SEMI-LAYER-2 PROTOCOLS

& HYBRID LAYER-2 PROTOCOLS

COMPRESSION-CHAINS

ROLL-UP

- 압축-체인(compression-chain) 기법
  - 온체인 트랜잭션의 크기를 줄이는 것에 초점
- 롤업(Roll-up)
  - 최소한의 신뢰를 가진 사이드 체인(side chain)

ROLL-UP

- 롤업에서 트랜잭션은 단 9바이트로 온체인에 등록
  - 서명을 분리
  - 서명의 유효성 등을 영지식증명(ZKP)을 통해 입증

ROLL-UP

- 커밋-체인
  - 운영자가 중앙화되어 있으므로 가용성을 보장할 수 없음
  - 그러나 탈출할 수는 있으므로 잔액 보안 속성은 제공
- 롤업
  - 데이터 가용성 문제를 해결함
  - 운영자가 악의적인 행동을 애초에 할 수 없음

ROLL-UP

- 롤업은 순수한 레이어-2 프로토콜이 아님
  - Semi-layer-2 protocol
- 확장성도 다르게 고려해야 함

SEMI-LAYER-2 PROTOCOLS

SEMI-LAYER-2 PROTOCOLS

- 상태 채널이나 커밋-체인
  - 레이어-2 확장성 솔루션
- 어플리케이션 설계가 복잡함
  - 특정한 사용처에서는 잘 작동
    - 페이먼트 채널
    - 플라즈마 캐시
  - 일반화가 어려움

SEMI-LAYER-2 PROTOCOLS

- 반(半)-레이어-2 프로토콜
  - 확장성의 이득이 적지만
  - 쉬운 일반화와
  - 보안 모델에 적합

SEMI-LAYER-2 PROTOCOLS

- 데이터 가용성을 검증하기 위해 주 체인이 사용됨
- 온체인 데이터 처리 속도가 곧 오버헤드이자 병목
  - 큰 확장성 이득은 없음 (수십-수백 배 가량)

SHADOW CHAIN

SHADOW CHAINS

- 그림자 체인 (shadow chains)
  - (그림자 체인의) 블록 데이터가 온체인에 공시되지만
  - 기본적으로 검증되지 않는 구조

SHADOW CHAINS

- (그림자 체인의) 블록은 임시적으로 받아들여 짐
  - 일정 기간(가령 2주)이 지나면 완결
  - 임시 기간 동안 챌린지를 받을 수 있음
- 챌린지를 받으면 블록 검증이 시작
  - 블록이 유효하지 않으면 해당 블록으로부터의 체인이 번복
  - 공시자의 보증금이 처벌됨

SHADOW CHAINS

- 전체 상태를 저장하지 않음
  - 상태 루트(root)만을 저장
- 사용자들은 스스로
  - 등록된 데이터(트랜잭션)를 처리해
  - 상태를 계산할 수 있음

SHADOW CHAINS

- 그림자 체인과 ZK 롤업과의 유사성을 강조하기 위해
  - 최근에는 (확장된) 그림자 체인을
  - 낙관적(optimistic) 롤업이라 부름

ZK ROLL-UP

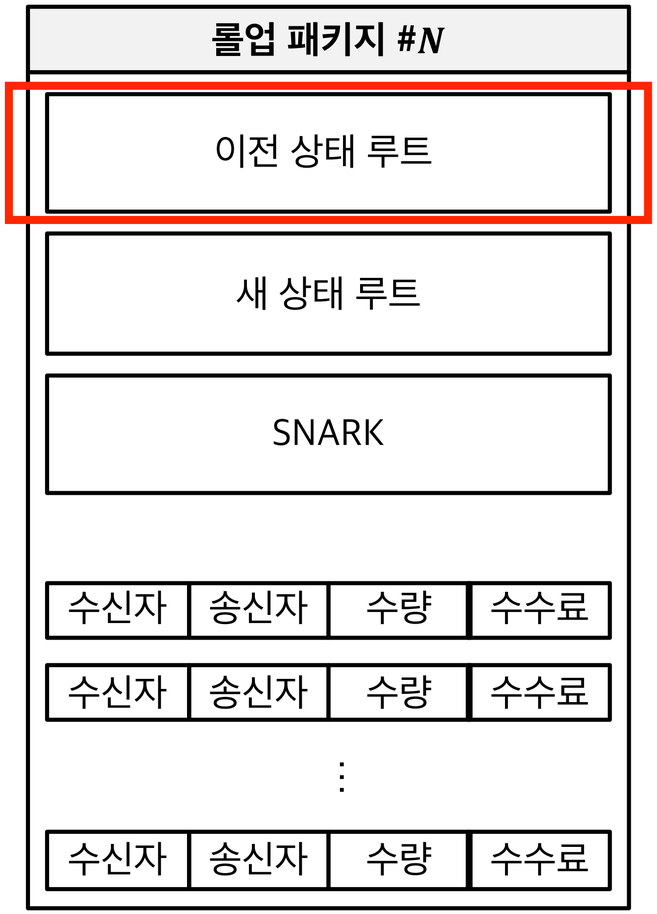
ZK ROLL-UP

- ZK 롤업
  - 챌린지 기간 없이 동일한 서비스 제공
  - 블록의 유효성 검증을 위해 zk-snarks 사용

ZK ROLL-UP

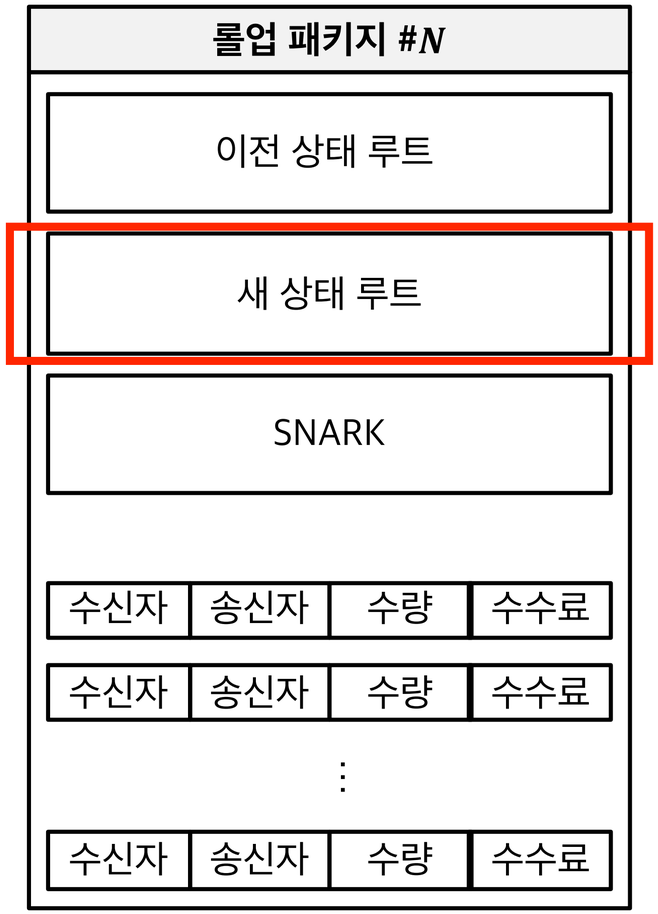
- ZK 롤업 시스템의 상태(즉, 계좌 잔액)
  - (수 백개의) 내부 트랜잭션들을 통해 상태 전이
  - 트랜잭션은 각 10바이트 정도의 크기
- 트랜잭션이 모두 유효함은 SNARK를 통해 압축된 형태로 등록
  - 대략 100~300 바이트

ZK ROLL-UP



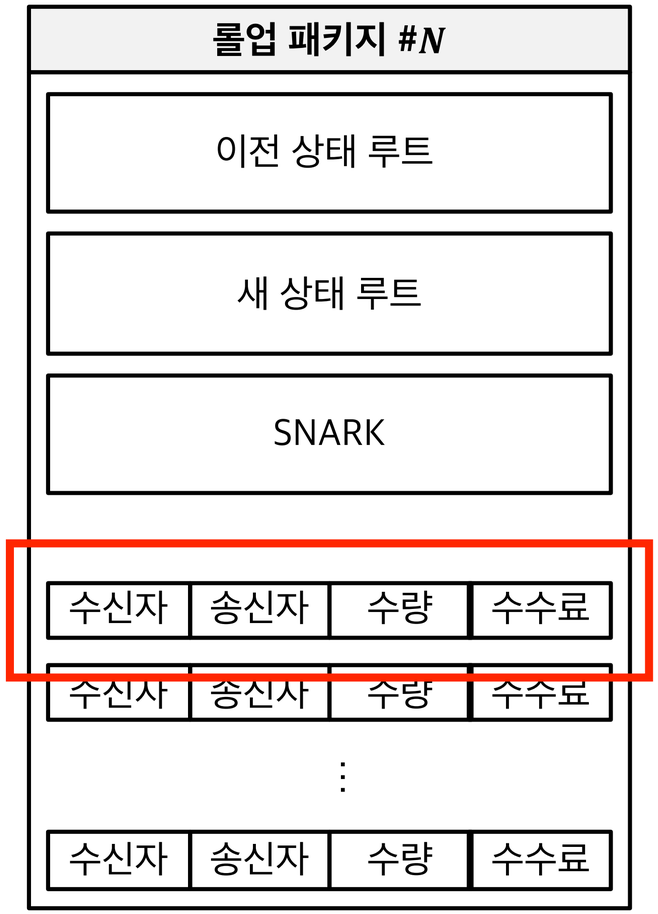
- 이전 상태 루트
  - ZK 롤업 컨트랙트는
  - 스마트 컨트랙트에 등록된 가장 최신 상태 루트가
  - 이전 상태 루트인 패키지만을 수용

ZK ROLL-UP



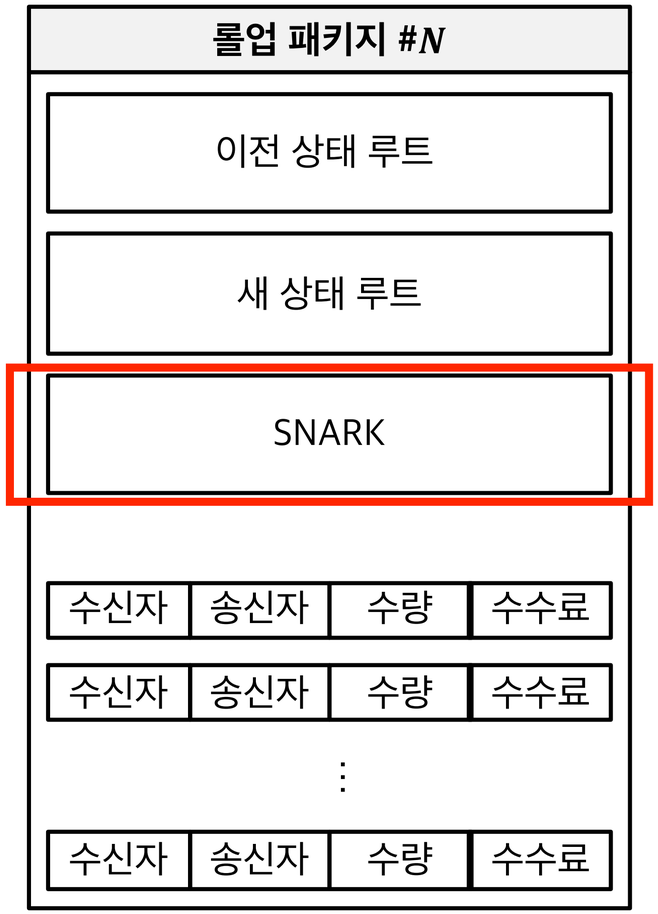
- 새 상태 루트
  - 패키지가 수용되면
  - 새 상태 루트가 컨트랙트에
  - 최신 상태 루트로 등록됨

ZK ROLL-UP



- 델타(delta)
  - 상태 델타
  - 어느 한 사용자가 개인적으로
  - 업데이트된 전체 상태를 계산하기에 충분한 정보
  - 패키지는 수 백 이상의 델타를 포함할 수 있음

ZK ROLL-UP



- SNARK
  - 짧은 (100-300 바이트) 암호학적 증명
  - 이전 상태 루트가 맞고
  - 델타를 적용하면 결과 상태 루트가
  - 새 상태 루트가 됨을 입증

ZK ROLL-UP

- BLS 통합(aggregate) 서명을 사용하면
  - 그림자 체인과 비슷한 (연산) 처리량으로
  - ZK 롤업 수행 가능
- 반대로, 낙관적 롤업 역시 BLS 통합 서명을 통해
  - 비슷한 수준의 확장성을 달성 가능

ZK ROLL-UP

- 트랜잭션을 10바이트 정도로 압축해
  - 500 TPS 달성 가능
- 이더리움 이스탄불(Istanbul) 하드포크로
  - 데이터(Calldata) 가스 비용이 바이트당 68에서 16으로 감소
  - ZK 롤업의 처리량을 4배 향상
  - 2,000 TPS 이상

DATA ON-CHAIN TECHNOLOGIES

DATA ON-CHAIN TECHNOLOGIES

- 데이터 온체인 기술
  - 블록(트랜잭션, 델타)이 온체인에 등록되는
  - ZK/낙관적 롤업 등을 지칭
- 데이터 오프체인 기술
  - 블록(트랜잭션, 델타)이 온체인에 등록되지 않는
  - 플라즈마 등을 지칭

DATA ON-CHAIN TECHNOLOGIES

- 데이터 온체인 기술 vs. 데이터 오프체인 기술
- 데이터 온체인 기술의 장점
  - 반(半)-신뢰 운영자가 필요하지 않음
  - 일반화하기 쉬움
  - 레이어-2 네트워크 인프라 요구사항이 적음

DATA ON-CHAIN TECHNOLOGIES

- 반(半)-신뢰 운영자가 필요하지 않음
- ZK 롤업에서 패키지의 유효성은 암호학적인 증명으로 검증
  - 악의적인 행동을 할 수 없음
  - 몇 초간 시스템을 중단시킴이 가장 해로운 행동

DATA ON-CHAIN TECHNOLOGIES

- 반(半)-신뢰 운영자가 필요하지 않음
- 낙관적 롤업에서는 나쁜 블록(패키지) 공시가 가능
  - 다음 제출자가 블록을 공시하기 전에 챌린지를 걸 것

DATA ON-CHAIN TECHNOLOGIES

- 반(半)-신뢰 운영자가 필요하지 않음
- ZK/낙관적 롤업에서
  - 제출된 델타들을 순서대로 처리하면
  - 누구나 내부 상태를 계산할 수 있을 정도로 충분한 데이터를
  - 온체인에 공시

DATA ON-CHAIN TECHNOLOGIES

- 일반화하기 쉬움
- 상태 전이 함수는 어느 것이나 될 수 있음
  - 페이먼트 이외의 기능
  - 한 블록의 가스 제한 안에서

DATA ON-CHAIN TECHNOLOGIES

- 일반화하기 쉬움
- ZK-SNARKs를 일반 목적의 연산으로 만들기 어렵긴 하지만
  - ZK 롤업 역시 이론적인 일반화는 충분히 가능

DATA ON-CHAIN TECHNOLOGIES

- 레이어-2 네트워크 인프라 요구사항이 적음
- 클라이언트 구축이 쉬움
  - 블록체인을 스캔하기만 하면 됨

DATA AVAILABILITY PROBLEM

DATA AVAILABILITY PROBLEM

- 데이터 온체인 기술의 장점은
  - 데이터 가용성으로부터 제공됨
- 데이터 가용성 문제
  - 고도의 기술적 이슈

DATA AVAILABILITY PROBLEM

- 레이어-2 시스템에서 사기를 시도하는 두 방법
  - 블록체인에 유효하지 않은 데이터를 공시
  - 아예 데이터를 공시하지 않음
    - 가령, 플라즈마에서
    - 새 플라즈마 블록의 루트 해시를 주 체인에 공시하지만
    - 블록의 내용을 공개하지 않음

DATA AVAILABILITY PROBLEM

- 공시되었지만 유효하지 않은 데이터
  - 다루기 쉬운 문제
- 데이터가 온체인에 공시되면
  - 유효 여부를 명확하게 가리는 여러 방법이 있음
- 유효하지 않은 제출은 명확히 유효하지 않기 때문에 중징계 가능

DATA AVAILABILITY PROBLEM

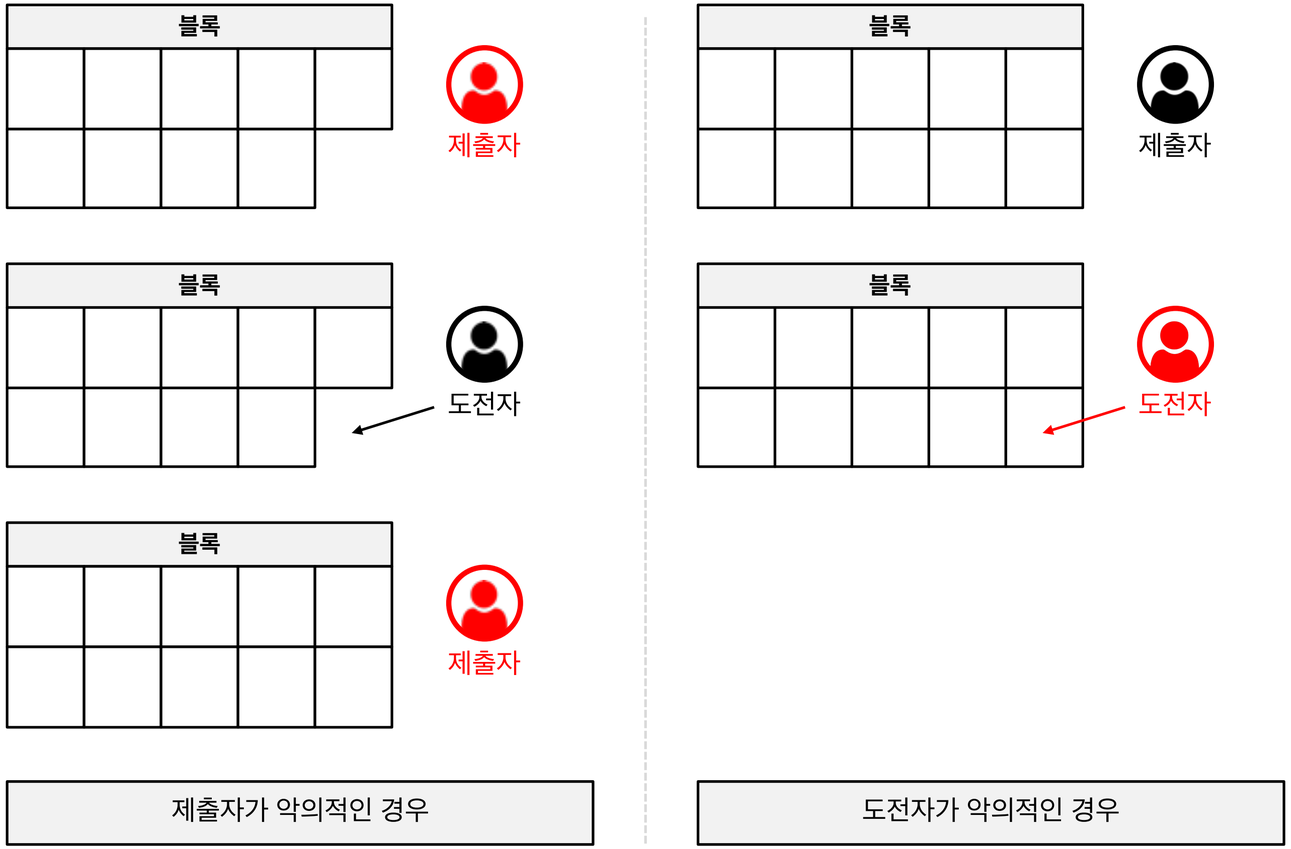
- 비가용한 데이터는 다루기 훨씬 어려움
  - 비공시(non-publication)가 누구의 잘못인지 판단이 어려움
- 특히 데이터가 기본적으로 보류되다가
  - 어떠한 검증을 위해 온디멘드(on-demand)하게 공개되는 경우
  - 더욱 어려움
  - 낚시꾼의 딜레마

FISHERMAN'S DILEMMA

- 낚시꾼의 딜레마 (Fisherman's dilemma)
  - 도전-응답(challenge-response) 게임이
  - 어떻게 악의적인 제출자와 악의적인 도전자를
  - 구분할 수 없는가

FISHERMAN'S DILEMMA

- 제출자와 도전자 중 누가 잘못했는지를 알 수 없는 상황



DATA AVAILABILITY PROBLEM

- 데이터 오프체인 기술
  - 플라즈마와 채널
  - 문제를 사용자에게 전가해 낚시꾼의 딜레마 해결

DATA AVAILABILITY PROBLEM

- 데이터 오프체인 기술
- 상호작용중인 다른 사용자가 공시해야 할 데이터를 공시 않으면
  - 채널에서 상대 거래자
  - 플라즈마 체인에서 운영자
  - 탈출은 개인의 책임

DATA AVAILABILITY PROBLEM

- 데이터 오프체인 기술
- 사용자로서 모든 이전 데이터를 가지고 있고
- 서명한 모든 트랜잭션들에 관한 데이터를 가지고 있어야 함
- 레이어-1 주 체인에게
  - 레이어-2 프로토콜에서 어떤 자산을 가지고 있었는지를 증명하기 위해
  - 그래야만 안전하게 레이어-2 시스템 밖으로 가져올 수 있음

DATA AVAILABILITY PROBLEM

- 데이터 오프체인 기술
- 모든 상태 객체는 논리적인 소유자(owner)를 가지고 있으며
  - 소유자의 동의 없이는 객체의 상태를 변경할 수 없다는 가정에 의존
- UTXO 기반의 지불에서는 잘 작동
  - 계좌 기반의 지불에서는 그렇지 않음
  - 다른 사람의 동의 없이 잔액을 늘어나게 편집할 수 있기 때문

DATA AVAILABILITY PROBLEM

- 데이터 오프체인 기술
- 다른 사람의 동의 없이 잔액을 늘어나게 편집할 수 있기 때문
- 유니스왑(Uniswap)과 같은 어플리케이션
  - 풀(pool)에 소유자가 없음
  - 풀의 잔액을 조정해 단가를 조정할 수 있음
- 심지어 합법적으로 객체 편집자가 여럿 있는 경우는 어떻게 반영하는가?

DATA AVAILABILITY PROBLEM

- 데이터 오프체인 기술
- 제출자(공시자)의 과실 여부를 입증할 수 없으므로
  - DoS 공격의 가능성을 배제하며
  - 임의의 제삼자에게 탈출을 허용할 방법이 없음

DATA AVAILABILITY PROBLEM

- 데이터 오프체인 기술
- 채널 고유 이슈
  - 채널에 속해있지 않은 사람에게는
  - 오프체인 트랜잭션을 허용하지 않음
- 플라즈마 고유 이슈
  - 사용자들이 많은 양의 히스토리 데이터를 보유해야 함
  - DEX에서처럼 서로 다른 자산이 엮여있을 경우 더욱 커짐

DATA AVAILABILITY PROBLEM

- 데이터 온체인 기술
- 데이터 가용성 문제가 없음
  - ZK/낙관적 롤업은 충분한 데이터를 온체인에 올리고
  - 사용자들이 시스템 전체 상태를 계산할 수 있도록 함

DATA AVAILABILITY PROBLEM

- 데이터 온체인 기술
- 유일한 단점은 온체인 연산 없이 검증한다는 점
  - 이는 데이터 가용성 문제 대비 훨씬 쉬운 문제
  - 영지식증명 등

HYBRID LAYER-2 PROTOCOLS

HYBRID LAYER-2 PROTOCOLS

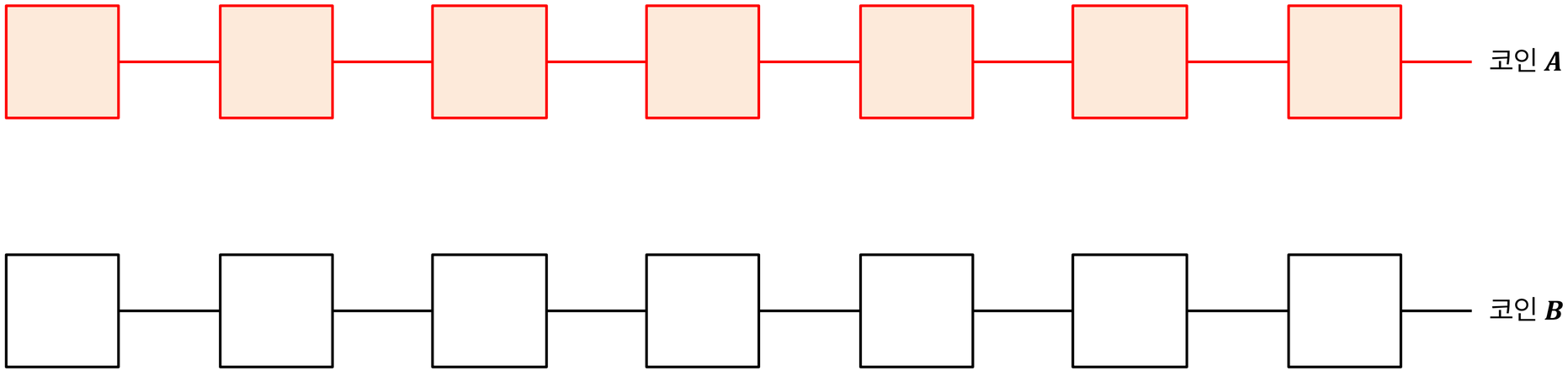
- 더 많은 확장성을 원한다면:
- 데이터-온체인 프로토콜과 데이터-오프체인 프로토콜 사이에는 큰 간극이 존재
  - 두 방법의 장점을 어느 정도 통합 제공하는
  - 하이브리드(hybrid) 접근 방식이 있음
  - Hybrid layer-2 protocols

HYBRID LAYER-2 PROTOCOLS

- 플라즈마 캐시의 DEX 구현체에서의 히스토리 폭발적 증대 문제
  - 온체인에 어떤 주문(orders)과 어떤 주문이 매칭되는지의
  - 매핑(mapping)을 공시해 방지할 수 있음

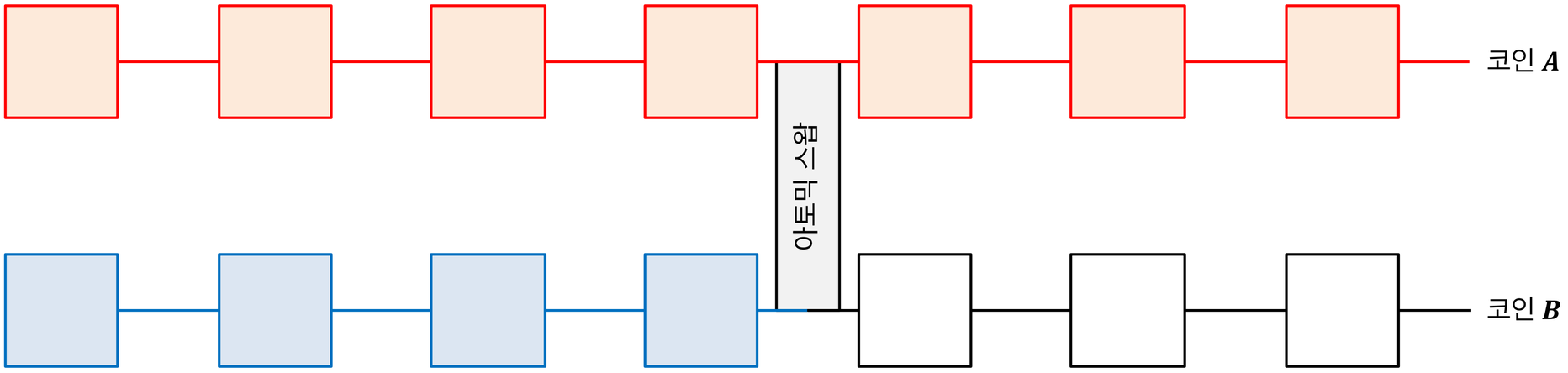
HYBRID LAYER-2 PROTOCOLS

- 플라즈마 캐시의 DEX 구현체에서의 히스토리 폭발적 증대 문제
  - 플라즈마 캐시 사용자가
  - 1코인을 소유하고 있는 경우
  - 필요로 하는 히스토리 데이터



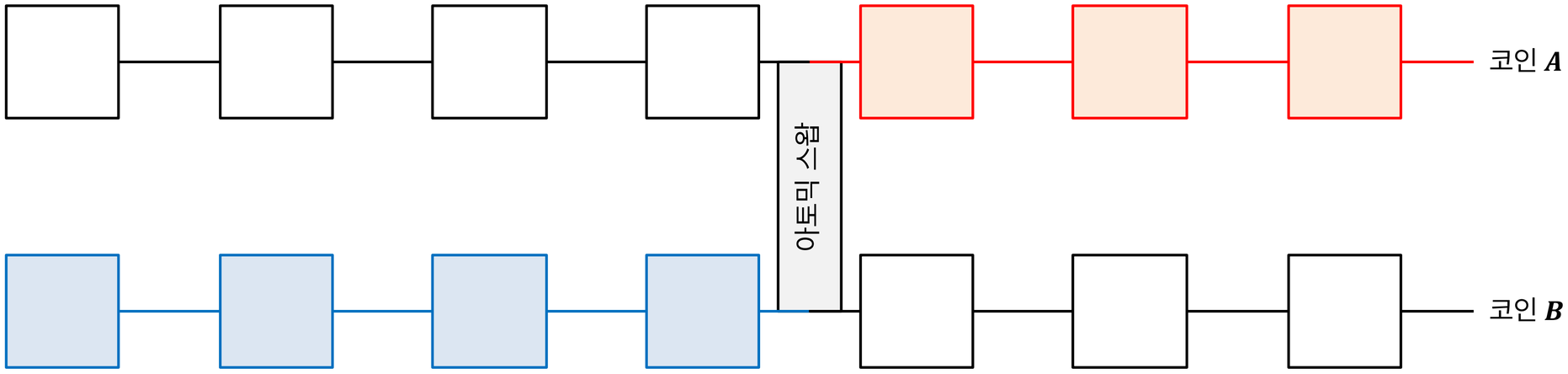
HYBRID LAYER-2 PROTOCOLS

- 플라즈마 캐시의 DEX 구현체에서의 히스토리 폭발적 증대 문제
  - 플라즈마 캐시 사용자가
  - 1코인을 아토믹 스왑(atomic swap)을 통해 다른 코인과 교환했을 경우
  - 필요로 하는 히스토리 데이터



HYBRID LAYER-2 PROTOCOLS

- 플라즈마 캐시의 DEX 구현체에서의 히스토리 폭발적 증대 문제
  - ~
  - 온체인에 주문 매칭이 공시된 경우
  - 필요로 하는 히스토리 데이터



HYBRID LAYER-2 PROTOCOLS

- 플라즈마 체인이
  - 주기적으로 어떠한 사용자별 데이터를 온체인에 게시하면
  - 플라즈마 사용자들이 필요로하는 히스토리의 양이 줄어듦

HYBRID LAYER-2 PROTOCOLS

- 플라즈마처럼 상태가 논리적 소유자를 가지고
- ZK/낙관적 롤업처럼 동작하는 플랫폼도 가능
  - 플라즈마 개발자들은 이미 이러한 종류의 최적화 작업에 착수
  - https://plasma.build/t/rollup-plasma-for-mass-exits-complex-disputes/90

HYBRID LAYER-2 PROTOCOLS

- 레이어-2 확장성 솔루션 개발자들이
- 사용자당 데이터를 온체인에 공시함으로써
  - 개발이 쉽고
  - 일반화가 쉽고
  - 보안성이 높고
  - 사용자당 부담을 줄일 수 있음

HYBRID LAYER-2 PROTOCOLS

- 하이브리드 접근 방식은
  - 상대적으로 빠르게
  - semi-layer-2 구조에서
  - 이더리움-형태의 스마트 컨트랙트 배포가
  - 가능하도록 할 것

SEMI-LAYER-2 PROTOCOLS

& HYBRID LAYER-2 PROTOCOLS