Software Architecture

# Distributed Architecture

Eunmi Choi
Kookmin University

# Distributed Architecture

분산 상황에 있는 아키텍쳐 구조. 분산 시스템 ( 컴퓨터가 모여있는 시스템. 서로 네트워크로 연결되어 어떤 일들을 함. 집합.)
특징은 분산시스템에 있는 컴퓨터는 몇개인지 신경 쓰지 않고 일을 받길 원함. = single image system
분산시스템 - 네트워크를 빼놓을 수가 없음.
메세지 패싱, 외부에 프로시져 콜 하고 외부의 메소드를 인보케이션, 웹 다 가능.

- Synopsis
  - A distributed system is a collection of computational and storage devices connected through a communications network.
  - The subsystems or components within a distributed system communicate with each other using a number of methods including message passing, remote procedure calls, and remote method invocation.

# Distributed Architecture

- Structure
  - Two important elements of designing a distributed system are:
    - network topology
      - the way in which entities are organized to form a connected network
    - communications mode
      - the method by which components communicate with each other.

# Distributed Architecture

- Structure
  - A distributed system can be modeled by the client-server architecture
    - the basis for multitier architectures.
  - Alternatives are
    - the broker architecture such as CORBA
    - the service-oriented architecture (SOA) such as web services and grid services
  - Key features of a distributed architecture
    - service location transparency
    - services reliability and availability
  - technology frameworks to support distributed architectures,
    - .NET, J2EE, CORBA, .NET web services, AXIS Java web services, and GloBus grid services.

클라이언트 - 서버 구조 : 리퀘스트 - 리스폰스

브로커가 어떤 구조?

소아

transparency : 투명성, 서비스가 위치되어 있는게 투명하다는 것. 어떤 일을 요청하면 결과를 예측할 수 있고 보여줌. 소프트웨어적으로 말하자면, 신뢰성이 있어야 하고 프로토콜이 맞춰진 상태에서 원하는 일이 반영되어 끝까지 반ㅇㅇ될 수 있고 그 결과를 예측할 수 있고 신뢰할 수 있음.

www.kookmin.ac.kr - > location transparency. (물리적인 위치가 어디에 있든 그 결과에 대해 믿을 수 있고 예측할 수 있고 url이 정해져 있음)

이런거도 분산 아키텍쳐에서 고려할 수 있음.

# Distributed Architecture

- Various distributed architecture styles
  - Client-server
  - Multitier
  - Proxy
  - Dispatcher (Load Balancer)
  - P2P
  - Broker
  - Service-oriented architecture
  - MicroService architecture