

3 Tier-Architecture

3 Tier-Architecture?

[Tier and Layer](#)

[해당 Architecture의 이점](#)

[다른 Architecture의 종류](#)

[1 Tier-Architecture](#)

[2 Tier-Architecture](#)

[N Tier-Architecture](#)

[출처](#)

3 Tier-Architecture?

3계층으로 나뉘 별도의 논리적/물리적인 장치에 구축 및 운용하는 형태 웹에 비유할때 아래와 같이 표현 가능

1. **프레젠테이션 계층(Presentation tier): Frontend**, 사용자와 대면하는 계층, 이 계층에서는 사용자의 데이터를 직접적으로 다루는 데이터는 포함하지 않는다.
2. **애플리케이션 계층(Application tier): Backend**, 프론트에서 온 **Request** 를 도메인 서비스에 맞게 처리 및 재가공 후 **Response** 해주는 영역
3. **데이터 계층(Data tier): Backend or DBA 영역**, 데이버베이스와 데이터를 읽거나 쓰는 것을 관리하는 영역

Tier and Layer

Tier 와 **Layer** 는 비슷하지만 다른 의미를 가진다. 계층은 다른 분할과 구분된 인프라에서 실행되는 소프트웨어의 기능적 분할을 의미합니다.

1. **Tier**: 컴포넌트의 물리적인 분리
2. **Layer**: 컴포넌트의 논리적인 분리

해당 Architecture의 이점

1. 각 계층을 서로 다른 팀에서 동시에 개발할 수 있다. 프로그래머는 각 계층에 최신 및 최상의 언어와 툴을 사용할 수 있다.
2. 어플리케이션을 **스케일업** 및 **스케일아웃** 가능
 - 별도의 백엔드 계층을 통해 다양한 데이터베이스에 구현 가능 또한 여러 웹 서버를 추가해 확장가능
3. 프레젠테이션 코드와 비즈니스 로직을 별도로 관리해 비즈니스 로직을 변경해도 프레젠테이션 계층에 영향을 미치지 않는다.
4. 프레젠테이션 계층과 데이터 계층이 직접 통신할 수 없어 SQL 인젝션 및 기타 악의적 공격을 방지할 수 있다.

다른 Architecture의 종류

1 Tier-Architecture

동일한 tier, 클라이언트 tier에 프레젠테이션, 애플리케이션, 데이터 layer가 존재하는 것을 의미한다.

- MP3 플레이어, MS Office 등이 예시로 볼 수 있다.

2 Tier-Architecture

클라이언트 tier에서 프레젠테이션 레이어와 어플리케이션 레이어를 모두 처리하고 서버는 데이터베이스 레이어를 처리한다.

- **클라이언트-서버 어플리케이션** 이라고 불리며 통신은 클라이언트 서버 간에 이뤄지며 클라이언트는 시스템에 Request를 전송하고 서버는 요청을 처리한 후 클라이언트에 Response한다.

N Tier-Architecture

분산 or 다중 계층 아키텍처라고 불린다. 3-tier 계층과 유사하지만 비즈니스 로직을 분산하기 위해 어플리케이션 서버의 수가 증가해 개별 계층으로 표시

출처

- 3계층 구조(3 Tier- Architecture) 이해하기 [스마트인재개발원]
- 3-Tier Architecture 정의 및 구성방식
- 3계층 아키텍처란?