

엔소(Enso)

해당 번역본은 이용자가 디지털자산 관련 정보를 편리하게 확인할 수 있도록
단순 참고용으로 번역된 것이며, 투자 권유를 목적으로 하지 않습니다.
그 내용이 정확하지 않을 수 있으므로 정확한 정보 습득을 위해서는
원문을 참고하시거나 원문 작성 주체 측에 문의하시기 바랍니다.
두나무는 해당 번역본 내용의 진실성 및 정확성을 보장하지 않고,
제공된 정보에 의한 투자 결과에 대하여 어떠한 책임도 지지 않으며,
투자 손실은 투자자에게 귀속됩니다.

엔소(Enso): 인텐트 엔진

Connor Howe, Peter Phillips, Milos Costantini

2024년 7월

버전 0.1

초록(Abstract). 본 문서에서는 다수의 블록체인, 롤업(Rollup), 앱체인(AppChain)에 걸쳐 스마트 컨트랙트를 실행 가능한 바이트코드로 생성하는 분산형 공유 네트워크인 엔소(Enso)를 소개한다. 엔소는 텐더민트(Tendermint) 기반 레이어-1 블록체인으로 동작하며, 모든 스마트 컨트랙트 상호작용을 매핑하여 개발자가 단일 소스를 통해 어떤 체인의 스마트 컨트랙트와도 결합·상호작용할 수 있도록 한다. 엔소 네트워크는 개발자가 원하는 결과를 명시하면 네트워크 참가자들이 이를 충족할 수 있는 해법을 조율하는 방식으로, 블록체인 생태계의 핵심 문제인 **사용성(usability)**을 해결한다. 조율 계층으로서 엔소를 활용함으로써 개발자들은 최소한의 마찰로 진정한 의미의 조합 가능하고 중단 없는 애플리케이션을 구축할 수 있다.

1 문제 정의: 사용성(Usability)

블록체인 세계는 지속적으로 변화하고 있다. 한때는 비트코인(Bitcoin)이라는 단순한 공유 네트워크 상태였던 것이, 이제는 수천 개의 블록체인과 수십만 개의 스마트 컨트랙트로 이루어진 복잡한 생태계로 진화했다. 이 생태계가 성숙해질수록 그 수는 계속 증가할 것이며, 이는 산업 내 복잡성과 단편화를 더욱 가중시킨다. 이러한 복잡성과 단편화는 스마트 컨트랙트를 통합하려는 개발자들에게 큰 도전 과제를 안겨주며, 혁신적인 애플리케이션과 새로운 프리미티브를 구축하는 데 마찰을 발생시켜 대규모 채택을 위한 확장 가능성을 저해한다.

엔소는 현재 블록체인 생태계에서 가장 큰 문제를 **사용성**이라 보고하며, 이는 블록체인과 스마트 컨트랙트의 단편화에서 비롯된다. 우리는 조합 가능성(composability)이 블록체인 기술의 가장 중요한 가치 제안 중 하나라고 믿는다. 따라서 개발자가 스마트 컨트랙트를 단순하고 편리하게 상호작용하고 조합할 수 있도록 하는 것은 블록체인 생태계를 성장시키는 데 필수적이다.

본 논문에서 소개하는 엔소 네트워크는 바로 이 **사용성** 문제를 해결한다. 엔소 이전에는 개발자가 다른 스마트 컨트랙트와 상호작용하기 위해 수작업으로 스마트 컨트랙트를 작성하고, 각 스마트 컨트랙트의 세부적인 특성을 이해하며, 통합을 유지하기 위한 맞춤형 인프라를 구축해야 했다. 이제 엔소를 사용하면 개발자는 모든 스마트 컨트랙트 상호작용의 복잡성을 추상화하는 단일 소스를 통해 상호작용할 수 있다. 이는 블록체인, 롤업, 롤업(rollapp)에 걸쳐 모든 스마트 컨트랙트 상호작용을 저장·구성하는 **공유 네트워크(shared network)** 상태를 통해 실행 가능하도록 소비될 수 있게 함으로써 이루어진다. 아래에서는 블록체인 생태계의 진화를 간략히 살펴보고, 애플리케이션 구축 시 개발자들이 직면하는 지속적인 과제를 제시한다.

비트코인(Bitcoin)은 최초의 공유 네트워크 상태를 창출했으며, 이는 블록체인으로 널리 알려져 있다. 비트코인은 금융기관의 개입 없이도 거래를 처리할 수 있는 P2P 전자 화폐로 설계되었다. 대신 독립적인 노드들이 송신자와 수신자의 상태 전이를 검증했다. 모든 상태는 단일 체인에 존재했으며, 비트코인은 주로 결제에 초점을 두었고 탈중앙 애플리케이션에 초점을 맞추지 않았기 때문에 개발자가 비트코인 연동을 위해 수행해야 하는 역할은 오늘날과 비교하면 사소한 수준이었다.

이더리움(Ethereum)은 비트코인이 발전시킨 공유 네트워크 상태 개념을 확장하여, 탈중앙 애플리케이션(DApp)의 개발을 가능케 하는 스마트 컨트랙트를 도입했다. 스마트 컨트랙트는 이더리움 가상머신(EVM)에 온전히 존재하는 코드이며, 상태 전이는 임의의 상호작용이 될 수 있다. 이를 통해 탈중앙 금융(DeFi), 탈중앙 물리 인프라 네트워크(DePin), 대체불가능토큰(NFT), 데이터 큐레이션 마켓플레이스 등 새로운 애플리케이션 카테고리가 구축되었다. 이러한 각 애플리케이션 카테고리는 공유 네트워크 상태 덕분에 이더리움 블록체인 위의 다른 스마트 컨트랙트와 통신하고 상호작용할 수 있었다. 여러 스마트 컨트랙트 상호작용이 상호 얹히고 조합된 새로운 프리미티브도 이제 만들어질 수 있었다.

레이어-1 블록체인들(예: 솔라나(Solana), 네오(Neo), 니어(Near), 아발란체(Avalanche) 등)은 최적화, 합의(consensus), 프로그래밍 언어와 같은 기술적 구현의 변형을 도입하면서도 공유 네트워크 상태를 유지했다. 이를 통해 개발자들은 다른 프레임워크에서 실행되는 스마트 컨트랙트와 상호작용하는 애플리케이션을 구축할 수 있었다.

레이어-2 블록체인은 사용자가 원하는 블록체인과 상호작용할 때 발생하는 거래 비용 문제를 해결했다. 채택이 지속적으로 증가함에 따라 더 많은 트랜잭션을 다음 블록에 포함시키기 위한 요청이 늘어나고, 그에 따라 가스비는 계속 상승했다. Conduit, Gelato, AltLayer와 같은 프로젝트는 RaaS(Rollups-as-a-Service)로 알려진 소프트웨어를 개발하여 애플리케이션 개발자가 5분 안에 블록체인을 배포할 수 있도록 했다. 이는 배포되는 블록체인의 수를 가속화했지만, 동시에 스마트 컨트랙트 조합성의 단편화를 초래하여 개발자들이 스마트 컨트랙트를 쉽게 조합하지 못하게 만들었고, 공유 네트워크 상태가 여러 블록체인에 분산됨으로써 사용성 문제를 더욱 악화시켰다.

앱체인(AppChain)은 애플리케이션 특화 블록체인(application-specific blockchain)이라고도 불리며, 특정 애플리케이션만을 위한 독립 체인을 만든다. 이를 통해 합의, 가스비, 검증자 세트, 기타 필요한 기능을 맞춤화할 수 있다. 그러나 이러한 애플리케이션과 연동하려는 개발자는 단지 하나의 새로운 애플리케이션을 얻기 위해 새로운 블록체인을 통합해야 하는 매우 번거로운 작업을 수행해야 한다.

수년간의 진화는 **사용성(usability)** 문제에 대한 우려스러운 상황을 만들어왔다. 개발자는 이제 어떤 체인을 통합할지, 어떤 스마트 컨트랙트와 연동할지를 선택해야 한다. 일단 개발자가 원하는 블록체인을 선택하면, 각 스마트 컨트랙트의 내부 동작을 이해하고 이를 다른 스마트 컨트랙트와 어떻게 조합할지 파악해야 하는 거대한 과제에 직면한다. 스마트 컨트랙트와 상호작용하기 위해서는 이전 상태가 요구될 수 있으며, 따라서 개발자는 의도한 결과의 명확성 여부와 무관하게 모든 필요한 상태 전이를 얻기 위해 하나가 아닌 여러 개의 스마트 컨트랙트 어댑터를 구축해야 한다. 이는 개발 자원의 막대한 소모를 초래하고, 간접적으로 벤더 종속(vendor lock-in)을 만들며, 개발자가 원하는 스마트 컨트랙트 기능을 전체 코드를 다시 작성하지 않고서는 빠르고 안전하게 통합할 수 없게 만든다. 이러한 복잡성의 결과로 혁신적인 제품의 개발이 저해되고, 현실 세계 채택 역시 가로막히고 있다.

엔소는 개발자가 어떤 체인의 어떤 스마트 컨트랙트와도 안전하고 단순하며 편리하게 연동하여 원하는 제품을 구축할 수 있어야 한다고 믿는다. 엔소 네트워크는 바로 이 핵심 문제를 해결한다. 개발자는 단일 네트워크와 연동함으로써 여러 체인의 여러 스마트 컨트랙트에 걸친 거래 가능한 데이터를 생성할 수 있으며, 원하는 스마트 컨트랙트와 연동하는 데 필요한 모든 이전 상태를 고려할 수 있다. 이제 개발자는 원하는 결과를 정의하고, 모든 기술적 구현 세부 사항을 엔소 네트워크 참가자에게 위임할 수 있다. 네트워크는 개발자들이 특정 스마트 컨트랙트를 어떻게 실행할지 기여하는 협력적 환경을 조성하며, 이를 통해 모든 스마트 컨트랙트 상호작용과 그 관계를 포함하는 공유 네트워크 상태가 형성된다.

2 엔소: 공유 네트워크 상태

개발자가 블록체인과 편리하게 연동하기 위해서는 각 스마트 컨트랙트의 모든 복잡성을 신경 쓸 필요가 없어야 하며, 원하는 결과를 인텐트(intent)로 명시하고 실행의 모든 복잡성이 추상화되어야 한다. 이를 달성하기 위해 엔소는 실행 가능한 거래 바이트코드(transactable bytecode)를 구축하기 위한 **최초의 공유 네트워크**를 도입한다. 스마트 컨트랙트 추상화는 **액션 제공자(action provider)**에 의해 공유 네트워크에 기여되며, 유효한 기여로 수락되면 맵의 엔티티로 저장된다. 각 기여가 추가될수록 스마트 컨트랙트 간의 관계와 연동 가능성이 더욱 명확해진다. 그 결과 실행 가능한 모든 관계의 방대한 맵이 형성된다.

스마트 컨트랙트 실행을 위한 거래 바이트코드를 구축하기 위해 네트워크는 함수 시그니처, 입력 토큰, 결과와 같은 바이트코드 생성에 필요한 개별 데이터뿐 아니라 바이트코드 빌드 전에 검증해야 하는 사전 요구사항 페처(fetcher)도 함께 저장한다. 예를 들어, 특정 NFT 민팅 추상화에 1,000개의 NFT 발행 한도가 있다면, 이 사전 요구사항 페처는 현재 발행된 NFT 수량과 발행 가능한 최대 수량을 가져온다. 결과적으로 소비자는 실행 가능한 유효한 바이트코드를 반환받는다.

엔소 네트워크에는 네 가지 유형의 참가자가 존재한다:

- **액션 제공자(Action providers):** 스마트 컨트랙트 추상화를 온체인에 게시하는 개발자이며, 그 대가로 소비 수수료의 일부를 얻는다. 네트워크는 자신의 기여가 소비된 정도에 비례해 액션 제공자를 보상한다.

- **그래퍼(Graphers):** 소비자의 요청을 해결하기 위해 알고리즘을 구축하는 숙련된 참여자다. 이들은 네트워크에 기여된 모든 추상화를 탐색하고 이를 최적의 순서로 결합해 실행 가능한 바이트코드를 생성한다. 그래퍼는 소비자 요청을 충족하는 해법은 단 하나만 선택되므로, 가장 최적의 솔루션을 지속적으로 찾도록 유인된다.
- **검증자(Validators):** 소비자 요청의 유효성을 검증하고, 기여된 추상화를 인증하며, 그래퍼가 제공한 솔루션 중 최종적으로 채택될 해법을 결정함으로써 엔소 네트워크를 보호한다.
- **소비자(Consumers):** 원하는 결과를 인텐트로 명시하여 네트워크에 요청을 제출하며, 선택적으로 수수료를 지불한다.

엔소 네트워크에는 세 가지 핵심 구성 요소가 있다:

- **요청(Requesting):** 소비자가 원하는 결과를 인텐트로 명시하여 바이트코드 해법을 얻는다.
- **액션 제출(Action submission):** 네트워크가 제공할 수 있는 해법의 수를 결정한다.
- **솔루션 집계(Solution aggregation):** 그래퍼가 제안한 해법 중 최적의 해법을 확립한다.

3 엔소: 요청(Requesting)

엔소 네트워크는 소비자가 결과를 직접 도출하는 방법을 명시하는 것이 아니라 원하는 결과를 요청한다는 핵심 전제를 기반으로 구축되었다. 이러한 사용성의 패러다임 전환을 **인텐트(intent)**라고 정의한다. 인텐트를 통해 소비자는 상호작용의 “**방식(how)**”을 지정하는 트랜잭션과 달리, “**무엇(what)**”을 얻고 싶은지를 명시할 수 있다. 트랜잭션을 생성하려면 특정 스마트 컨트랙트의 제약과 특수성을 깊이 이해해야 하지만, 인텐트는 이러한 복잡성을 제거하고 소비자가 원하는 결과를 얻기 위한 해법 창출을 외부에 위임할 수 있게 한다. 인텐트는 반드시 달성해야 하는 변수들과 그 변수에 대한 제약 조건을 정의하며, 이는 탁월한 개발자 경험을 제공한다.

엔소 네트워크에 요청이 제출되면 네트워크 참가자들이 조율하여 해법을 결정하고, 해법은 다음과 같은 형식으로 반환된다:

```
Unset
RequestResponse {
  IntentId string
  From string
  To string
  Data string
  Value string
  AmountOut string
  Gas string
}
```

4 엔소: 추상화(Abstraction)

엔소 네트워크가 스마트 컨트랙트 상호작용에 대한 가장 방대한 맵을 보유하도록 하기 위해, 인센티브가 결합된 기여 메커니즘이 개발되었다. 이를 통해 개발자(액션 제공자)는 네트워크에 추상화를 기여할 수 있으며, 해당 추상화가 고객 요청에 대한 해법을 생성할 때 소비되면 보상을 받는다. 이는 개발자가 생태계에서 활용될 것으로 기대하는 추상화를 기여하는 경제를 형성하며, 추상화가 상세할수록 이미 기여된 적이 있을 가능성이 낮아진다. 추상화 선점(squatting)을 방지하기 위해, 네트워크는 동일한 목적을 가진 여러 추상화를 기여할 수 있도록 허용하며, 솔루션 생성 시에는 엔소 토큰이 가장 많이 스테이킹된 추상화가 선택된다.

그래퍼(Graphers)는 네트워크의 추상화를 소비하여 해법을 구축한다. 소비의 확장성을 보장하기 위해 각 추상화는 반드시 연관된 액션 타입(action type)을 가져야 한다. 하나의 추상화는 단일 액션 타입에만 연결될 수 있으며, 하나의 액션 타입은 다수의 추상화를 가질 수 있다. 액션 타입은 연관된 추상화들이 수행하는 특정 동작을 정의하는 래퍼(wrapper)이다. 예를 들어, 하나 또는 여러 개의 토큰을 전송하면서 그 대가로 다른 하나 또는 여러 개의 토큰을 받는 디파이(DeFi) 스마트 컨트랙트는 "입금(deposit)"으로 분류된다. 추상화를 기여할 때는 우선 메시지 액션 타입(message action type)으로 정의된다:

```
Unset
MsgAddDeposit {
  Protocol string
  TokensIn []string
  PositionOut string
  Primary string
  SupportsReceiver bool
}
```

메시지가 추가되면, 다음과 같은 액션 타입(action type)으로 파싱된다:

```
Unset
Action {
  ActionId string
  ActionType string
  Protocol string
  PositionsIn []string
  PositionsOut []string
  SupportsReceiver bool
  Provider string
  Args map[string]string
}
```

스마트 컨트랙트와 상호작용할 때는 다양한 동적 결과가 발생하기 때문에, *positions* 변수는 토큰(token), NFT, 상태 전이(state transition) 등 실행 결과로 반환될 수 있는 여러 유형을 포괄한다. 특정 스마트 컨트랙트는 결과를 다른 수신자 주소로 위임할 수 있도록 허용하므로, 이를 지원하기 위해 SupportsReceiver가 구현되었다. 이러한 방식을 따름으로써, 엔소 위에 구축된 인공지능 실행 봇과 같이 특정 실행 한계를 가진 소비자(consumers) 유형을 열어 두어, 요청을 받고 제공된 권한으로 정산(settle)할 수 있도록 한다.

새로운 *PositionTypes*는 거버넌스 모듈을 통해 네트워크에 추가될 수 있다. 포지션은 다음 요소들로부터 결정적 해시(deterministic hash)로 생성된다:

```
Unset
Position {
    PositionType string
    Chain uint64
    Address string
    TokenId string
}
```

네트워크에서 생성된 해법을 검증하기 위해, 각 연관된 추상화에는 *ViewType*이 첨부된다. 여기에는 온체인 getter를 포함시켜 원하는 결과와 실제 결과를 교차 검증할 수 있다:

```
Unset
ViewFunction {
    PositionType string
    Fragment string
    Target string
    Args []string
    ResponseIndex uint64
}
```

추상화가 연관된 액션 타입과 함께 기여되면, 새로운 추상화 제안은 네트워크 상에서의 유효성을 보장하기 위해 다음 단계를 거친다:

- **1단계:** 액션 제공자가 액션을 제출한다.
- **2단계:** 검증자는 새로운 액션 제출을 감지하고, 액션 제공자가 온체인 제공자 목록에 등록되어 있는지 검증한다.
- **3단계:** 액션 메시지 데이터가 액션 타입 형식으로 파싱되며, 다른 값들은 포지션(Position) 타입 형식으로 변환된다.
- **4단계:** 액션 제공자의 인덱싱 인프라와 인터페이스하여 액션 정의를 수신한다.
- **5단계:** 액션 정의로부터 트랜잭션 바이트코드를 생성한다.
- **6단계:** 트랜잭션 바이트코드를 시뮬레이션하고 실행하여, 특정 포지션에서 결과가 올바르게 동작하는지 확인한다.
- **7단계:** 포지션이 온체인에 저장되었는지 검증하고, 데이터가 누락된 경우 포지션 데이터를 수정한다.
- **8단계:** 액션을 온체인에 저장한다.
- **9단계:** ActionId를 이벤트로 발생시킨다.

이 단계가 완료되면 추상화는 엔소 네트워크 맵의 일부가 되어 해법 생성에 활용될 수 있다.

5 솔루션 집계(Solution aggregation)

엔소 네트워크에 요청이 제출되면, 네트워크 참가자들은 소비자에게 최적의 결과를 제공하기 위해 해법 제안에서 협력하고 경쟁한다. 경쟁은 그래퍼들 간의 솔루션 제공 과정에서 발생하며, 이는 엔소 네트워크가 시간이 지나도 정체되지 않고 지속적으로 최적의 해법을 제공하게 만든다. 결과적으로 소비자들은 실행 가능한 데이터의 단일 소스로서 엔소를 계속 사용하게 되며, 다른 출처로 이동할 필요가 없다.

검증자는 네트워크에 제출된 요청을 청취하며 올바른 형식이 사용되었는지 확인한다. 제출이 올바르면 요청은 네트워크의 메모리풀(mempool)에 전파된다. 이후 그래퍼들은 고유 알고리즘을 실행하여 네트워크 내 모든 액션을 탐색하고 가장 최적의 경로를 찾는다. 이때 사용되는 알고리즘은 매우 복잡할 수도 있고, 단순히 K 최단 경로(K shortest path)와 같이 간단할 수도 있다.

솔루션이 바이트코드로 형성될 수 있다면, 이는 그래퍼의 유효한 제안으로 간주된다. 솔루션이 생성되면, 해당 바이트코드는 해결하려는 *IntentID*와 연결된다. 검증자의 역할은 이제 각 솔루션을 시뮬레이션하는 것이다. 이를 위해 원하는 정산 체인(settlement chain)을 최근 상태로 포킹(forking)하고, 바이트코드를 실행하여 실행 가능 여부를 확인하며, 각 액션의 *ViewFunction* 헬퍼를 호출하여 올바른 상태 전이가 일어났는지 검증한다. 검증자는 또한 잔고 확인, 오버라이드, delegate call 등 맞춤 모듈을 실행할 수도 있다.

각 검증자가 상태 전이를 개별적으로 감지하면, 각 검증자는 적절한 *IntentID*와 *SolutionID*와 함께 자신의 결과를 메모리풀에 다시 전파한다. 이후 가장 높은 달러 환산 산출액과 가장 낮은 실행 비용을 가진 솔루션이 최종 승자로 선택된다. 나머지 솔루션 제안은 메모리풀에서 제거된다. 그 결과, 미래에 관련 없는 솔루션은 네트워크 내 불필요한 데이터 누적을 일으키지 않으며, 이는 일종의 내장 캐싱 계층으로 볼 수 있다. 최종 선택된 해법은 다시 소비자에게 반환되어 실행된다.

엔소 네트워크는 다양한 블록체인에서 운영되며, 솔루션에는 실행을 위해 바이트코드에 수수료가 내장될 수 있다. 모든 수수료를 엔소 블록체인으로 다시 브리징하는 시스템을 구축하는 대신, 엔소 네트워크에서 경매가 진행된다. 이 경매에서 특정 주소가 해당 블록체인에서 발생한 수수료를 청구할 권리를 얻기 위해 엔소 토큰으로 입찰할 수 있다. 이러한 방식을 통해 수수료 확보에 사용된 엔소 토큰은 추상화 소비, 솔루션 채택, 검증 등 기여 활동에 따라 네트워크 참가자들에게 분배된다. 따라서 엔소는 지속적으로 배포되는 블록체인의 확산에도 대응 가능한 확장성과 적응성을 가진 내장 순환 경제를 제공한다.

6 애플리케이션(Applications)

엔소의 아키텍처는 어떤 블록체인에서든 스마트 컨트랙트와 상호작용하고, 최적의 실행을 달성하려는 모든 애플리케이션을 가능하게 한다. 엔소는 네트워크를 통해 상호작용할 수 있는 스마트 컨트랙트의 유형에 구애받지 않기 때문에, 엔소를 기반으로 구축될 수 있는 애플리케이션의 다양성은 무한하다. 모든 블록체인의 모든 스마트 컨트랙트와 상호작용할 수 있는 보편적 게이트웨이를 제공함으로써, 개발자들은 진정으로 중단 없고 조합 가능한 애플리케이션을 만들 수 있다.

엔소를 사용하면 무엇이든, 어떻게든, 어디에서든 구축할 수 있다.

7 토큰(Token)

엔소는 텐더민트(Tendermint) 기반 네트워크 내에서 사용되는 네이티브 토큰이다. 엔소 토큰은 엔소 생태계의 핵심 구성 요소이며, 토큰 없이는 네트워크가 작동할 수 없다. 토큰의 주요 기능은 다음과 같다:

- **가스(Gas):** 요청 제출과 네트워크 상태 변경을 가능하게 한다.
- **거버넌스(Governance):** 프로토콜 업그레이드가 토큰 투표를 통해 진행된다.
- **네트워크 참여(Network participation):** 검증자(Validators), 그래퍼(Graphers), 액션 제공자(Action Providers)는 모두 엔소 토큰을 스테이킹해야 하며, 잘못된 행위 시 슬래싱(slash)이 적용된다.
- **위임(Delegation):** 스테이킹을 네트워크 참가자에게 위임함으로써 네트워크를 더욱 안전하게 보호할 수 있다.

엔소 토큰의 총 발행량은 100,000,000개이다.

8 로드맵(Roadmap)

엔소 네트워크는 여러 단계에 걸쳐 출시될 예정이다. 첫 번째 단계는 초기 네트워크 출시로, 독립적인 검증자들이 바이트코드 솔루션을 시뮬레이션하며, 이 시기에는 네트워크와 병행하여 중앙화된 엔소 서비스가 공존한다. 이 단계에서 액션 제공자가 되고자 하는 개발자는 엔소가 호스팅하는 중앙화된 서비스에 추상화를 기여할 수 있으며, 이를 통해 네트워크가 올바르게 작동하고 기여 절차가 원활히 이루어지도록 한다. 두 번째 단계에서는 액션 제공자들이 완전히 퍼미션리스(permissionless) 환경에서 기여할 수 있으며, 다양한 그래퍼들이 네트워크에 참여할 수 있게 된다. 이 시점에서 엔소 네트워크는 완전히 지속 가능한 생태계가 되며, 이더리움 가상머신(EVM)뿐 아니라 솔라나 가상머신(SVM), 무브 가상머신(MVM) 프레임워크로 확장되어 이러한 생태계에서의 개발자 경험을 더욱 향상시킬 것이다.

9 결론(Conclusion)

본 논문에서는 개발자가 단일 소스를 통해 어떤 블록체인의 스마트 컨트랙트와도 상호작용할 수 있도록 하는 **최초의 공유 네트워크 상태(shared network state)**를 소개했다. 이러한 패러다임 전환적인 개념은 업계 최초이며, 생태계 내 지속적인 단편화로 인해 발생한 블록체인의 사용성 문제를 해결한다. 각 네트워크 참가자의 역할과 기여가 이루어지고 검증되는 과정이 명확히 설명되었다. 엔소는 모든 생태계 사이의 인터페이스 계층으로 작동하여, 개발자가 블록체인 기술과 상호작용할 수 있는 활력 있는 환경을 조성하고, 블록체인 기술 전체의 활용도와 사용성을 한층 높인다.

엔소를 사용하면 무엇이든, 어떻게든, 어디에서든 구축할 수 있다.