

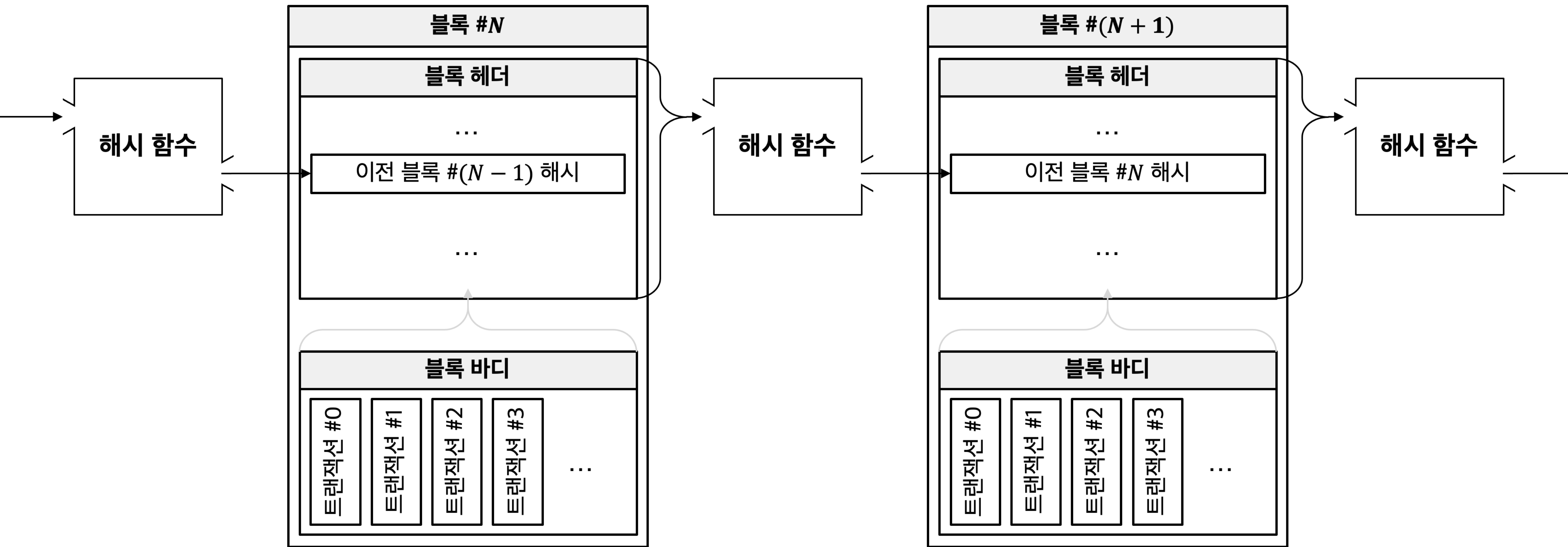
WHAT IS BLOCKCHAIN

TRANSACTION

ABSTRACT

- ▶ 블록체인(blockchain): 블록(block)의 체이닝(chain-ing)
 - ▶ 블록은 바디라는 트랜잭션들의 집합과 헤더라는 메타데이터로 구성
 - ▶ 현재 블록의 헤더에 이전 블록의 해시값을 포함함으로써 체인 구조를 형성

ABSTRACT



TRANSACTION

TRANSACTION

- ▶ 데이터베이스 관리 시스템(Database Management System, DBMS)
- ▶ 트랜잭션(transaction): 상호작용 및 작업 수행의 논리적 단위
 - ▶ DBMS는 트랜잭션에 대해 ACID 속성들을 보장하고자 한다
 - ▶ Haerder, Theo, and Andreas Reuter.
"Principles of transaction-oriented database recovery."
ACM computing surveys (CSUR) 15.4 (1983): 287-317.

ACID

- ▶ 원자성(Atomicity)은 트랜잭션과 관련된 작업이 모두 수행 or 모두 수행되지 않음을 보장
 - ▶ 자금 이체는 성공하거나 실패할 수만 있음
 - ▶ 한쪽에서 돈을 출금했으나 다른 쪽에서 입금에 실패하는 등 중간 단계까지만 수행되는 일이 없어야 함

ACID

- ▶ 일관성(Consistency) 있는 DBMS는 트랜잭션 처리 이후에도 일관성 있는 DB 상태를 유지
 - ▶ 무결성을 해치는 트랜잭션은 중단
 - ▶ 계좌는 0 이상의 잔액을 가져야 한다는 제약이 있는 경우, 가진 돈보다 큰 금액을 송금하는 트랜잭션은 중단

ACID

- ▶ 고립성(Isolation)은 한 트랜잭션의 처리 도중에 다른 작업이 끼어들지 못하도록 보장
 - ▶ 트랜잭션 외부에서는 중간 단계 상태를 읽을 수 없음
 - ▶ 직렬성(serializability): 여러 트랜잭션의 처리 결과가 순차 처리했을 때와 같음

ACID

- ▶ 영구성(Durability)은 성공적으로 처리된 트랜잭션의 작업이 DB에 영구적으로 반영됨을 보장

ACID

- ▶ 트랜잭션이 안전하게 처리됨을 보장
- ▶ 실제 서비스 환경에서는 성능 향상 등의 이유로 ACID를 유연하게 적용
 - ▶ 고립성 수준(isolation level)을 완화해 병행처리를 허용
 - ▶ 초당 트랜잭션 처리량(Transaction Per Second, TPS)을 증대

ISOLATION LEVEL

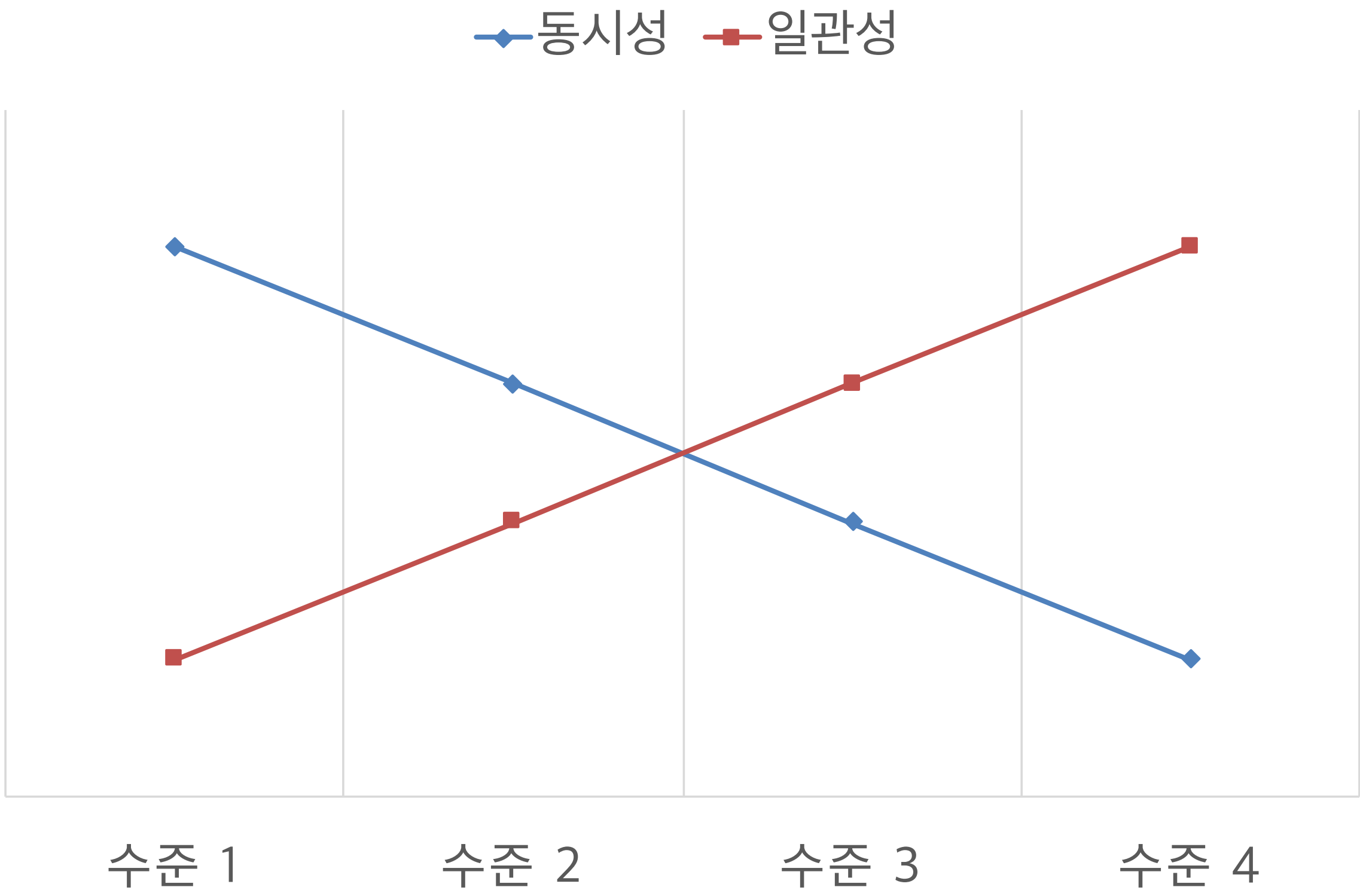
- ▶ ANSI SQL-92 표준에 따라 고립성 수준과 발생할 수 있는 현상을 정리
- ▶ <http://www.contrib.andrew.cmu.edu/%7Eshadow/sql/sql1992.txt>
- ▶ Berenson, Hal, et al. "A critique of ANSI SQL isolation levels." ACM SIGMOD Record 24.2 (1995): 1-10.

ISOLATION LEVEL

고립성 수준		현상		
		오염된 읽기 (Dirty Read)	비-반복가능한 읽기 (Non-repeatable Read)	유령 데이터 (Phantom)
1	비확정 데이터 읽기 (Read Uncommitted)	발생 가능	발생 가능	발생 가능
2	확정된 데이터 읽기 (Read Committed)	불가능	발생 가능	발생 가능
3	반복 가능한 읽기 (Repeatable Read)	불가능	불가능	발생 가능
4	직렬화 (Serializable)	불가능	불가능	불가능

ISOLATION LEVEL

► Concurrency & Consistency



ACID VS BASE

BASE

- ▶ BASE는 ACID와는 대조적으로 일관성과 고립성을 희생하고 가용성과 성능을 중시
- ▶ Brewer, Eric A. "Towards robust distributed systems." PODC. Vol. 7. 2000.

BASE

- ▶ 기본적으로 가용한(Basically Available)
 - ▶ 다수의 저장소에 복사본을 저장하는 등의 방법으로 높은 수준의 가용성을 보장
 - ▶ 일부 서버가 장애 상황에 처하더라도 나머지가 요청을 처리

BASE

- ▶ 유연한 상태(Soft-state)
 - ▶ 유연한 상태 하에서 데이터 복사본은 일관적이지 않을 수 있음
 - ▶ 시스템의 상태가 시간에 따라 변화할 수 있음
 - ▶ ‘궁극적 일관성’에 의해 입력이 없는 상황에서도 상태가 변할 수 있음
- ▶ Chandra, Deka Ganesh. "BASE analysis of NoSQL database." Future Generation Computer Systems 52 (2015): 13-21.

BASE

- ▶ 궁극적 일관성(Eventually consistency)
 - ▶ 한 서버에서의 변경사항은 반드시 다른 장치들에게 전파되어야 함
 - ▶ 과정에서 상태는 일시적으로 비일관적일 수 있으나, 궁극적으로는 일관성을 찾음
- ▶ Sullivan, Dan. “NoSQL for mere mortals.” Addison-Wesley Professional, 2015.

ACID VS BASE

- Lotfy, Ayman E., et al. "A middle layer solution to support ACID properties for NoSQL databases." Journal of King Saud University-Computer and Information Sciences 28.1 (2016): 133-145.

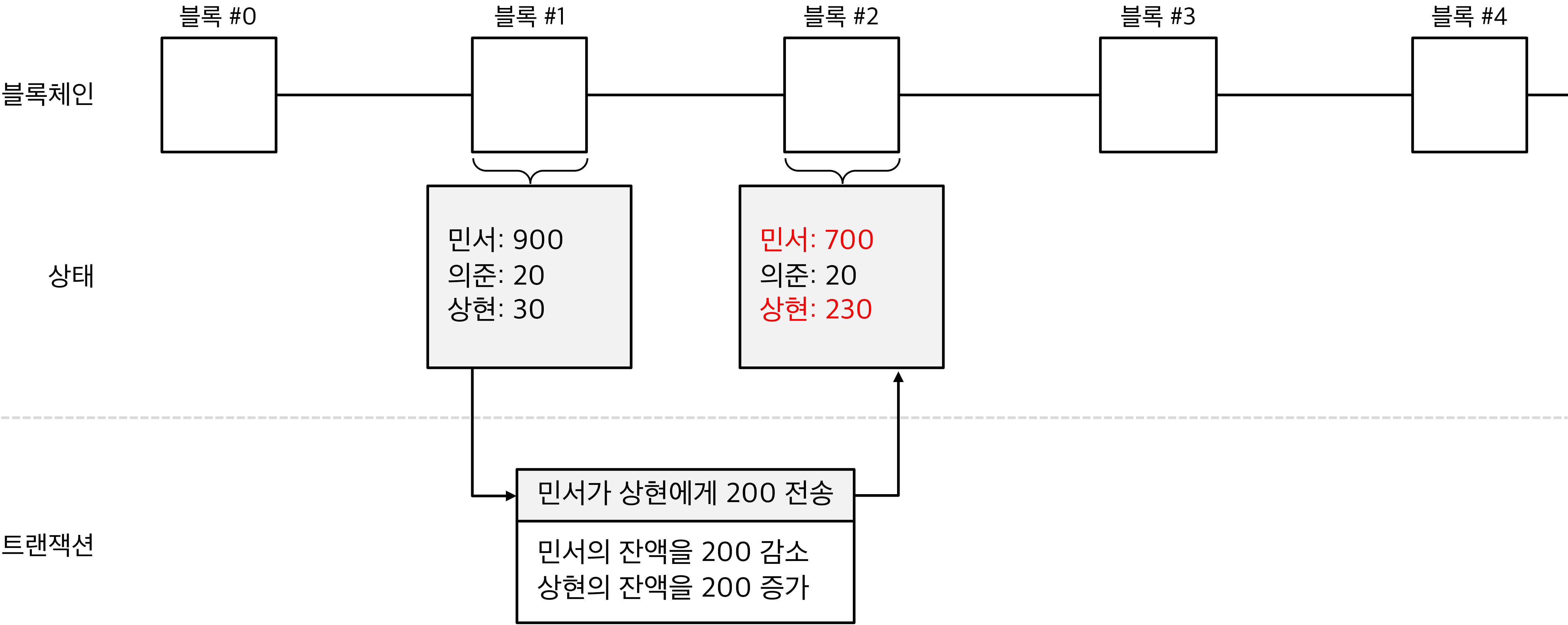
항목	ACID		BASE	
적용분야	RDBMS/일반적인 시스템		NoSQL/분산 시스템	
일관성	강한 일관성		약한 일관성: 오래된 데이터(state data)	
고립성	강한 고립성		약한 고립성	
병행수행	보수적		적극적(낙관적)	
중점사항	커밋: 엄격한 데이터 관리		성능 및 가용성: 대략적인(부정확한) 응답 허용	
수평적 확장성 (Horizontal Scalability)	낮음(어려움)		높음(쉬움)	
발전의 어려움 (Evolution difficulty)	어려움		상대적으로 쉬움	

BLOCKCHAIN TRANSACTION

BLOCKCHAIN TRANSACTION

- ▶ 블록체인의 트랜잭션 역시 상호작용 및 작업 수행의 논리적 단위
 - ▶ 블록체인상의 모든 활동은 트랜잭션을 통해 이뤄짐
 - ▶ 추상적인 관점에서 트랜잭션은 블록체인의 상태 변화를 야기하는 일련의 작업을 내포

BLOCKCHAIN TRANSACTION



BLOCKCHAIN TRANSACTION

- ▶ 블록체인은 근본적으로 분산 시스템
 - ▶ BASE의 특성을 보임

BLOCKCHAIN TRANSACTION

- ▶ 기본적으로 가용한
 - ▶ 블록체인 네트워크의 노드들은 각기 원장(ledger)의 복사본을 저장
 - ▶ 일부 노드의 장애 상황을 허용

BLOCKCHAIN TRANSACTION

- ▶ 유연한 상태
 - ▶ 각 노드가 저장 중인 원장은 일관적이지 않을 수 있음
 - ▶ 그러나 ‘궁극적 일관성’에 의해 점차 하나의 상태로 합의

BLOCKCHAIN TRANSACTION

- ▶ 궁극적 일관성
 - ▶ 트랜잭션과 블록은 송신자로부터 다른 노드들에게 브로드캐스트(broadcast)
 - ▶ 상태는 일시적으로 비일관적일 수 있음
 - ▶ 직/간접 참조하는 블록이 많아지면서 궁극적으로는 일관성을 찾음

BLOCKCHAIN TRANSACTION

- ▶ 궁극적 일관성에 의해 트랜잭션이 높은 수준의 일관성을 확보
 - ▶ ACID에 가까운 특성
 - ▶ 합의된 과거의 상태를 신뢰할 수 있음

BLOCKCHAIN TRANSACTION

- ▶ 강한 고립성
 - ▶ 블록체인의 트랜잭션은 블록에 담긴 순서에 따라 순차적으로(직렬적으로) 수행
 - ▶ 가장 높은 수준의 고립성

BLOCKCHAIN TRANSACTION

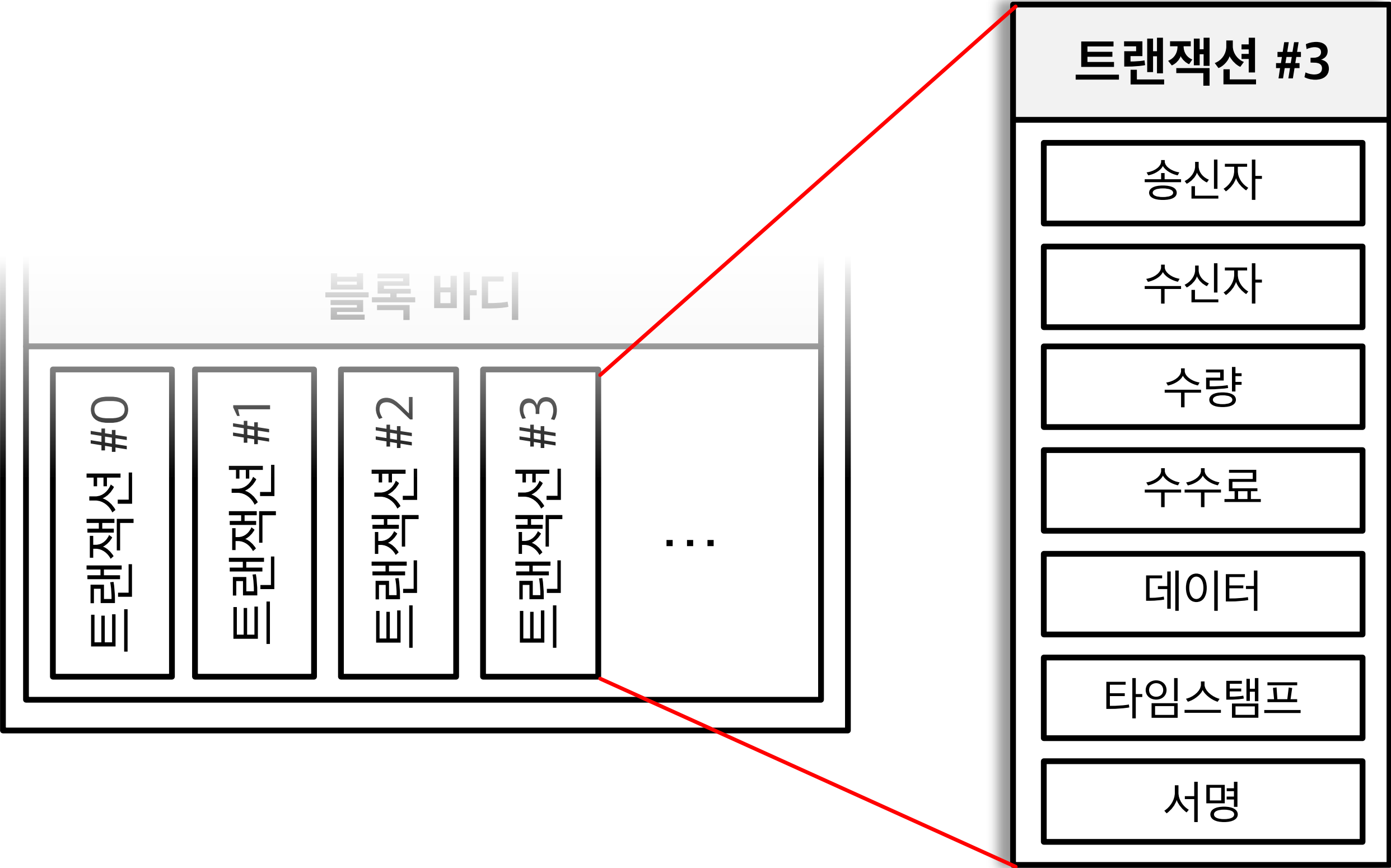
- ▶ 보수적 병행수행
 - ▶ 트랜잭션은 순차적으로 수행
 - ▶ 이전 트랜잭션의 처리가 완료되기 전까지 다음 트랜잭션은 수행될 수 없음

BLOCKCHAIN TRANSACTION

- ▶ 영구성

- ▶ 트랜잭션은 블록에 담겨 모든 다음 블록들의 상태에 영구적으로 반영

BLOCKCHAIN TRANSACTION



WHAT IS BLOCKCHAIN

TRANSACTION