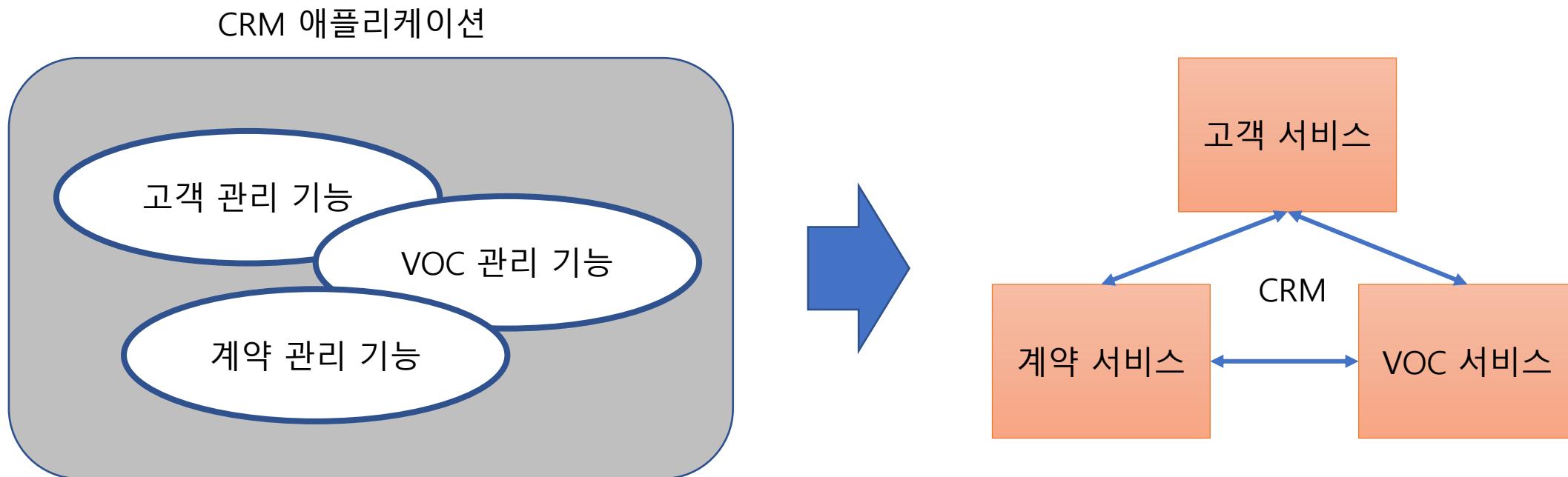




What is the Microservice Architecture?

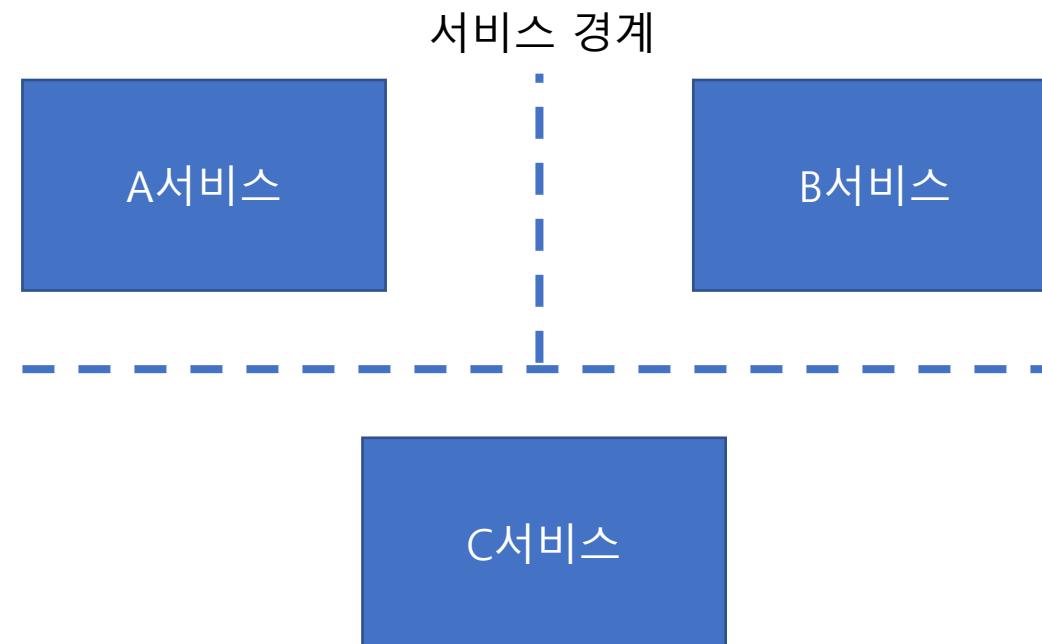
Microservice 특징

- 작은 서비스
 - ~~독립된 비즈니스~~ → **독립된 서비스**
 - Eric Evans "Domain Driven Design"
 - Bounded Context: 하나의 서비스에 담을 수 있는 기능들의 그룹, 서비스 수준의 경계
 - Ubiquitous Language: 같은 업무를 하는 사람들은 업무상 같은 단어와 용어를 사용



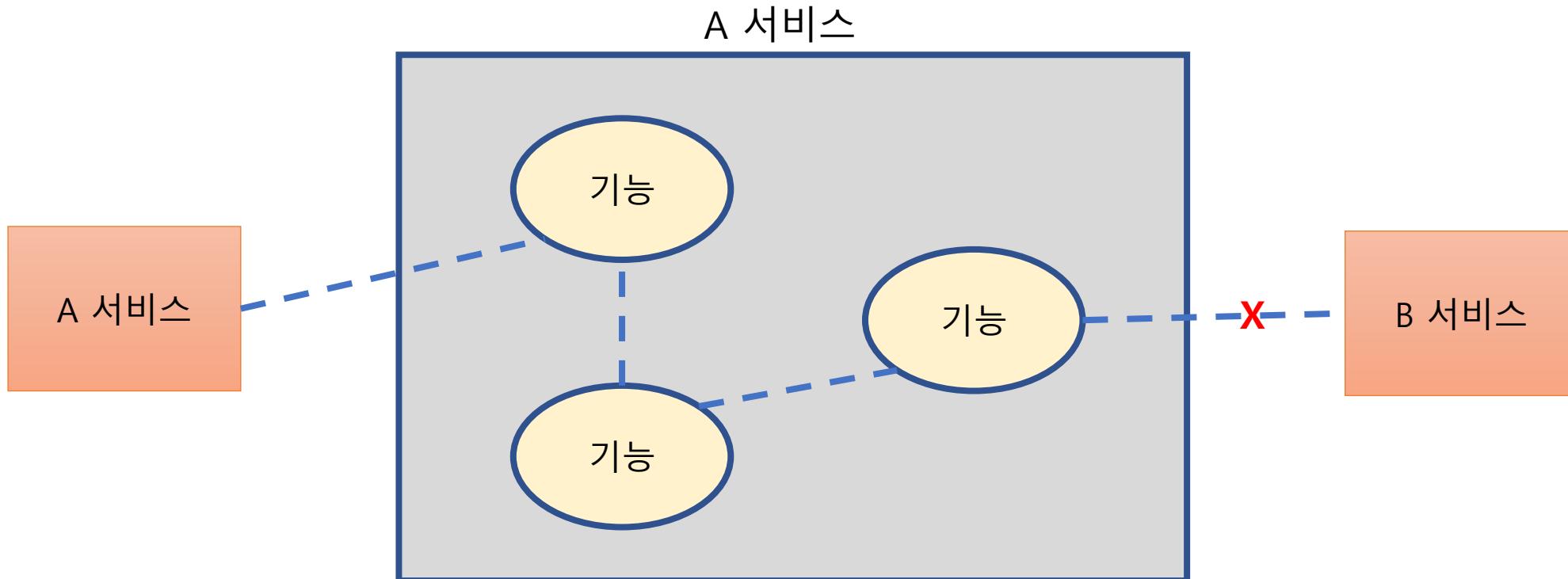
Microservice 특징

- 독립된 서비스
 - 독립적 실행, 다른 서비스 결합이 없는 서비스
 - 구현, 배포 실행, 장애에 대한 영향 받지 않음
- ex) "고객 관리 서비스" 중 고객 등록 기능에 문제가 생겨도, 이와 관계없이 예약 가능



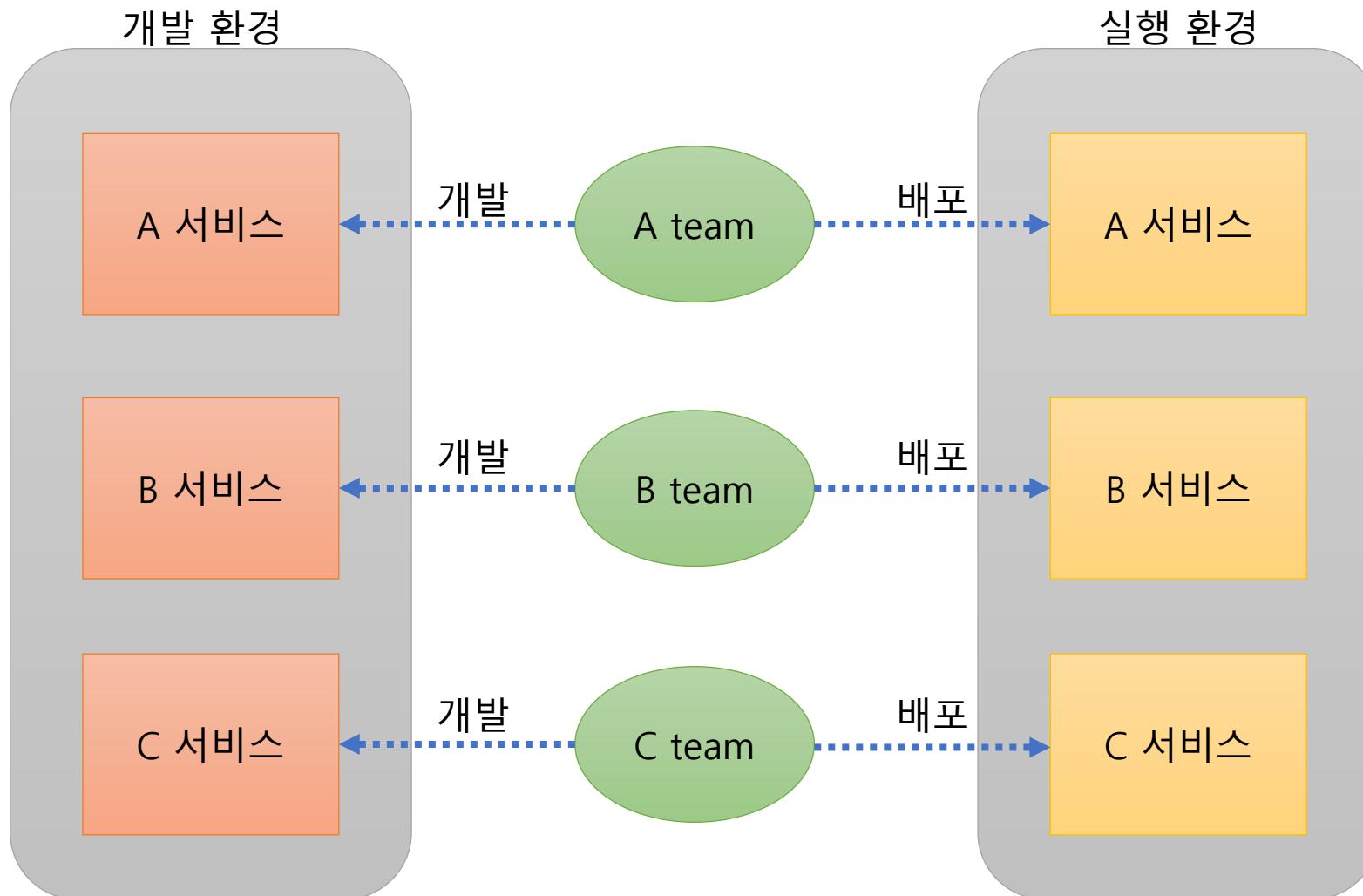
Microservice 특징

- 응집된 서비스
 - 하나의 서비스는 기능적으로 응집
 - 서비스의 역할이 한 가지 일을 위해 묶여야 함 → 단순 명확, 오류 최소화
ex) 수강 신청 시스템



Microservice 특징

- 자율적 서비스
 - 서비스의 기획, 개발, 테스트, 배포 및 서비스의 운영까지 담당 조직이 독립적으로 의사결정 → Ownership



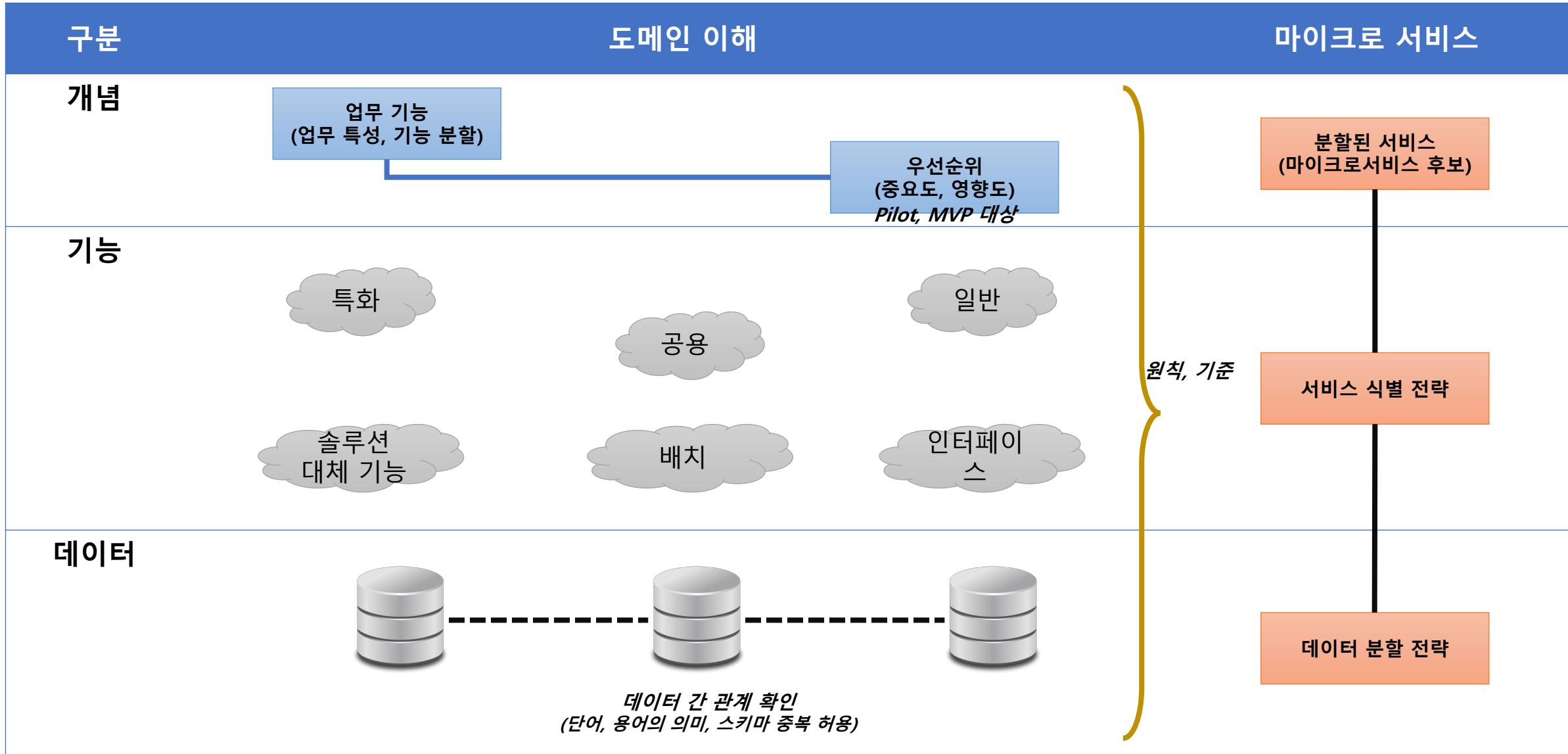
Microservice 기획

- Microservice 식별 전략



Microservice 기획

- Microservice 식별 전략



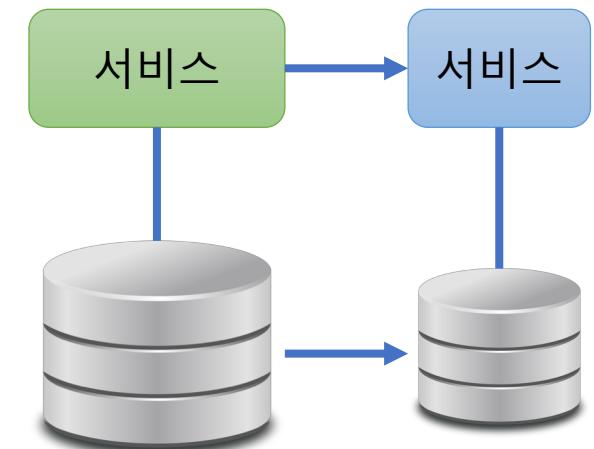
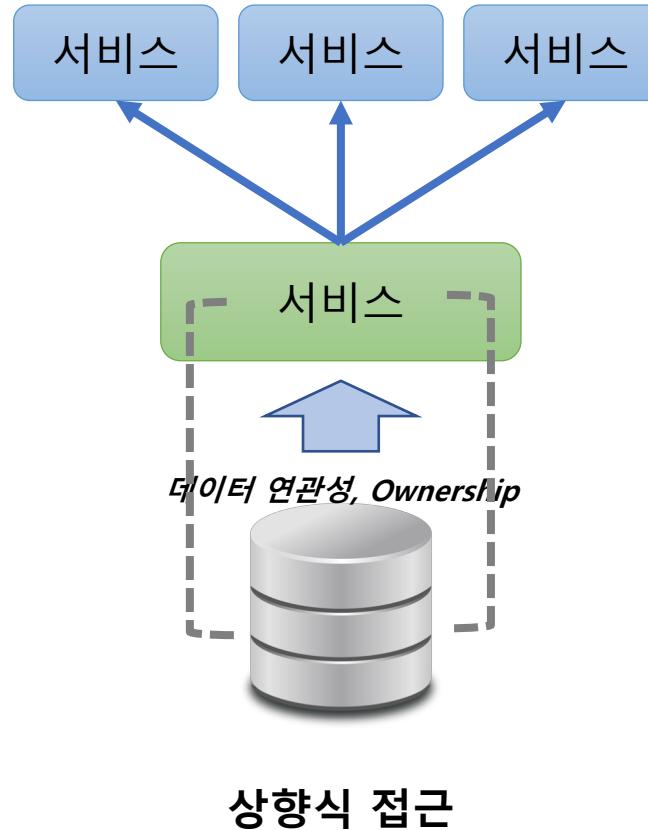
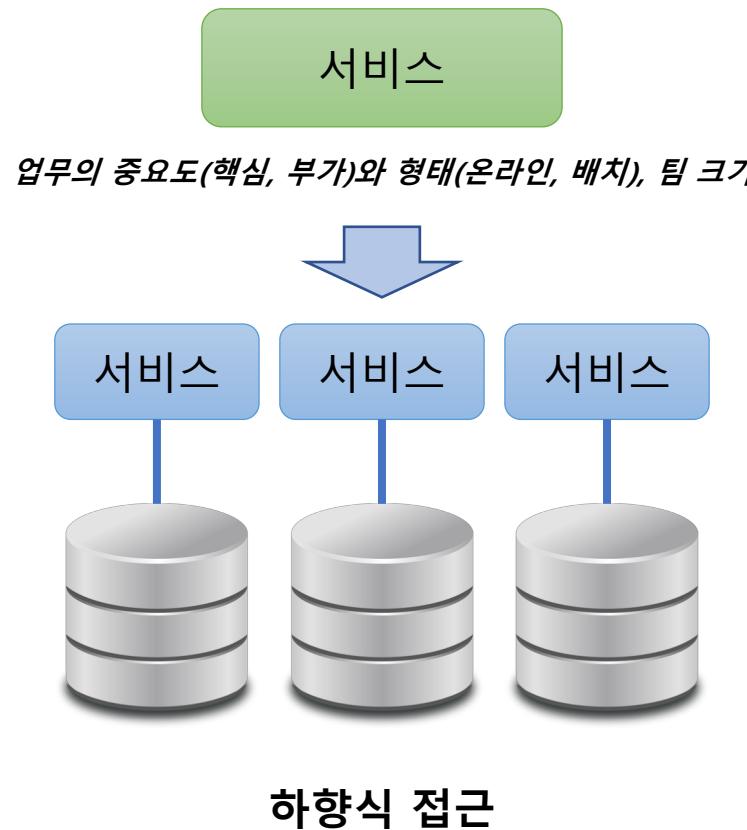
Microservice 기획

- Microservice 식별 전략



Microservice 기획

- Microservice 식별 전략



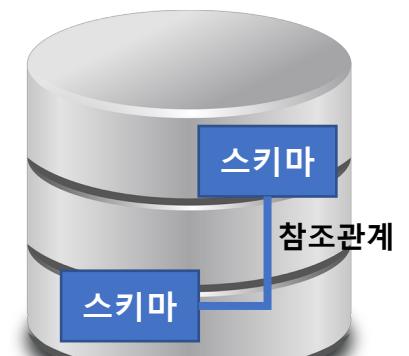
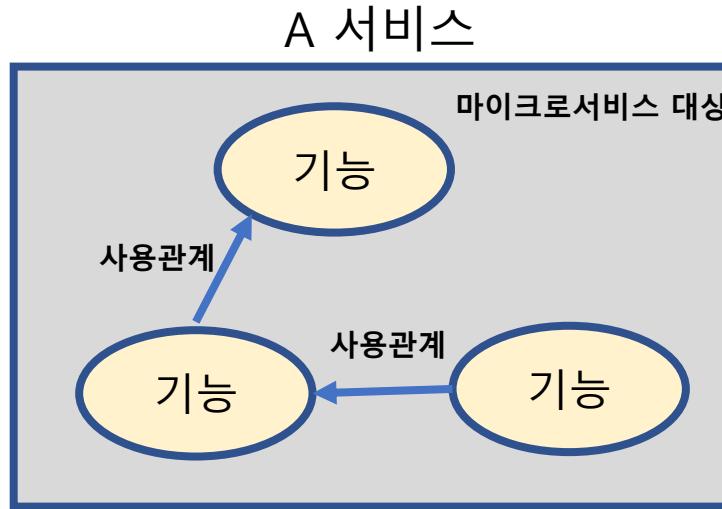
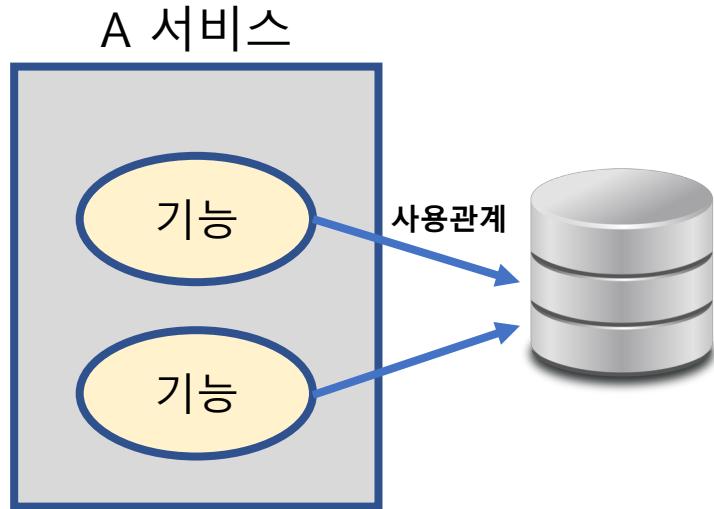
Microservice 기획

- Microservice 식별 전략



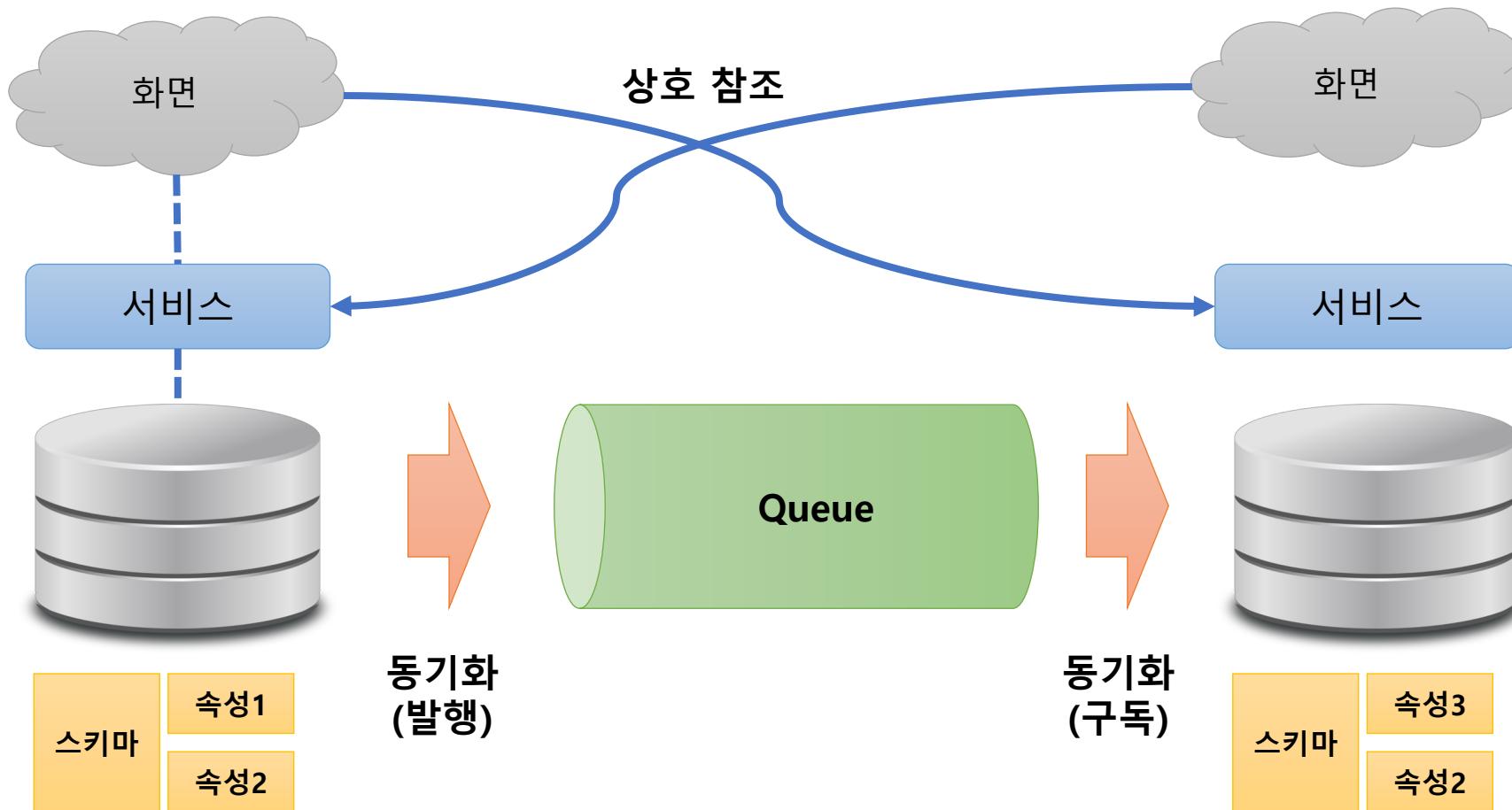
Microservice 기획

- Microservice 식별 전략



Microservice 기획

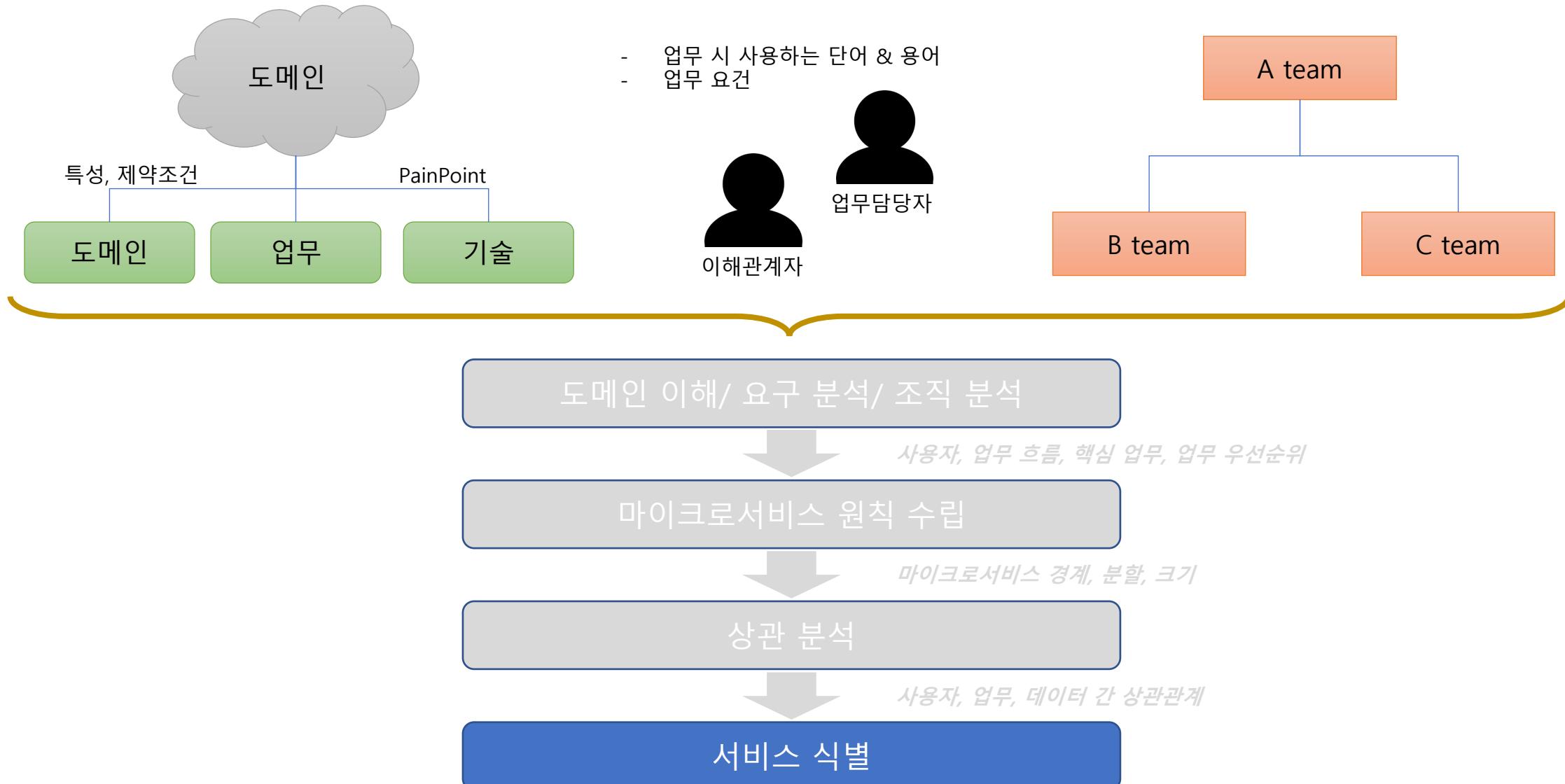
- Microservice 식별 전략



마이크로서비스 간 데이터 연계

Microservice 기획

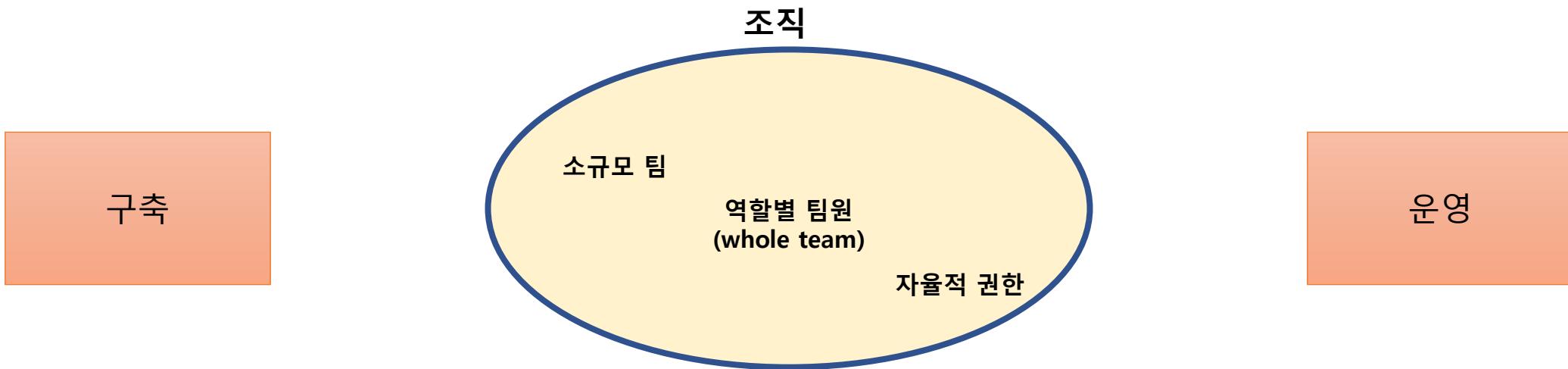
- Microservice 식별 전략



Microservice 기획

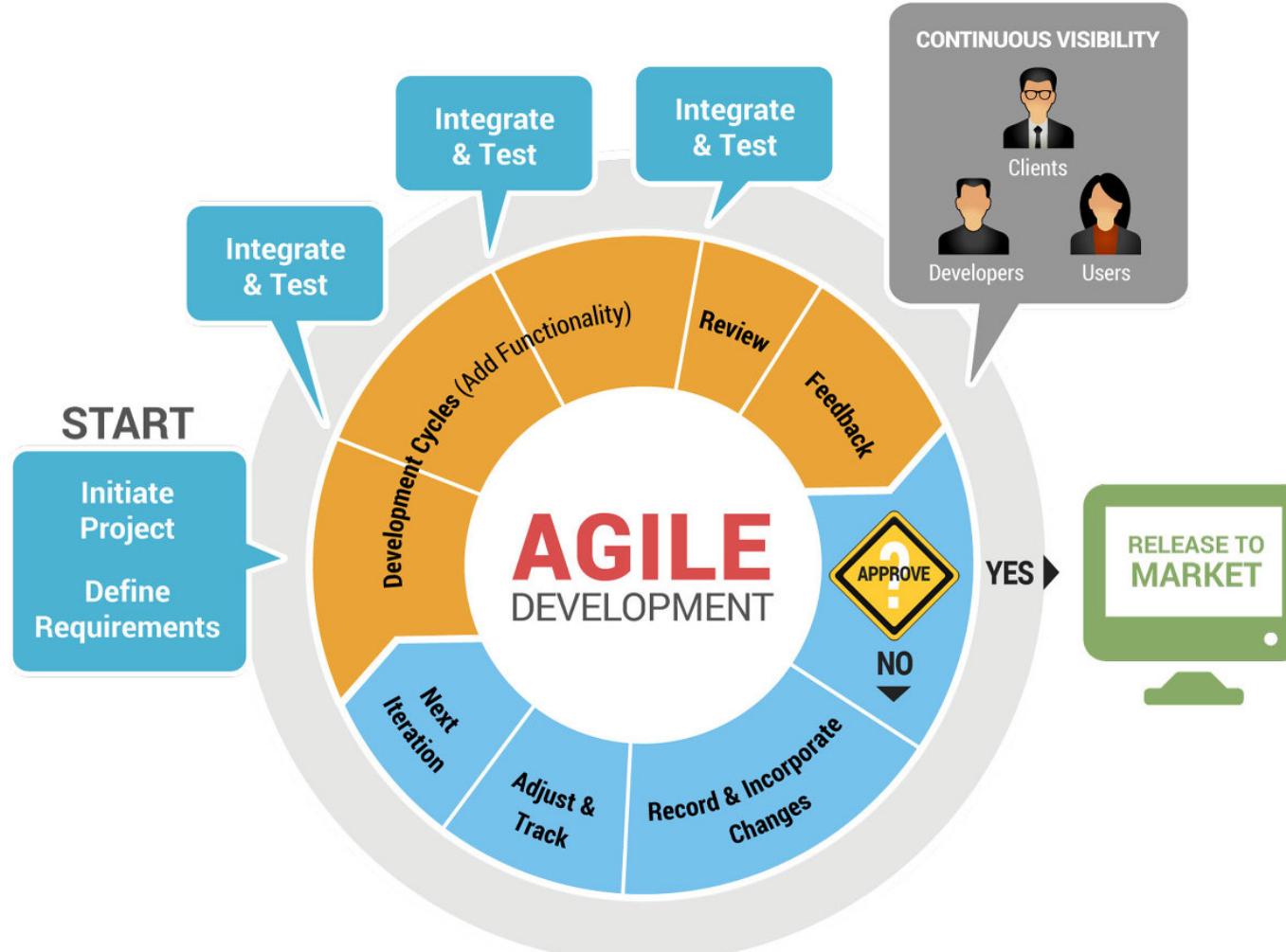
- Microservice 고려 사항

- 1) 조직의 구성



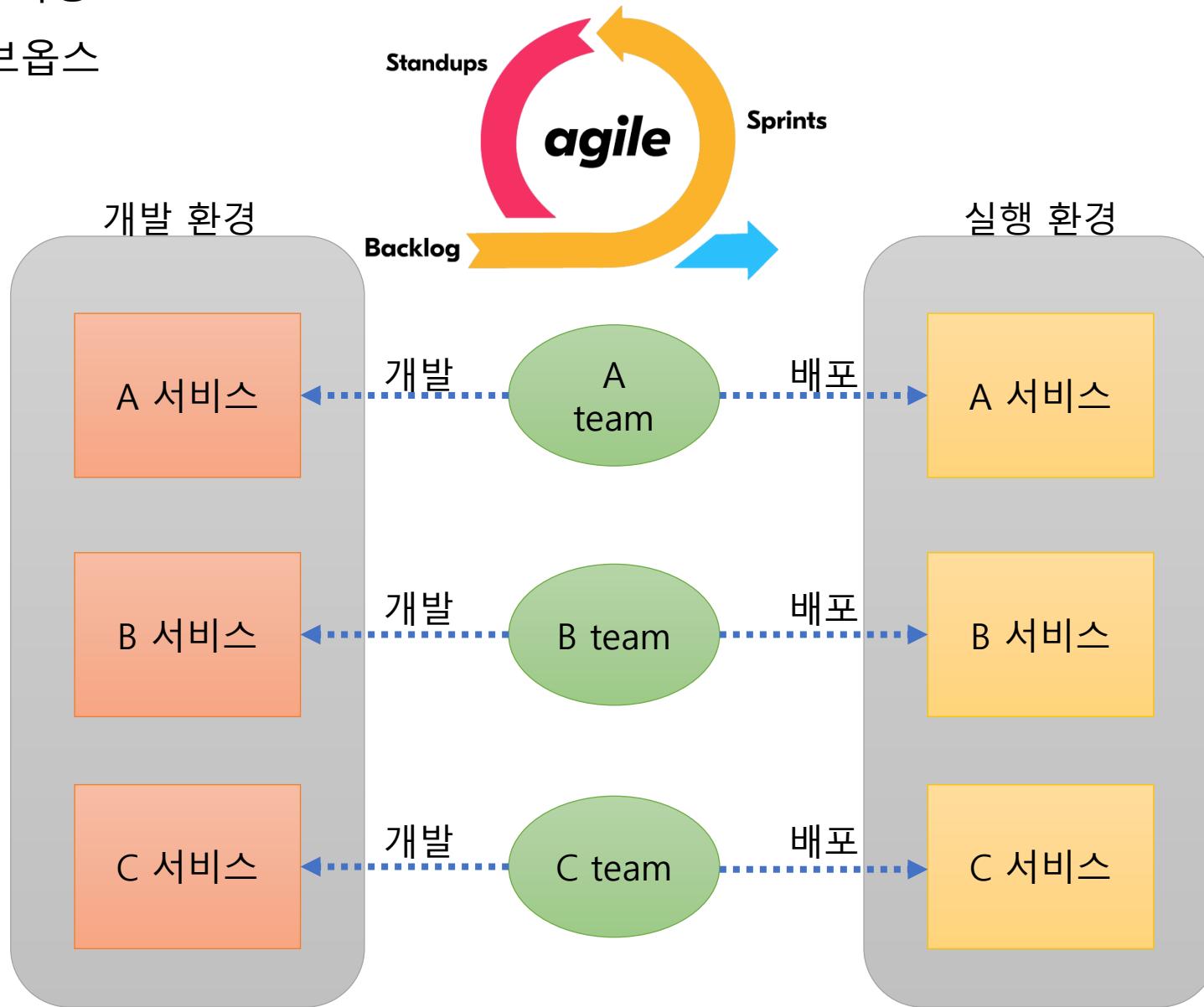
Microservice 기획

- Microservice 고려 사항
2) 애자일과 데브옵스



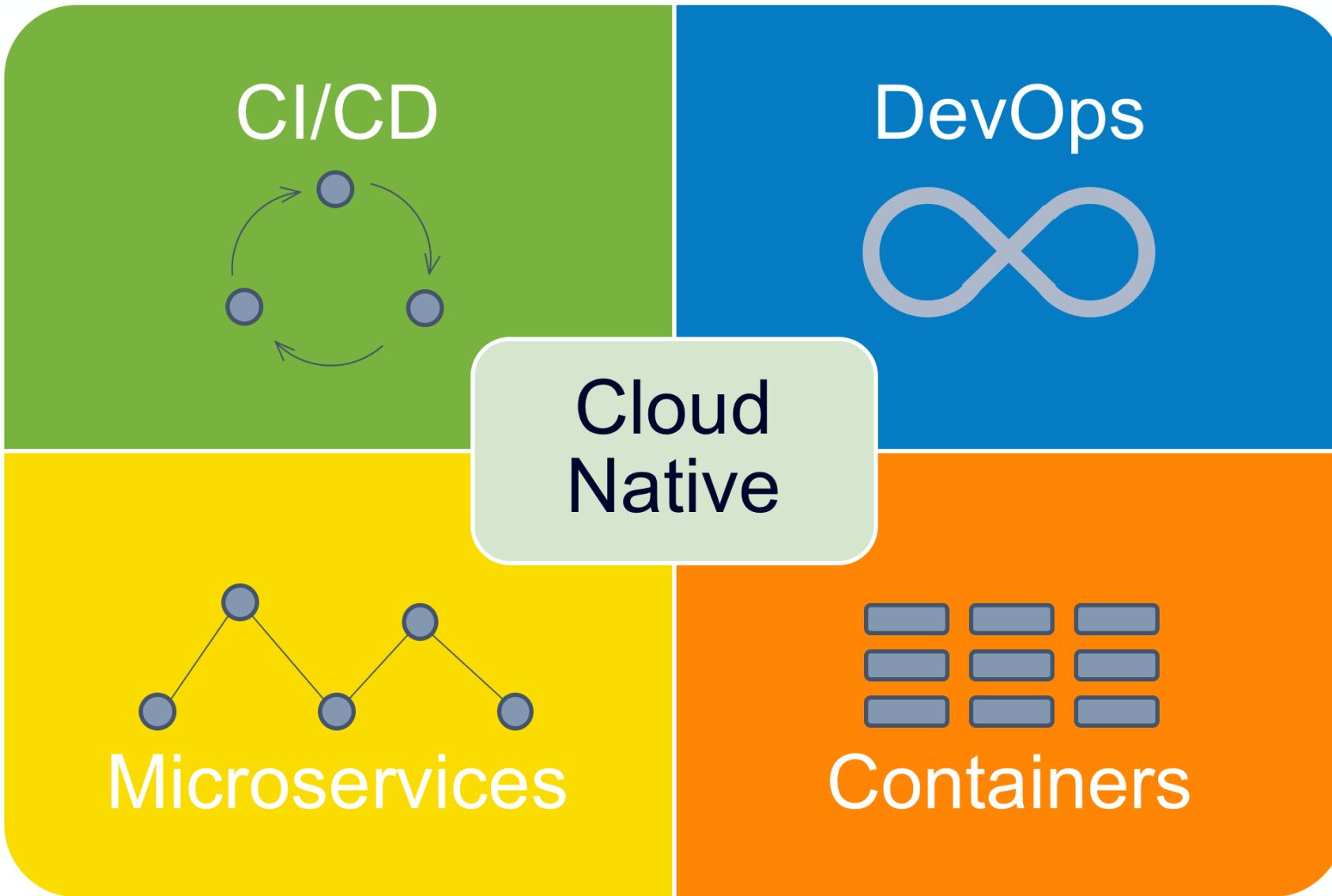
Microservice 기획

- Microservice 고려 사항
- 2) 애자일과 데브옵스



Microservice 기획

- Microservice 고려 사항
 - 3) 클라우드 네이티브 기술 환경



Microservices Architecture and Polyglot Persistence

