

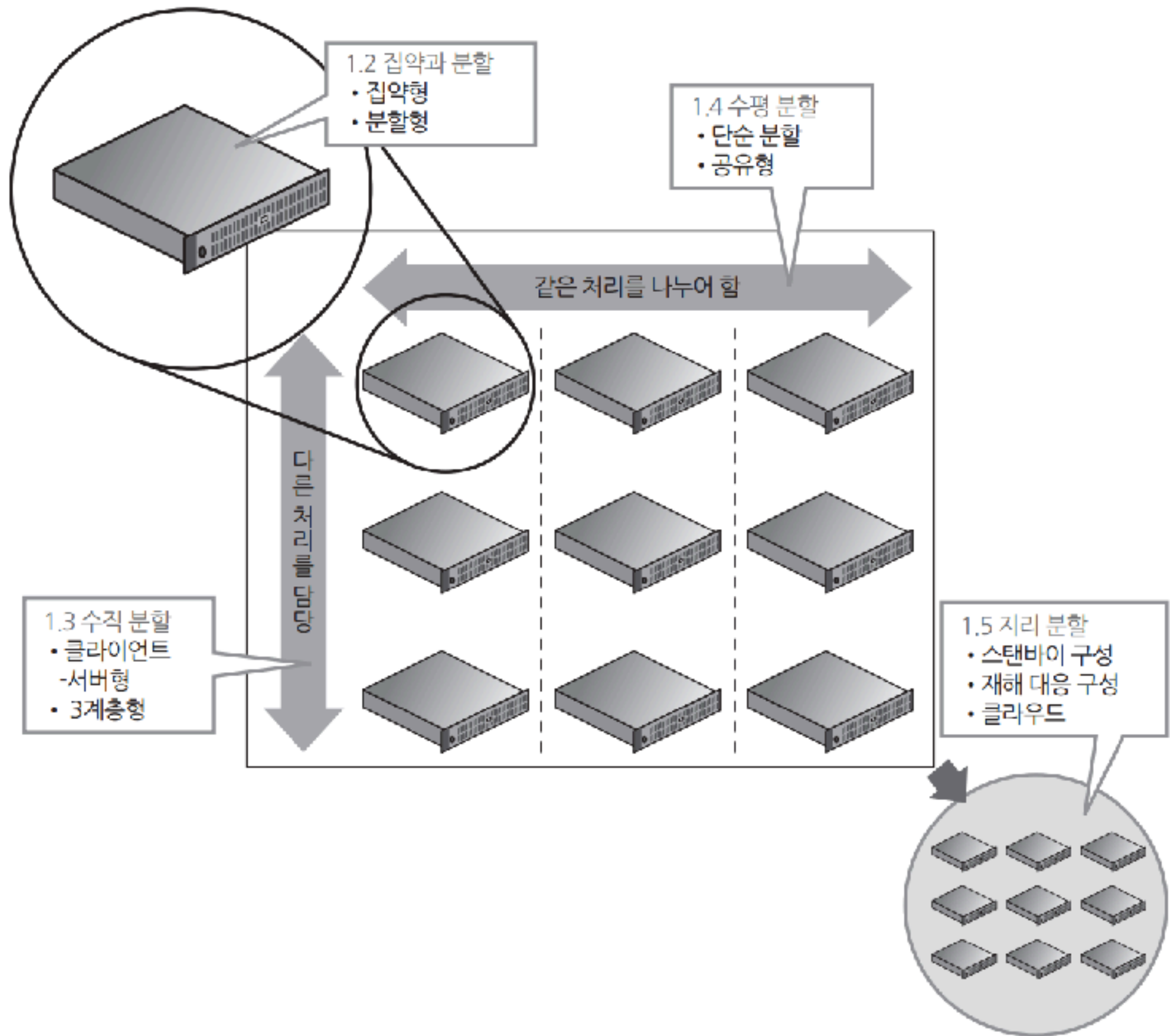
01 인프라 아키텍처

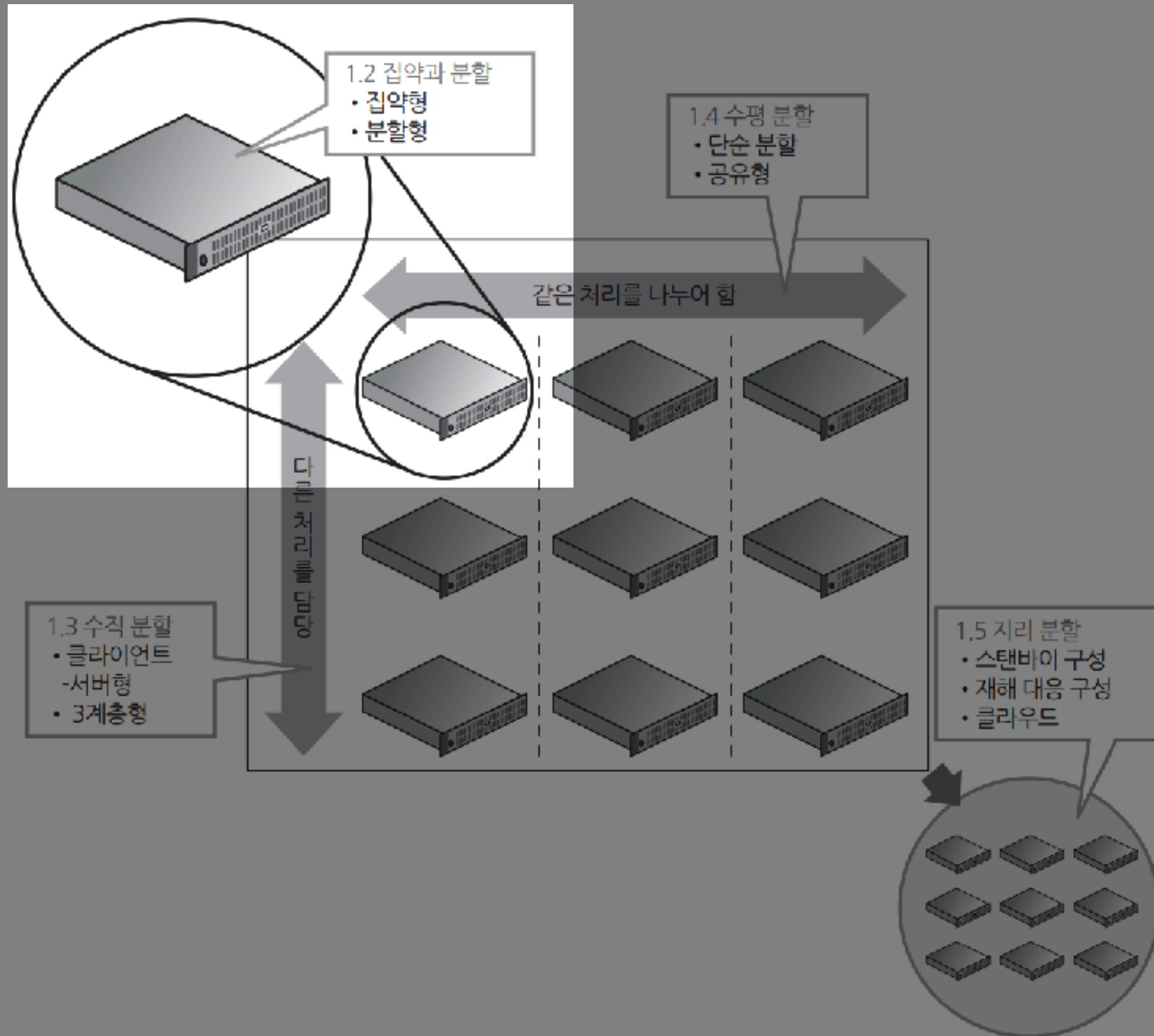
대표적 인프라 아키텍처 소개

인프라(infra) : 기반을 형성하는 시설·제도

IT인프라 : IT(information technology; 정보 통신 기술 산업)의 기반

인프라 아키텍처 : IT인프라의 구조





집약형 아키텍처

대형 컴퓨터(=범용 장비, 호스트, 메인 프레임)로 모든 업무 처리

안정성 보완 위해

-> CPU 및 구성 부품의 다중화 (재해 대비)

-> 유한 리소스 관리 (멀티 프로세싱)

집약형 아키텍처

장점

1. 한대의 컴퓨터만 있으면 되므로 구성 간단
2. 대형 컴퓨터의 리소스 관리, 이중화에 의해 안정성 높고 고성능

단점

1. 도입비용과 유지 비용 큼
2. 확장성에 한계

분할형 아키텍처

대형 컴퓨터가 했던 처리를 다수의 소형 컴퓨터로 분할,
여러 컴퓨터가 연결돼 있어서 데이터 서로 교환

표준 OS나 개발 언어를 이용하기 때문에 ‘오픈시스템’이라 한다.
(분산시스템 이라고도 함)

수직형과 수평형이 있음

분할형 아키텍처

장점

1. 낮은 비용으로 시스템 구축 (개별 컴퓨터를 저가 장비로)
2. 서버 대수를 늘릴 수 있어 확장성 좋음

단점

1. 관리 구조가 복잡해짐
2. 서버 간 영향 범위를 최소화 하기 위해 서버 별 역할(구조) 검토

물리서버와 논리서버

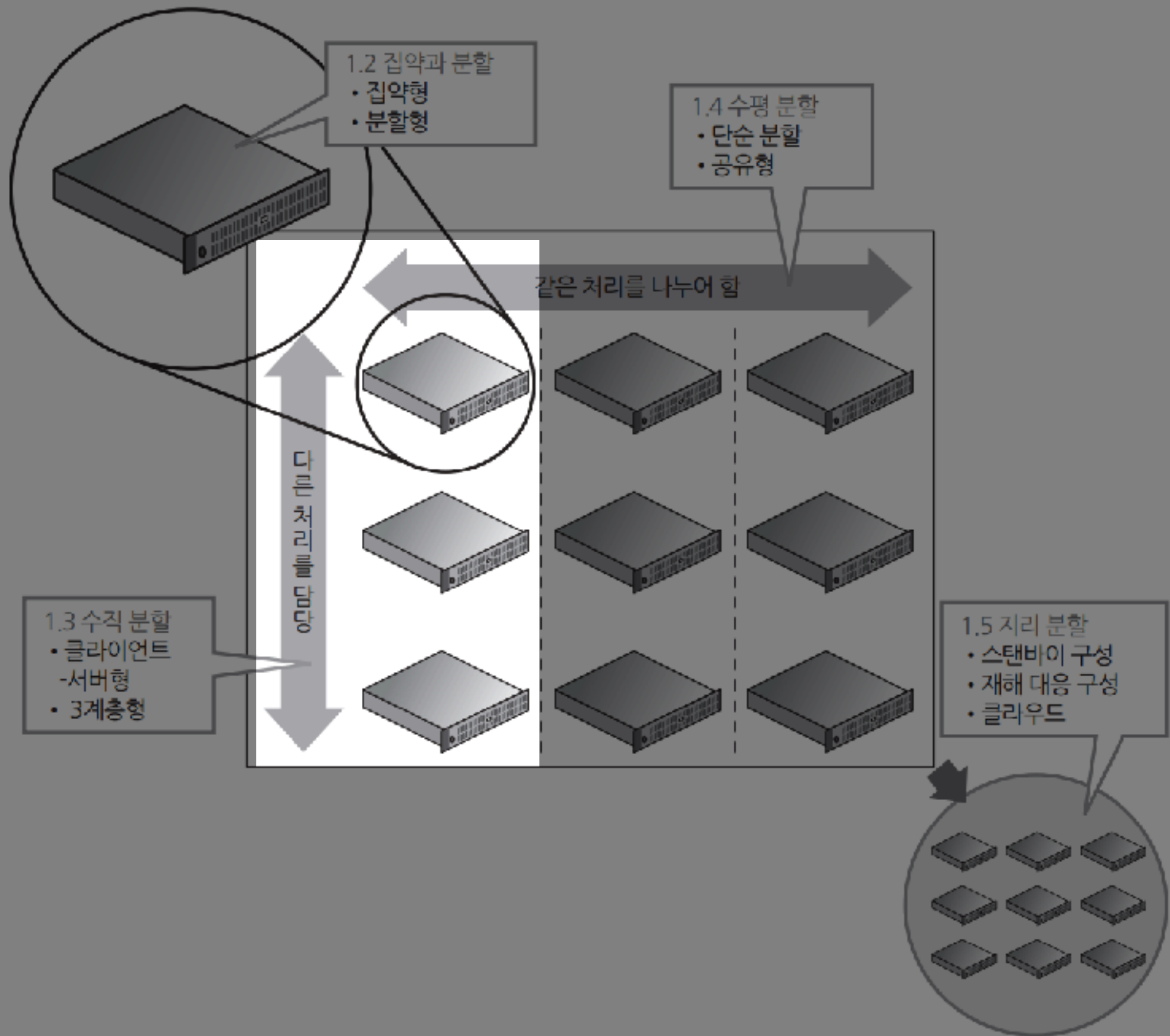
서버: 특정 역할에 특화된 것

분할형 아키텍트에 사용하는 컴퓨터=서버라고 함

물리 서버 : 동작 하고 있는 물리적인 컴퓨터

논리 서버: 컴퓨터 내부에서 동작하는 소프트웨어들

ex) 웹서버, DB서버 ...



수직 분할형 아키텍처

분할 된 서버 별로 다른 역할을 담당함
역할에 따라 위-아래 계층으로 나뉨

1. 클라이언트 - 서버형 아키텍처
2. 3계층형 아키텍처

수직 분할형 아키텍처 - 클라이언트 - 서버형

물리서버 상에서 주요 소프트웨어가 운영되고,
클라이언트(=단말)라 불리는 소형 컴퓨터가 접속하여 이용하는 형태.
클라이언트(pc, 모바일, 태블릿 등)에 전용 소프트웨어를 설치해야함

화면표시, 단순계산은 클라이언트에서 처리
데이터 입출력 등의 요청 서버에서 처리

수직 분할형 아키텍처 - 클라이언트 - 서버형

장점

1. 소수 서버로 다수 클라이언트 처리

단점

1. 클라이언트 측 소프트웨어 정기 업데이트 필요
2. 확장성에 한계

수직 분할형 아키텍처 - 3계층형

클라이언트- 서버형의 발전 모델

인터넷 대부분이 사용

프레젠테이션 계층 (웹 서버)

- 사용자 입력을 받는다

- 웹브라우저 화면 표시

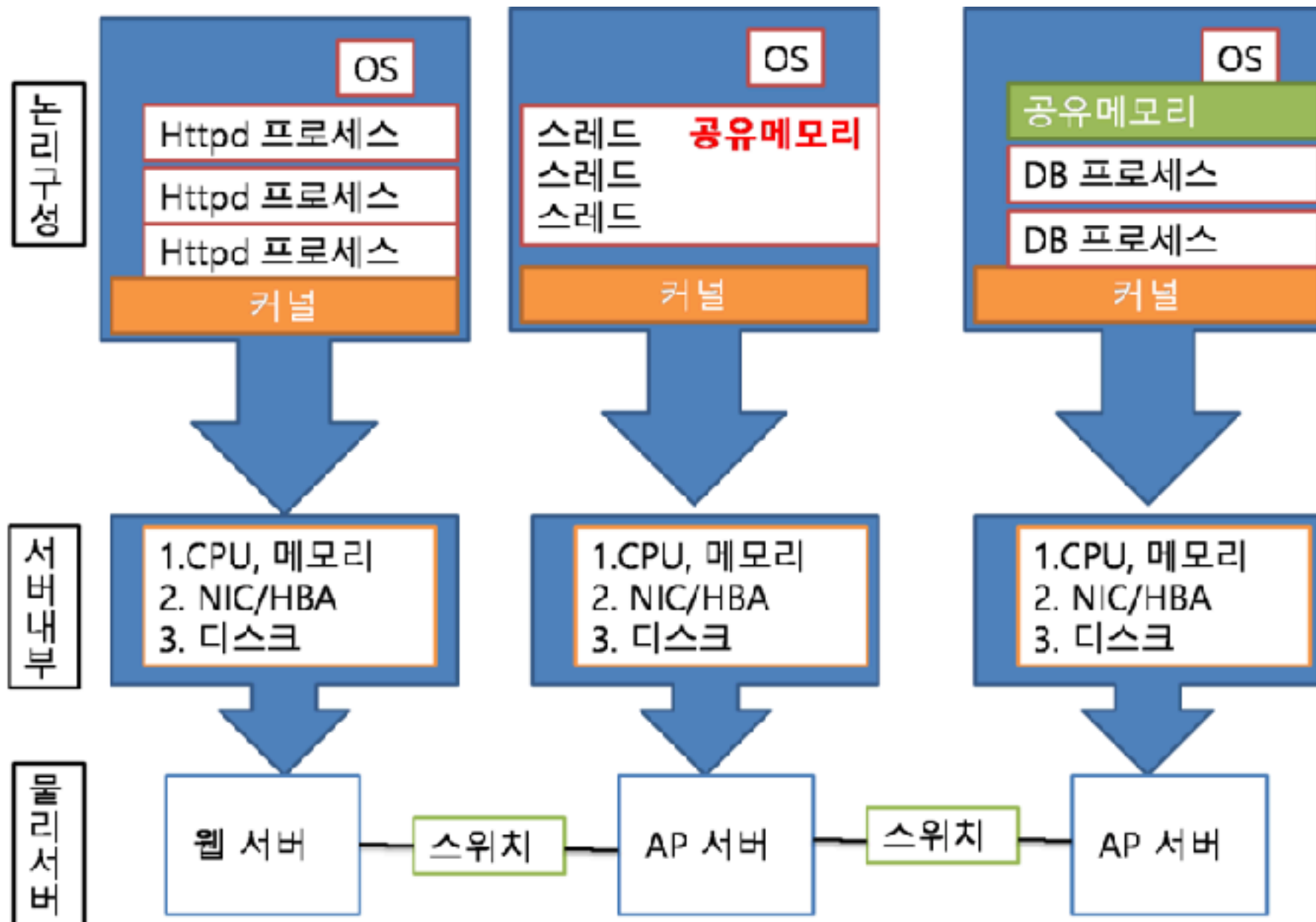
애플리케이션 계층 (애플리케이션 서버; AP서버)

- 사용자 요청에 따라 업무 처리

데이터 계층 (DB 서버)

- 애플리케이션 계층 요청에 따라 데이터 입출력

수직 분할형 아키텍처 - 3계층 시스템 구성도



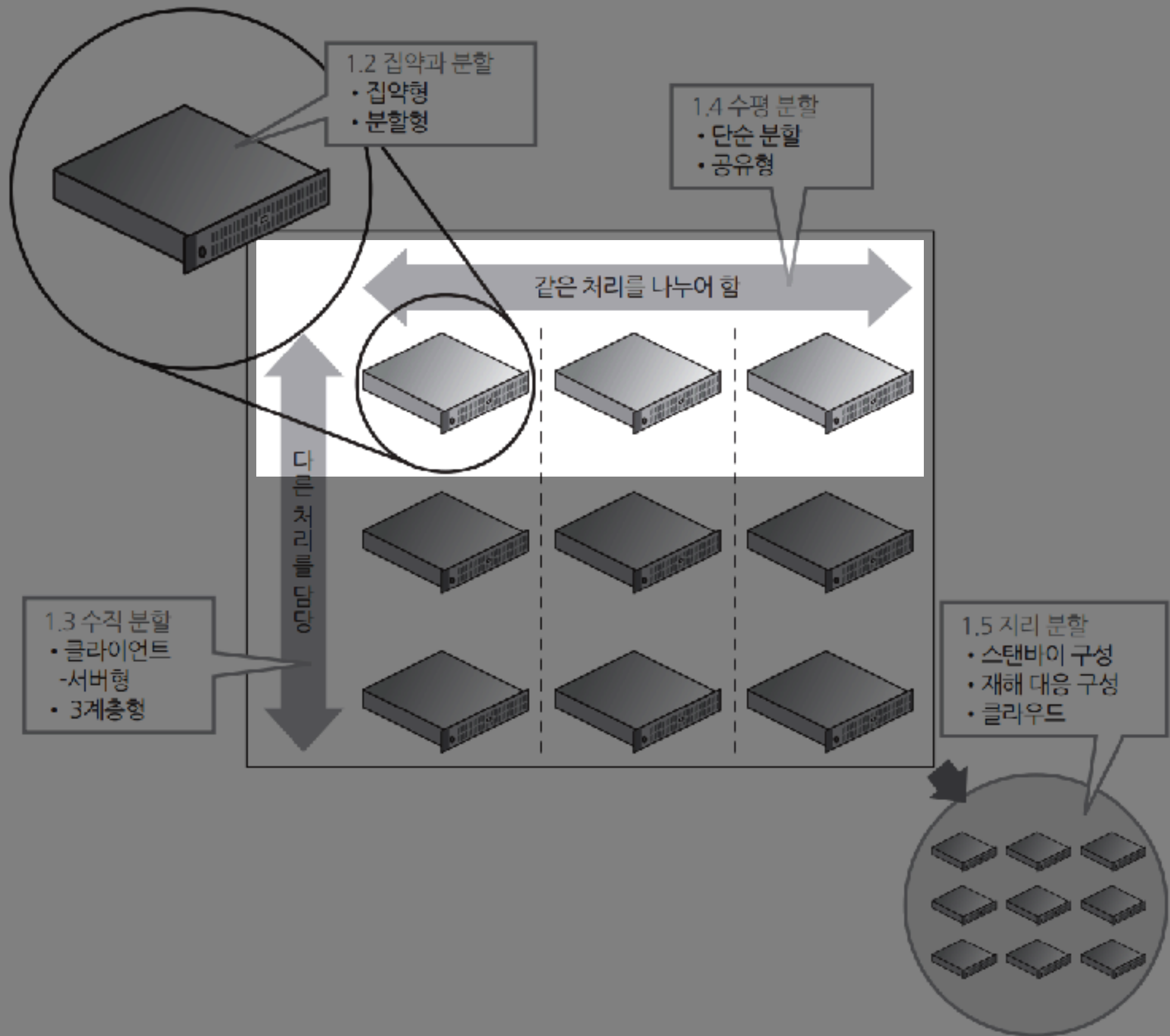
수직 분할형 아키텍처 - 3계층형

장점

1. 특정 서버에 부하 집중되는 문제 해결
2. 클라이언트의 정기 업데이트 불필요
3. '처리 반환'에 의한 서버 부하 절감(3계층까지 다 갈 필요 없음)

단점

1. 구조가 클라이언트-서버 구성보다 복잡



수평 분할형 아키텍처

용도가 같은 서버를 늘려나가는 방식
안정성, 성능 향상

1. 단순 수평 분할형
2. 공유형

수평 분할형 아키텍처 - 단순 수평 분할형

ex) 같은 구조지만 서울 본사와 부산 지사 2개의 시스템으로 분할

샤딩(sharding), 파티셔닝(partitioning)으로 부르기도 함

SNS 웹서비스에서 사용자 ID기준으로 서버 샤딩하는 경우 있음

지리적으로 멀리 떨어진 시스템에 적합

완전히 독립된 운영을 하고 있는 경우 적합

수평 분할형 아키텍처 - 단순 수평 분할형

장점

1. 확장성 향상 (서버 한대의 영향력 작아짐, 안정성과 성능 향상)
2. 분할한 시스템이 독립적으로 운영되므로 영향 주지 않음

단점

1. 데이터 일원화 불가
2. 애플리케이션 업데이트 동시에 해주어야함
3. 처리량 균등하지 않으면 서버에 따라 부하 집중 됨

수평 분할형 아키텍처 - 공유형

단순 분할형과 같지만 일부 계층에서 상호 접속 이루어짐

ex) 데이터 계층의 상호 접속

장점

1. 수평으로 서버 늘리기 때문에 확장성 향상
2. 분할한 시스템이 서로 다른 시스템의 데이터 참조 가능

단점

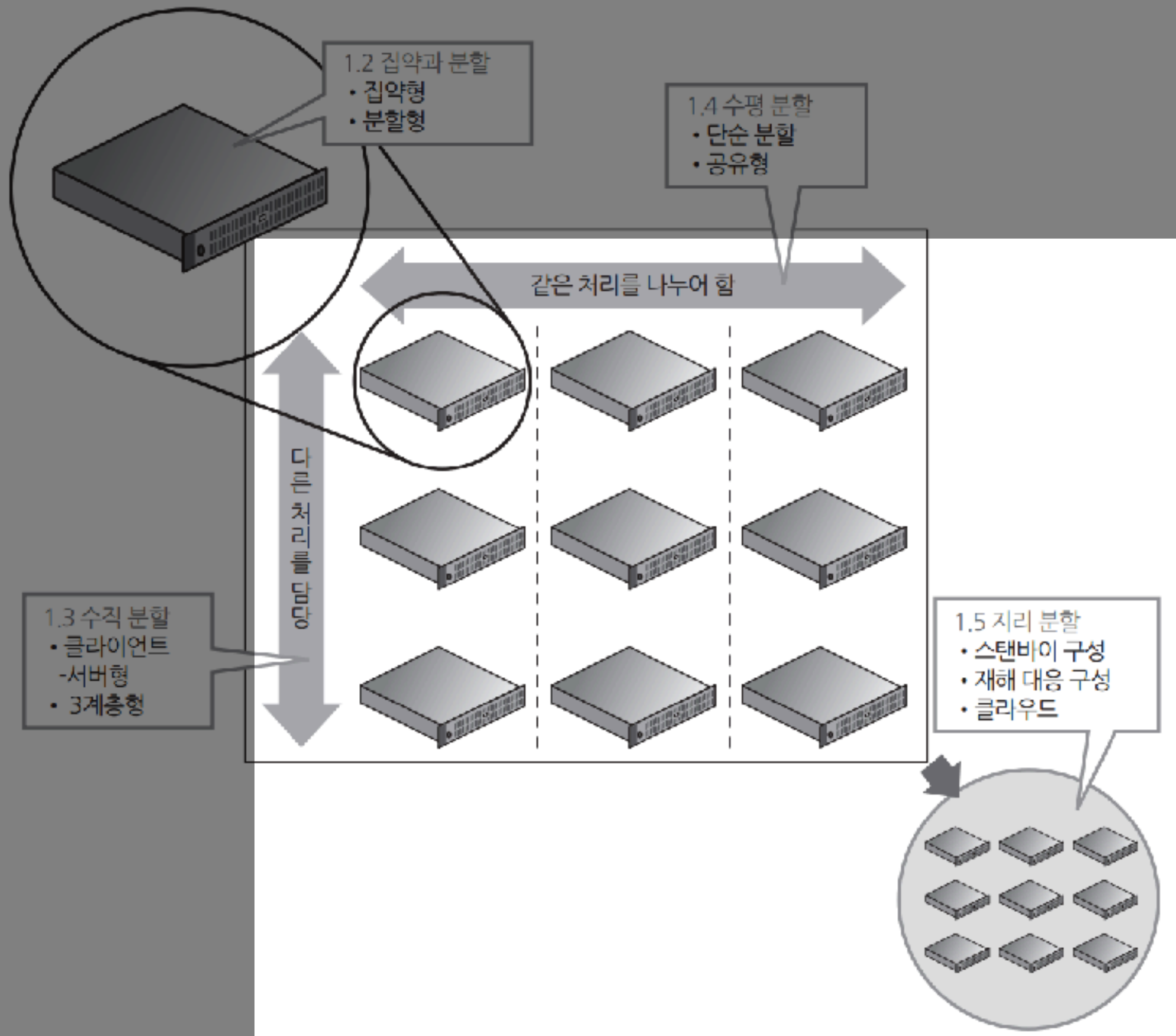
1. 분할한 시스템 간 독립성이 낮아짐
2. 공유한 계층의 확장성이 낮아짐

집약형과 분할형의 장점만 취하는 ‘가상화’

물리 서버를 가상화 기능으로 여러 대의 가상 서버로 분할하는 방식
+물리서버의 이중화

장점

- 1.물리서버 이중화로 안정성 향상
- 2.가상서버의 환경을 옮길 수 있어 확장성 보장



지리 분할형 아키텍처

수직, 수평 분할 아키텍처를 조합하여 목적에 적합한 구성

- 1.스탠바이형 아키텍처
- 2.재해 대책형 아키텍처
- 3.클라우드형 아키텍처

스탠바이형 아키텍처

=HA(High Availability) 구성, 액티브-스탠바이 구성

물리서버 최소 두대 준비하여 한 대 고장나면 소프트웨어를 다른 서버로 옮겨서 운영

-이 때 소프트웨어 재시작을 자동으로 하면 '페일오버'

-리소스 낭비 때문에 스탠바이를 따로 두지 않고 양쪽 서버 교차 이용하기도 함

재해 대책형 아키텍처

특정 환경 고장 시 다른 사이트의 재해 대책 환경에서 업무 처리
재개함

ex) 서울본사 데이터 센터 고장 시 부산 지사 시스템이용

어플리케이션 최신화

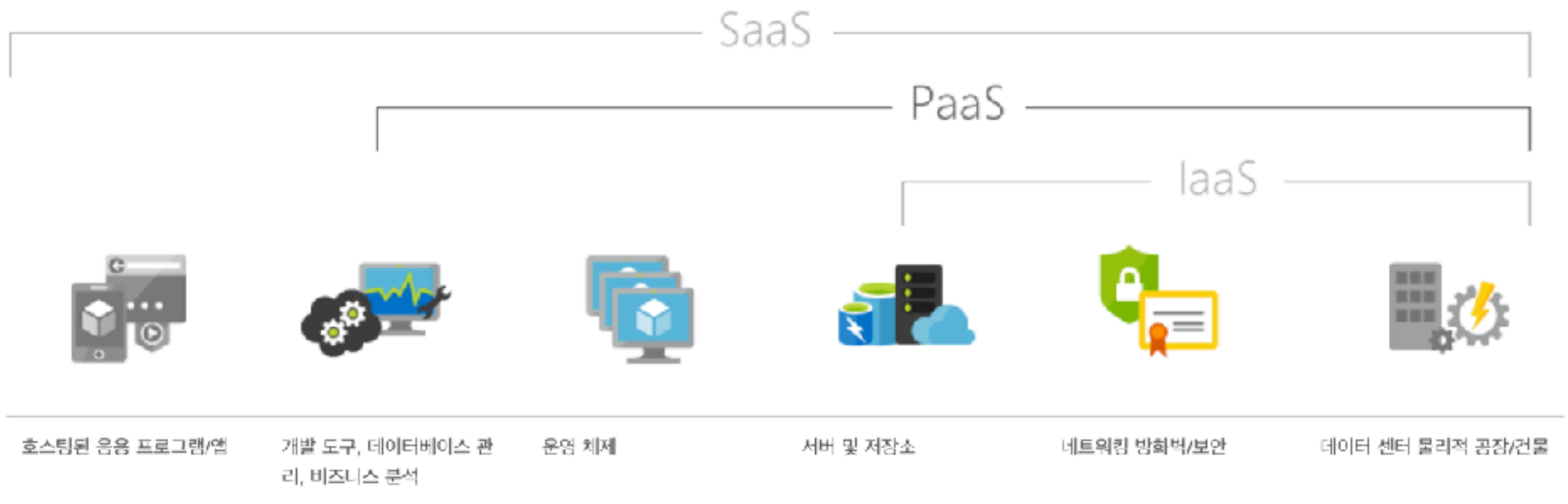
사이트 간 데이터 동기 처리 필요

클라우드형 아키텍처

3계층형 시스템의 일부 또는 전부가 클라우드 서비스가 보유한 물리서버에서 동작

인프라 고려하여 설계할 필요 있음

ex) 물리서버 구입에 비해 간단, 비용 낮지만 보안문제



:) 1장 끝~