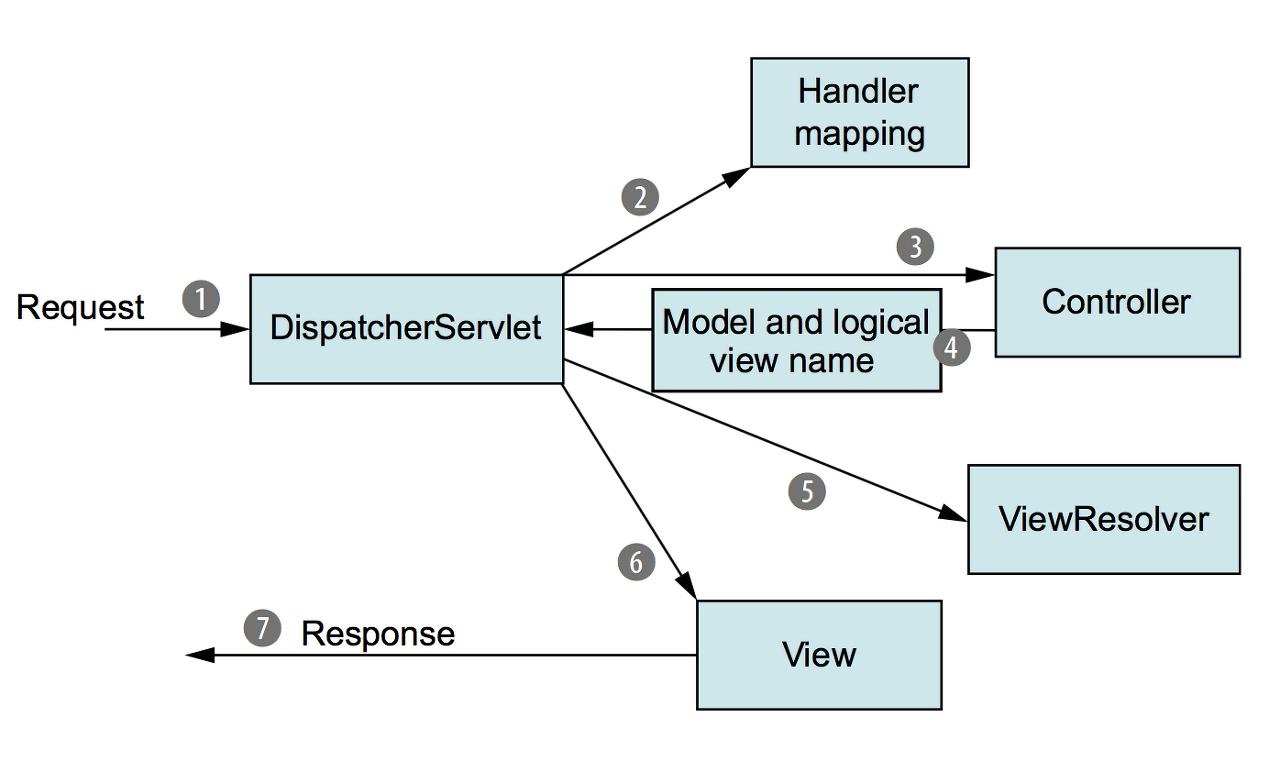
**Web Server**

* (하드웨어적 의미) Web Server 소프트웨어 + Website(HTML, CSS , JavaScript 파일)의 컴포넌트 파일들을 저장하는 하드웨어
* (소프트웨어적 의미) Client요청 해석 및 알맞은 정적 응답을 보내주는 소프트웨어.
* Client -> HTTP 프로토콜 사용 요청 -> (웹서버: http 요청 해석. 그 다음, 요청에 대한 정적 데이터 응답
* (장점) 처리 속도가 빠르며 트래픽의 과부하를 잘 처리할 수 있다는 장점이 있다. (단순 정적데이터를 반환하기 때문)
* (초창기) 정적 데이터에 대한 수요가 높았기 때문에 기능적으로 어플리케이션 서버를 따로 나누지 않고 **웹 서버**라는 개념을 통칭해서 사용
* (현재) 더 많은 서비스 기능을 제공 + 서버의 부하를 줄이기 위해 따로 서버를 나누게 되었다 **-> 어플리케이션 서버 등장**

**어플리케이션 서버(Application Server)**

* 동적데이터를 처리하며, 주로 데이터베이스 서버와 같이 수행
* (동적 데이터란?) 상황에 따라 비즈니스 로직(영속 저장, 계산, 분기 등) 에 따라 서버에서 그때그때 달라지는 데이터

**WAS(Web Application Server)**

* web application server의 약자 (Application서버의 웹에 특화된 한 종류)
* 그러나 보통 J2EE(자바로 어플리케이션을 만드는데, 필요한 기술 <https://gyrfalcon.tistory.com/entry/J2EE>)스펙을 이용해서 서블릿이나 JSP로 작성된 어플리케이션 소프트웨어
* 💡**현 Jakarta\_EE(구 J2EE(Java Enterprise Edition))의 구성요소**
* **Servlet** : 클라이언트의 요청 처리 및 그 결과를 반환하는 Servlet 클레스의 구현 규칙을 지킨 자바 웹 프로그래밍 기술(ex. MVC패턴에서의 dispatcher서블릿)
* **JSP(Java Server Pages)**: HTML이나 Java 코드를 써서 사용자에게 정보를 보여 줍니다. JSP가 처음 실행될 때 Servlet 엔진이 이것을 Servlet으로 컴파일시켜서 내부적으로는 Servlet으로 동작합니다.
* **EJB(Enterprise Java Beans)** : Java에서 제공하는 분산 컴포넌트 기술로 비즈니스 로직이나 데이터, 메시지를 처리할 수 있습니다.
* **Remote Method Invocation(RMI)**: 프록시를 써서 원격에 있는 Java 객체의 메소드를 실행시키기 위한기술입니다.
* **Java Naming Directory Interface(JNDI)**: 자바 기술로 만들어진 객체에 이름을 붙여 찾을 수 있도록 단일한 인터페이스를 제공합니다.
* **Java Database Connector(JDBC)**: 여러 종류의 데이터베이스 시스템에 접근하는 단일한 인터페이스를 제공합니다. 각각의 데이터베이스에 맞는 JDBC 드라이버가 있어야 합니다.
* **Java Connector Architecture(JCA)**: 다른 기종 플랫폼을 통합할 수 있도록 플랫폼 독립적인 인터페이스를 제공합니다.
* **Java Message Service (JMS)**: 여러 가지 메시징 시스템에 대한 플랫폼 독립적인 인터페이스를 제공합니다.

**웹 컨테이너(Web Container)**

**동적 데이터 생성**

**CGI(Common Gateway Interface)**

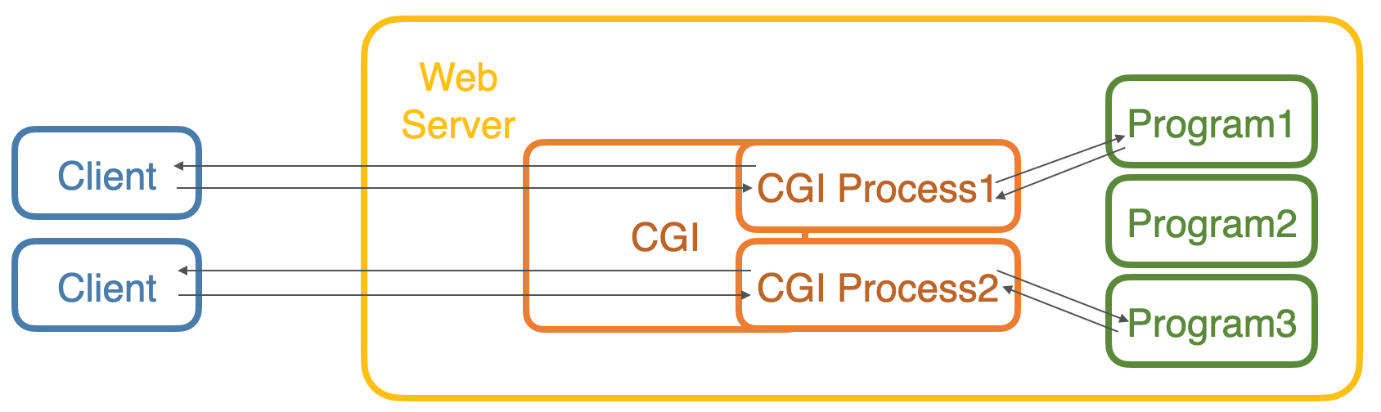
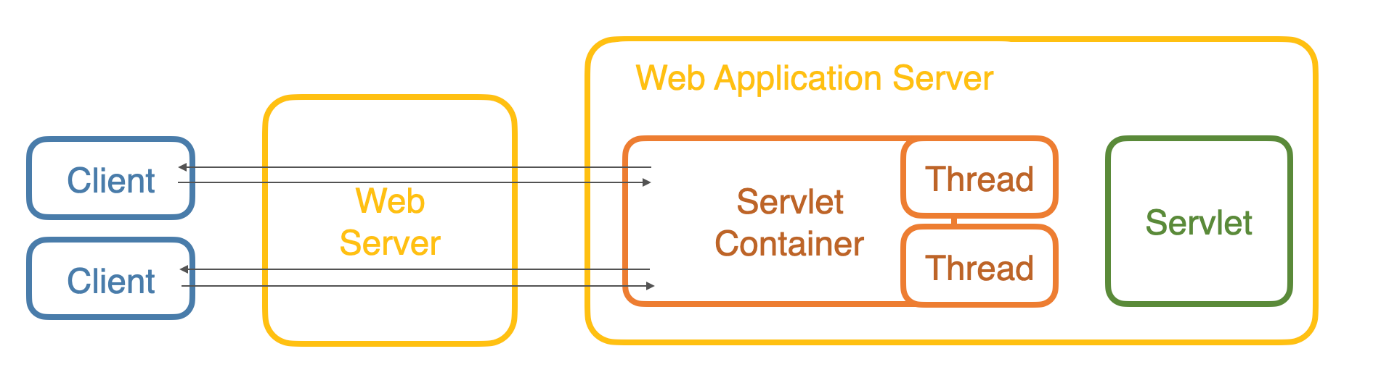
* 서버와 외부 프로그램 사이에서 정보를 주고 받는 표준 스펙 Interface
* 자바에서는 이 역할을 서블릿이 담당함

**서블릿**

* “클라이언트의 요청을 받아 동적 데이터를 제공하는(CGI)” 자바 클래스
* 쓰레드 사용 동작

**CGI와 서블릿의 차이점**

* (포괄적 개념)CGI는 매 요청이 들어올 때마다 프로세스가 생성되고 각각의 CGI 구현체를 통해 처리한다.
* (자바에서 CGI역할을 위해 만든 클래스)서블릿은 각 요청마다 스레드가 생성되거나 스레드 풀에서 기존의 스레드를 사용하여 동작한다.(보통은 스레드 풀 사용!)  
  -> 쓰레드가 무거운 프로세스보다 생성되는데 시간 상, 공간 상 이득

CGI을 이용한 동적 데이터 처리Servlet을 이용한 동적 데이터 처리

**웹 컨테이너(Web Container)**

**서블릿 컨테이너**

* 웹컨테이너와 같은 말이다
* was = 웹서버+웹컨테이너
* 서블릿을 생성, 소멸 및 관리해주는 곳
* 서블릿 컨테이너 내에 쓰레드 풀이 존재함 -> 서블릿이 실행될 멀티 쓰레드또한 관리해준다