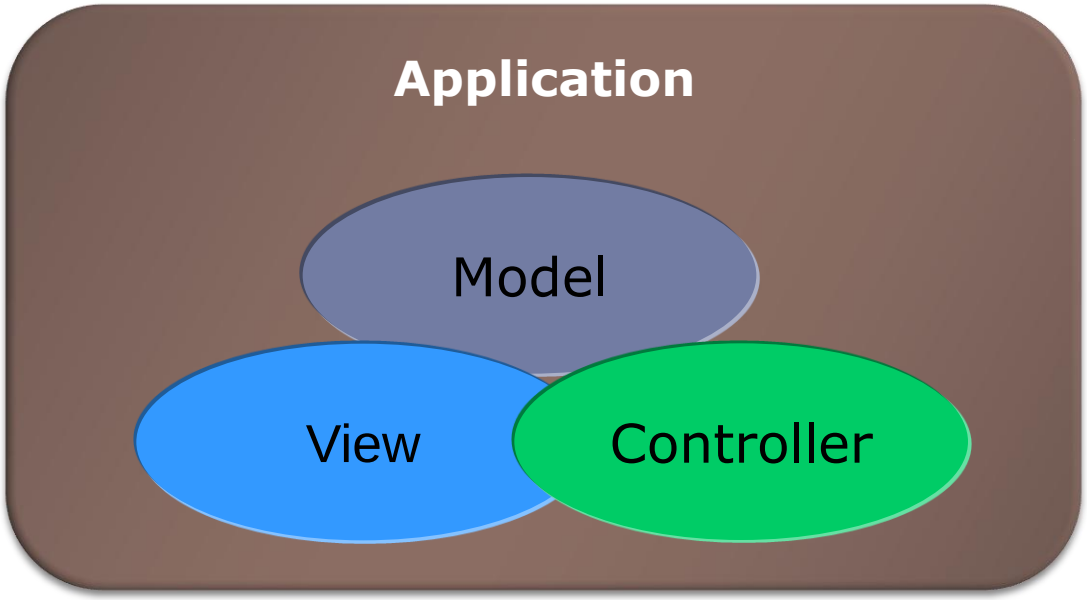
4. MVC Design Pattern

MVC Design Pattern

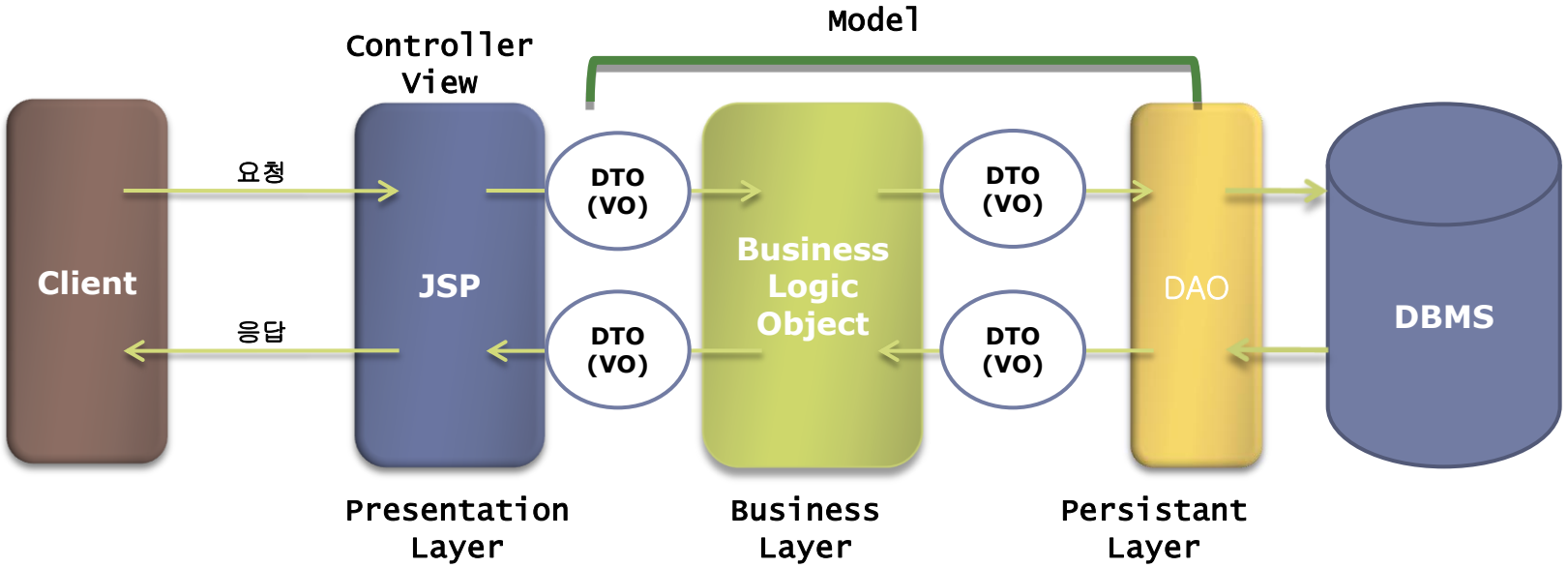
- 애플리케이션을 Model, View, Controller 3가지 영역으로 구분하고 영역간의 결합도를 최소화하여 개발하는 패턴



MVC Design Pattern

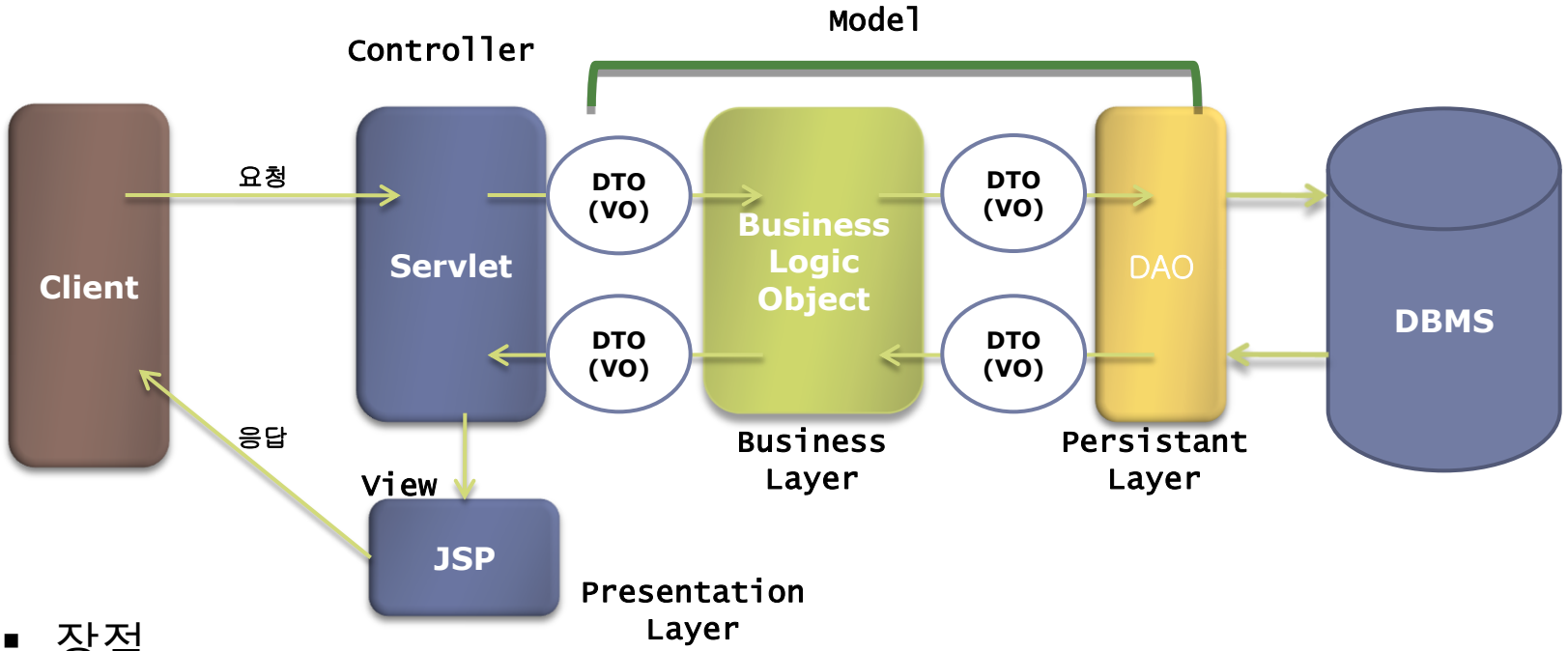
- Model
  - 애플리케이션에서 서비스하는 Business Logic , 사용되어지는 데이터를 다룬다.
- View
  - 사용자에게 보여질 Presentation Logic을 담당한다.
- Controller
  - 흐름을 관리하는 역할로써, 사용자의 요청을 받아 적절한 Model영역의 Business Logic을 호출하고 그 결과에 따른 적절한 View를 선택하여 보여질 수 있도록 한다.

MVC Model 1



- 장점
  - 쉬운 개발 방법으로 인한 개발 시간 단축
  - 성능면에서 유리
- 단점
  - View의 재사용이 어려움
  - Controller와 View의 코드가 섞여 있어 View의 변경이 용이하지 않음

MVC Model 2



- 장점
  - 비즈니스 로직의 결과에 따라 View 수정이 용이
  - Presentation 계층과 Business 계층의 독립성 보장
  - View의 재 사용성이 뛰어나고, 유지 보수 및 확장성이 유리

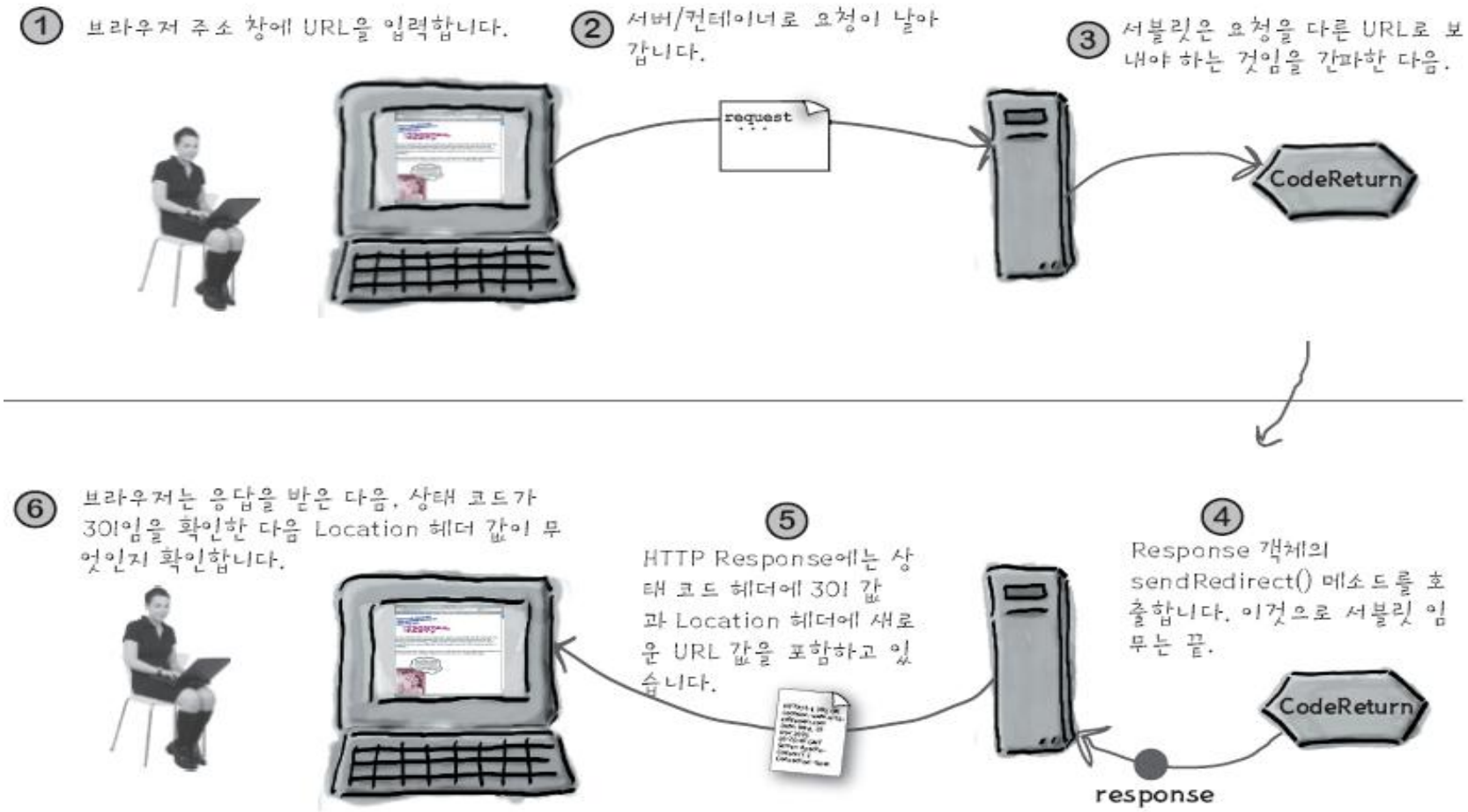
- 단점
  - 아키텍처에 대한 높은 이해 필요
  - Controller인 Servlet의 추가적인 개발이 필요

Page 이동 방식

- Redirect 방식
  - 서버측에서 이동할 페이지의 정보를 브라우저로 응답을 보낸다.
  - 브라우저는 응답정보에 있는 url로 다시 요청을 보낸다.
  - 따라서, 브라우저의 URL이 변화한다.

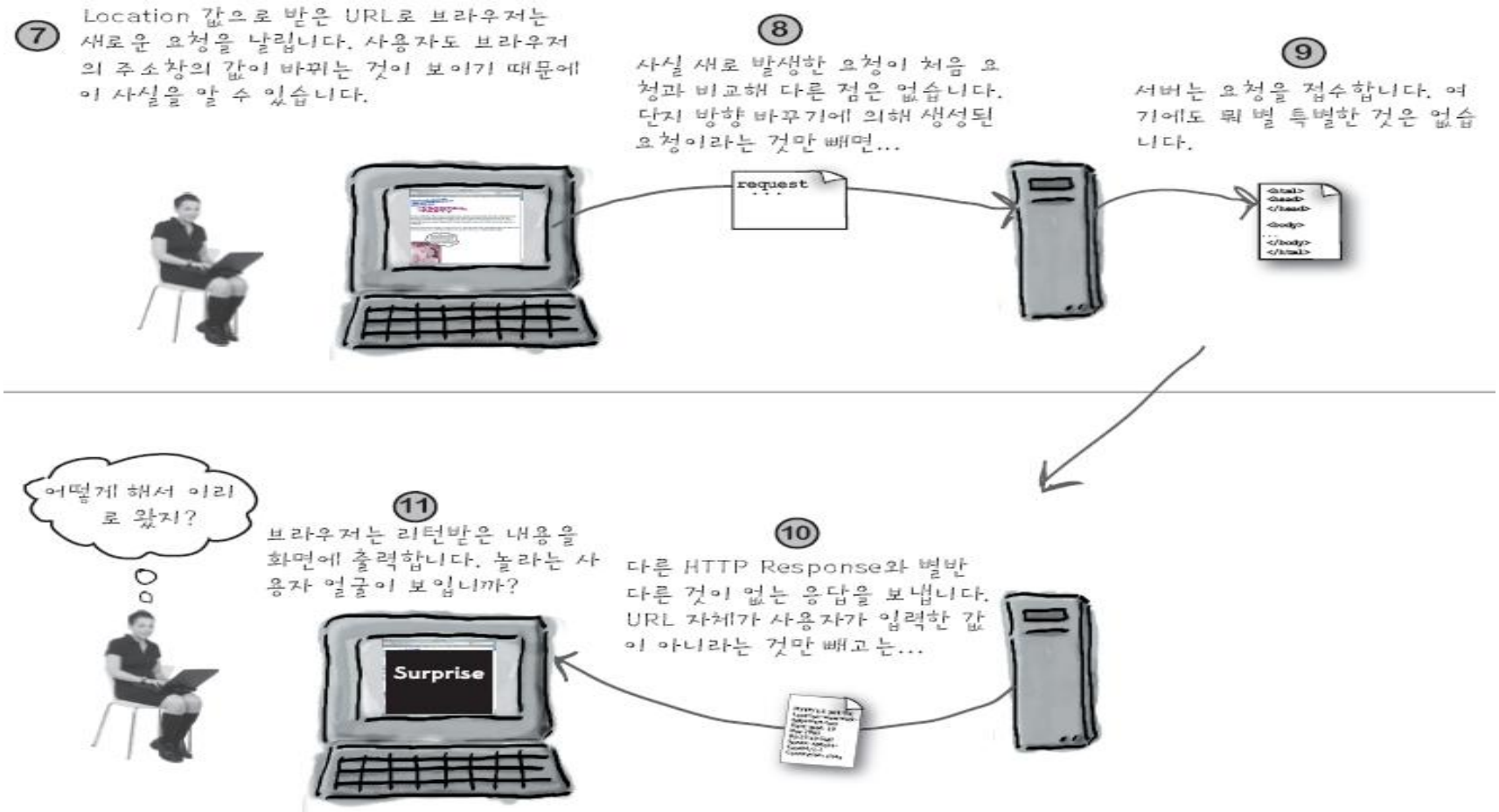
Page 이동 방식

- Redirect 방식



Page 이동 방식

- Redirect 방식



Page 이동 방식

- Forward 방식
  - 서버측에서 이동할 페이지의 정보를 브라우저로 응답을 보내지 않고 직접 페이지를 이동 시킨다.
  - 이때, 이동되어지는 페이지는 기존 요청,응답을 그대로 전달받는다.
  - 따라서, 브라우저의 URL은 변화없다.

Page 이동 방식

- Forward 방식