



하이티센 3주차

웹서버란 무엇인가

was란 무엇인가

웹서버랑 was의 차이

웹서버끼리의 비교 : 아파치 엔진엑스

was끼리 비교 : 톰캣 jboss

+서블릿 컨테이너

웹 서버란 무엇인가?

웹의 동작원리

클라이언트(사용자)가 웹 브라우저(크롬, 익스플로러 등등)를 사용하여 서버에 요청을 보내고 서버가 해당 요청에 대해 응답하는 것이 Web 기본 동작이다

[그림]

기본 동작을 그림으로 간단하게 표현하면 다음과 같다

웹브라우저에서 검색하고 주문을 하고 페이지를 새로고침한다.

이런 행위들은 사실 매번 서버에게 새로운 요청을 하고 그 응답에 대한 결과물로 웹 브라우저의 화면을 접한다

[그림]

Web Server

웹 서버는 말 그대로 작성된 html 페이지 등을 네트워크망에 종속되지 않고, 웹서비스를 할 수 있도록 하는 어플리케이션이다.

[그림]

- 웹서버는 소프트웨어와 하드웨어로 구분된다.

1) 소프트웨어 웹서버 : 웹 브라우저 클라이언트로부터 HTTP 요청을 받아들이고, 정적인 콘텐츠 (HTML, .jpeg, .css 문서)와 같은 웹 페이지에서 흔히 찾아 볼 수 있는 자료 콘텐츠에 따라 HTTP 에 반응하는 컴퓨터 프로그램

2) 하드웨어 웹서버 : 위에 언급한 기능을 제공하는 컴퓨터 프로그램을 실행하는 컴퓨터

- 클라이언트가 서버가 준 HTML 을 볼 때 안에 삽입된 이미지를 한 번에 가져온 것이 아니라 우선 HTML 문서를 먼저 받고 그에 맞게 필요한 이미지 파일들을 다시 서버에 요청하면 그 때 이미지 파일들을 받아온다

[그림]

- Web Server 의 기능

- HTTP 프로토콜을 기반으로 하여 클라이언트(웹 브라우저 또는 웹 크롤러) 의 요청을 서비스하는 기능을 담당한다.

- 요청에 따라 다음의 기능 중 적절하게 선택하여 수행한다.

- * 기능 1) 정적인 콘텐츠 제공

WAS 를 거치지 않고 바로 자원을 제공한다.

- * 기능 2) 동적인 콘텐츠 제공을 위한 요청 전달

클라이언트의 요청을 WAS 에 보내고, WAS 가 처리한 결과를 클라이언트에게 전달 (응답)한다.

클라이언트는 일반적으로 웹 브라우저를 의미한다.

- Web Server 의 예 : Apache Server, Nginx [사진]

++ 이때 웹서버를 이용하면 Application Server 까지 가지 않고 빠르게 다시 응답할 수 있다. 웹서버에 서는 정적 콘텐츠만 처리하도록 기능을 분배하여 서버의 부담을 줄이는 것이다.

WAS 란 무엇인가?

- WAS 란 DB 조회나 다양한 로직 처리를 요구하는 동적인 콘텐츠를 제공하기 위해 만들어진 Application Server 다.
- HTTP 를 통해 컴퓨터나 장치에 애플리케이션을 수행해주는 미들웨어 (소프트웨어 엔진) 이다.
- “웹 컨테이너(Web Container)” 혹은 “서블릿 컨테이너(Servlet Container)” 라고도 불린다.
 - 컨테이너란 JSP, Servlet 을 실행시킬 수 있는 소프트웨어를 말함
 - 즉, WAS 는 JSP, Servlet 구동환경을 제공한다.
- 웹 페이지에는 정적 콘텐츠(Static Content) 뿐만 아니라 동적 콘텐츠(Dynamic Content) 가 모두 존재한다. 만약 위의 웹서버만 이용할 시 미리 모든 데이터를 만들어 놓고 클라이언트를 기다려야 한다 (하지만 이는 불가능)
- 그래서 보통 WAS는 요청에 맞는 데이터를 DB에서 가져와 비즈니스 로직에 맞게 그때그때 결과를 만들어 클라이언트에게 제공한다.
- 이로써 한정된 자원을 좀 더 효율적으로 사용할 수 있다.

역할

WAS = WEB SERVER + WEB CONTAINER

- Web Server 기능들을 구조적으로 분리하여 처리하고자 하는 목적으로 제시되었다.
 - 분산 트랜잭션, 보안, 메시징, 쓰레드 처리 등의 기능을 처리하는 분산환경에서 사용된다.
 - 주로 DB서버와 같이 수행됨

[정리]

- 프로그램 실행 환경과 DB 접속기능 제공
- 여러 개의 트랜잭션(논리적인 작업 단위 관리 기능)
- 업무를 처리하는 비즈니스 로직 수행

WAS 의 예 : Jboss, (Tomcat)

Web Server 와 WAS 의 차이

대표적인 웹서버

대표적인 WAS

서블릿 컨테이너

출처 : <https://maenco.tistory.com/entry/Web-Server-와-WAS>

<https://helloworld-88.tistory.com/71>

https://velog.io/@han_been/서블릿-컨테이너Servlet-Container-란

<https://gmlwjd9405.github.io/2018/10/27/webserver-vs-was.html>

=====

서버라는 말이 다른 의미로도 사용되는데 어떤 컴퓨터로 하여금 서버 역할을 하도록 해주는 소프트웨어를 무슨무슨 서버라고 부르기도 한다. 그래서 백엔드 개발자가 서버를 개발한다고 한다.

이 웹서버도 소프트웨어의 개념이다 - 아파치, NginX 가 대표적인 제품

웹 사이트가 서비스 될 때 필요한 것들이 무엇이 있을까

: 웹사이트는 브라우저에서 돌아가는 것 크롬이나 사파리나 파이어폭스, 엣지 같은 것

그 브라우저가 읽을 수 있는 파일들 HTML, CSS , 자바스크립트 파일들이랑 각종 이미지, 기타 여러 데이터들을 갖다가 서버에서 사용자의 컴퓨터로다가 보내줄 수 있어야 한다.

이 파일들은 원래 서버 컴퓨터에 저장돼 있다. 이 서버의 특정 폴더, 디렉토리에 이것들을 넣어두면 이 폴더를 외부에서 접근 가능하도록 개방해서 서버에 지정된 웹사이트 주소로 접속하면 이것들을 받아갈 수 있도록 하는 것이 웹서버의 기본적인 역할 중 하나이다.

이 아파치, 엔진엑스 이 프로그램들로 서버 컴퓨터에 있는 어떤 폴더를 개방을 해가지고다가 거기 들어 있는 HTML 등의 파일들로 웹사이트를 제공할 수 있는 것

서버에 정해진 사이트 주소로 접속을 하면 그 파일들을 꺼내가서 웹사이트를 띄우게끔 - 이것은 정적 웹이다.

동적웹을 제공하는 것도 웹서버의 고유 역할로 정의해야할 지는 애매하지만 아파치나 엔진엑스의 모듈로 할 수 있다.

오랫동안 사용되어온 방법으로 아파치랑 PHP, MySQL 을 연동시켜서 동적인 PHP 웹사이트를 제공하는 방식이 있다. - 아파치에 PHP 를 해석할 수 있는 모듈을 세팅해놓으면 웹접근이 있을 때마다 PHP 코드에 적힌 레시피대로 MySQL 에 있는 정보들을 가져와서 아파치가 만들어줌 엔진엑스도 가능

요즘은 스프링부트에 톰캣이 내장되기 때문에 직접은 많이 안접하지만 자바랑 JSP 로 만든 웹 또는 API 어플리케이션을 실행할 때 이 톰캣이 사용된다.

WAS - 동적사이트를 전문적으로 처리해주는 것

reverse proxy

아파치는 다중 프로세스, 엔진 엑스는 이벤트로 일을 처리함

아파치 - MPM, 멀티 프로세스 모듈 방식으로 일함

손님이 올때마다 프로세스를 새로 생성하는 방식이 있고 한 프로세스 안에서 스레드를 새로 생성하는 방식이 있다.

mpm prefork, 즉 손님마다 프로세스를 두는 방식은 손님이 새로 올 때마다 상담 테이블을 따로 가져와서 앉히는 것 그리고 이 테이블을 계속 돌아다니면서 여러 손님들을 동시에 상대

mpm worker, 한 프로세스에서 손님마다 스레드를 생성한다는 건 가로로 길다란 테이블을 하나 두고, 손님이 올 때마다 나란히 앉힌 다음 역시 좌우로 돌아다니면서 동시에 업무를 처리하는 것

반면 NginX 의 event driven 방식을 비유하자면 작은 데스크 하나만 두고 손님들을 한 줄 로 쪽 세우는 것

그리고 다음 손님이 오는 대로 업무별로 집중해서 일을 처리하는 것

<https://www.youtube.com/watch?v=Zimhvf2B7Es>