

Spring 2023



소프트웨어 아키텍처 패턴: Peer-to-Peer

Seonah Lee

Gyeongsang National University

Peer-to-Peer 패턴

- ▶ 패턴 정의
- ▶ 패턴 예제
- ▶ 패턴 설명
- ▶ 패턴 컴포넌트, 구조 및 행위
- ▶ 패턴 구현
- ▶ 패턴 코드
- ▶ 패턴 장단점

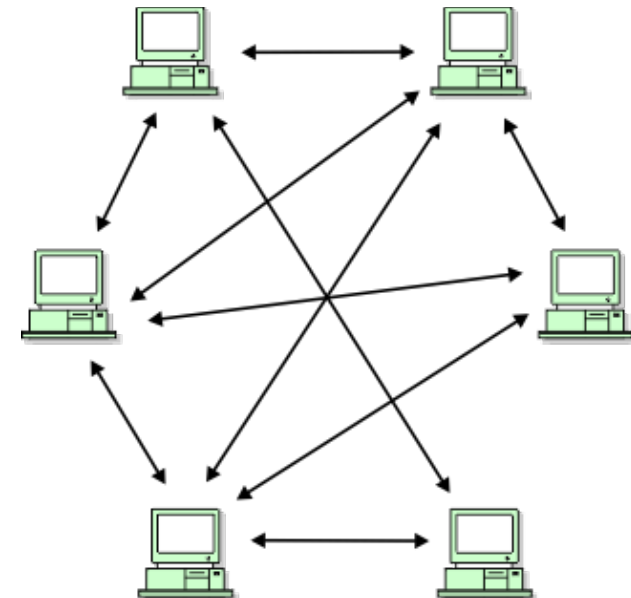
Peer-to-Peer Pattern: Definition

▶ 정의

- ▶ 네트워크의 개별 노드가 리소스의 공급자와 소비자 역할을 모두 수행하는 분산 및 분산 네트워크 아키텍처의 유형
 - ▶ 클라이언트 노드가 액세스를 요청하는 중앙 집중식 클라이언트-서버 모델과는 다름

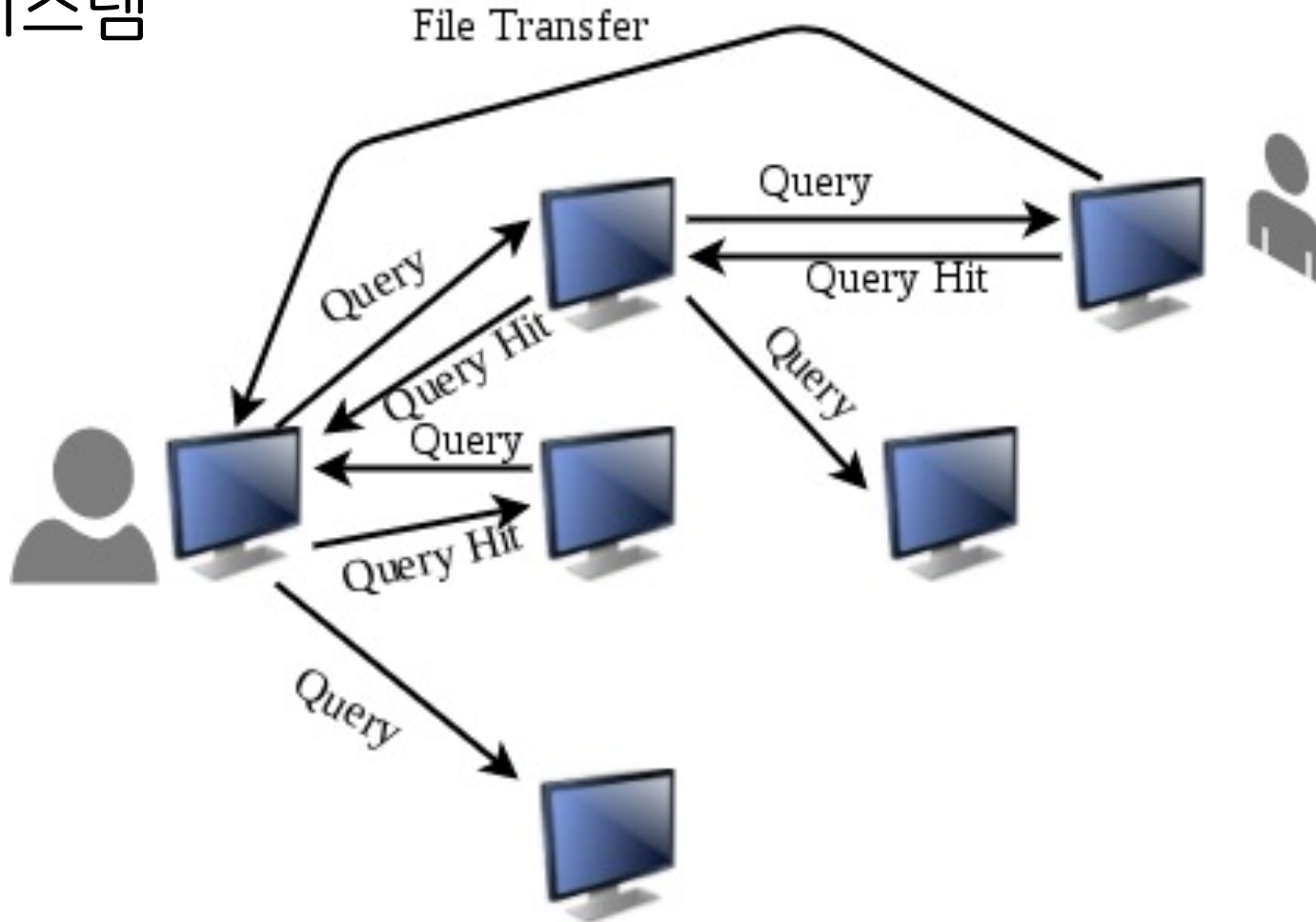
▶ 예제

- ▶ **BitTorrent, Gnutella** 등 파일 공유 네트워크
- ▶ **Spotify** 등 멀티미디어 어플리케이션



Peer-to-Peer Pattern: Example

▶ 파일 공유 시스템



Peer-to-Peer Pattern: Description

▶ 정황 (Context)

- ▶ 분산 네트워크 환경에서 중앙 조정 없이 다른 노드 참가
- ▶ 여러 시스템이 공통 정보 및 응용 프로그램을 공유

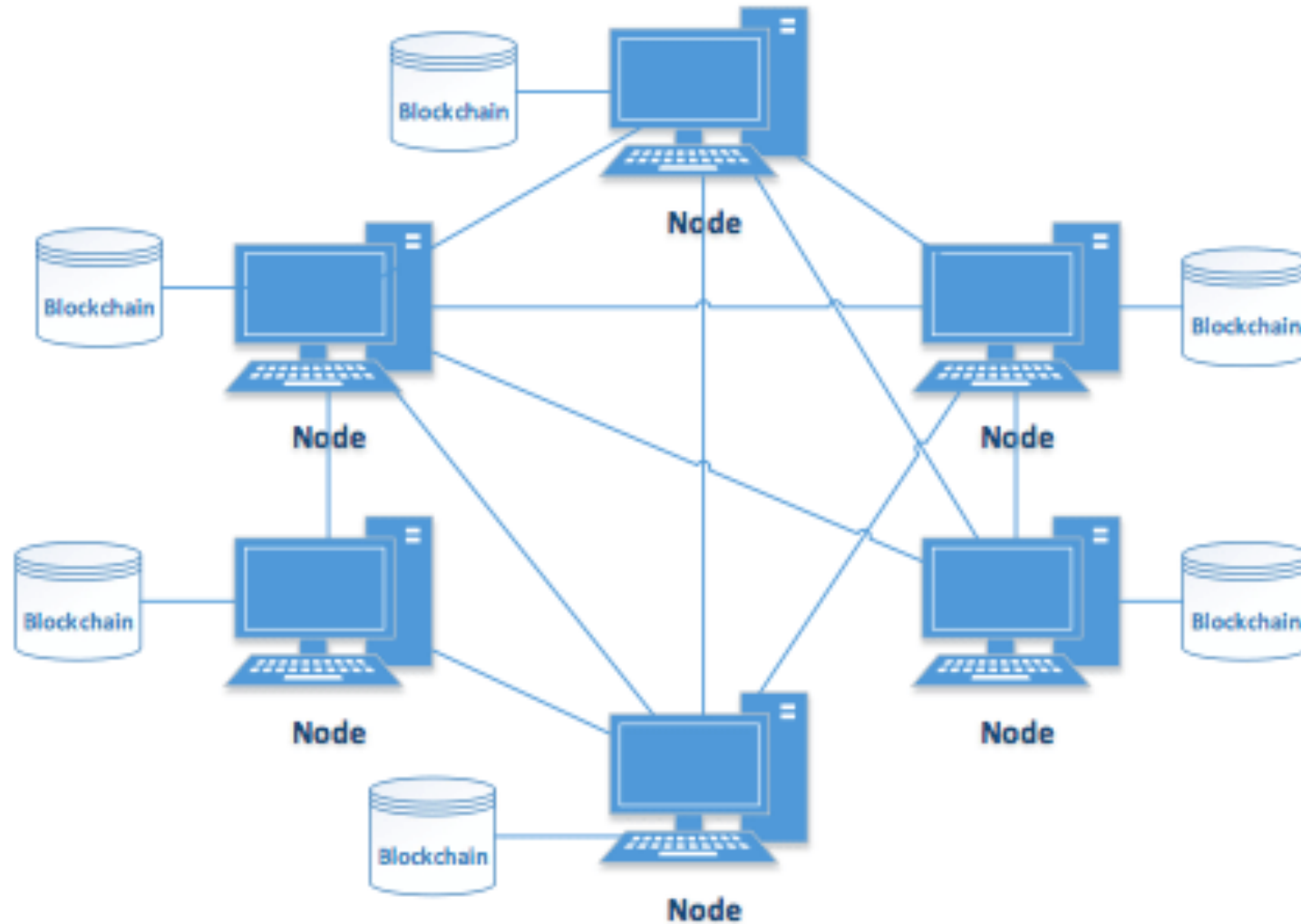
▶ 문제 (Problem)

- ▶ 피어 간에 작업 또는 작업 부하를 분할하는 분산 응용 프로그램 필요
- ▶ 피어는 리소스의 공급자이자 소비자여야 함

▶ 해법 (Solution)

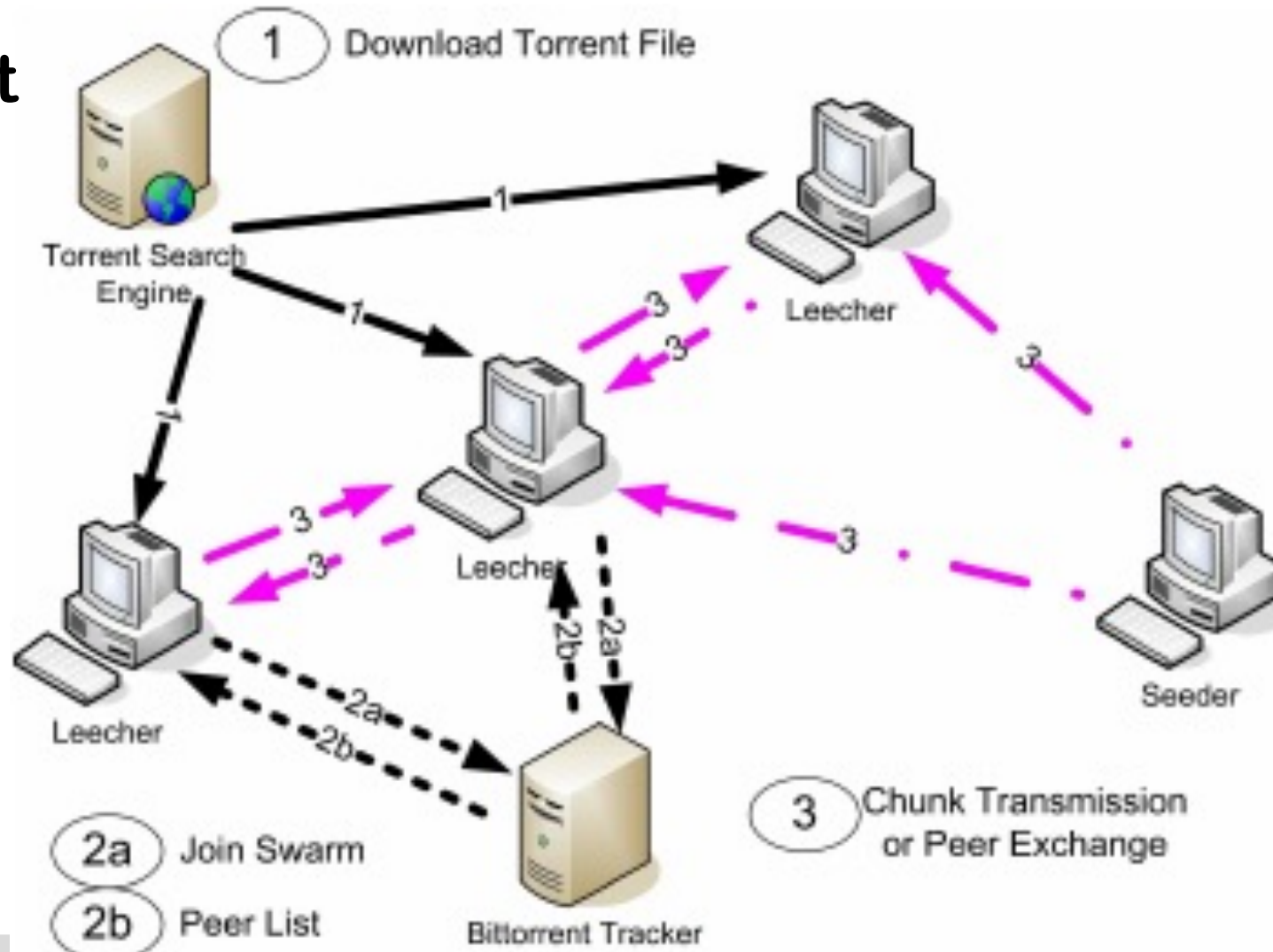
- ▶ **P2P** 네트워크의 노드는 “클라이언트”와 “서버”로 작동

Peer-to-Peer Pattern: Case Studies



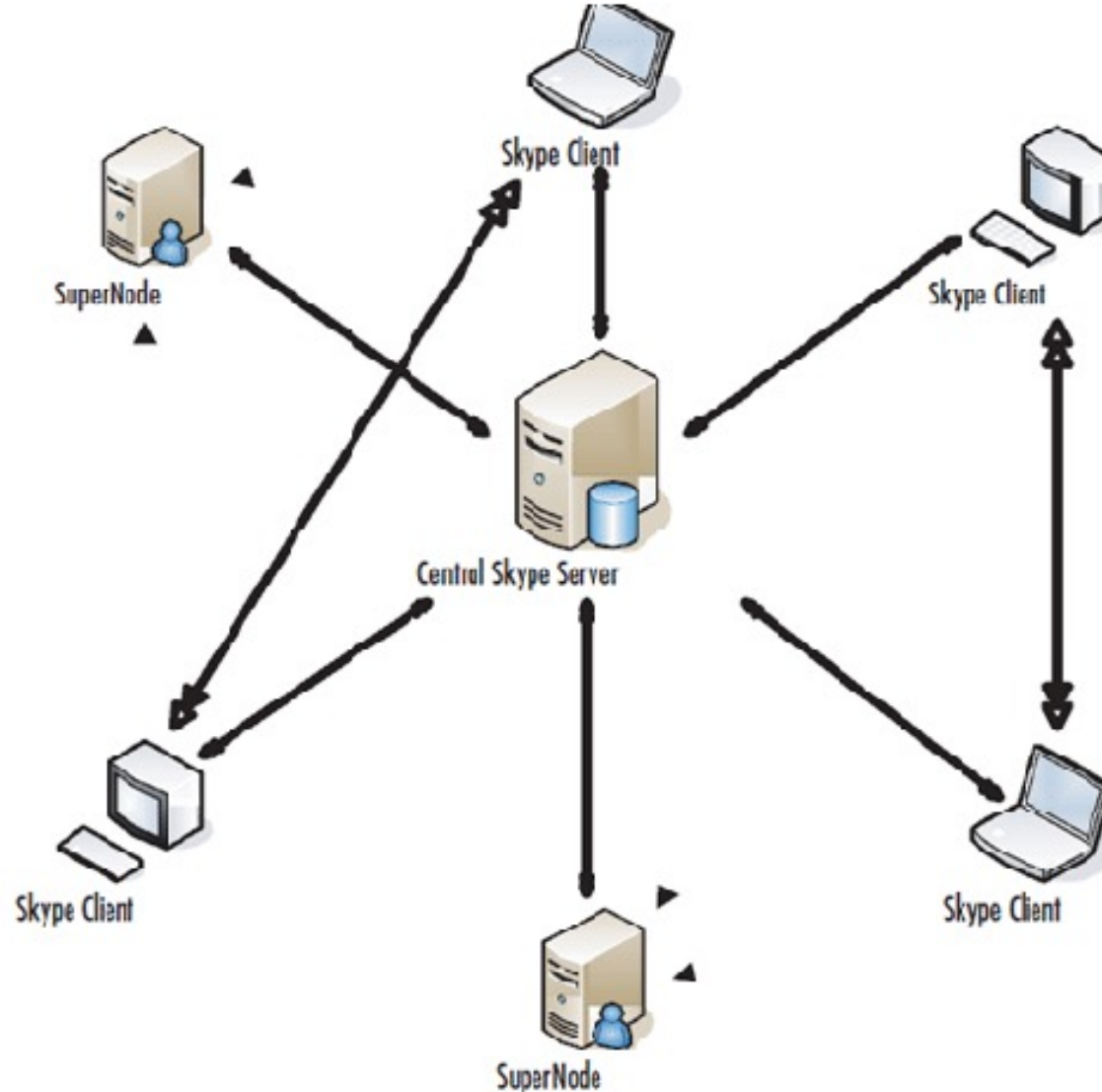
Peer-to-Peer Pattern: Case Studies

► BitTorrent



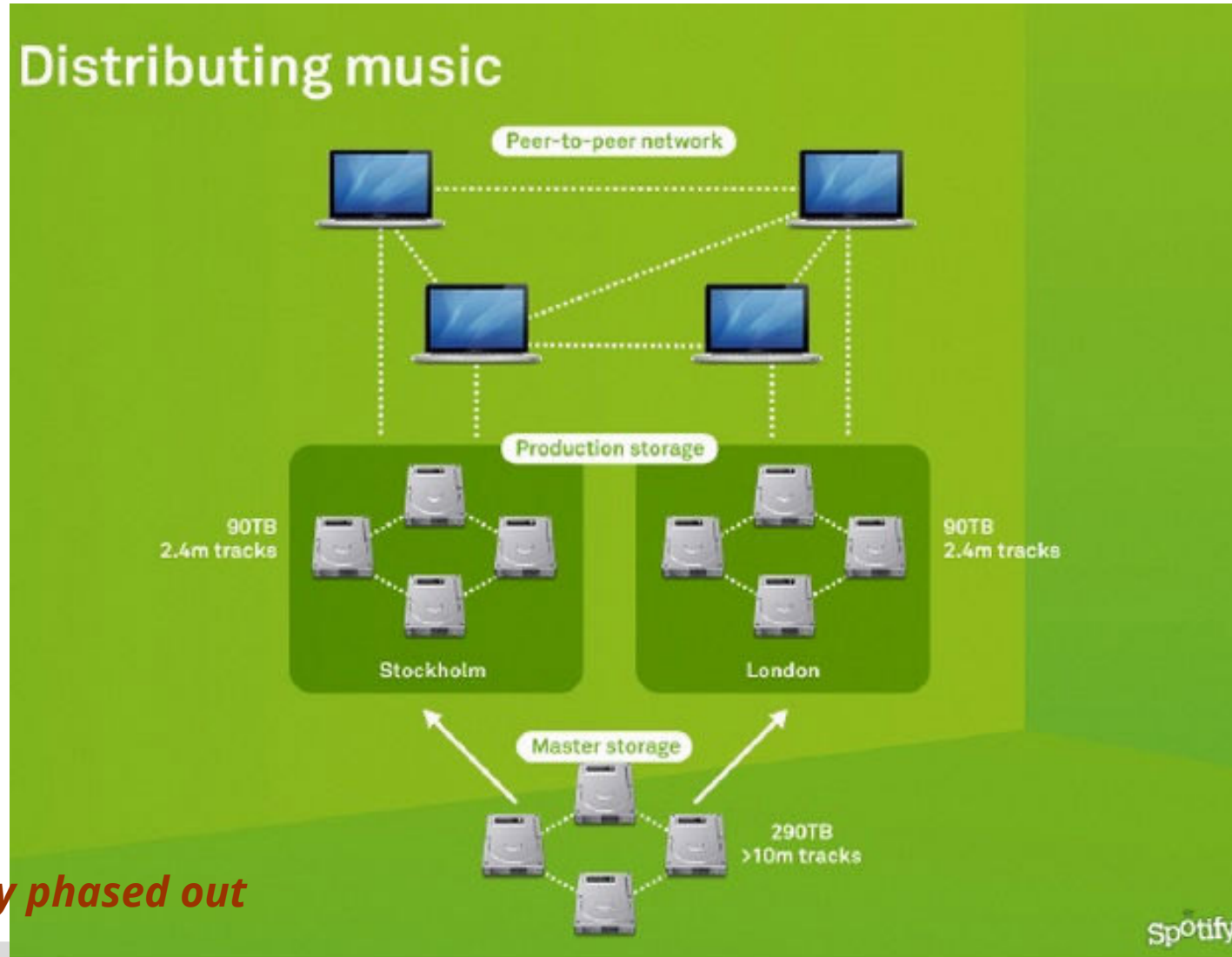
Peer-to-Peer Pattern: Case Studies

► Skype



Peer-to-Peer Pattern: Case Studies

► Spotify



This is the model Spotify phased out

Peer-to-Peer Pattern: Benefits

- ▶ 탈중앙화된 컴퓨팅 지원
 - ▶ 이러한 분산 네트워크 컴퓨팅은 좀 더 경제적이 될 수 있음
- ▶ 특정 노드 장애 감내
 - ▶ 특정 노드가 실패했다고 해서 전체 시스템의 실패가 되지 않음
- ▶ 리소스 및 컴퓨팅의 확장성 높음
 - ▶ **Peer**의 변경이 쉬움.
 - ▶ 하나의 **Peer**가 네트워크에서 단절되어도 전체 네트워크에는 별 영향이 없음
 - ▶ 일단, 많은 **Peer**들이 네트워크에 있다면, 이러한 부하의 증대를 쉽게 처리할 수 있음

Peer-to-Peer Pattern: Liabilities

▶ 보안이 힘들

- ▶ 하나의 **Peer**가 바이러스에 감염되었을 때 다른 **Peer**로의 전파가 빠를 수 있음
- ▶ **Peer**들이 네트워크에 접근할 적절한 권한이 있는지 확인이 힘들

▶ 서비스 품질 보장 어려움

- ▶ 실시간 영상 제공 등에서의 적절한 성능 보장이 어려움

▶ 노드 갯수에 따라 성능 좌우



Question?



Seonah Lee
saleese@gmail.com