

## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

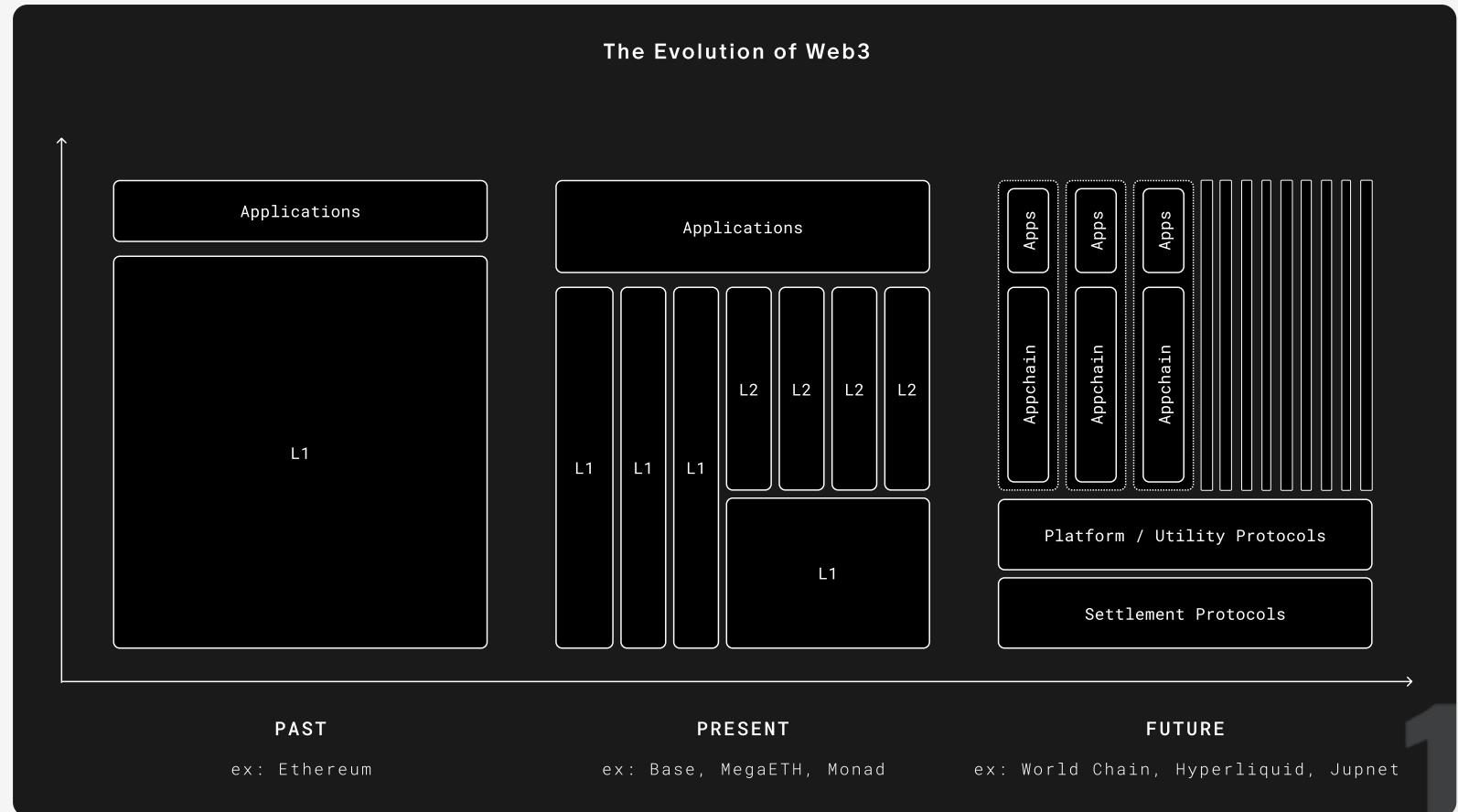
## CORE CONCEPTS

시퀀싱 모드

## 왜 앱체인인가

프로그래밍 가능한 온체인 시퀀서를 갖춘 앱체인이 해결하는 문제와 웹3를 주류로 이끌 수 있는 혁신적 잠재력을 이해해보세요.

## 역사적 맥락: 애플리케이션 중심 체인으로의 진화



지난 10년 동안 블록체인은 주입역 사용자로 확장하기 위해 세 가지 뚜렷한 단계를 거쳐 진화해왔습니다:

1. **범용 체인:** 모든 애플리케이션이 동일한 블록스페이스와 기본 경제 시스템 위에 구축됨 — 혼잡, 높은 수수료, 애플리케이션 레이어에서 제한된 가치 포착으로 이어짐.
2. **범용 체인 확장:** 성능 향상과 수수료 인하 — 하지만 여전히 특정 애플리케이션과 생태계를 위한 네트워크 또는 시퀀서 수준의 커스터마이징을 허용하지 않음.
3. **애플리케이션 중심 체인:** 이제 우리는 개발자가 자신의 애플리케이션과 생태계에 맞게 네트워크와 경제를 맞춤화할 수 있는 애플리케이션 중심 블록체인(앱체인) 시대에 진입했습니다.

유니스왑, 월드, 주피터, 에테나, 젠신, 하이퍼리퀴드 같은 애플리케이션들과 매일 더 많은 프로젝트들이 네트워크, 수수료, 경제에 대한 더 큰 주도권, 통제력, 소유권을 위해 자체 체인을 출시하고 있습니다. 가치가 애플리케이션 레이어로 이동함에 따라 범용 L1/L2의 언번들링이 가속화되고 있습니다.

궁극적으로, 앱체인은 웹3의 최종 목표이며 블록체인이 기술적으로나 경제적으로 애플리케이션 및 토큰화된 커뮤니티와 최대한 정렬되어 10억 사용자로 확장하는 방법입니다. 신디케이트는 더 나아가 앱체인이 원자적으로 구성 가능하고, 프로그래밍 가능하며, 커뮤니티 소유가 가능하도록 함으로써 앱체인의 궁극적인 최종 목표를 실현합니다.

## 앱체인 예시

앱체인은 아직 비교적 새로운 카테고리이지만, 다양한 영역에서 실제 사용 사례가 등장하고 있습니다. 앞으로 더 많이 볼 수 있을 것으로 예상되는 몇 가지 예시는 다음과 같습니다:

- 커머스 앱체인 온체인 충성도 및 멤버십 상태에 기반하여 판매가 입증 가능하게 정렬됨
- 제조 앱체인 온체인 평판에 기반하여 공급망과 수수료가 공유됨
- 금융 앱체인 온체인 거래량에 기반하여 기관과 수수료가 프로그래밍 방식으로 공유됨
- 게임 앱체인 온체인 이벤트에 기반하여 보상이 실시간으로 프로그래밍 방식으로 분배됨
- 소셜 앱체인 네트워크 수준의 사용량과 필요에 따라 인센티브가 자율적으로 할당됨

이는 사용량이 증가할 것으로 예상되는 애플체인의 몇 가지 예시일 뿐입니다. 하지만 애플체인의 가능성과 성장 기회는 무한합니다.

## 큰 트레이드오프: 맞춤화 또는 구성 가능성?

오늘날 애플리케이션은 핵심적인 트레이드오프에 직면해 있습니다: 더 넓은 생태계와 구성 가능하도록 범용 L1/L2에 구축하되 네트워크 설계, 경제 및 인센티브에 대한 통제력을 희생해야 할까요? 아니면 애플리케이션과 커뮤니티에 더 깊이 부합하는 더 많은 경제적 기회를 가진 맞춤형 애플체인을 출시하되 다른 앱 및 프로토콜과의 기본 구성 가능성을 잃어야 할까요? 이로 인해 개발자들은 광범위한 구성 가능성과 네트워크 맞춤화, 통제 및 소유권을 통한 차별화 및 경제적 성장 사이에서 선택해야 합니다. 매일 더 많은 애플리케이션이 애플체인을 출시하기로 선택하고 있습니다.

그러나 신디케이트는 애플체인에 완전한 맞춤화/통제와 원자적 구성 가능성을 제공함으로써 이 근본적인 트레이드오프를 깨뜨립니다.

## 문제: 중앙화된 시퀀서가 애플체인이 할 수 있는 일을 제한함

오늘날 대부분의 애플체인은 많은 애플리케이션별 네트워크를 위해 설계되지 않은 오프체인, 중앙화된 시퀀서나 제공자에 의존합니다. 이러한 중앙화는 통제, 확장성, 경제성 및 주권 측면에서 주요 제한을 도입하여 애플체인이 그들의 잠재력을 완전히 실현하는 것을 방해합니다.

- **제한된 제어 및 커스터마이징:** 오프체인 시퀀서는 업그레이드하기 어려운 고정된 로직으로 작동합니다. 개발자는 맞춤형 트랜잭션 포함 또는 정렬 규칙을 정의하거나, 사용 사례에 맞게 블록 생성을 조정하거나, 시간이 지남에 따라 시퀀서를 쉽게 발전시킬 수 없습니다. 모든 변경은 간단한 온체인 업데이트 대신 고위험 인프라 수준의 조정이 필요합니다. [신디케이트가 이를 해결하는 방법 알아보기](#)
- **단일 장애 지점:** 대부분의 롤업은 하나의 엔티티나 제공자가 운영하는 단일 시퀀서에 의존합니다. 해당 시퀀서가 실패하거나, 오작동하거나, 정렬이 어긋나면 애플체인은 복구하거나 마이그레이션할 쉬운 방법 없이 멈춰버립니다. 이는 운영상 심각한 취약성과 벤더 종속을 초래합니다. [신디케이트가 이를 해결하는 방법 알아보기](#)
- **경제적 가치 손실:** 중앙화된 시퀀서를 사용하면 수수료 시장, MEV 및 토큰이코노믹스를 애플체인의 생태계에 맞게 커스터마이징할 수 없으며, 토큰 보유자에게 신뢰할 수 있고, 증명 가능하며, 프로그래밍 방식으로 가치가 흐를 수 없습니다. 더 나쁜 것은 가치가 중

종 시스템의 다른 행위자에게 유출된다는 점입니다. 신디케이트가 이를 해결하는 방법 알아보기

- **확장 비용 증가:** 오프체인 시퀀서 인프라는 운영 및 유지하는 데 비용이 많이 들어, 네트워크가 성장함에 따라 오버헤드와 필요한 중복성이 증가합니다. 비용과 가격이 함께 증가하는 중앙화된 제공자에 대한 의존도가 높아지면 큰 문제가 됩니다. 신디케이트가 이를 해결하는 방법 알아보기
- **컴포저빌리티 손실:** 크로스체인 상호작용은 느리고, 신뢰할 수 없으며, 원자적이지 않습니다. 원자적 포함 보장 없이는 애플리케이션이 체인 간에 쉽게 또는 안전하게 구성될 수 없어 UX/DX가 분열되고, 위험이 증가하며, 네트워크 효과가 약화됩니다. 신디케이트가 이를 해결하는 방법 알아보기
- **데이터 저장 및 가용성:** 대부분의 롤업에서 트랜잭션 데이터는 약 18일 후에 만료되는 블록에 저장됩니다. 역사적인 계약 및 사용자 상태 데이터가 적극적으로 보관되지 않으면 사라집니다. 중앙화된 제공자에 의존하는 앱체인은 특히 데이터 손실 위험이 높습니다. 신디케이트가 이를 해결하는 방법 알아보기
- **진정한 커뮤니티 제어 및 소유권 부재:** 시퀀서는 일반적으로 단일 엔티티에 의해 제어됩니다. 제어권과 거버넌스를 커뮤니티에 점진적으로 분산시킬 수 있는 기본 경로가 없어, 네트워크의 가장 중요한 부분이 중앙화된 상태로 남게 됩니다. 신디케이트가 이를 해결하는 방법 알아보기

이러한 모든 이유와 그 이상으로, 중앙화된 오프체인 시퀀서는 피해야 하며 혁신이 필요합니다. 이것이 바로 신디케이트의 프로그래밍 가능한 온체인 시퀀서가 하는 일입니다.

## 미래: 프로그래밍 가능한 온체인 시퀀서를 갖춘 앱체인

신디케이트의 프로그래밍 가능한 온체인 시퀀서는 앱체인을 위한 새로운 설계 공간을 열어줍니다 — 유연성, 회복력, 커뮤니티 소유권으로 정의되는 공간입니다. 중앙화의 제약에 묶이는 대신, 개발자와 그들의 커뮤니티는 간단한 스마트 컨트랙트 모듈 업데이트를 통해 네트워크를 완전히 소유하고 발전시킬 수 있습니다.

### 완전한 맞춤화와 제어

트랜잭션 포함 및 정렬 규칙, 권한 부여/거버넌스, 경제 모델을 정의하고 업데이트하세요 — 모두 간단한 스마트 컨트랙트 모듈을 통해 가능합니다. 비용이 많이 드는 새로운 인프라나 중앙화된 조정 없이 프로그래밍 방식으로 네트워크 설계와 경제를 발전시킬 수 있





습니다.

## 단일 장애점 없음

시퀀싱 로직은 완전히 온체인에 존재합니다. 시퀀싱이 스마트 컨트랙트에 인코딩되어 있기 때문에, 개별 노드나 참여자가 실패하더라도 시스템은 계속 작동합니다. 이는 중앙화된 병목 현상을 제거하고 애플체인을 설계상 더 회복력 있고 검열 저항적으로 만듭니다.

## 맞춤형 경제 모델 및 내장된 프로토콜

자체 가스 토큰을 만들고, 스테이킹 기반 정렬 규칙을 만들고, 동적 수수료를 구현하고, 오라클과 마켓플레이스와 같은 프로토콜을 시퀀서에 직접 내장하세요. 수익 공유, 스테이킹 보상 또는 인센티브 프로그램을 통해 가치를 커뮤니티에 직접 전달하거나 — 또는 가치 있는 사용자와 활동에 보조금을 지급하세요.

## 운영 및 확장 비용 절감 (10-100배)

시퀀싱과 데이터를 온체인으로 이동함으로써 비용이 많이 드는 오프체인 인프라의 필요성을 줄입니다. 애플체인을 중단하면서도 그 역사를 무비용으로 유지하세요. 전통적인 롤업보다 훨씬 적은 비용(10배-100배)으로 애플체인을 운영하면서 — 더 나은 성능과 유연성을 얻으세요.

## 원자적 크로스체인 구성성

원자적 포함 모듈을 통해 여러 애플체인이나 샤드의 트랜잭션을 동일한 블록에 포함시킬 수 있습니다. 브릿지, 배치 윈도우 또는 릴레이어에 의존하지 않고 애플체인 간의 즉각적이고 신뢰 없는 조정을 가능하게 합니다. 부분 실행의 위험을 크게 줄이고 새로운 크로스체인 애플리케이션을 활성화합니다.

## 온체인 데이터 저장

모든 트랜잭션 및 정렬 데이터는 중앙 집중식 서버나 만료되는 임시 블록이 아닌 직접 온체인에 영구적으로 저장됩니다. 역사적 상태, 스마트 계약 데이터 및 사용자 활동이 항상 이용 가능하고, 검증 가능하며, 손실에 강하도록 보장합니다.

## 토근화된 커뮤니티로의 점진적 탈중앙화

온체인 시퀀서의 권한 부여 모듈에 대한 간단한 업데이트를 통해 거버넌스, 경제 및 운영의 원활하고 점진적인 탈중앙화를 커뮤니티에 제공합니다. 너무 늦기 전에 처음부터 커뮤니티 소유권으로 가는 경로를 만드세요.

## 시작하기

앱체인이 어떻게 당신의 애플리케이션과 생태계에 도움이 될 수 있는지 탐색할 준비가 되셨나요?

1. 신디케이트 스택 탐색하기로 기술 아키텍처 이해하기
2. 사례 연구 검토하기로 다른 사람들이 앱체인을 어떻게 사용하고 있는지 확인하기

이러한 기회나 도전을 탐색하고 계시다면, 저희 팀에 연락하세요. 저희는 이 새로운 기술로 가능한 것의 경계를 넓히기 위해 선도적인 프로젝트들과 협력합니다.

< 소개

신디케이트가 어떻게 프로그래밍 가능한 온체인 시퀀싱 레이어를...

신디케이트 스택 >

신디케이트는 원자적으로 구성 가능하고 프로그래밍 가능한 앱체...

## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

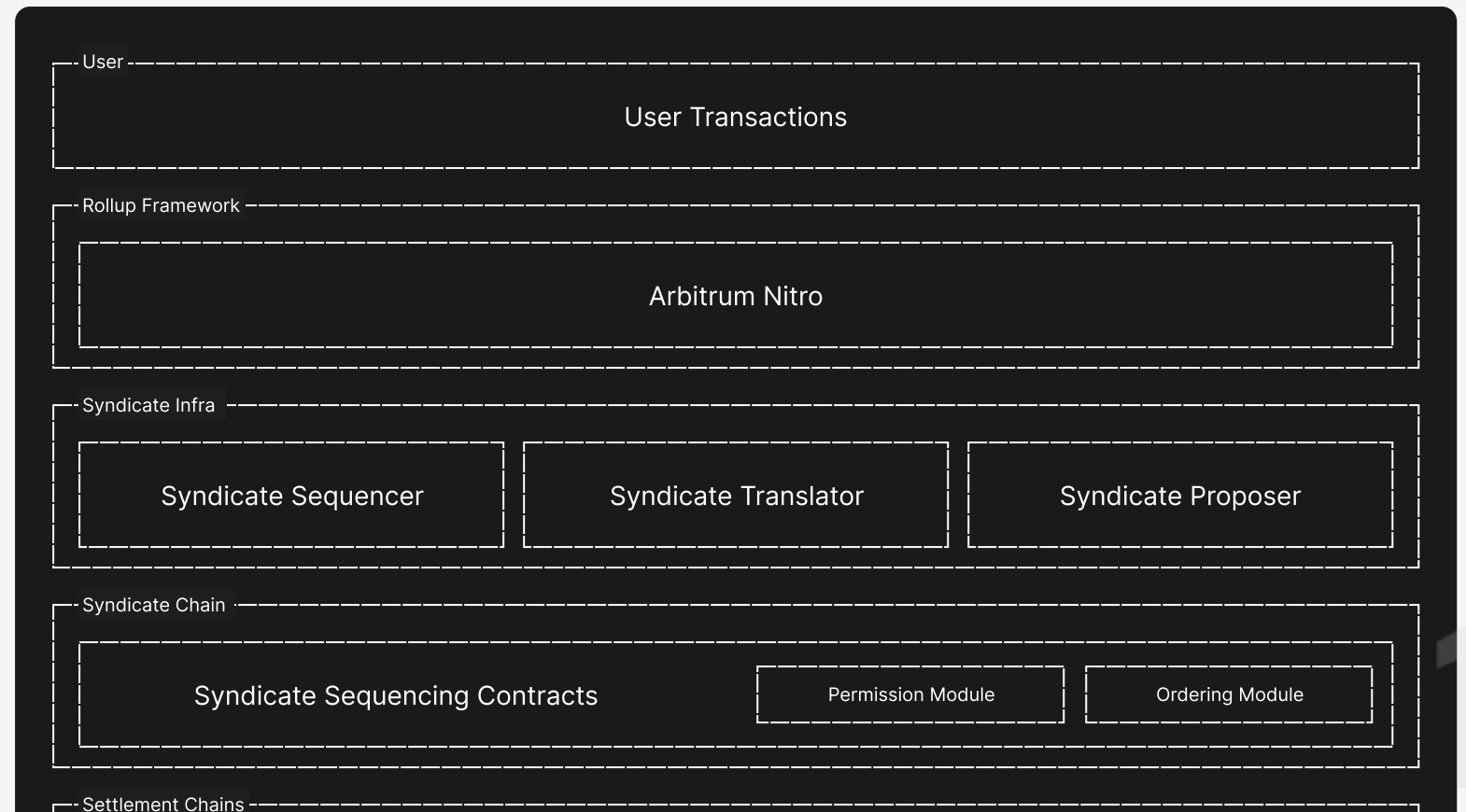
시퀀싱 모드

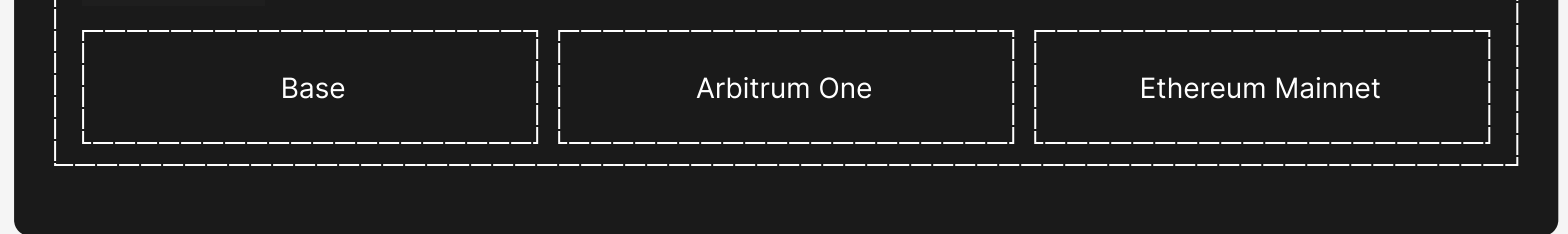


# 신디케이트 스택

신디케이트는 원자적으로 구성 가능하고 프로그래밍 가능한 앱체인을 통해 web3를 수평적으로 확장하는 샤딩된 네트워크 오브 네트워크입니다.

신디케이트 스택은 EVM 생태계 및 그 이상의 기존 잘 알려진 롤업 솔루션과 통합하기 위해 신디케이트가 만든 소프트웨어로 구성됩니다. 우리의 스택은 맞춤형 구성 요소와 업계 표준 기술을 결합하여 전체 스택 제어, 경제적 주권 및 커뮤니티 소유권을 제공하면서 분산 애플리케이션 구축 및 운영의 주요 과제를 해결합니다.





신디케이트 애플체인은 이러한 모든 구성 요소를 결합한 전체 스택 배포입니다.

## 트랜잭션 흐름: 쓰기 경로

단계별 과정:

1. 사용자: 트랜잭션 시작 (지갑/dApp)
2. 신디케이트 멤풀 (선택사항): 트랜잭션 수집 및 배치
3. 신디케이트 시퀀서: 배치된 트랜잭션을 신디케이트 체인에 제출
4. 신디케이트 체인: 온체인 시퀀싱, 순서 지정, 권한 부여
5. 신디케이트 트랜슬레이터: 실행을 위한 블록 준비
6. 롤업 프레임워크: 트랜잭션 실행 (예: Arbitrum Nitro)
7. 신디케이트 프로포저: 정산 체인에 데이터 제출
8. 정산 체인: 최종성 및 보안 (Base, Arbitrum One, 이더리움 메인넷 등)

## 핵심 구성 요소

### 0. 신디케이트 멤풀

(선택사항) 신디케이트에서 유지 관리

멤풀은 시퀀서에 제출하기 전에 트랜잭션을 수집합니다. 자동 배치 처리를 통해 트랜잭션 제출을 더 효율적으로 만듭니다. 이는 선택

적 구성 요소로, 체인이 분산화와 트랜잭션 개인 정보 보호를 유지할 수 있게 합니다. 원하는 경우 언제든지 멤폴을 건너뛰고 자신의 트랜잭션을 직접 시퀀싱할 수 있습니다.

## 1. 신디케이트 시퀀서

*신디케이트에서 유지 관리*

시퀀서는 시퀀싱 계약에 트랜잭션 제출을 처리합니다. 사용자로부터 트랜잭션을 수집하고 시퀀싱 계약에 정의된 규칙에 따라 신디케이트 체인에 제출합니다. 이 구성 요소는 트랜잭션이 실행 환경으로 전달되기 전에 적절하게 정렬되고 형식화되도록 보장합니다.

## 2. 신디케이트 체인

*신디케이트에서 유지 관리*

신디케이트 체인은 앱체인 전체 생태계를 위한 온체인 시퀀싱을 가능하게 하는 전용 블록체인입니다. 이 체인은 트랜잭션 순서 지정에 대한 규칙과 권한을 정의하는 시퀀싱 스마트 계약을 호스팅하여, 각 앱체인이 경제적 주권을 유지하면서 트랜잭션 흐름을 관리하는 기본 규칙을 설정합니다.

## 3. 신디케이트 트랜슬레이터

*신디케이트에서 유지 관리*

트랜슬레이터는 신디케이트 체인과 정산 체인을 통합합니다. 신디케이트 체인에서 블록을 수집하고, 정의된 규칙에 따라 정산 체인 데이터와 결합한 후, 실행을 위해 표준 형식으로 준비합니다. 트랜슬레이터는 고수준의 시퀀싱 결정을 실행 환경에서 처리할 수 있는 형식으로 효과적으로 변환합니다.

## 4. 롤업 프레임워크

*오프체인 랩스에서 개발, 확장 호환성 제공*

현재 아비트럼 Nitro로 구축되는 롤업 프레임워크는 트랜잭션이 실제로 처리되는 곳입니다. 트랜슬레이터에서 시퀀싱된 블록을 가

현재 이머트럼 디프로로 검증되는 클립 프레임워크는 트랜잭션이 실제로 처리되는 것입니다. 트랜슬레이터에서 지원되는 클립을 가져와 트랜잭션을 실행하고, 체인의 상태를 유지하며 모든 상태 전환을 검증합니다. 완전한 EVM 호환성을 갖춘 이 프레임워크는 모든 프로토콜 규칙을 따르면서 효율적으로 트랜잭션을 처리합니다.

트랜슬레이터의 유연한 아키텍처를 통해, 우리는 옵티미즘의 OP 스택과 추가 실행 환경에 대한 지원을 확장하면서도 보안과 성능에 대한 동일한 높은 표준을 유지할 것입니다.

## 5. 신디케이트 프로포저

신디케이트에서 유지 관리 프로포저는 브리징과 출금에 필요한 데이터를 정산 체인에 제출합니다. 실행 환경의 상태 변경이 정산 체인에 올바르게 커밋되도록 보장하여 시스템 전반의 보안과 최종성을 유지합니다. 프로포저는 현재 TEE 기반이지만, 시간이 지남에 따라 다른 검증 방법이 추가될 예정입니다.

## 6. 정산 체인

*기존 L2 솔루션과의 통합*

정산 체인은 일반적으로 Base, Arbitrum One 또는 이더리움 메인넷과 같은 확립된 L2로, 앱체인에 보안과 최종성 백본을 제공합니다. 이는 트랜잭션 최종성과 데이터 가용성을 보장하는 궁극적인 진실의 원천 역할을 합니다. 검증된 L1/L2 인프라를 기반으로 구축함으로써, 앱체인은 자체 실행 환경과 경제적 주권을 유지하면서 강력한 보안 보장을 상속받습니다.

## 엔드투엔드 앱체인 관리

신디케이트 스택은 기술적으로 정교하지만, 우리는 이를 간단하게 운영할 수 있도록 만들었습니다. 우리의 관리형 인프라는 배포부터 지속적인 운영까지 앱체인 실행의 모든 복잡성을 처리합니다:

- **인프라:** 신디케이트 앱체인 스택은 시퀀서, 트랜슬레이터, 프로포저를 포함한 모든 구성 요소를 자동으로 관리하는 프로덕션 준비 인프라 레이어를 제공합니다. 깊은 블록체인 운영 전문 지식 없이도 업데이트, 모니터링 및 확장을 처리합니다.
- **관리 콘솔:** 관리 콘솔을 통해 우리가 기술적 운영을 처리하는 동안 앱체인의 경제와 거버넌스에 대한 완전한 제어권을 유지할 수 있

습니다. 간단한 인터페이스를 통해 성능을 모니터링하고, 시퀀싱 규칙을 조정하며, 네트워크 매개변수를 관리할 수 있습니다.

이는 앱체인이 안정적이고 효율적으로 실행되도록 보장하는 동안 애플리케이션 구축과 생태계 성장에 집중할 수 있다는 것을 의미합니다. 게임 네트워크, 소비자 애플리케이션 또는 AI 플랫폼을 출시하든, 운영 오버헤드 없이 맞춤형 블록체인의 이점을 얻을 수 있습니다.

## 시작하기

신디케이트 스택은 다양한 수준의 커스터마이징과 제어를 위해 구성될 수 있습니다:

1. 표준 구성 - 기본 설정으로 빠르게 시작합니다. 이는 롤업 프레임워크로서의 Arbitrum Nitro + 정산을 위한 Base + 가스 토큰으로서의 ETH + 허용 목록 시퀀싱 모듈입니다.
2. 맞춤 구성 - 특정 요구에 맞게 구성 요소를 조정합니다. 이는 롤업 프레임워크로서의 Arbitrum Nitro + 정산을 위한 선택한 레이어 + 커스텀 가스 토큰 + 허용 목록 시퀀싱 모듈입니다.
3. 고급 커스터마이징 - 특수 시퀀싱 로직과 경제적 메커니즘을 구현합니다. 이는 롤업 프레임워크로서의 Arbitrum Nitro + 정산을 위한 선택한 레이어 + 커스텀 가스 토큰 + 커스텀 시퀀싱 모듈입니다.

Syndicate Stack으로 구축을 시작하려면:

1. 다양한 시퀀싱 접근 방식 이해하기와 이것이 주요 앱체인 이점에 미치는 영향
2. 트랜잭션 순서 지정을 위한 시퀀싱 모듈 탐색하기

< 왜 앱체인인가

프로그래밍 가능한 온체인 시퀀서를 갖춘 앱체인이 해결하는 문...

시퀀싱 아키텍처 >

시퀀싱 아키텍처의 이해와 신디케이트의 접근 방식 비교

Q Search



K

☐ Syndicate Stack



## GET STARTED

소개

왜 애플체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모드



# 시퀀싱 아키텍처

시퀀싱 아키텍처의 이해와 신디케이트의 접근 방식 비교

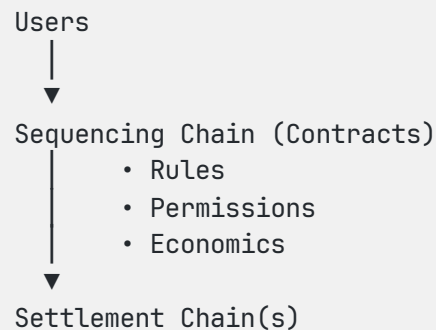
## 소개

시퀀싱은 블록체인 네트워크가 트랜잭션을 정렬하고 처리하는 핵심입니다. 이 문서는 비용 효율성, 성능, 프로그래밍 가능한 정렬, 맞춤형 수수료 시장, 원자적 호환성, 탈중앙화와 같은 주요 애플체인 이점에 미치는 영향을 기준으로 다양한 시퀀싱 접근 방식을 비교합니다.

## 시퀀싱 접근 방식 비교

### 온체인 시퀀싱 (신디케이트)

신디케이트의 온체인 시퀀싱은 트랜잭션 정렬 로직을 중앙화된 서비스에서 EVM 호환 롤업에 배포된 투명한 스마트 계약으로 이동 시킵니다. 이 접근 방식은 애플리케이션에 트랜잭션 정렬, 경제성, 거버넌스에 대한 완전한 제어권을 제공하면서 프로그래밍 가능한 트랜잭션 정렬, 맞춤형 수수료 시장, 체인 간 원자적 구성 가능성과 같은 고유한 기능을 가능하게 합니다.



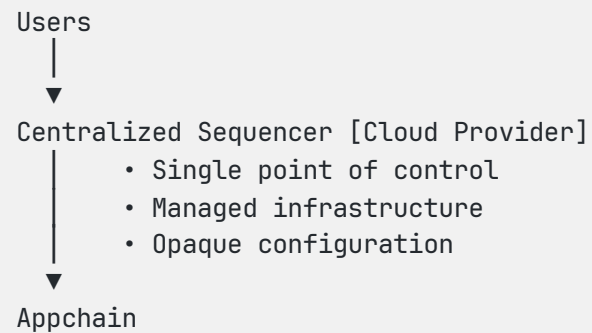


- Execution
- State changes
- Security/finality

주요 장점	주요 단점
<ul style="list-style-type: none"> <li>• 프로그래밍 가능한 트랜잭션 정렬</li> <li>• 경제적 주권과 수수료 시장에 대한 통제</li> <li>• 체인 간 원자적 구성 가능성</li> <li>• 점진적 탈중앙화 경로</li> <li>• 온체인에서 투명하고 검증 가능한 규칙</li> <li>• 거버넌스에 대한 완전한 통제</li> </ul>	<ul style="list-style-type: none"> <li>• 온체인 개발 전문 지식 필요</li> <li>• 시퀀싱 계약 작성 및 유지 관리 필요</li> <li>• 초기 설정 복잡성 증가</li> </ul>

## 중앙화된 시퀀서

중앙화된 시퀀서는 클라우드 제공업체를 통해 중앙화된 롤업 호스팅을 제공하여 앱체인에 가장 간단한 배포 모델을 제공하지만 상당한 중앙화가 수반됩니다.



주요 장점	주요 단점
-------	-------

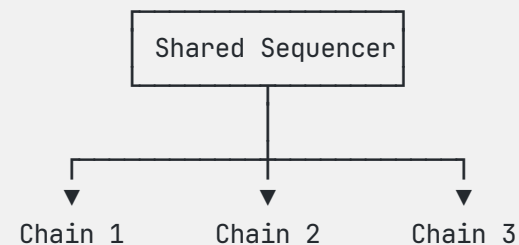
- 쉬운 설정 및 배포
- 고도로 구성 가능한 인프라
- 단순한 정신 모델(중앙화된 클라우드)
- 다양한 롤업 프레임워크 지원
- 관리형 인프라

- 완전히 중앙화된 운영
- 활성 실패 위험
- 데이터 손실 위험
- 불투명한 구성 변경
- 제공업체가 관리하는 키(상당한 신뢰 필요)

**Syndicate**의 온체인 시퀀싱 관련성: 온체인 시퀀싱은 중앙화된 앱체인 호스팅과 잘 어울리며 중앙화된 시퀀서 구성 요소를 대체합니다. 중요 정보를 중앙화된 클라우드 서비스 대신 온체인에 저장함으로써 활성 실패 및 데이터 손실 위험을 줄입니다. 중앙화된 호스팅은 여전히 RPC, 개발자 도구, 브릿지 UI 등을 통해 가치를 제공할 수 있습니다.

## 공유 시퀀싱

공유 시퀀싱은 여러 앱체인의 개별 시퀀서를 결합하여 트랜잭션을 공동으로 정렬함으로써 크로스체인 구성 가능성을 활성화하고 공유 블록 경매와 같은 메커니즘을 통해 잠재적으로 수익을 창출합니다.



- Common sequencer for multiple chains
- Standardized transaction ordering
- Coordinated block production

주요 장점

주요 단점

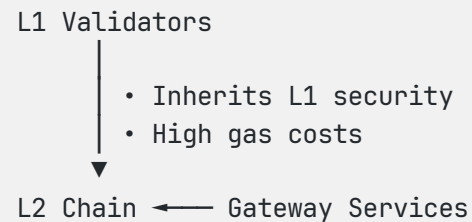


- |  |  |
|--|--|
| <ul style="list-style-type: none"> <li>• 네트워크 전반에 걸친 표준화된 접근 방식</li> <li>• 원자적 구성 가능성 활성화</li> <li>• 크로스체인 수익 기회 창출</li> <li>• 긍정적인 네트워크 효과</li> <li>• 공유 블록 경매 잠재력</li> </ul> | <ul style="list-style-type: none"> <li>• 하향식 시장 채택 필요</li> <li>• 경매 메커니즘 수정 어려움</li> <li>• 롤업이 시퀀서 제어권 포기를 꺼림</li> <li>• 승자 독식 역학</li> </ul> |
|--|--|

**Syndicate**의 온체인 시퀀싱 관련성: 온체인 시퀀싱은 설계상 원자적 구성 가능성을 활성화하여 네트워크가 규칙 세트를 선택적으로 동기화함으로써 "상향식" 공유 시퀀싱을 구축할 수 있게 합니다. 이 접근 방식은 모든 사람이 동일한 시스템에 참여해야 하는 하향식 명령이 아닌, 체인이 자체 조건에 따라 상호 운용성을 점진적으로 채택할 수 있게 합니다.

## 베이스드 시퀀싱

베이스드 시퀀싱은 L1 검증자를 사용하여 트랜잭션을 정렬하고 게시함으로써 트랜잭션 정렬이 레이어 1 보안 및 활성을 상속받을 수 있게 하여 이론적으로 더 강력한 탈중앙화를 제공합니다.



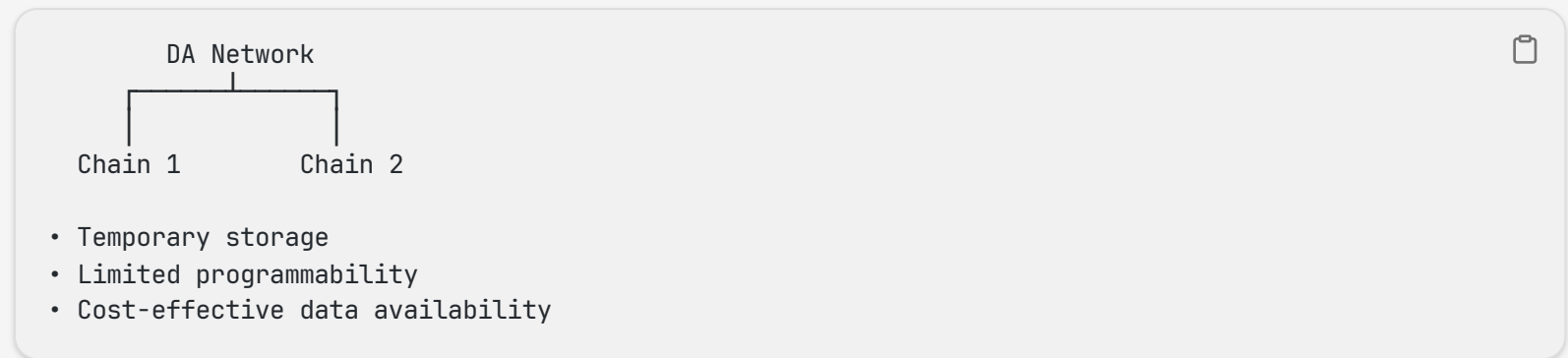
주요 장점	주요 단점
<ul style="list-style-type: none"> <li>• 전통적인 롤업보다 더 온체인 지향적</li> <li>• 강력한 활성 보장</li> <li>• L1 보안 상속</li> <li>• 견고한 이론적 탈중앙화</li> </ul>	<ul style="list-style-type: none"> <li>• 이더리움 사용으로 인한 높은 가스 비용</li> <li>• 게이트웨이 서비스에 의존</li> <li>• 실제로는 덜 탈중앙화됨</li> <li>• 사전 합의 시스템이 중앙화를 재도입</li> </ul>

**Syndicate의 온체인 시퀀싱 관련성:** Syndicate의 접근 방식은 L2 레벨에서 기반 시퀀싱 원칙을 적용하여 누구나 작성할 수 있는 스마트 컨트랙트에 대한 Vitalik의 원래 비전과 일치합니다. 이는 높은 L1 가스 비용을 피하면서 기반 시퀀싱의 많은 이점을 제공합니다.

Syndicate는 또한 시퀀싱 체인 L2를 위해 기반 시퀀싱을 활용할 수 있으며, 이는 더 높은 수수료를 대가로 활성도를 향상시킬 수 있습니다—이는 특정 애플리케이션에 대해 가치 있는 트레이드오프가 될 수 있습니다.

## DA 레이어

Celestia 및 EigenDA와 같은 데이터 가용성(DA) 레이어는 블록체인 데이터를 저렴하고 효율적으로 저장하기 위한 특수 솔루션을 제공하지만, 프로그래밍 가능성이 제한되고 임시 저장소만 제공합니다.

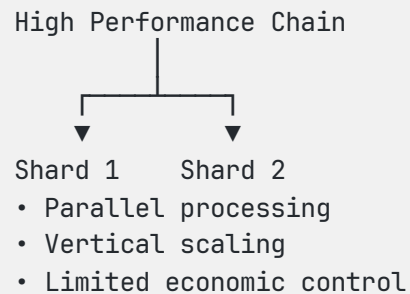


주요 장점	주요 단점
<ul style="list-style-type: none"> <li>• 저비용 데이터 가용성</li> <li>• 높은 트랜잭션 처리량</li> <li>• 광범위한 생태계 수용</li> <li>• 많은 롤업 프레임워크에서 사용</li> </ul>	<ul style="list-style-type: none"> <li>• 제한적이거나 프로그래밍 불가능</li> <li>• 데이터가 일정 기간 후 만료됨</li> <li>• 종종 중앙화된 데이터 게시 필요</li> <li>• 트랜잭션 정렬 로직 없음</li> </ul>

**Syndicate**의 온체인 시퀀싱 관련성: Syndicate의 시퀀싱 체인은 프로그래밍 가능성과 영구 저장소를 추가하면서 데이터 가용성을 제공합니다. DA 레이어는 여전히 중요한 역할을 합니다—Syndicate는 시퀀싱 체인에서 EigenDA를 사용합니다—하지만 온체인 시퀀싱은 순수 DA 레이어가 제공할 수 없는 추가 기능을 제공합니다.

## 수직 L1/L2

MegaETH 및 Monad와 같은 대체 L1/L2 구현은 성능 향상을 위해 병렬화 또는 특수 하드웨어와 같은 기술을 통한 수직 확장에 중점을 둡니다.

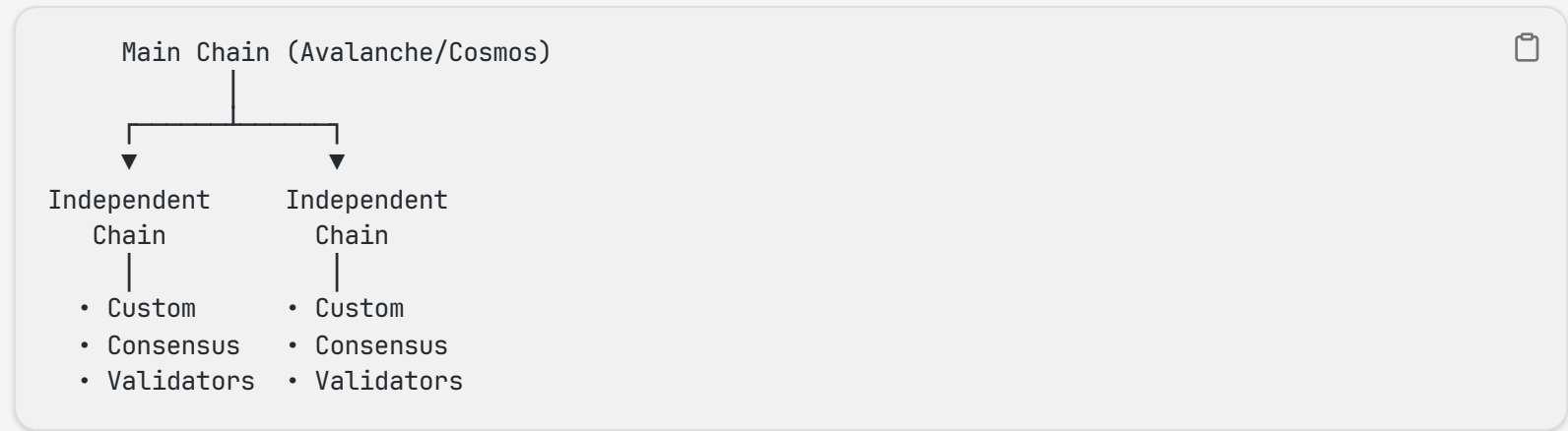


주요 장점	주요 단점
<ul style="list-style-type: none"><li>• 간소화된 배포 프로세스</li><li>• 잠재적으로 더 나은 성능</li><li>• 강력한 네트워크 효과 잠재력</li><li>• 특수 하드웨어 최적화</li></ul>	<ul style="list-style-type: none"><li>• 높은 엔드투엔드 지연 시간</li><li>• 애플리케이션이 경제성을 제어할 수 없음</li><li>• 기존 롤업 인프라보다 검증이 덜 된 상태</li></ul>

**Syndicate**의 온체인 시퀀싱 관련성: Syndicate는 소수의 고성능 체인을 통한 수직적 확장보다는 다수의 애플리케이션별 체인을 통한 수평적 확장에 베팅합니다. 이 접근 방식은 애플리케이션에 경제 및 거버넌스에 대한 통제권을 부여하며, 특히 기반 체인에 가치를 양도하기보다 자체 생태계 내에서 가치를 창출하고자 하는 개발자들에게 매력적입니다.

## 아발란체 서브넷과 코스모스 체인

이러한 접근 방식은 개발자가 각 프레임워크 위에 새로운 체인을 생성할 수 있게 해주며, 아발란체는 EVM 호환 서브넷에 중점을 두고 코스모스는 더 넓은 범위의 체인 아키텍처를 가능하게 합니다.



주요 장점	주요 단점
<ul style="list-style-type: none"><li>• 체인 설계의 완전한 유연성</li><li>• 더 적은 의존성(독립적인 체인)</li><li>• 애플리케이션에 대한 최고 수준의 주권</li><li>• 맞춤형 합의 메커니즘 가능</li></ul>	<ul style="list-style-type: none"><li>• 합의 부트스트래핑 필요</li><li>• 체인 간 복잡한 브리징</li><li>• 복잡한 크로스체인 메시징 설정</li><li>• 상당한 운영 오버헤드</li></ul>

**Syndicate의 온체인 시퀀싱 관련성:** 온체인 시퀀싱은 합의를 부트스트래핑할 필요 없이 애플리케이션별 체인을 출시할 수 있게 해주어, 새롭거나 작거나 토큰 이전 네트워크의 진입 장벽을 크게 낮춥니다. 코스모스 체인이 최대 주권을 제공하지만, 출시하고 유지하는 데 상당히 더 많은 작업이 필요합니다.

## 주요 토크인도유포

## 주요 트레이드오프

각 시퀀싱 접근 방식은 뚜렷한 트레이드오프를 제공합니다:

- 중앙화된 시퀀서: 단순하지만 중앙화됨
- 공유 시퀀싱: 조정되지만 하향식 채택 필요
- 온체인 시퀀싱: 프로그래밍 가능하고 주권이 있지만 더 복잡함
- 베이스드 시퀀싱: 안전하지만 비용이 많이 듦
- **DA 레이어**: 비용 효율적이지만 일시적
- 수직적 **L1/L2**: 고성능이지만 제한된 제어
- 아발란체 서브넷과 코스모스 체인: 최대 주권이지만 높은 운영 복잡성

신디케이트의 온체인 시퀀싱은 스마트 계약의 프로그래밍 가능성과 L2 효율성을 결합하여, 애플리케이션이 경제적 주권을 유지하면서도 수평적 성장을 통해 확장성을 달성할 수 있게 합니다.

### < 신디케이트 스택

신디케이트는 원자적으로 구성 가능하고 프로그래밍 가능한 앱체...

### 빠른 시작 >

Syndicate를 통해 몇 분 안에 첫 번째 앱체인 배포하기

Q Search



K

☐ Syndicate Stack



## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모드



# 빠른 시작

Syndicate를 통해 몇 분 안에 첫 번째 앱체인 배포하기

이 가이드는 새로운 앱체인의 주요 구성 옵션을 안내합니다. **Syndicate**가 배포 및 인프라를 처리합니다 - 여러분은 체인이 어떻게 작동해야 하는지에 대한 중요한 결정만 내리면 됩니다.

## 구성할 항목

1. 체인 매개변수 설정: 앱체인의 기본 설정 정의
2. 시퀀싱 모듈: 트랜잭션이 정렬되는 방식과 참여 가능한 사용자
3. 수수료 모델 구성: 커스텀 토큰 사용을 포함한 트랜잭션 수수료 작동 방식
4. 거버넌스 매개변수 설정: 업그레이드 및 중요 작업 관리 방법
5. 실행 환경 구성: 특정 실행 요구사항
6. 정산 레이어 선택: 정산에 사용할 기본 레이어(L1/L2)

시작해 봅시다!

## 1. 체인 매개변수 설정

이러한 기본 설정은 앱체인의 정체성과 더 넓은 생태계와의 연결 방식을 정의합니다.



- 체인 이름: 체인의 사람이 읽을 수 있는 이름(예: my-appchain).
- 체인 ID: 체인의 고유한 숫자 식별자. 510로 시작하는 체인 ID가 앱체인에 자동으로 할당됩니다.
- 블록 시간: 체인에서 블록 간의 목표 시간(밀리초). 소유자가 불필요하게 높은 컴퓨팅 및 저장 요구사항으로 실행 노드(RPC) 머신을 프로비저닝하는 부담을 갖지 않도록 1초가 좋은 기본값입니다. [자세히 알아보기](#)

## 2. 시퀀싱 모듈

시퀀싱 모듈은 누가 체인에 트랜잭션을 제출할 수 있고 어떤 순서로 처리되는지를 제어합니다. 이는 앱체인의 가장 강력한 커스터마이징 포인트 중 하나입니다.

인기 있는 구성 옵션:

모듈 유형	옵션	최적 사용 사례
권한	허용 목록	초대 전용 체인
	토큰 게이트	커뮤니티 소유 체인
	항상 열림	공개 체인
정렬	선착순	간단하고 공정한 정렬
	수수료 우선순위	수익 최적화
	무작위	MEV 보호
경매	밀봉 입찰 경매	경쟁적 시퀀싱
원자적	크로스체인 조정	복잡한 멀티체인 애플리케이션

자의 임은 위어의 요구 사항에 따라 인격된 모듈을 구현할 것입니다. 더 고급 사용자 정의가 필요한 경우, 위어의 사용에 맞는 맞춤형 모듈도 구축할 수 있습니다.

각 모듈에 대한 더 자세한 기술 정보는 [시퀀싱 모듈 문서](#)를 참조하세요.

### 3. 수수료 모델 구성

체인에서 트랜잭션 수수료가 작동하는 방식을 선택하세요. 여기에는 가스 지불에 사용되는 토큰과 경제 모델의 구조가 포함됩니다.

수수료 토큰 옵션:

토큰 유형	설명	최적 사용처
ETH	가스 지불을 위한 표준 ETH	단순성과 호환성
커스텀 토큰	네이티브 통화로서의 자체 ERC-20	커뮤니티 경제와 맞춤형 사용 사례

팁: 간단하게 시작하고(예: 모든 수수료를 시퀀서나 금고로) 커뮤니티가 성장함에 따라 모델을 발전시키세요.

### 4. 거버넌스 매개변수 설정

앱체인에서 업그레이드와 중요한 결정이 관리되는 방식을 정의하세요. 좋은 거버넌스는 보안을 보장하면서 미래의 개선을 가능하게 합니다.

거버넌스 모델:

모델	설명	최적 사용처
단일 관리자	한 주소가 모든 업그레이드를 제어	초기 단계 또는 테스트 체인

멀티시그	변경 사항을 승인하기 위해 여러 서명자 필요	소규모 팀 또는 신뢰할 수 있는 그룹
DAO/투표	변경 사항에 대한 커뮤니티 투표	커뮤니티 소유 체인

지정해야 할 사항:

- 거버넌스 모델
- 관리자 주소 또는 계약 주소
- 변경 사항에 대한 타임락 기간(있는 경우)

프로젝트가 발전함에 따라 나중에 거버넌스 모델을 변경할 수 있지만, 이는 초기 거버넌스 프로세스를 거쳐야 함을 참고하세요.

## 5. 실행 환경 구성

앱체인은 기본적으로 업계 선도적인 이더리움 롤업인 **Arbitrum Nitro**에서 실행됩니다. 다른 실행 스택(OP Stack 등)에 대한 지원은 곧 제공될 예정입니다.

### Arbitrum Nitro의 장점

- 낮은 수수료: 이더리움 메인넷보다 10~100배 저렴하며, 데이터 가용성 위원회를 사용하면 추가 비용 절감이 가능합니다.
- 이더리움 호환성: 이더리움과 동일한 스마트 계약, 워크플로우 및 도구(Solidity, Hardhat, Foundry 등)를 사용할 수 있습니다.
- 유연한 스마트 계약: 표준 Solidity 또는 Stylus를 사용한 고성능 WASM 계약을 배포할 수 있습니다.
- 상호 운용성: 이더리움 메인넷 또는 EVM 롤업과의 신뢰할 수 있는 브리징 및 더 넓은 이더리움 생태계와의 원활한 통합이 가능합니다.
- 완전 관리형 인프라: 노드 운영, 업그레이드 및 확장을 처리하므로 애플리케이션 로직에 집중할 수 있습니다.

참고: 사전 컴파일, 커스텀 가스 한도 또는 기타 고급 실행 매개변수에 대한 특정 요구 사항이 있는 경우 구성 중에 알려주세요. 앱체인

이 귀하의 요구 사항을 충족하도록 함께 노력하겠습니다.

## 6. 정산 레이어 선택

정산 레이어는 앱체인이 보안을 위해 최종적으로 상태를 고정하는 곳입니다. 이는 비용, 보안 및 상호 운용성에 영향을 미치는 중요한 결정입니다.

정산 옵션 비교:

레이어	보안	비용	속도	최적 용도
 <b>Base</b>	이더리움에 데이터를 게시하지만 중앙화된 시퀀서/검증자에 의존함	중간	빠름	광범위한 배포
 <b>Arbitrum</b>	이더리움에 의해 시행되는 부정 증명을 통한 무신뢰 L2 보안. <b>Arbitrum</b> 으로 정산하면 시퀀싱 수수료가 없습니다.	중간	빠름	DeFi 및 고가치 앱
 <b>Ethereum</b>	네이티브 L1 보안	높음	느림	최고 수준의 보안 요구사항

앱체인에 사용할 정산 레이어를 결정하세요. 저희 팀이 선택한 정산 레이어에 앱체인을 연결하는 모든 기술적 세부 사항을 처리할 것입니다.

## 7. 배포

앱체인이 배포되면 다음에 대한 자격 증명과 접근 권한을 받게 됩니다:

리소스	설명
RPC 엔드포인트	<code>https://rpc.testnet.{chainName}.syndicate.io</code>
익스플로러	<code>https://explorer.testnet.{chainName}.syndicate.io</code>
관리자 콘솔	모니터링 및 관리용
지원 채널	질문이나 문제에 대한 지원

참고: RPC 엔드포인트에 대해 `rpc.mydomain.com` 와 같은 커스텀 서브도메인도 제공합니다.

## 8. 첫 상호작용

앱체인이 활성화되면 다음을 수행하게 됩니다:

### MetaMask에 앱체인 추가하기:

1. 체인의 블록 익스플로러 방문: `https://explorer.testnet.{chainName}.syndicate.io`
2. 원클릭 네트워크 설정을 위해 "**MetaMask에 추가**" 버튼을 클릭하세요.
3. 자세한 내용이나 대체 방법은 지갑 구성을 참조하세요.

### 체인 소유자 지갑에 자금 추가하기:

네이티브 토큰(ETH 또는 커스텀 토큰)을 체인 소유자 지갑 주소로 보내세요.

이를 통해 새 앱체인과 즉시 상호작용할 수 있습니다.

## 다음 단계

앱체인의 커스터마이징 및 최적화에 대해 더 자세히 알아보려면 다음 리소스를 확인하세요:

- 체인 매개변수: 체인에 대한 모든 구성 가능한 매개변수와 고급 설정 옵션에 대해 알아보세요.
- 시퀀싱 모듈: 트랜잭션 흐름과 권한을 제어하는 데 사용할 수 있는 시퀀싱 로직과 모듈의 전체 범위를 탐색하세요.
- 프레임워크 선택: 사용 가능한 다양한 실행 및 정산 프레임워크를 이해하고 필요에 맞는 최적의 스택을 선택하는 방법을 알아보세요.

이 가이드는 Syndicate 앱체인을 최대한 활용하고 고급 배포를 준비하는 데 도움이 될 것입니다.

< 시퀀싱 아키텍처

시퀀싱 아키텍처의 이해와 신디케이트의 접근 방식 비교

체인 설정 >

앱체인 배포 시 구성할 수 있는 체인 매개변수 이해하기

## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모드



# 체인 설정

앱체인 배포 시 구성할 수 있는 체인 매개변수 이해하기

체인 매개변수는 블록체인의 핵심 정체성과 운영 규칙을 정의합니다. 새로운 Syndicate 앱체인을 설정할 때, 다음 세 가지 주요 매개변수를 구성하는 것부터 시작할 수 있습니다:

1. 체인 이름: 체인의 사람이 읽을 수 있는 이름
2. 체인 ID: 체인의 고유 숫자 식별자
3. 블록 시간: 체인에서 블록이 생성되는 빈도

이 가이드는 각 매개변수와 선택 시 고려해야 할 사항을 설명합니다.

## 1. 체인 이름

앱체인의 사람이 읽을 수 있는 이름입니다(공백이나 특수 문자 없음). 이는 지갑, 익스플로러 및 사용자 인터페이스에서 사용됩니다. 프로젝트나 커뮤니티에 맞는 짧고, 기억하기 쉬우며, 고유한 이름을 선택하세요. 체인 이름은 한번 설정되면 지갑, 익스플로러 및 사용자 인터페이스에서 참조되므로 앱체인을 다시 배포하지 않고는 변경할 수 없습니다.

## 2. 체인 ID

앱체인의 고유 숫자 식별자입니다. 510 로 시작하는 체인 ID가 앱체인에 자동으로 할당됩니다. 체인 ID는 한번 설정되면 지갑, 익스플로러 및 사용자 인터페이스에서 참조되므로 앱체인을 다시 배포하지 않고는 변경할 수 없습니다.

### 3. 블록 시간

블록 시간은 체인에서 새 블록이 생성되는 빈도를 결정합니다. 이는 사용자 경험, 처리량, 주로 네트워크 지연 시간 및 인프라 요구 사항에 영향을 미칩니다.

짧은 블록 시간(예: 250ms)은 더 빠른 트랜잭션 확인을 가능하게 하여 고빈도 애플리케이션의 사용자 경험을 향상시킵니다. 그러나 짧은 블록 시간은 노드에 더 큰 컴퓨팅 및 네트워킹 부담을 주어 안정성에 영향을 미치거나 더 강력한 인프라가 필요할 수 있습니다. 긴 블록 시간(예: 1-2초)은 노드에 대한 부담을 줄이지만 트랜잭션 확인이 느려집니다.

지원 범위: Arbitrum Nitro가 지원하는 기능을 기반으로 블록 시간을 최소 250밀리초(0.25초)까지 설정할 수 있습니다.

고려 사항 및 모범 사례에 대한 자세한 내용은 [블록 시간에 관한 종합 가이드](#)를 참조하세요.

< 빠른 시작

Syndicate를 통해 몇 분 안에 첫 번째 앱체인 배포하기

커스텀 가스 토큰 추가하기 >

Syndicate 앱체인에서 커스텀 토큰을 설계, 구현 및 통합하는 방...



## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모드



# 커스텀 가스 토큰 추가하기

Syndicate 앱체인에서 커스텀 토큰을 설계, 구현 및 통합하는 방법을 알아보세요.

Syndicate 앱체인은 트랜잭션 수수료("가스")를 지불하는 데 사용되는 토큰을 선택할 수 있는 유연성을 제공합니다. ETH로만 제한되지 않고, 표준 ERC-20, 커스텀 구현 또는 브릿지된 자산을 체인의 가스 토큰으로 선택할 수 있습니다. 이를 통해 프로젝트의 경제성, 사용자 경험 및 커뮤니티 참여에 새로운 가능성이 열립니다.

다음 중에서 선택할 수 있습니다:

옵션	설명	최적 사용 사례
ETH	가스 지불을 위한 표준 ETH	단순성과 호환성
커스텀 토큰	네이티브 통화로서의 자체 ERC-20	커뮤니티 경제 및 커스텀 사용 사례

기본적으로 Syndicate 앱체인은 네이티브 가스 토큰으로 커스텀 ERC-20 토큰을 사용합니다. 이 접근 방식은 비용 효율성이 훨씬 높습니다: 커스텀 토큰 초기화 비용은 약 \$0.01인 반면, ETH는 브릿지 초기화 요구 사항으로 인해 \$1,000 이상이 듭니다.

## 커스텀 가스 토큰을 사용하는 이유?

- **낮은 비용:** 커스텀 가스 토큰은 트랜잭션 수수료를 크게 줄일 수 있습니다. 예를 들어, Degen Chain은 자체 DEGEN 토큰을 가스로 사용하여 초저가 수수료를 가능하게 합니다—때로는 L2 비용의 아주 작은 일부에 불과합니다.
- **커뮤니티 연계:** 프로젝트에 이미 네이티브 토큰과 커뮤니티가 있다면, 이를 가스로 사용하면 참여도를 높이고 온보딩을 더 쉽게 만

를 수 있습니다(사용자가 비싼 L1 트랜잭션을 건너뛰고 L2에서 직접 입금할 수 있음).

- **경제적 유연성:** 커스텀 가스 토큰을 통해 인센티브를 설계하고, 특정 사용자나 앱에 보조금을 지급하며, 생태계 성장을 형성할 수 있습니다.

## 주요 고려사항

- **토큰 선택은 고정적임:** 출시 후 ETH와 커스텀 가스 토큰 간 전환은 복잡합니다—신중하게 선택하세요.
- **토큰노믹스:** 가스 토큰이 인플레이션, 디플레이션 또는 고정 공급량인지 결정하세요.
- **업그레이드 가능성 및 거버넌스:** 토큰과 수수료 모델이 시간이 지남에 따라 어떻게 발전할 수 있는지 계획하세요.
- **호환성:** 선택한 토큰이 수수료 모델 및 인프라와 호환되는지 확인하세요.

< 체인 설정

앱체인 배포 시 구성할 수 있는 체인 매개변수 이해하기

시퀀싱 구성하기 >

앱체인을 위한 시퀀싱 구성 방법 알아보기

Q Search



K

☐ Syndicate Stack



## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모듈



# 시퀀싱 구성하기

앱체인을 위한 시퀀싱 구성 방법 알아보기

## 온체인 시퀀싱 작동 방식

1. 사용자가 트랜잭션을 제출하여 앱체인의 멤풀에 전송합니다
2. 실행 엔진이 트랜잭션을 검증하고 조정하여 처리합니다
3. 시퀀싱 체인의 시퀀싱 모듈이 권한, 순서, 경매 및 원자적 실행을 제어합니다
4. 정산 레이어가 주기적인 커밋으로 상태를 최종 확정합니다

이 아키텍처는 실행과 트랜잭션 순서 지정을 분리하여 기존 실행 환경과의 호환성을 유지하면서 시퀀싱 혁신을 가능하게 합니다.

## 기존 모듈 선택하기

프로덕션 준비가 완료된 모듈 중에서 선택하세요:

카테고리	모듈	설명	최적 사용 사례
권한	허용 목록	관리자 제어 접근 목록	테스트, 비공개 네트워크, 제어된 출시
권한	토큰 잔액	토큰 보유량 기반 권한	토큰 게이트 시퀀싱, 경제적 보안
권한	항상 허용	테스트를 위한 개방형 접근	개발 및 테스트

경매	밀봉 입찰	시퀀싱 권한에 대한 경쟁 입찰	시장 기반 시퀀서 선택
원자적	원자적 시퀀서	크로스 체인 트랜잭션 조정	복잡한 다중 체인 작업

## 커스텀 모듈 구축하기

다음은 인터페이스를 사용하여 구축할 수 있는 모듈입니다:

카테고리	모듈	설명	최적 사용 사례
권한	DAO 투표	커뮤니티 주도 시퀀서 선택	분산형 거버넌스
권한	다중 서명	다중 승인 요구사항	고보안 애플리케이션
권한	평판	과거 성과 기반 권한	품질 중심 네트워크
권한	타임락	예약된 시퀀싱 권한	예측 가능한 순환
순서	수수료 우선순위	가스 가격별 순서 지정	수익 최적화
순서	랜덤	무작위 트랜잭션 순서 지정	MEV 보호
순서	배치 경매	시간 창 기반 순서 지정	공정한 가격 발견
순서	동적 크기	적응형 배치 크기 조정	네트워크 최적화
경매	네덜란드식	가격 하락 경매	빠른 가격 발견
경매	영국식	가격 상승 경매	최대 가치 추출
경매	연속	롤링 경매 기간	지속적인 가용성

## 컴포즈 모듈

팩토리 계약을 사용하여 모듈을 결합해 정교한 시퀀싱 솔루션을 만들 수 있습니다:

```
// Deploy the factory
SyndicateFactory factory = new SyndicateFactory();

// Create a RequireAll module for combining multiple checks
RequireAllModule permissions = RequireAllModule(
    factory.createAllContractsWithRequireAllModule(
        admin,
        manager,
        l3ChainId,
        salt
    ).permissionModule
);

// Add permission checks
permissions.addCheck(
    address(new TokenBalanceSequencingModule(token, minBalance)),
    true // add to head of list
);
permissions.addCheck(
    address(new AllowListSequencingModule(admin)),
    false // add to tail of list
);
```

### 구성 기능

- **RequireAll**: 모든 검사가 통과해야 함(AND 로직)
- **RequireAny**: 최소 하나의 검사가 통과해야 함(OR 로직)
- 연결 리스트 구조: 검사는 순서가 있는 리스트에 저장됨
- 동적 업데이트: 런타임에 검사 추가 또는 제거 가능
- 관리자 제어: 관리자만 구성을 수정할 수 있음



## 시작하기

1. **모듈 선택하기:** 사용 가능한 모듈을 검토하고 보안, 공정성 및 성능 요구 사항에 맞는 모듈을 선택하세요. 필요한 경우, 특정 요구 사항을 충족하는 커스텀 모듈을 계획하세요.
2. **커스텀 로직 구현하기:** 인터페이스를 구현하여 새로운 모듈을 만들고 개발 도구를 사용하여 철저히 테스트하세요.
3. **체인 구성하기:** 선택한 모듈로 체인을 배포하고 초기화하세요.
4. **모니터링 및 업그레이드:** 성능 지표를 추적하고 최적의 체인 운영을 보장하기 위해 모듈식 아키텍처를 사용하여 필요에 따라 모듈을 업그레이드하세요.

< 커스텀 가스 토큰 추가하기

Syndicate 앱체인에서 커스텀 토큰을 설계, 구현 및 통합하는 방...

체인 사용하기 >

앱체인과 상호작용하기 위한 집중 가이드

## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모드



# 체인 사용하기

앱체인과 상호작용하기 위한 집중 가이드

이 가이드는 앱체인과 상호작용하는 네 가지 주요 측면을 다룹니다:

1. 트랜잭션 제출: 앱체인에서 트랜잭션을 보내는 방법.
2. 가스 추정: 트랜잭션에 대한 가스를 추정하고 관리하는 방법.
3. 오류 처리: 앱체인에서 오류를 처리하는 방법.
4. 복구 절차: 문제에서 복구하고 앱체인 특정 복구 메커니즘을 사용하는 방법.

## 트랜잭션 제출

앱체인에서는 지갑이 설정되고 연결되면 트랜잭션 제출은 다른 EVM 체인과 매우 유사하게 작동하지만, 앱체인 특정 트랜잭션 유형이나 메타데이터가 포함될 수 있습니다.

```
// Example: Connect to your appchain's RPC
const provider = new ethers.JsonRpcProvider('https://your-appchain-rpc-url');
const chainId = 12345; // Your appchain's unique chain ID

// Make sure your wallet/signer is set up for the appchain
const signer = provider.getSigner();

// Submit a transaction (could be ETH, tokens, or custom logic)
const tx = await signer.sendTransaction({
  to: recipientAddress,
  value: ethers.utils.parseEther('0.1')
```



```

    value: otherStatus.parsedValue(0.1),
    chainId // ensure correct chain
  });
  const receipt = await tx.wait();
  console.log('Confirmed on appchain:', receipt.transactionHash);

```

## 가스 추정

앱체인은 공개 체인과 비교하여 다른 가스 가격 모델을 사용하거나 최적화가 있을 수 있습니다. 일부는 기본 수수료가 낮거나, 고정 가스 가격을 가지거나, 심지어 보조금이 지급되는 트랜잭션을 가질 수 있습니다.

```

// Estimate gas using your appchain's provider
const gasEstimate = await provider.estimateGas({
  to: contractAddress,
  data: contract.interface.encodeFunctionData('method', [args])
});
// Appchains may recommend a different buffer or have a fixed gas limit
const gasLimit = gasEstimate.mul(120).div(100);

```

## 오류 처리

앱체인은 공개 체인에서 찾을 수 없는 사용자 정의 오류 코드, 되돌림 이유 또는 동작을 도입할 수 있습니다. 앱체인 특정 로직, 거버넌스 또는 상태 전환과 관련된 오류가 발생할 수 있습니다.

```

try {
  const tx = await signer.sendTransaction(txData);
  const receipt = await tx.wait();
  if (receipt.status === 0) throw new Error('Transaction failed');
} catch (error) {
  // Appchain-specific error handling
  if (error.code === 'APPCHAIN_CUSTOM_ERROR') {
    // Handle custom error
  } else if (error.code === 'INSUFFICIENT_FUNDS') {

```



```
// Not enough balance
} else {
    throw error;
}
}
```

**트랜잭션 시뮬레이션:** 트랜잭션 시뮬레이션은 현재 체인 상태를 기반으로 트랜잭션이 성공할지 여부만 확인합니다. 그러나 시뮬레이션이 성공하더라도 표준 시뮬레이션 도구가 감지할 수 없는 시퀀싱 계약의 사용자 정의 로직으로 인해 트랜잭션이 나중에 실패할 수 있습니다.

## 복구 절차

앱체인은 온체인 거버넌스, 관리자 키 또는 사용자 정의 복구 계약과 같은 고유한 복구 메커니즘을 제공할 수 있습니다. 이는 공개 체인 복구 프로세스와 크게 다를 수 있습니다.

- **멈춘 트랜잭션:** 앱체인은 멈춘 트랜잭션을 해결하기 위해 거버넌스나 관리자 개입을 허용할 수 있습니다.
- **계정 복구:** 일부 앱체인은 소셜 복구, 다중서명 또는 사용자 정의 계정 추상화를 지원합니다.
- **비상 절차:** 앱체인은 회로 차단기, 일시 중지 가능한 계약 또는 기타 앱 특정 안전 기능을 가질 수 있습니다.

```
// Example: Using an appchain's emergency pause feature
await contract.pause(); // Only available if your appchain implements this
```



< 시퀀싱 구성하기

앱체인을 위한 시퀀싱 구성 방법 알아보기

자산 브릿지 >

자산 브릿지 방법 알아보기

Q Search



K

☐ Syndicate Stack



## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모드



# 자산 브릿지

## 자산 브릿지 방법 알아보기

브릿징은 개발자가 네트워크 간에 자산을 이동시키는 방법으로, 정산 체인(예: 이더리움 또는 베이스)에서 앱체인으로 유동성을 가져오고 필요할 때 사용자가 다시 나갈 수 있게 합니다. 시스템 지갑에 자금을 공급하고, 유동성을 시드하고, 원활한 크로스 네트워크 제품 경험을 구축하는 데 사용하세요.

입금은 표준 브릿지 흐름을 따르고, 출금은 신디케이트의 빠르고 안전한 출금 경로를 사용합니다.

## 브릿지 UI 안내

1. 브릿지 접속: 아비트럼 브릿지를 엽니다.
2. 지갑 연결: 메타마스크나 다른 지원되는 지갑을 사용합니다.
3. 앱체인 추가(한 번만): 대상 네트워크 선택기에서 '커스텀 체인 추가'를 선택하고 앱체인 세부 정보(체인 ID, 이름, RPC URL, 익스플로러 URL)를 제공합니다.
4. 네트워크 선택: 소스(예: 이더리움/베이스)와 대상(앱체인) 네트워크를 선택합니다.
5. 자산 및 금액 선택: 토큰을 선택하고 브릿지하려는 금액을 입력합니다.
6. 트랜잭션 제출: 세부 정보를 검토하고 지갑에서 트랜잭션을 확인합니다.

## 입금 과정(부모 체인 → 앱체인)

```

+-----+ +-----+ +-----+
| Grant Token Allowance (ERC-20) |-->| Submit Deposit |-->| Receive Bridged Funds |
+-----+ +-----+ +-----+

```

1. 토큰 허용량 부여(ERC-20만 해당): 지갑에서 브릿지 컨트랙트가 선택한 토큰을 사용할 수 있도록 승인합니다 - 네이티브 자산 (예: ETH)의 경우 허용량이 필요하지 않습니다.
2. 입금 제출: 금액을 입력하고, 세부 정보를 검토한 후 소스 네트워크에서 입금 트랜잭션을 확인합니다.
3. 브릿지된 자금 수령: 확인 후, 브릿지된 자산은 몇 분 내에 앱체인에 나타납니다.

## 출금 과정(앱체인 → 부모 체인)

```

+-----+ +-----+ +-----+
| Submit Withdrawal |-->| Fast Validation |-->| Finalize on Parent Chain |
+-----+ +-----+ +-----+

```

출금은 신디케이트의 속도와 보안을 위한 맞춤형 시스템으로 구동됩니다:

1. 출금 제출: 앱체인에서 자산과 금액을 선택한 다음 출금 트랜잭션을 확인합니다.
2. 빠른 검증: 시스템은 TEE와 ZKVM 증명을 사용하여 속도와 보안을 위해 요청을 검증합니다.
3. 부모 체인에서 완료: 표준 브릿지의 챌린지 기간이 끝난 후, 지갑을 사용하여 부모 체인에서 수동으로 출금을 완료/청구해야 합니다. 자금은 자동으로 이체되지 않습니다.

더 깊은 이해를 위해 자산 인출 핵심 개념을 참조하세요.

## 개발자 보안 모범 사례

- 브릿지 활동을 모니터링하고 비정상적인 패턴에 대한 알림 설정



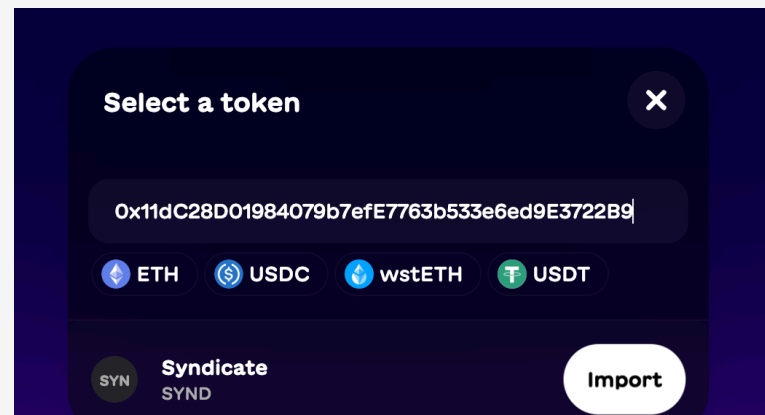
- 사용 사례에 적합한 속도 제한 및 가치 상한선 구현
- 브릿지 계약 구성을 정기적으로 검토 및 감사
- 잠재적 보안 사고에 대한 비상 대응 절차 마련
- 메인넷 배포 전 테스트넷에서 브릿지 기능을 철저히 테스트

## Base로 SYND 토큰 브리징하기

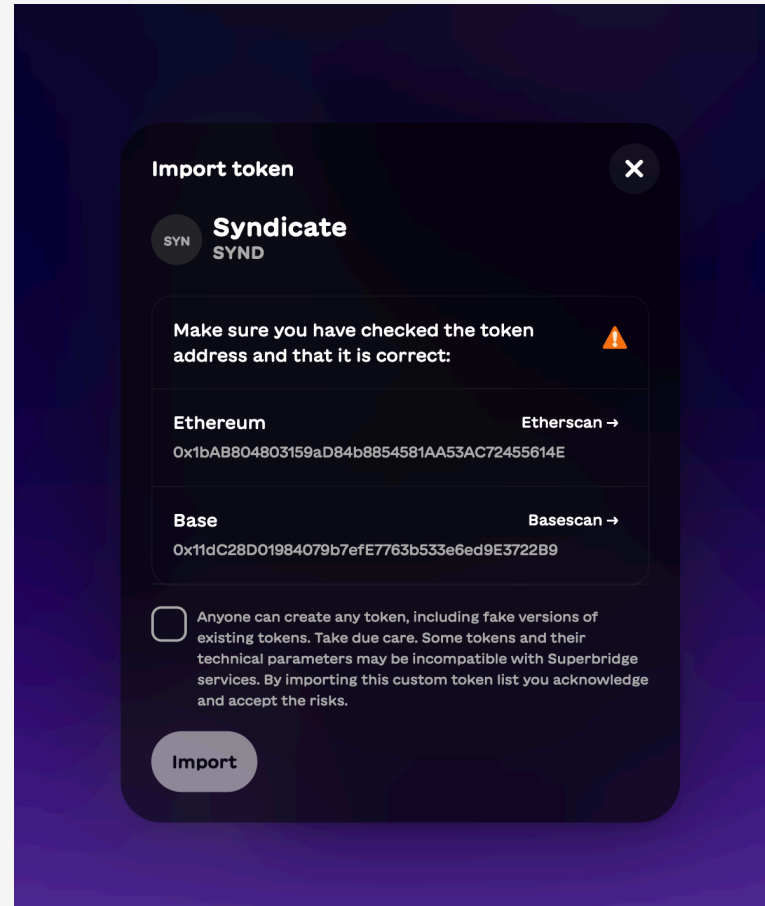
Superbridge라는 사용자 친화적인 브리징 인터페이스를 사용하여 이더리움 메인넷에서 Base로 SYND 토큰을 브리징할 수 있습니다.

### Superbridge 사용하기

1. 브리지 UI 접속하기: Superbridge로 이동하여 사전 구성된 SYND 토큰 매개변수를 확인하세요.
2. 지갑 연결하기: 지갑이 이더리움 메인넷에 연결되어 있는지 확인하세요.
3. SYND 토큰 추가하기: 토큰 검색창에 SYND 토큰 주소를 붙여넣으세요:
  - Base 네트워크의 SYND: `0x11dC28D01984079b7efE7763b533e6ed9E3722B9`



4. 토큰 가져오기: 토큰을 찾는 데 문제가 있다면, ETH 메인넷의 SYND 주소  
( 0x1bAB804803159aD84b8854581AA53AC72455614E )를 붙여넣어야 할 수도 있습니다.



5. 브리지 완료하기: 표준 브리징 과정을 따라 SYND 토큰을 Base로 전송하세요.

## 토큰 주소

- 이더리움의 SYND: 0x1bAB804803159aD84b8854581AA53AC72455614E

- **Base의 SYND:** 0x11dC28D01984079b7efE7763b533e6ed9E3722B9

< 체인 사용하기

앱체인과 상호작용하기 위한 집중 가이드

지갑 설정 >

앱체인에 지갑을 연결하고 최종 사용자 지갑 UX를 설계하는 방법



Q Search



K

Syndicate Stack



## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모드



○ 앱체인의 블록 탐색기에 접근하는 방법:



## 체인 모니터링

신디케이트 앱체인을 모니터링하기 위한 블록 탐색기 사용 방법

블록 탐색기는 앱체인에 필수적입니다: 투명성을 제공하고, 트랜잭션을 추적하며, 컨트랙트를 검증하고, 네트워크 상태를 모니터링할 수 있습니다. 신디케이트 앱체인의 경우, 기본적으로 호스팅된 **Blockscout** 탐색기가 제공되어 트랜잭션을 모니터링하고, 컨트랙트를 검증하며, 앱체인의 데이터를 탐색할 수 있습니다.

### 앱체인의 블록 탐색기에 접근하는 방법:

- 신디케이트 플랫폼에서 앱체인 관리 콘솔을 방문하세요.
- "블록 탐색기" 링크를 찾으면 앱체인 전용 탐색기 인스턴스로 연결됩니다.

### 왜 Blockscout인가?

Blockscout은 EVM 체인, 롤업 및 앱체인을 위한 모든 기능을 갖춘 오픈소스 블록체인 탐색기입니다. 다음과 같은 기능을 제공합니다:

- 실시간 트랜잭션 추적
- 스마트 컨트랙트 검증 및 상호작용
- 토큰 지원 (ERC-20, ERC-721, ERC-1155)
- 디코딩된 컨트랙트 호출 및 로그

- 맞춤형 뷰 및 API
- 오픈소스 및 멀티체인 지원

Blockscout의 기능에 대해 더 알아보려면 [공식 문서](#)를 참조하세요.

#### < 지갑 설정

앱체인에 지갑을 연결하고 최종 사용자 지갑 UX를 설계하는 방법

#### 시퀀싱 모듈 >

맞춤형 트랜잭션 순서 지정, 권한, 경매 및 크로스체인 조정을 위...



Q Search

⌘

K

☐ Syndicate Stack

↕

## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모듈

한글

☀

🌙

# 시퀀싱 모듈

맞춤형 트랜잭션 순서 지정, 권한, 경매 및 크로스체인 조정을 위한 모듈식 스마트 계약

신디케이트는 별도의 블록체인에 배포된 스마트 계약을 통해 블록체인 시퀀싱을 온체인으로 가져와 변화시킵니다. 이러한 모듈식 접근 방식은 트랜잭션 순서 지정 프로세스에 대한 전례 없는 제어, 투명성 및 커뮤니티 소유권을 가능하게 합니다.

신디케이트 시퀀싱 모듈은 특정 애플리케이션 요구 사항을 충족하기 위해 독립적으로 구성할 수 있는 여러 핵심 구성 요소로 이루어져 있습니다.

## 권한 모듈

허용 목록 및 토큰 보유량과 같은 메커니즘을 통해 트랜잭션 시퀀싱을 제어할 수 있는 사람 관리

## 순서 지정 모듈

최적의 처리를 위해 트랜잭션이 정렬되고 일괄 처리되는 방식 정의

## 경매 모듈

밀봉 입찰 경매와 같은 메커니즘을 통한 경쟁적 시퀀싱 권한 활성화

## 원자적 모듈

복잡한 작업을 위해 여러 체인에 걸친 조정된 시퀀싱 허용

## 권한 모듈

권한 모듈은 앱체인에서 트랜잭션 시퀀싱을 수행할 수 있는 사람을 제어합니다. 이들은 주소가 시퀀싱할 수 있는지 여부를 반환하는

`isAllowed` 함수가 있는 간단한 인터페이스를 구현합니다:

```
function isAllowed(address proposer) external view returns (bool);
```



사용 가능한 모듈

#### 사용 가능한 모듈

- 허용 목록: 관리자가 제어하는 허용된 시퀀서의 간단한 목록
- 토큰 잔액: 시퀀서가 최소한의 토큰 양을 보유하도록 요구
- 밀봉 입찰 경매: 시퀀싱 권한에 대한 경쟁적 입찰

#### 예시: 토큰 잔액 모듈

다음은 시퀀서가 토큰을 보유해야 하는 간단한 예시입니다:

```
contract TokenBalanceSequencingModule is PermissionModule {  
    address public immutable tokenAddress;  
    uint256 public immutable minimumBalance;  
  
    function isAllowed(address proposer) external view override returns (bool) {  
        return IERC20(tokenAddress).balanceOf(proposer) >= minimumBalance;  
    }  
}
```

#### 순서 지정 모듈

트랜잭션 순서 지정은 현재 기본 `SyndicateSequencingChain` 계약에 의해 처리되며, 제출된 순서대로 트랜잭션을 처리합니다. 이는 간단하고 예측 가능한 트랜잭션 순서 지정 메커니즘을 제공합니다.

#### 사용 가능한 기능

- 단일 트랜잭션 처리: `processTransaction` 를 통한 개별 트랜잭션 처리
- 대량 트랜잭션 처리: `processBulkTransactions` 로 여러 트랜잭션을 한 번에 처리
- 원시 트랜잭션 지원: `processTransactionRaw` 로 압축된 트랜잭션 처리

#### 예시: 트랜잭션 처리

```
contract SyndicateSequencingChain is SequencingModuleChecker {
```



```

event TransactionProcessed(address indexed sender, bytes data);

function processTransaction(bytes calldata data) external onlyWhenAllowed(msg.sender) {
    emit TransactionProcessed(msg.sender, prependZeroByte(data));
}

function processBulkTransactions(bytes[] calldata data) external onlyWhenAllowed(msg.sender) {
    for (uint256 i = 0; i < data.length; i++) {
        emit TransactionProcessed(msg.sender, prependZeroByte(data[i]));
    }
}
}

```

## 경매 모듈

경매 모듈은 봉인된 입찰을 통해 공정성을 보장하기 위해 입찰자가 암호화된 입찰가를 제출하는 경쟁적 시퀀싱 권한을 가능하게 합니다.

### 사용 가능한 모듈

- 봉인 입찰 경매: 입찰자가 먼저 암호화된 입찰가를 제출하고 나중에 공개하는 2단계 경매

### 예시: 봉인 입찰 경매

```

contract SealedBidAuctionSequencingModule is PermissionModule {
    // Track bids and auction state
    mapping(address => bytes32) public sealedBids;
    address public highestBidder;
    uint256 public highestBid;

    // Phase 1: Submit sealed bids
    function bid(bytes32 sealedBid) external payable {
        require(msg.value > 0, "Must send deposit");
        sealedBids[msg.sender] = sealedBid;
    }

    // Phase 2: Reveal bids
    function revealBid(uint256 bidAmount, string memory salt) external {
        bytes32 computedHash = keccak256(abi.encodePacked(bidAmount, salt));
    }
}

```



```

        bytes32 computedHash = keccak256(abi.encodePacked(bidAmount, salt));
        require(computedHash == sealedBids[msg.sender], "Invalid bid reveal");

        if (bidAmount > highestBid) {
            highestBid = bidAmount;
            highestBidder = msg.sender;
        }
    }

    // Only highest bidder can sequence
    function isAllowed(address proposer) external view override returns (bool) {
        return proposer == highestBidder;
    }
}

```

## 원자적 모듈

원자적 모듈은 여러 체인에 걸쳐 조정된 트랜잭션 처리를 가능하게 하여 모든 트랜잭션이 함께 처리되거나 전혀 처리되지 않도록 보장합니다.

### 사용 가능한 모듈

- 원자적 시퀀서: 여러 체인에 걸쳐 트랜잭션을 원자적으로 처리

### 예시: 원자적 시퀀서

```

contract AtomicSequencer {
    // Process transactions on multiple chains atomically
    function processTransactionsAtomically(
        SyndicateSequencingChain[] chains,
        bytes[] transactions
    ) external {
        // Ensure input arrays match
        require(chains.length == transactions.length, "Length mismatch");

        // Process each transaction on its respective chain
        for (uint256 i = 0; i < chains.length; i++) {
            chains[i].processTransaction(transactions[i]);
        }
    }
}

```



```

    }

    // Process multiple transactions per chain atomically
    function processBulkTransactionsAtomically(
        SyndicateSequencingChain[] chains,
        bytes[][] transactions
    ) external {
        require(chains.length == transactions.length, "Length mismatch");

        for (uint256 i = 0; i < chains.length; i++) {
            chains[i].processBulkTransactions(transactions[i]);
        }
    }
}

```

원자적 시퀀서는 모든 지정된 체인에서 모든 트랜잭션이 성공적으로 처리되거나 아무것도 처리되지 않도록 보장합니다. 이는 일관성을 유지해야 하는 크로스체인 작업에 유용합니다.

#### < 체인 모니터링

신디케이트 애플체인을 모니터링하기 위한 블록 탐색기 사용 방법

#### 실행 프레임워크 >

우리의 실행 프레임워크가 Arbitrum Orbit으로 시작하는 이유...

## GET STARTED

[소개](#)[왜 앱체인인가](#)[신디케이트 스택](#)[시퀀싱 아키텍처](#)[빠른 시작](#)

## GUIDES

[체인 설정](#)[커스텀 가스 토큰 추가하기](#)[시퀀싱 구성하기](#)[체인 사용하기](#)[자산 브릿지](#)[지갑 설정](#)[체인 모니터링](#)

## CORE CONCEPTS

[시퀀싱 모드](#)

# 실행 프레임워크

우리의 실행 프레임워크가 Arbitrum Orbit으로 시작하는 이유와 제공하는 주요 기술적 이점

우리가 롤업 프레임워크로 Arbitrum Orbit을 선택한 이유는 성능, 유연성 및 사용자 경험을 향상시키는 다섯 가지 주요 기술적 이점 때문입니다.

1. 빠른 블록 시간
2. 우수한 지연 시간 성능
3. 내장된 맞춤화 가능성
4. Stylus WASM 환경
5. 브리징 인프라

## 빠른 블록 시간

Arbitrum Orbit은 빠른 블록 시간을 통해 경쟁사에 비해 현저히 빠른 트랜잭션 처리를 가능하게 하여 사용자와 애플리케이션에 명확한 성능 이점을 제공합니다.

- 권장 최소값: 250 밀리초
- 달성 가능 범위: 100-150 밀리초 (주의 필요)
- 권장하지 않음: 100ms 미만, 50ms에서는 강력히 권장하지 않음

## 우수한 지연 시간 성능

사용자가 실제로 인식하고 신경 쓰는 더 빠른 트랜잭션 영수증. 블록 시간은 내부 체인 작업에 도움이 되지만, 사용자가 보고 느끼는 것은 엔드투엔드 트랜잭션 영수증입니다.

엔드투엔드 트랜잭션 지연 시간: Arbitrum Orbit의 지연 시간은 일반적으로 1초 정도입니다(OP Stack 지연 시간의 1/2에서 1/3).

## 내장된 맞춤화 가능성

Arbitrum Orbit은 처음부터 맞춤화 가능한 기반으로 설계되었습니다. 주요 맞춤화 가능 구성 요소:

- 커스텀 가스 토큰: 완전 지원 (주목할 만한 점은 Optimism이 이 기능을 중단했다는 것)
- 시퀀싱 레이어: 나머지 기술과 더 분리된 구성 요소이기 때문에 OP Stack보다 더 유연함
- 프리컴파일: 쉬운 통합 및 지원

## Stylus WASM 환경

Arbitrum Orbit의 Stylus WASM 환경은 계약 개발에 상당한 유연성을 제공하여 전통적인 EVM 기반 체인을 넘어서는 새로운 가능성을 열어줍니다.

- 언어 지원: Solidity와 함께 Rust로 계약 작성 가능
- 계산 이점: Rust 계약은 Solidity만으로는 처리할 수 없는 훨씬 더 복잡한 계산을 처리할 수 있습니다. (저장 능력을 증가시키지는 않지만 계산 능력을 증가시킵니다)
- 사용 사례: 순수 Solidity에서는 불가능하거나 비실용적인 기능을 가능하게 함 (예: 복잡한 완전 온체인 NFT 메타데이터 생성, TEE 증명 검증과 같은 고급 암호화 기능)

## 인프라 연결

- 네이티브 브릿지 **UI**: 모든 Arbitrum 체인은 동일한 친숙한 브릿지 인터페이스를 활용할 수 있습니다
- 사용자 혜택: 사용자는 이미 알고 신뢰하는 브릿지와 상호작용합니다
- 출금 관리: 출금 처리를 위한 유연한 옵션 (*Syndicate가 출금을 처리하는 방법에 대해 자세히 알아보세요*)

## 요약

Arbitrum Orbit는 빠른 트랜잭션과 친숙한 브릿지 인터페이스를 통해 우수한 사용자 경험을 유지하면서 맞춤화를 위해 필요한 유연성을 갖춘 강력하고 고성능 기반을 제공합니다.

### < 시퀀싱 모듈

맞춤형 트랜잭션 순서 지정, 권한, 경매 및 크로스체인 조정을 위...

### 자산 출금 >

TEE + zk 증명을 사용한 빠르고 안전한 앱체인 출금 시스템으로 ...



## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모드



## 자산 출금

TEE + zk 증명을 사용한 빠르고 안전한 앱체인 출금 시스템으로 네이티브 브릿지를 활용합니다. 주요 개념과 필요한 구성을 알아보세요.

상호운용성은 앱체인이 도달 범위, 유틸리티 및 사용자 경험을 극대화하는 데 중요합니다. Syndicate의 출금 시스템은 신뢰할 수 있는 실행 환경(TEE)과 zk 검증 증명을 결합하여 허가 없이 비대화형 검증을 가능하게 합니다. 이 시스템은 챌린지 기반 부정 증명을 대체하여 네이티브 브릿지 보안으로 거의 즉각적인 완결성을 제공하며, 포크나 추가 운영 오버헤드 없이 빠른 브릿지의 속도와 표준 브릿지의 안전성을 결합합니다.

## 설계 근거

우리의 맞춤형 시퀀싱 레이어는 트랜잭션이 처리, 정렬 및 파생되는 방식을 변경하여 두 가지 옵션을 만들었습니다:

1. Arbitrum Orbit를 포크하고 부정 증명 시스템을 맞춤화(우리의 무포크 원칙을 위반하고, 업데이트를 지연시키며, 모든 롤업 프레임워크에 맞춤 작업 필요)
2. 모든 롤업 프레임워크와 호환되는 일반화된 솔루션 구축(포크 없음, 맞춤형 부정 증명 없음)

우리는 일반화된 솔루션을 선택했습니다.

## 핵심 구성 요소

### TEE (AWS Nitro):

안전한 컴퓨팅 엔클레이브를 제공합니다. (참고: *Arbitrum Nitro가 아닌 AWS Nitro TEE*)

- 역할: 파생을 실행하고, 시퀀싱/앱체인 전환을 검증하며, 서명된 출금 주장을 출력합니다.
- 신뢰: zk 검증된 증명(PCR 측정)을 가진 엔클레이브만 온체인에서 허용 목록에 포함됩니다.
- 온체인 확인: `TeeModule` 는 ECDSA 서명을 검증하고; `TeeKeyManager` 는 zk 검증기에 의해 증명된 키를 수락합니다.
- 개발자 경험: 평소처럼 네이티브 브릿지를 사용하세요; 제안자는 엔클레이브와 상호작용합니다. 업그레이드는 앱 변경 없이 키를 재증명/허용 목록에 추가하는 것만 필요합니다.

## zkVM (Succinct SP1)

- 역할: 전체 문서를 공개하지 않고 AWS Nitro 증명이 유효함을 증명합니다.
- 온체인: `AttestationDocVerifier` 는 SP1 증명(Groth16/Plonk)을 확인하고 예상 루트 인증서, PCR 및 유효 기간을 적용하여 허용 목록에 추가할 `tee_signing_key` 를 반환합니다.
- 공개 출력: 루트 인증서 해시, 유효 기간, PCR0/1/2 및 `tee_signing_key` .
- 개발자 경험: zkVM을 직접 호출할 필요가 없습니다. 증명 제출자가 증명을 생성하고 온체인에서 엔클레이브 키를 등록/교체합니다.

### 향후 업그레이드

- 다양한 클라우드 제공업체 간의 멀티 TEE
- 여러 제공업체가 있는 멀티 zkVM 시스템
- 여러 당사자의 승인 필요
- 시스템 간 불일치 시 이의 제기 메커니즘 포함

## 출금 프로세스 흐름

```

[User/Appchain] -- withdrawal request
|
v
[Proposer] -- validate request + build trusted input
|
v
[TEE Enclave] -- verify state + sign assertion --> [TeeModule on settlement]
|
|-- attestation verified on-chain -----> [TeeKeyManager + zk verifier]
|
[TeeModule] -- finalize -----> posts to rollup (e.g., Base) -> withdrawals enabled

```

1. 요청 수신: 시스템이 출금 요청을 수신
2. 검증 및 승인:
  - 출금 요청의 모든 상태 전환이 유효한지 확인
  - 출금 요청이 재구성 저항성 방식으로 정산 체인에 상태 루트로 커밋되었는지 확인
  - 표준 검사 수행: 사용자 잔액 확인, 상태 유효성, 상태 통화
3. **TEE 서명**: 검증되면 TEE가 내부 키를 사용하여 서명
4. 키 검증: ZKVM 증명이 검증했기 때문에 TEE의 키가 유효한 것으로 간주됨
5. 브릿지 통합: 빠른 출금 메커니즘을 통해 Arbitrum의 네이티브 브릿지에 연결(데이터 가용성 위원회 인터페이스 사용, 하지만 DA 레이어 대신 우리의 출금 시스템이 승인 제공). Arbitrum 브릿지 자체에 대한 수정 없음.

## 주요 이점

- 프레임워크 독립적: 맞춤화 없이 모든 롤업 프레임워크와 작동
- 포크 유지 관리 불필요: 고객이 즉시 Arbitrum Orbit 업데이트를 통합 가능
- 네이티브 브릿지 보안: 자금을 보유하는 커스텀 브릿지가 필요 없음. 자금은 Arbitrum의 브릿지에 보관되며 출금을 가능하게 하는 기존 기능 사용

- 단계적 보안 강화: 분쟁 사례에 대한 이의 제기 메커니즘이 있는 빠른 출금
- 분산 운영: TEE는 핵심 팀뿐만 아니라 누구나 운영 가능
- 다중 제공업체 중복성: 향후 다중 TEE/다중 zkVM 아키텍처로 단일 장애점 방지

## 현재 상태

- 출금 시스템은 현재 개발 중
- 하나의 TEE와 하나의 zkVM 구현으로 시작
- 추가 제공업체는 점진적으로 추가될 예정

“참고: 이것은 개발 중인 새로운 시스템입니다. 아키텍처는 안정적이지만, 세부 사항은 변경될 수 있습니다. 질문이나 피드백이 있으면, **Syndicate** 팀에게 연락하세요.”

### < 실행 프레임워크

우리의 실행 프레임워크가 Arbitrum Orbit으로 시작하는 이유...

### 토큰 브리징 >

신디케이트 앱체인에서 토큰 브리징이 작동하는 방식 알아보기

Q Search

⌘

K

☐ Syndicate Stack

⌵

## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모드

한글

☀

🌙

# 토큰 브리징

신디케이트 앱체인에서 토큰 브리징이 작동하는 방식 알아보기

신디케이트 앱체인은 토큰 브리징을 위해 아비트럼 스탠다드 브릿지를 사용합니다. 이는 아비트럼에서 사용하는 표준 잠금-발행 메커니즘으로, 앱체인 자산의 호환성과 보안을 보장합니다.

## 아비트럼 스탠다드 브릿지 작동 방식

아비트럼 스탠다드 브릿지를 사용하면 별도의 커스텀 브릿지 계약 없이 정산 체인(부모 체인, 예: 이더리움)과 앱체인(자식 체인) 사이에서 ERC-20 토큰을 이동할 수 있습니다. 다음은 높은 수준에서 작동 방식입니다:

1. 정산 체인에 **ERC-20** 토큰을 배포하거나 기존 토큰을 사용합니다.
2. 브릿지 게이트웨이 계약이 정산 체인에서 토큰을 사용할 수 있도록 승인합니다.
3. 브릿지 **UI** 또는 **SDK**를 사용하여 입금을 시작합니다. 브릿지는 정산 체인에서 토큰을 잠그고(에스크로) 앱체인에 메시지를 보냅니다.
4. 앱체인에 해당하는 토큰 계약이 자동으로 생성되고(아직 존재하지 않는 경우), 동등한 양의 토큰이 사용자 주소로 발행됩니다.
5. 출금은 반대로 작동합니다: 앱체인에서 토큰이 소각되고, 챌린지 기간이 지난 후 정산 체인의 사용자 주소로 동등한 양의 토큰이 반환됩니다.

주요 구성 요소:

- 라우터 계약: 브릿지 요청을 올바른 게이트웨이로 라우팅합니다.

- **게이트웨이 계약:** 소스 체인에서 토큰을 에스크로하거나 소각하고 대상 체인에서 발행하거나 반환합니다.
- **자동 토큰 배포:** 토큰이 처음 브릿지될 때 앱체인에 표준 토큰 계약이 자동으로 배포됩니다.

단계별 가이드와 더 자세한 기술 정보는 [아비트럼 스탠다드 브릿지 문서](#)를 참조하세요.

< 자산 출금

TEE + zk 증명을 사용한 빠르고 안전한 앱체인 출금 시스템으로 ...

블록 타임 >

블록 타임에 대해 자세히 알아보기

## GET STARTED

[소개](#)[왜 애플체인인가](#)[신디케이트 스택](#)[시퀀싱 아키텍처](#)[빠른 시작](#)

## GUIDES

[체인 설정](#)[커스텀 가스 토큰 추가하기](#)[시퀀싱 구성하기](#)[체인 사용하기](#)[자산 브릿지](#)[지갑 설정](#)[체인 모니터링](#)

## CORE CONCEPTS

[시퀀싱 모드](#)

# 블록 타임

블록 타임에 대해 자세히 알아보기

## 권장 블록 타임

다음은 일반적인 사용 사례에 따라 블록 시간을 선택하기 위한 일반적인 가이드라인입니다. 참고로, Arbitrum Orbit 체인은 일반적으로 250ms 블록 시간을 사용하는 반면, OP Stack 체인은 일반적으로 2초를 사용합니다.

블록 타임	효과 및 트레이드오프	최적 사용 사례	고려사항
250ms – 500ms	거의 즉각적인 확인, 높은 네트워크 요구사항	대부분의 사용 사례	고성능 하드웨어와 낮은 지연 시간의 네트워크 연결이 필요합니다. 체인 재구성을 모니터링하세요.
500ms – 1s	균형 잡힌 응답성과 안정성	NFT 마켓플레이스, 소셜 플랫폼	속도와 신뢰성의 이상적인 균형. 블록 타임이 사용자 기대치를 충족하는지 철저히 테스트하세요.
1s – 2s	낮은 노드 요구사항, 합리적인 확인 속도	거버넌스, 데이터 아카이빙	덜 강력한 하드웨어에서도 잘 작동합니다. 실시간 요구사항이 없는 비용 중심 배포에 완벽합니다. <i>(Arbitrum 생태계 외부의 많은 롤업에서 일반적인 블록 타임)</i>

## 고려해야 할 요소

블록 시간을 선택할 때 다음 사항을 고려하세요:

애플리케이션 유형, 기하급수적인 블록 크기, 트랜잭션 확인이 필요한지(게이밍 및 거래의 경우) 또는 다른 지연을 허용할 수 있는 지(가

- 애플리케이션 유형: 귀하의 dApp이 빠른 트랜잭션 확인이 필요한지(게임 및 거래와 같은) 또는 더 긴 지연을 처리할 수 있는지(거버넌스 및 기록 보관과 같은)?
- 트랜잭션 볼륨: 높은 처리량의 dApp은 빠른 처리를 위해 짧은 블록 타임이 더 적합하며, 낮은 볼륨의 앱은 더 긴 간격으로도 잘 작동할 수 있습니다.
- 노드 인프라: 귀하의 검증자는 선택한 블록 타임, 특히 최소 250ms에서 처리할 수 있는 적절한 하드웨어와 네트워크 대역폭이 필요합니다.
- 사용자 경험: 더 빠른 블록 타임은 더 반응이 빠른 dApp을 만들지만, 지나치게 짧은 간격은 재구성이나 네트워크 혼잡을 일으킬 수 있습니다.

## 모범 사례

- 보수적으로 시작하기: 확실하지 않다면 1초 블록 타임으로 시작하세요—대부분의 dApp에 좋은 균형을 제공합니다. 성능 테스트를 기반으로 미세 조정할 수 있습니다.
- 철저히 테스트하기: 선택한 블록 타임이 트랜잭션 지연 시간, 노드 성능 및 사용자 경험에 미치는 영향을 평가하기 위해 먼저 테스트넷에 앱체인을 배포하세요.
- 노드 상태 모니터링: 짧은 블록 타임(특히 250ms)은 강력한 인프라를 요구합니다. 검증자가 최적으로 작동하도록 CPU, 메모리 및 네트워크 사용량을 추적하세요.
- 최종성 고려하기: 블록 타임이 확인 속도에 영향을 미치지만, 트랜잭션 최종성은 합의 메커니즘에 따라 달라집니다. 자세한 내용은 Syndicate의 합의 문서를 참조하세요.
- 확장성 계획하기: dApp의 트랜잭션 볼륨은 시간이 지남에 따라 증가할 가능성이 높습니다. 미래 수요를 처리할 수 있는 블록 타임을 선택하거나, 체인 구성을 적절히 최적화할 준비를 하세요.

< 토큰 브리징

신디케이트 앱체인에서 토큰 브리징이 작동하는 방식 알아보기

셀레네 체인 >

셀레네 체인은 신디케이트의 프로그래밍 가능한 온체인 시퀀서의...



Q Search



K

☐ Syndicate Stack



## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

이식성



# 셀레네 체인

셀레네 체인은 신디케이트의 프로그래밍 가능한 온체인 시퀀서의 라이브 데모입니다.

셀레네 체인은 자연 주기, 이 경우에는 달의 위상에 의해 통제되는 프로그래밍 가능한 온체인 시퀀서의 라이브 데모입니다. 트랜잭션 시퀀싱을 달의 주기에 고정함으로써, 셀레네는 수동 개입을 제거하고 자율적이면서도 의도적으로 독특한 시스템을 가능하게 합니다.

신디케이트의 인프라를 기반으로 구축된 셀레네는 시퀀싱 로직이 어떻게 표현력 있고, 시간 기반이며, 완전히 온체인으로 만들어질 수 있는지 보여주며, 권한 부여, 순서 지정 및 맞춤형 트랜잭션 규칙에 대한 새로운 가능성을 열어줍니다.



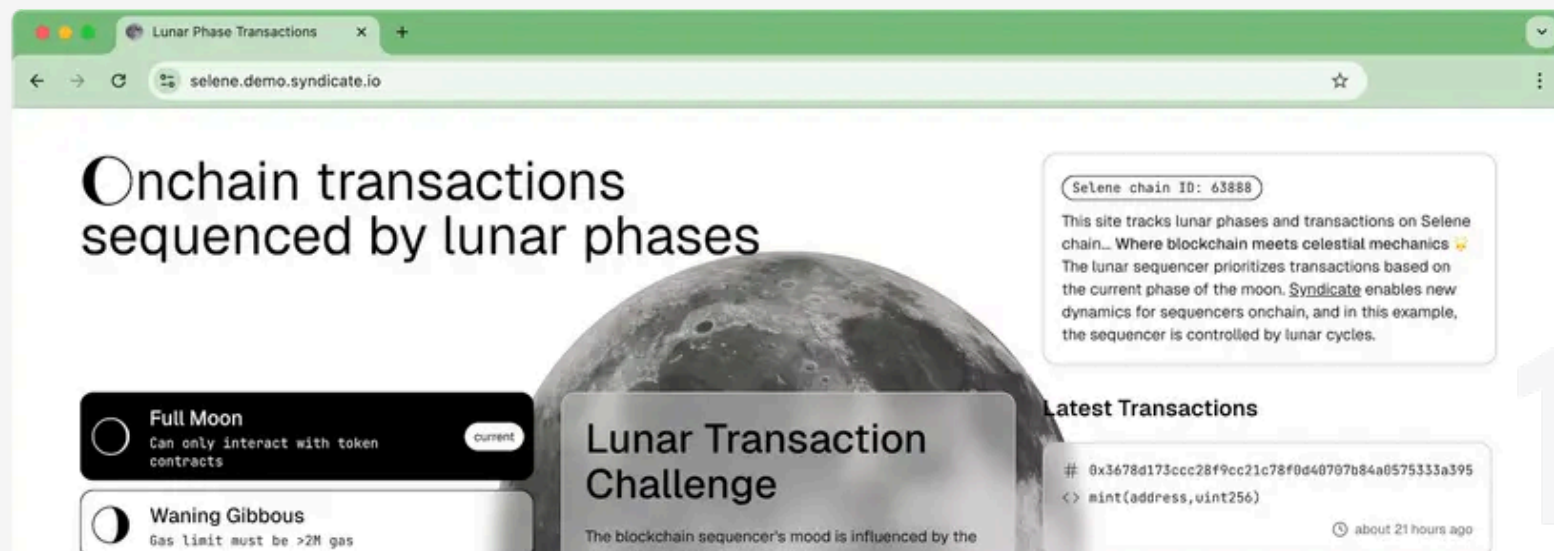
셀레네 체인 데모

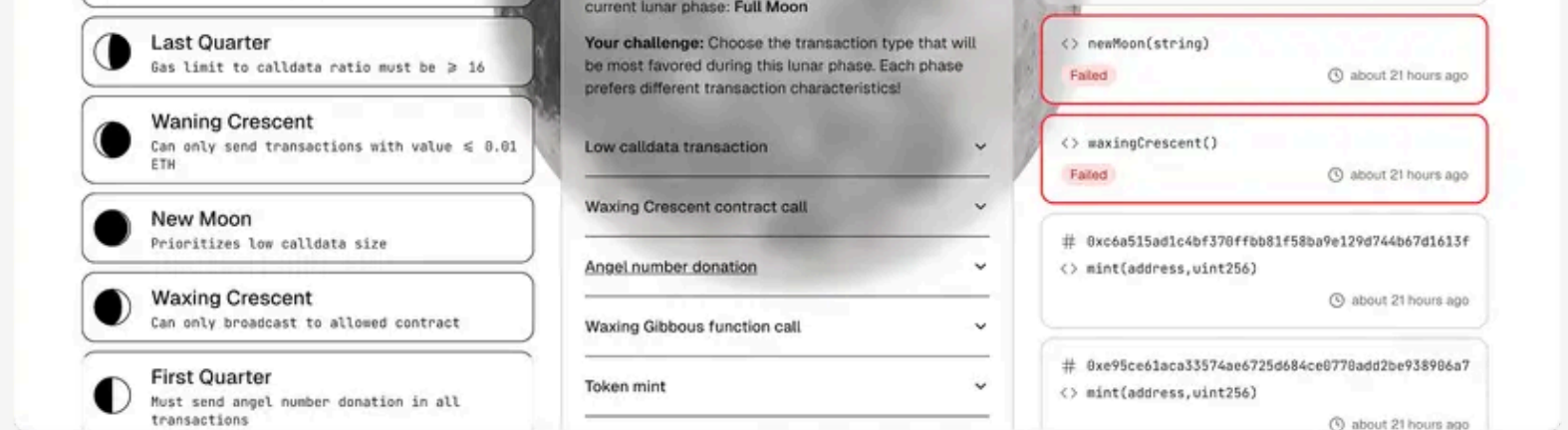
데모 사이트 보기



syndicateprotocol/selene-chain

데모 리포지토리 보기





스크린샷: 셀레네 체인 데모 인터페이스, 달의 위상 기반 트랜잭션 프로토타입 표시.

## 작동 방식: 트랜잭션 라이프사이클

1. 제출: 사용자가 셀레네 데모 인터페이스를 통해 트랜잭션을 제출합니다.
2. 시퀀싱: 온체인 시퀀서 모듈이 현재 달의 위상을 결정하고 해당 규칙을 트랜잭션에 적용합니다.
3. 검증: 트랜잭션이 활성 위상의 기준과 일치하면 시퀀싱되어 온체인에 포함됩니다. 그렇지 않으면 거부되고 기록됩니다.
4. 피드백: 사용자는 성공 시 트랜잭션 해시 또는 거부 메시지와 같은 즉각적인 피드백을 받습니다. 모든 거부된 시도는 투명성과 분석을 위해 PostgreSQL 데이터베이스에 저장됩니다.

## 시퀀싱 모듈: 달의 위상 권한

신디케이트의 프로그래밍 가능한 시퀀서에서 모듈은 트랜잭션이 어떻게 선택되고, 정렬되며, 온체인에 포함되는지 정의하는 플러그인 구성 요소입니다. 셀레네 데모는 맞춤형 권한 중심 시퀀싱 모듈을 구현합니다. 모든 트랜잭션 포함 규칙은 시퀀서에 의해 온체인에서 정의되고 시행됩니다. 각 달의 위상은 고유한 권한 부여 규칙 세트를 활성화합니다. 예를 들어, 그믐달 기간에는 저가치 트랜잭션만 허용됩니다.

이웃한 게릴라가 Syndicate를 지칭하여 트로이를 제공해서 직접 사용되는 공격이고 관계 기반 트랜잭션 공격을 뜻한 표현이었고 자율적인 인프라를 구축하는 방법을 보여줍니다.

## 달의 위상 계산 및 규칙

이 모듈은 타임스탬프에서 현재 달의 위상을 계산하여 8개의 규칙 세트 중 하나에 매핑합니다.

공식:

$$\text{Elapsed Time} = \text{timestamp} - 947182440$$

$$\text{Phase Time} = \text{Elapsed Time} \bmod 2551443$$

$$\text{Phase Index} = \left\lfloor \frac{\text{Phase Time} \times 30}{2551443} \right\rfloor$$

📄 MoonphasePermissionModule.sol

```
/// @notice Computes the phase index for a given timestamp
/// @param timestamp The timestamp to compute the phase index for
/// @return phaseIndex The phase index
function getPhaseIndex(uint256 timestamp) internal pure returns (uint8 phaseIndex) {
    // Known new moon: Jan 6, 2000 at 18:14 UTC
    uint256 synodicMonth = 2551443; // seconds in a synodic month (~29.5306 days)
    uint256 newMoonReference = 947182440; // reference new moon timestamp
    uint256 elapsed = timestamp - newMoonReference;
    uint256 phaseTime = elapsed % synodicMonth;

    // Calculate the phase from 0 to 29
    return uint8((phaseTime * 30) / synodicMonth);
}
```

그런 다음 `phaseIndex`를 `getPhaseName` 함수에 전달하여 달의 위상 이름을 반환합니다:

📄 MoonphasePermissionModule.sol

```
function getPhaseName(uint8 phase) internal pure returns (string memory name) {
    if (phase == 0 || phase == 29) return "New Moon";
```

```

if (phase == 0 || phase == 29) return "New Moon";
if (phase >= 1 && phase <= 6) return "Waxing Crescent";
if (phase == 7) return "First Quarter";
if (phase >= 8 && phase <= 13) return "Waxing Gibbous";
if (phase >= 14 && phase <= 15) return "Full Moon";
if (phase >= 16 && phase <= 21) return "Waning Gibbous";
if (phase == 22) return "Last Quarter";
if (phase >= 23 && phase <= 28) return "Waning Crescent";
}

```

## 규칙 표:

달의 위상	규칙 설명
신월(New Moon)	낮은 콜데이터 크기 선호
초승달(Waxing Crescent)	허용된 계약에만 브로드캐스트
상현달(First Quarter)	천사 숫자 기부 필요(예: 111, 333 등)
차가는 만월(Waxing Gibbous)	<code>waxingGibbous()</code> 에 대한 함수 호출만 허용
만월(Full Moon)	토큰 상호작용만 허용(ERC20, ERC721, ERC1155)
기우는 만월(Waning Gibbous)	높은 가스 한도 필요( $\geq 21k$ )
하현달(Last Quarter)	콜데이터/가스 한도 비율 $\geq 16$ 강제
그믐달(Waning Crescent)	낮은 가치 트랜잭션만 허용( $\leq 0.01$ ETH)

이러한 각 규칙은 `MoonphasePermissionModule.sol` 에서 구현되고 시퀀싱 계층에서 시행됩니다.

## 데모 아키텍처 및 기술 스택

**\*\*셀레네 체인(Selene Chain)\*\***은 달의 위상 로직이 온체인에서 어떻게 시행되고 현대적인 웹 애플리케이션에서 표현될 수 있는지 보여주는 모듈식, 확장 가능한 아키텍처로 구축되었습니다. 다음은 각 요소가 어떻게 연결되는지 보여줍니다:

- **시퀀서:** 탈중앙화된 온체인 시퀀싱
- **시퀀싱 모듈:** 모든 달의 위상 로직은 Solidity 모듈(`MoonphasePermissionModule.sol`)을 통해 온체인에서 시행되며, 현재 위상에 따라 트랜잭션 유효성을 결정합니다.
- **프론트엔드:** **Next.js, TypeScript, Tailwind CSS**로 구축되어 사용자에게 대화형 프롬프트와 실시간 피드백을 제공합니다.
- **백엔드:** 트랜잭션 로깅에는 **PostgreSQL**을, 데이터베이스 접근에는 **Drizzle ORM**을 사용합니다.
- **개발 및 테스트:** 계약 개발 및 테스트에는 **Foundry**가 사용됩니다.

## 시작하기

Syndicate로 앱체인을 구축할 준비가 되셨나요?

1. [라이브 데모 시도하기](#)
2. [코드베이스 탐색하기](#)로 아키텍처를 이해하고 모듈을 확장하세요
3. [온체인 시퀀싱 모듈 보기](#)로 자신만의 앱체인을 위한 주문, 권한 및 경제적 메커니즘 커스터마이징을 시작하세요
4. 지원이 필요하거나 실험 결과를 공유하려면 [Syndicate 팀에 연락하세요!](#)

< 블록 타임

블록 타임에 대해 자세히 알아보기

자주 묻는 질문 >

Syndicate 앱체인, 시퀀싱, 프레임워크 및 데이터 가용성에 관한...



## GET STARTED

소개

왜 앱체인인가

신디케이트 스택

시퀀싱 아키텍처

빠른 시작

## GUIDES

체인 설정

커스텀 가스 토큰 추가하기

시퀀싱 구성하기

체인 사용하기

자산 브릿지

지갑 설정

체인 모니터링

## CORE CONCEPTS

시퀀싱 모드



## 자주 묻는 질문

Syndicate 앱체인, 시퀀싱, 프레임워크 및 데이터 가용성에 관한 일반적인 질문에 대한 답변.

### RaaS(Rollup-as-a-Service)인가요?

아니요, 저희는 RaaS가 아니지만 RaaS 제공업체와 협력합니다. 저희는 다음과 같은 방법으로 의존성이 적은 체인을 쉽게 구축할 수 있는 기능을 제공합니다:

- 중앙화된 시퀀서를 탈중앙화된 온체인 시퀀서로 대체
- 중앙화된 백업을 온체인 데이터 저장소로 대체

저희가 RaaS가 필요하지 않은 일부 서비스(시퀀서, RPC 노드, 블록 탐색기 등)를 직접 호스팅하지만, 이는 개발자 사용만을 위한 것입니다. 앱이 라이브 준비가 되면 RaaS 제공업체를 강력히 권장합니다. 온체인 시퀀싱은 이 과정을 단순화하고 더 간단하고 신뢰할 수 있는 스택으로 RaaS 제공업체로 이동하거나 제공업체 간 이동을 쉽게 만들어 줍니다.

### 롤업 프레임워크인가요?

아니요, 저희는 롤업 프레임워크가 아닙니다. 저희는 기존 롤업 프레임워크와 통합하여 프로그래밍 가능한 시퀀싱과 장기 데이터 저장을 제공합니다. 저희가 Arbitrum Orbit부터 시작하는 이유는 실행 프레임워크 페이지에서 확인하세요.

### DA(Data Availability) 레이어인가요?

아니요, 저희는 데이터 가용성 레이어가 아닙니다. 저희는 내부적으로 데이터 가용성 레이어(EigenDA와 같은)를 사용합니다.

Syndicate Chain 위에서 온체인 시퀀싱을 사용하면 데이터가 거기에 저장되므로 별도의 데이터 가용성 레이어가 필요하지 않습니다. 기본적으로 저희가 내부적으로 하나를 사용하므로 사용자는 그럴 필요가 없습니다.

## 공유 시퀀서인가요? 공유 시퀀싱 또는 기반 시퀀싱과 어떻게 다른가요?

자세한 설명은 [시퀀싱 아키텍처 비교](#)를 참조하세요.

## 어떤 프레임워크가 애플체인을 지원하나요?

현재 EVM 블록체인을 위한 Arbitrum Orbit를 지원합니다. [신디케이트 스택](#)에서 자세히 알아보세요. 곧 non-EVM 체인을 포함한 더 많은 스택이 추가될 수 있습니다.

## 어떤 가스 토큰을 사용할 수 있나요?

ETH 또는 커스텀 ERC-20 가스 토큰을 사용할 수 있습니다. 설정 비용이 훨씬 저렴한 커스텀 가스 토큰을 권장합니다.

## 어떤 정산 레이어를 사용할 수 있나요?

모든 정산 레이어를 사용할 수 있습니다. 가장 일반적인 것은 이더리움 메인넷과 베이스입니다.

## 서비스를 사용하기 위해 특별히 해야 할 일이 있나요?

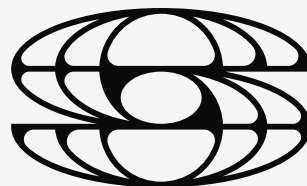
아니요. 기존의 모든 도구와 지갑은 수정 없이 작동해야 합니다.

## 롤업 프레임워크를 위해 코드를 포크하나요?

기능적으로는 아닙니다. 우리는 Arbitrum Orbit 스택을 통해 Arbitrum Nitro를 실행하며, 몇 가지 추가 읽기 전용 기능만 노출하는 최소한의 추가적인 포크를 사용합니다. 실행 로직이나 합의를 변경하지 않습니다. Arbitrum 스마트 계약에서는 작은 수정만 합니다 (예: 사소한 성능 조정 및 Orbit을 위한 추가 온체인 로깅). 이러한 변경 사항은 몇 줄에 불과하며 업스트림과 호환되므로 업스트림 업데이트를 받는 능력에 영향을 미치지 않아야 합니다.







# 자체 앱체인으로 구축, 배포 및 확장하세요

확장 가능하고 프로그래밍 가능하며 원자적으로 구성 가능한 앱체인을 위한 인프라. 네트워크, 시퀀서 및 경제에 대한 완전한 제어권을 제공합니다.

## 신디케이트란 무엇인가요?

신디케이트는 프로그래밍 가능한 온체인 시퀀서로 앱 체인을 구축하고 확장하기 위한 인프라입니다.

[Read more](#)

## 왜 앱체인인가요?

앱체인은 특정 애플리케이션이나 생태계 및 그 요구사항에 맞게 커스터마이징된 블록체인입니다.

[Read more](#)

## 신디케이트 스택

신디케이트는 프로그래밍 가능한 온체인 시퀀서로 앱 체인을 쉽게 구축하고 확장할 수 있게 해줍니다.

[Read more](#)

Q Search



K

Syndicate Chains



Commons Chain

Syndicate Chain

# Commons Chain

신디케이트 네트워크의 스테이킹 허브 역할을 하는 대표 애플체인

## Commons Chain이란 무엇인가요

Commons Chain은 신디케이트 네트워크의 대표 애플체인으로, 네트워크 기술의 시연과 커뮤니티 참여를 위한 허브 역할을 합니다. Arbitrum Orbit 스택을 사용하여 구축되고 Base에 정산되는 Commons Chain은 애플체인이 커스텀 가스 토큰과 프로그래밍 가능한 시퀀싱을 활용하면서 필수적인 네트워크 기능을 제공하는 방법을 보여줍니다.

SYND 스테이킹 및 배출 분배를 위한 주요 장소로서, Commons Chain은 네트워크의 경제 모델에서 중요한 역할을 합니다. 초기 예치부터 배출 청구까지 모든 스테이킹 작업은 Commons Chain의 전용 인프라에서 이루어집니다. SYND를 네이티브 가스 토큰으로 사용함으로써 사용자가 거래 수수료를 위해 ETH나 다른 토큰을 보유할 필요가 없어져 사용자 경험이 간소화되고 SYND 토큰에 추가적인 유틸리티가 생성됩니다.

## 연결 세부 정보

Commons Chain은 Base에 정산되는 레이어 3 블록체인으로 운영되며, 표준 이더리움 호환 도구와 지갑을 통해 접근할 수 있습니다.

## 네트워크 구성

매개변수	값
네트워크 이름	Syndicate Commons

네트워크 이름	Syndicate Commons
체인 ID	510003
스택	Arbitrum Orbit
정산 체인	Base (8453)
통화 기호	SYND
통화 소수점	18
RPC URL	<a href="https://commons.rpc.syndicate.io">https://commons.rpc.syndicate.io</a>
블록 탐색기	<a href="https://commons.explorer.syndicate.io">https://commons.explorer.syndicate.io</a>

프로덕션 애플리케이션은 안정적인 성능을 보장하기 위해 전용 RPC 엔드포인트 사용을 고려해야 합니다. 공개 RPC 엔드포인트는 모든 사용자에게 공정한 접근을 보장하기 위해 속도 제한을 구현합니다.

## 핵심 계약

다음 계약들은 Commons Chain의 필수 기능을 제공합니다:

### 시스템 컨트랙트

컨트랙트	주소	설명
SYND 토큰	<a href="#">0x11dC28D01984079b7efE7763b533e6ed9E3722B9</a>	네이티브 SYND 토큰 구현
롤업 컨트랙트	<a href="#">0xf12832C23a94db0E966C4776DCce50ADe0697A32</a>	핵심 롤업 상태 관리
어드민 프록시	<a href="#">0x7b0ab8F15940F573C1D82C2238CFa8368EdBf9c6</a>	관리 작업 및 업그레이드

업그레이드 실행자	<u>0xC924178de522feB6147df2668E44fe7683852cEE</u>	컨트랙트 업그레이드 실행
검증자 지갑 생성기	<u>0x1a7F2Ef9Ed9A084D8b07d1601F9545e453E4e19C</u>	검증자 지갑 생성 및 관리
인박스	<u>0xAE824E2d20F21B222932aFC6079cDaA1EB5b2F00</u>	L2에서 메시지 수신
아웃박스	<u>0xdBFF2d1df0E30F6e5dDa325C29EC0bC8944ebA7b</u>	L2로 메시지 전송
시퀀서 인박스	<u>0x916eb8bbA622b547341843fBcF6D5Fa71C6B950E</u>	시퀀서 작업 관리
롤업 이벤트 인박스	<u>0xB817645bb7E4580B7DeF79760932DbDAF06d434E</u>	롤업 이벤트 처리
챌린지 매니저	<u>0x273D2eF882A097f4408B02F64745BcB34eE52F7d</u>	부정 증명 및 챌린지 관리

## 브릿지 인프라

컨트랙트	주소	설명
L2 브릿지	<u>0x64d426e64B7191a1fCf48C89d2D12772FbFB297f</u>	Base에서 자산 수신
L2 라우터	<u>0x2e5B0Df3F931caeF3e330c15819008472D47b6f0</u>	L2에서 브릿지 작업 라우팅
L3 라우터	<u>0xd5889D7cD6735359676B5Eb5540bB323E156F74c</u>	L3에서 브릿지 작업 라우팅
표준 게이트웨이 (L2)	<u>0xbb03Ac04c9Dd48a8F93f4a843762e4d9BDfd5093</u>	L2에서 표준 토큰 브릿징
표준 게이트웨이 (L3)	<u>0x272d785336CdE55a1C4E9443Acc2f63c7A8157Ec</u>	L3로 표준 토큰 브릿징
커스텀 게이트웨이 (L2)	<u>0x2dFf0C44AeDeDAbc3285c6590D24c96D077C733A</u>	L2에서 커스텀 토큰 브릿징
커스텀 게이트웨이 (L3)	<u>0x15961fd104438a6aB998Bd90187593A0A75FAdD2</u>	L3로 커스텀 토큰 브릿징





Q Search



K

Syndicate Chains



Commons Chain

Syndicate Chain

# Syndicate Chain

신디케이트 생태계를 위한 전용 블록체인

## Syndicate Chain이란 무엇인가요

Syndicate Chain은 앱체인 전체 생태계를 위한 온체인 시퀀싱을 가능하게 하는 전용 블록체인입니다. 트랜잭션 순서 지정에 대한 규칙과 권한을 정의하는 시퀀싱 스마트 계약을 호스팅하여 트랜잭션 흐름을 관리하는 기본 규칙을 설정하는 동시에 각 앱체인이 경제적 주권을 유지할 수 있도록 합니다.

## 연결 세부 정보

Syndicate Chain은 이더리움 L1에 정산되는 레이어 2 블록체인으로 운영되며, 표준 이더리움 호환 도구와 지갑을 통해 접근할 수 있습니다.

## 네트워크 구성

매개변수	값
네트워크 이름	Syndicate Chain
체인 ID	510
스택	Arbitrum Orbit
정산 체인	이더리움 L1 (Ethereum L1)

생산 체인	이더리움 L1 (1)
통화 기호	SYND
통화 소수점	18
RPC URL	<a href="https://rpc.syndicate.io">https://rpc.syndicate.io</a>
블록 탐색기	<a href="https://explorer.syndicate.io">https://explorer.syndicate.io</a>

프로덕션 애플리케이션은 안정적인 성능을 보장하기 위해 전용 RPC 엔드포인트 사용을 고려해야 합니다. 공용 RPC 엔드포인트는 모든 사용자에게 공정한 접근을 보장하기 위해 속도 제한을 구현하고 있습니다.

## 핵심 계약

다음 계약들은 Syndicate Chain의 필수 기능을 제공합니다:

## 시스템 계약

계약	주소	설명
SYND 토큰	<a href="#">0x1bAB804803159aD84b8854581AA53AC72455614E</a>	네이티브 SYND 토큰 구현
롤업 계약	<a href="#">0x451bD7813909B899DA6EbEC55E8fF823c057e14A</a>	핵심 롤업 상태 관리
관리자 프록시	<a href="#">0x481D290473e4f6929AA45CFb7Ef7c7847aBeD007</a>	관리 작업 및 업그레이드
업그레이드 실행자	<a href="#">0xFA4d1D308f4B4f6E6F836Db2B77Db549606A460c</a>	계약 업그레이드 실행
검증자 지갑 생성기	<a href="#">0x87f3f4620B4e8Dd170c7cCe84d1F8bEAD9f79b92</a>	검증자 지갑 생성 및 관리
인박스	<a href="#">0x5EA55Fd41D42Eb307D281bdE78E4e7572A35ea13</a>	L2에서 메시지 수신

아웃박스	<u>0xf555Bc86D1C953414F676479Bf7C979b1A737E8C</u>	L2로 메시지 전송
시퀀서 인박스	<u>0x12ad349e5d72B582856290736e0f13FE5fA57Aa4</u>	시퀀서 작업 관리
롤업 이벤트 인박스	<u>0x82E761873714cDe47C594aA6F23E6b1844CD98dB</u>	롤업 이벤트 처리
챌린지 매니저	<u>0xABf2988264170a7f94E6Fa76ECA5965B906E229d</u>	부정 증명 및 챌린지 관리

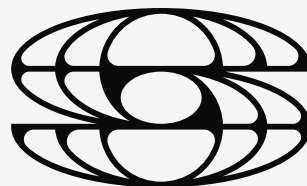
## 브릿지 인프라스트럭처

컨트랙트	주소	설명
L1 브릿지	<u>0x3C8cF0ae6E89AC0796f29B3a58e7dEa1cD072277</u>	이더리움 L1에서 자산을 수신
L1 라우터	<u>0x534Eb1F79C8df3aB1E507e408EeF4e99D53A1239</u>	L1에서 브릿지 작업을 라우팅
L2 라우터	<u>0xD96323e3ce7556Dc5119653DCe1BB73F22562F6C</u>	L2에서 브릿지 작업을 라우팅
표준 게이트웨이(L1)	<u>0x6CA109706c6EBE5379c45f20B3311441D50cb711</u>	L1에서 표준 토큰 브릿징
표준 게이트웨이(L2)	<u>0xBE9eF55Cb610d13324341654bd623B0B5EC11976</u>	L2로 표준 토큰 브릿징
커스텀 게이트웨이(L1)	<u>0x85C88ddc2f525459E243AB27D6059f55Ce80EB0c</u>	L1에서 커스텀 토큰 브릿징
커스텀 게이트웨이(L2)	<u>0xdb0a2E87eE6bAe0fa6Ce6f213B6B6664174a4416</u>	L2로 커스텀 토큰 브릿징

### < Commons Chain

신디케이트 네트워크의 스테이킹 허브 역할을 하는 대표 앱체인





# 자체 앱체인으로 구축, 배포 및 확장하세요

확장 가능하고 프로그래밍 가능하며 원자적으로 구성 가능한 앱체인을 위한 인프라. 네트워크, 시퀀서 및 경제에 대한 완전한 제어권을 제공합니다.

## 신디케이트란 무엇인가요?

신디케이트는 프로그래밍 가능한 온체인 시퀀서로 앱 체인을 구축하고 확장하기 위한 인프라입니다.

[Read more](#)

## 왜 앱체인인가요?

앱체인은 특정 애플리케이션이나 생태계 및 그 요구사항에 맞게 커스터마이징된 블록체인입니다.

[Read more](#)

## 신디케이트 스택

신디케이트는 프로그래밍 가능한 온체인 시퀀서로 앱 체인을 쉽게 구축하고 확장할 수 있게 해줍니다.

[Read more](#)

Q Search



K

🔍 SYND and emissions



기본 정보

토큰 이코노믹스

스테이킹 &amp; 배출

브리징

계약

라이트페이퍼

MiCAR 백서

## 기본 정보

SYND 토큰의 개요 - 목적, 사양, 배포 및 현재 상태.

## 토큰 세부 정보

SYND 토큰은 신디케이트 네트워크의 기본 디지털 자산으로, 애플체인을 구동하고, 거래를 촉진하며, 네트워크를 보호하고, 생태계 전반의 이해관계자들을 연결하도록 설계되었습니다.

## 핵심 정보

SYND는 총 10억 개의 고정 공급량을 가진 ERC-20 토큰으로 배포됩니다. 이 토큰은 신디케이트 네트워크와 커먼스 체인의 가스 토큰 역할을 하며, 모든 애플체인 시퀀싱 트랜잭션과 네트워크 운영을 가능하게 합니다. 가스 토큰과 스테이킹 자산으로서의 유틸리티를 통해 SYND는 애플체인 개발자, 사용자 및 더 넓은 신디케이트 생태계 간의 경제적 연계를 형성합니다.

## 토큰 사양

- 심볼: SYND
- 소수점: 18
- 토큰 표준: ERC-20
- 총 공급량: 1,000,000,000 SYND (고정)
  - 920,000,000은 제네시스에서 발행

- 80,000,000은 30일 간격의 48개 에포크에 걸쳐 배출량으로 발행 예정

[토큰 이코노믹스 >](#)

SYND 토큰 분배, 발행 일정 및 공급 역학

Q Search

⌘

K

🔍 SYND and emissions

↕

기본 정보

토큰 이코노믹스

스테이킹 &amp; 배출

브리징

계약

라이트페이퍼

MiCAR 백서

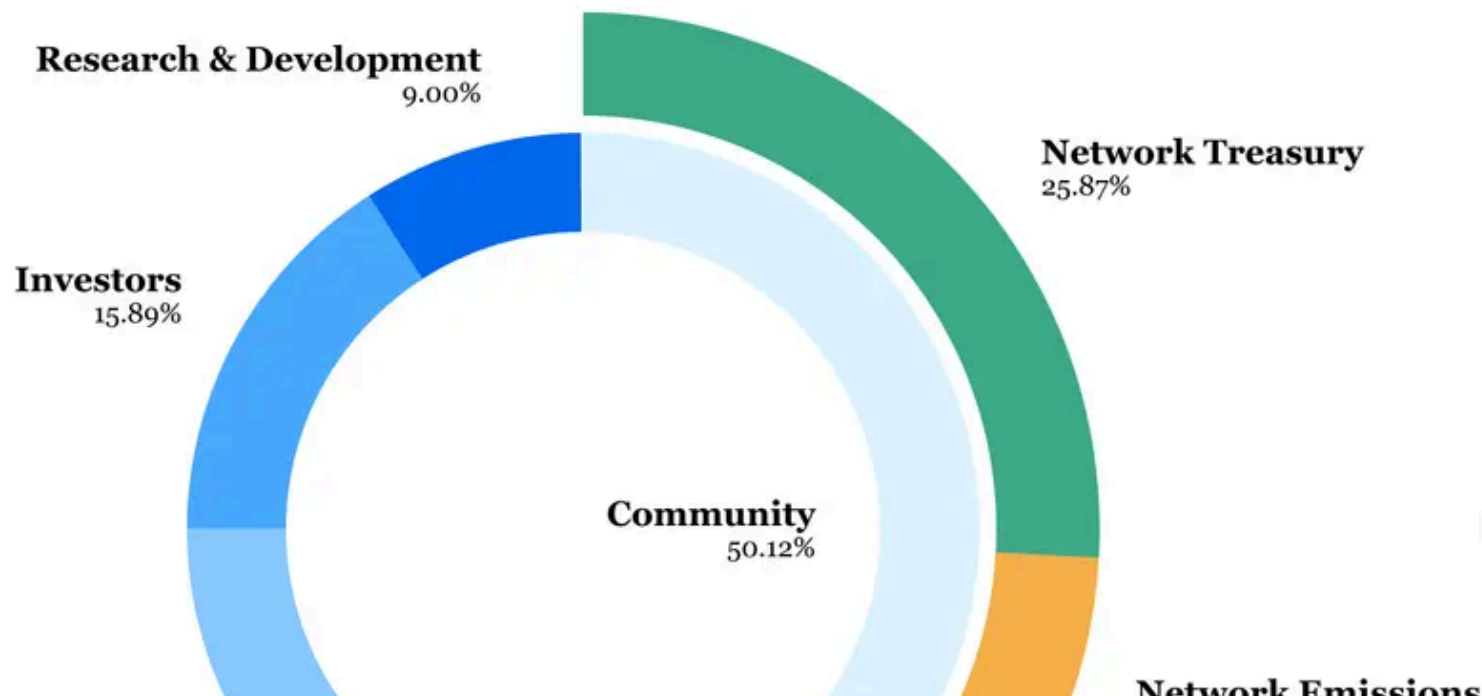
## 토큰 이코노믹스

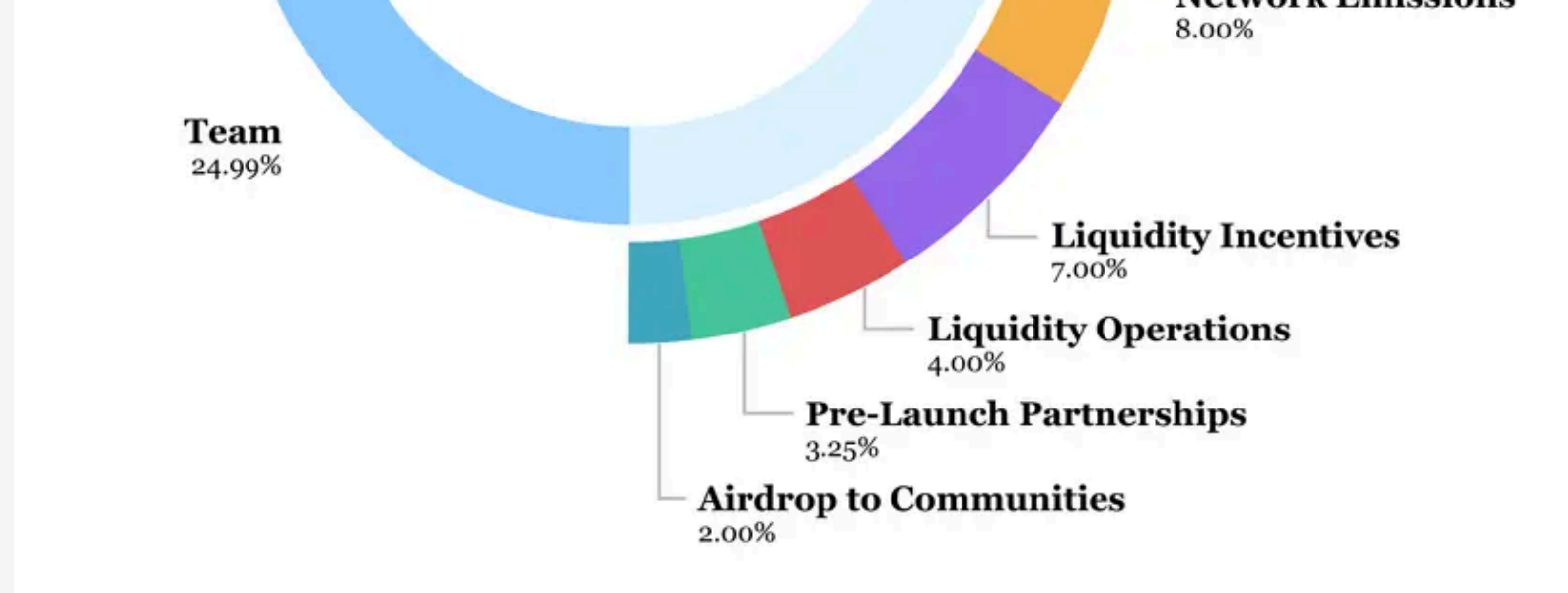
SYND 토큰 분배, 발행 일정 및 공급 역학

### 토큰 분배

SYND는 10억 개의 고정된 총 공급량을 가지고 있습니다. 분배는 장기적인 개발과 성장을 지원하면서 커뮤니티 소유권(공급량의 50.12%)을 우선시합니다.

### 할당 내역



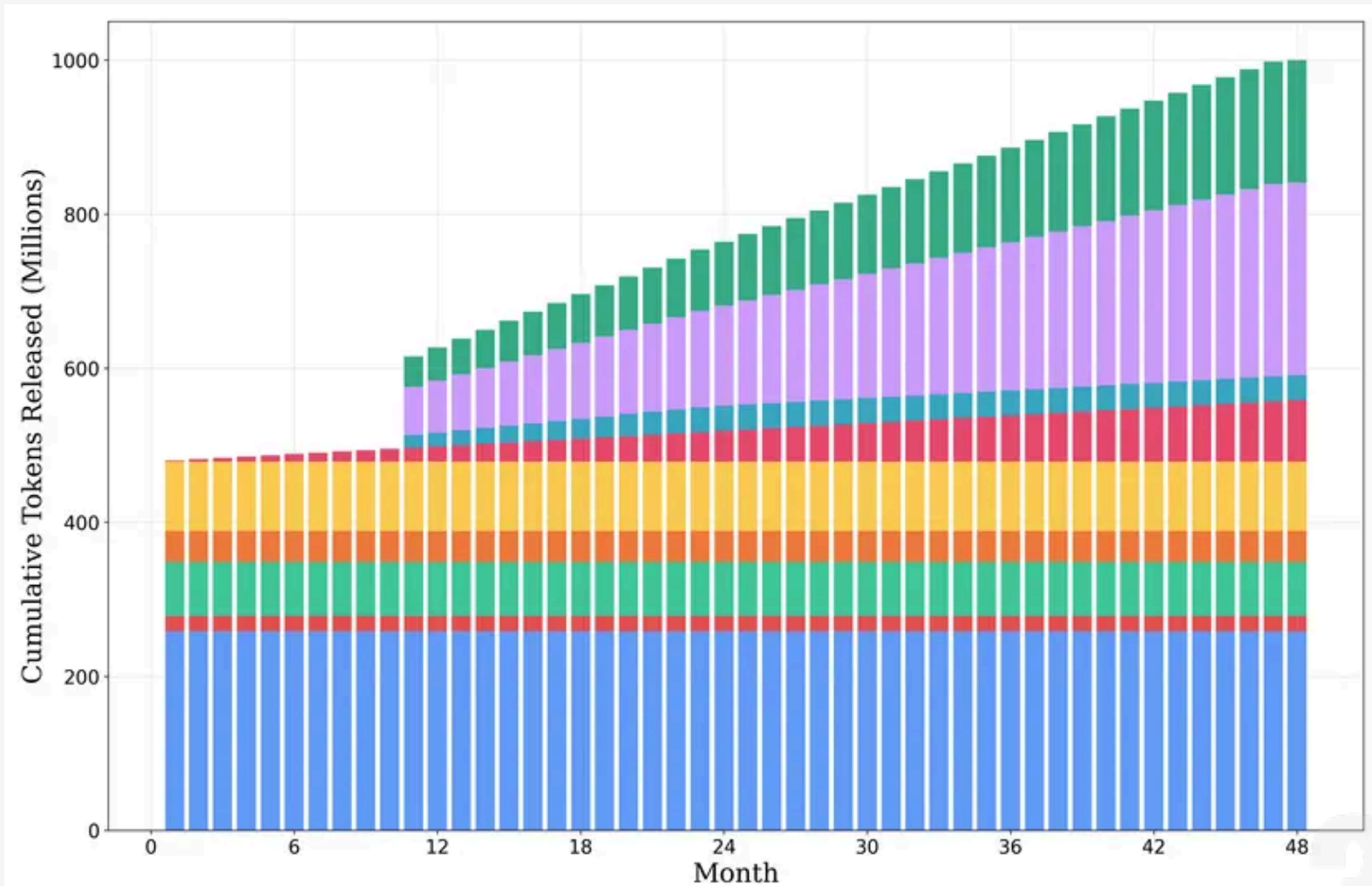


카테고리	토큰 수	공급량 대비 %	비고
재무부	258,700,000	25.87%	거버넌스를 통해 SYND 보유자가 제어
팀	249,900,000	24.99%	1년 유예 기간이 있는 4년 베스팅
투자자	158,900,000	15.89%	1년 유예 기간이 있는 4년 베스팅
R&D	90,000,000	9.00%	지속적인 네트워크 개발
네트워크 배출	80,000,000	8.00%	48개 에포크에 걸쳐 분배
유동성 인센티브	70,000,000	7.00%	초기 유동성 부트스트래핑
유동성 운영	40,000,000	4.00%	지속적인 유동성 관리
출시 전 파트너십	32,500,000	3.25%	마일스톤 기반 베스팅
에어드롭	20,000,000	2.00%	2025년 8월 15일 배포

가장 큰 단일 할당은 재무부(25.87%)로, 온체인 거버넌스를 통해 SYND 보유자가 제어합니다. 자세한 내용은 [라이트페이퍼](#)를 참조하세요.

## 토큰 출시 일정

다음 차트는 4년 발행 기간 동안 모든 할당 카테고리에 걸친 누적 토큰 출시 일정을 보여줍니다.



Treasury	Liquidity operations	Pre-launch partnerships
Airdrop	Research & Development	Team
Liquidity incentives	Network emissions	Investors

## 발행 일정

네트워크 배출은 48개의 30일 에포크에 걸쳐 80,000,000 SYND(총 공급량의 8%)를 분배합니다. 네트워크는 현재 예측 가능한 토큰 분배를 보장하기 위해 에포크당 동일한 배출량으로 운영되지만, 배출 공식은 향후 거버넌스 조정을 위한 유연한 감소 요소를 지원합니다.

## 감소 공식

각 에포크의 발행량은 구간별 기하급수적 감소를 사용하여 계산됩니다:

에포크 **0-46**에 대해: 배출량 = 남은 토큰 × (1 - 감소\_계수) / (1 - 미래\_감소\_곱)

에포크 **47(마지막)**에 대해: 배출량 = 모든 남은 토큰

**decay\_factor = 1.0**(현재 구성)인 경우: 배출량 = 80,000,000 / 48 = 1,666,667.67 토큰/에포크

여기서:

- **Remaining:** 에포크 시작 시 할당되지 않은 배출 토큰
- **decay\_factor:** 현재 동일 분배를 위해 1.0(거버넌스에 의해 조정 가능)
- **future\_decay\_product:** 모든 미래 감소 요소의 곱

이 공식은 향후 감소 요소 조정에 관계없이 정확히 8천만 토큰이 분배되도록 보장합니다.

## 현재 48-에포크 일정

현재 1.0 감소 요소의 경우:

현재 1.0 감소 요소의 경우.

- 모든 에포크: 1,666,667.67 SYND(총 공급량의 0.167%)
- 총 배출량: 80,000,000 SYND(마지막 에포크는 반올림 조정됨)

거버넌스를 통해 사용 가능한 대체 구성:

- 감소 요소 **0.98**: 초기 집중형(2,577,259 → 997,205 SYND/에포크)
- 감소 요소 **1.02**: 후기 집중형(997,205 → 2,577,259 SYND/에포크)

▶ 전체 배출 일정 보기(현재 구성)

#### < 기본 정보

SYND 토큰의 개요 - 목적, 사양, 배포 및 현재 상태.

#### 스테이킹 & 배출 >

SYND 스테이킹 작동 방식, 배출 분배 및 보상 계산



Q Search



K

SYND and emissions



기본 정보

토큰 이코노믹스

스테이킹 &amp; 배출

브리징


계약

라이트페이퍼

MiCAR 백서

## 스테이킹 & 배출

SYND 스테이킹 작동 방식, 배출 분배 및 보상 계산

“ 신디케이트 네트워크 스테이킹은 현재 알파 에포크—시그널 스테이킹 단계입니다

배출은 현재 활성화되지 않았으며 **2025년 9월 30일 화요일 23:00:00 GMT+0000**에 에포크 0부터 시작됩니다. 참고: 출금은 현재 에포크가 종료된 후 처리됩니다. 지금 스테이킹된 **SYND**는 에포크 0이 시작될 때까지 출금할 수 없습니다.

커먼스 체인 스테이킹 및 배출은 단계적으로 진행됩니다. 예상 일정:

현재 진행 중 - 알파 에포크(시그널 스테이킹 기간)

- **SYND** 보유자는 커먼스 체인에서 스테이킹을 시작할 수 있습니다.
- 이 설정 기간 동안에는 아직 보상이 분배되지 않습니다.
- 사용자는 아직 특정 앱체인을 향해 스테이킹할 수 없습니다.

**10월 1일 — 에포크 0 시작**

- 첫 번째 공식 에포크가 시작되며, 네트워크 배출의 **100%**가 기본 풀로 전달됩니다.
- 이는 앱체인 스테이킹이 가능해지기 전에 스테이킹 시스템을 부트스트랩합니다.

**10월 30일 — 에포크 0 종료**

- 스테이커의 에포크 0 배출 지분은 스테이크 크기와 기간에 따라 계산됩니다.

### 10월 31일 — 에포크 1 시작(앱체인 스테이킹 활성화)

- 에포크 0의 보상은 에포크 1이 시작된 직후 청구할 수 있습니다.
- 앱체인 스테이킹은 이 에포크에 출시될 예정입니다. 앱체인 스테이킹은 10월 중순에 활성화될 것으로 예상됩니다. 활성화되면 스테이커는 에포크가 시작되기 전에 특정 앱체인을 선택하여 배출을 해당 앱체인으로 지정할 수 있습니다.

### 11월 29일 — 에포크 종료

- 배출이 완전한 3폴 모델로 확장됩니다.
  - 30% 기본 풀(네트워크 기반)
  - 30% 성능 풀(네트워크 효율성과 연계)
  - 40% 앱체인 풀(개별 앱체인의 성장 지원)
- 스테이커는 기본 + 성능 풀에서 배출을 청구할 수 있으며, 앱체인은 자신에게 지정된 스테이크와 앱체인 활동에 기반하여 자신의 몫을 청구합니다.

### 11월 30일 — 에포크 2 시작

- 스테이킹이 완전히 장기 설계로 전환됩니다. 모든 스테이크가 앱체인 네트워크를 지원하고, 모든 배출이 신디케이트에서 구축하는 커뮤니티로 다시 흐르는 시스템입니다.

## 개요

SYND 스테이킹 시스템은 토큰 보유자가 네트워크 배출량을 앱체인으로 유도하면서 보상을 얻을 수 있게 합니다. 이 프로토콜은 30일 단위의 에포크로 운영되며, 48개의 에포크(약 4년) 동안 세 가지 풀 배출 시스템을 통해 8천만 SYND 토큰을 분배합니다.

## 작동 방식

1. SYND 보유자는 토큰을 커먼스 체인으로 브릿지하고 [commons.syndicate.io](https://commons.syndicate.io)에서 호스팅되는 UI를 통해 스테이킹 계약에 예치합니다
2. 스테이커는 전체 에포크 기간 동안 자신의 스테이크를 등록된 앱체인으로 유도합니다
3. 배출량은 스테이크 금액, 기간 및 앱체인 성능에 따라 세 개의 풀을 통해 분배됩니다
4. 스테이커는 기본 배출량과 성능 기반 보상 모두에서 수익을 얻습니다
5. 언스테이킹 요청은 에포크 종료 시 적용되며, 토큰은 현재 에포크 동안 계속 수익을 창출합니다

## 토큰노믹스

SYND는 에포크 기반 배출이 있는 방향성 스테이킹 모델을 사용합니다:

- **SYND** — 10억 개의 고정 공급량을 가진 ERC-20 토큰
- 스테이킹 — 커먼스 체인에 SYND를 잠그고 배출량을 유도하며 배출량을 획득
- 에포크 — 첫날 00:00 UTC에 시작하는 30일 기간, 2025년 10월 1일로 예상

스테이커는 자신의 토큰을 등록된 모든 앱체인에 할당할 수 있습니다. 이러한 할당은:

- 전체 에포크 기간 동안 잠깁니다
- 배출량 분배에 직접적인 영향을 미칩니다
- 성능 기반 보상을 결정합니다
- 변경되지 않는 한 자동으로 롤오버됩니다

## 세 가지 풀 배출 시스템

각 에포크의 배출량은 세 개의 풀로 나뉩니다:

### 기본 풀(배출량의 30%)

- 스테이크 금액과 기간에 비례하여 모든 스테이커에게 분배됩니다
- 스테이크-블록 곱(토큰 × 스테이킹된 블록)을 사용하여 계산됩니다
- 앱체인 선택에 관계없이 보장된 기본 보상을 제공합니다

### 성과 풀(총 발행량의 30%)

- 지원한 앱체인의 성공에 따라 스테이커에게 보상 제공
- 분배는 앱체인 풀 할당을 비례적으로 반영
- 전략적 앱체인 선택과 리서치를 장려

### 앱체인 풀(총 발행량의 40%)

- 네트워크 기여도에 따라 앱체인에 직접 분배
- 거래 수수료(0.4)와 유치된 스테이크(0.2)에 따라 가중치 부여
- 작은 앱체인을 지원하기 위해 로그 재분배 방식 사용

## 예상 발행 일정

네트워크는 거의 선형적인 감소 일정을 따릅니다:

- 첫 번째 에포크: 2,577,259 SYND (총 공급량의 0.26%)
- 중간 지점(에포크 24): ~1,590,000 SYND
- 마지막 에포크: 997,205 SYND (총 공급량의 0.10%)
- 총 발행량: 48개 에포크 동안 80,000,000 SYND

발행된 토큰은 이더리움에서 권한 없는 기능을 통해 발행되고 분배를 위해 커먼스 체인으로 브릿지됩니다.

## 스테이킹 메커니즘

스테이킹 과정:

1. SYND를 커먼스 체인으로 브릿지
2. 스테이킹 컨트랙트에 토큰 예치
3. 선택한 앱체인에 스테이크 할당
4. 할당과 성과에 따라 발행량 획득

언스테이킹 과정:

1. 언스테이킹 요청 시작
2. 현재 에포크가 끝날 때까지 토큰은 스테이킹 상태 유지
3. 에포크 완료까지 발행량 계속 획득
4. 에포크 종료 후 토큰이 사용자의 커먼스 체인 주소로 반환

주요 특징:

- 지정된 스테이크는 전체 에포크 기간 동안 잠김
- 수정하지 않는 한 할당은 자동으로 이월됨
- 청구되지 않은 보상은 무기한 사용 가능
- 보상 청구 지연에 대한 페널티 없음

## 보상 계산

기본 풀 분배 개인 지분 = (사용자의 스테이크-블록 곱 / 전체 네트워크 스테이크-블록 곱) × 기본 풀 발행량

예시: 전체 에포크(216,000 블록) 동안 100,000 SYND 스테이킹 = 21.6B 스테이크-블록 곱

성과 풀 분배 개인 지분 =  $\Sigma$ (앱체인에 대한 사용자 할당 / 앱체인에 대한 총 할당) × 앱체인의 성과 풀 지분

예시: 풀의 45%를 획득하는 앱체인 A에 60%가 할당된 경우 = 비례 성과 배출량의 27%

앱체인 배출량 다음을 포함하는 지배 요소 기반:

- 지불된 거래 수수료(40% 가중치)
- 유치된 스테이킹(20% 가중치)
- 작은 앱체인 지원을 위한 로그 재분배 함수

## 보상 청구하기

스테이커 보상:

- 에포크 종료 직후 청구 가능
- 만료 또는 몰수 없음
- 여러 에포크에 걸쳐 누적 가능
- 사용자 편의에 따라 언제든지 청구 가능

앱체인 보상:

- 1년 선형 베스팅을 통해 분배
- 청구 가능한 부분 즉시 이용 가능

- 청구 가능한 부분 즉시 이용 가능
- 미확정 금액은 계속 스트리밍됨
- 청구 빈도에 대한 페널티 없음

#### 앱체인 리워드:

- 1년 선형 베스팅을 통해 분배
- 즉시 청구 가능한 부분 제공
- 베스팅되지 않은 금액은 계속 스트리밍됨
- 청구 빈도에 대한 페널티 없음

< 토큰 이코노믹스

SYND 토큰 분배, 발행 일정 및 공급 역학

브리징 >

체인 간 SYND 브리징

Q Search



K



SYND and emissions



기본 정보

토큰 이코노믹스

스테이킹 &amp; 배출

브리징

계약

라이트페이퍼

MiCAR 백서

## 브리징

체인 간 SYND 브리징

## SYND 브리징

SYND는 이더리움 메인넷, 베이스, 그리고 커먼스 체인에서 사용 가능합니다. 커먼스 체인은 SYND를 네이티브 가스 토큰으로 사용하며, 사용자들은 트랜잭션 비용을 지불하고 SYND를 스테이킹하기 위해 SYND를 커먼스로 브리징해야 합니다.

## 브리지 옵션

브리지	링크	설명	경로	상태
신디케이트 브리지	<a href="https://bridge.syndicate.io">bridge.syndicate.io</a>	호스팅된 릴레이 브리지 (즉시)	메인넷/베이스, 메인넷/커먼스, 베이스/커먼스	라이브
릴레이	<a href="https://relay.link">relay.link</a>	릴레이 공식 브리지 (즉시)	메인넷/베이스, 메인넷/커먼스, 베이스/커먼스	라이브
슈퍼브리지	<a href="https://superbridge.app">superbridge.app</a>	슈퍼브리지 (정식)	메인넷/베이스	라이브-수동
아비트럼	<a href="https://bridge.arbitrum.io">bridge.arbitrum.io</a>	아비트럼 공식 (정식)	베이스/커먼스	라이브

신디케이트는 이 페이지에 나열되거나 나열되지 않은 어떤 브리지에서도 수수료를 받지 않습니다. 나열된 브리지는 정보 제공 목적으로만 제공됩니다. 사용자는 자체 조사를 수행하고 자신의 책임 하에 브리지를 사용해야 합니다.



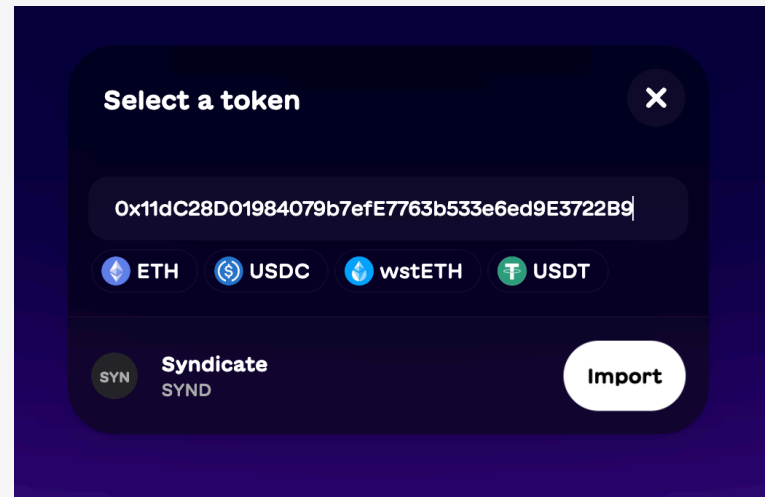
# 슈퍼브리지를 사용하여 메인넷과 베이스 간 SYND 토큰 브리징하기

사용자는 사용자 친화적인 브리징 인터페이스인 슈퍼브리지를 사용하여 이더리움 메인넷에서 베이스로 SYND 토큰을 브리징할 수 있습니다.

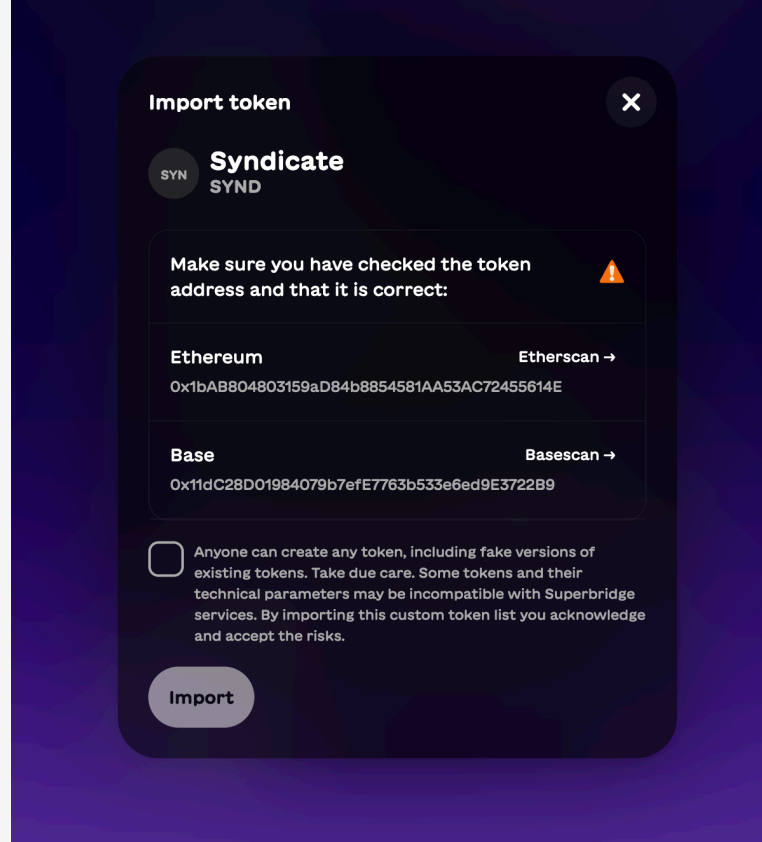
## 슈퍼브리지 사용하기

1. 브리지 UI 접속하기: SYND 토큰 매개변수가 미리 구성된 슈퍼브리지로 이동합니다.
2. 지갑 연결하기: 지갑이 이더리움 메인넷에 연결되어 있는지 확인합니다.
3. **SYND** 토큰 추가하기: 토큰 검색창에 SYND 토큰 주소를 붙여넣습니다:

- 베이스 네트워크의 **SYND**: `0x11dC28D01984079b7efE7763b533e6ed9E3722B9`



4. 토큰 가져오기: 토큰을 찾는 데 문제가 있는 경우, ETH 메인넷의 SYND 주소 (`0x1bAB804803159aD84b8854581AA53AC72455614E`)도 붙여넣어야 할 수 있습니다.



5. 브리지 완료하기: 표준 브리징 과정을 따라 SYND 토큰을 베이스로 전송합니다.

### 토큰 주소

- 이더리움의 SYND: 0x1bAB804803159aD84b8854581AA53AC72455614E
- 베이스의 SYND: 0x11dC28D01984079b7efE7763b533e6ed9E3722B9

## 아비트럼 브리지를 사용하여 베이스와 커먼스 체인 간 SYND 토큰 브리징하기

사용자는 아비트럼 브리지를 사용하여 베이스와 커먼스 체인 간에 SYND 토큰을 브리징할 수 있습니다.

베이스 → 커먼스 체인: [bridge.arbitrum.io/?destinationChain=commons&sanitized=true&sourceChain=base](https://bridge.arbitrum.io/?destinationChain=commons&sanitized=true&sourceChain=base)

커먼스 체인 → 베이스: [bridge.arbitrum.io/?destinationChain=base&sanitized=true&sourceChain=commons](https://bridge.arbitrum.io/?destinationChain=base&sanitized=true&sourceChain=commons)

< 스테이킹 & 배출

SYND 스테이킹 작동 방식, 배출 분배 및 보상 계산

계약 >

SYND 토큰, 스테이킹 및 배출 시스템을 구동하는 핵심 계약

Q Search

⌘

K

🔍 SYND and emissions

↕

기본 정보

토큰 이코노믹스

스테이킹 & 배출

브리징

계약

라이트페이퍼

MiCAR 백서

## 계약

SYND 토큰, 스테이킹 및 배출 시스템을 구동하는 핵심 계약

## 핵심 계약

다음 계약들은 이더리움과 베이스에서 SYND 토큰 발행을 지원합니다:

### 토큰 및 거버넌스

계약	네트워크	주소	설명
SYND 토큰	이더리움	<u>0x1bAB804803159aD84b8854581AA53AC72455614E</u>	10억 고정 공급량을 가진 ERC-20 토큰 계약
SYND 토큰	베이스	<u>0x11dC28D01984079b7efE7763b533e6ed9E3722B9</u>	베이스에 크로스체인 배포
SYND 토큰	Commons Chain	해당 없음(네이티브 토큰)	Commons L3의 네이티브 토큰

### 스테이킹 및 발행

계약	네트워크	주소	설명
			다양한 애플체인과 에포크에 걸쳐 사용자 스테이킹을 관리하는 핵심 스테이킹 계약으로, 배

SyndStaking	Commons Chain	<u>0xF9637B60f27AF139FC46EAa655cFBbe4E731BCdF</u>	이코를 관리하는 핵심 스테이킹 계약으로, 발행 계산을 위한 스테이크 금액과 기간을 추천합니다
BasePool	Commons Chain	<u>0x71cF8bf70Bb4f5ba8e4B4588bacB5ee108f3Ed10</u>	스테이커가 스테이크 금액과 기간에 비례하여 에포크 발행량의 30%를 받는 베이스 풀 보상 분배 시스템을 구현합니다. 퍼포먼스 및 베이스 풀이 출시될 때까지(2025년 10월 30일 에포크 1 예정), 베이스 풀은 (에포크 0에서) 발행량의 100%를 받게 됩니다
Refunder	Commons Chain	<u>0x9BE716F21428a254a2e4825cfa1d8A0893B9827B</u>	브릿지 작업에서 남은 SYND를 복구하고 현재 에포크에 맞는 적절한 풀에 예치하는 유틸리티 계약

## 발행 인프라

계약	네트워크	주소	설명
L1Relayer	이더리움	<u>0x96f93df52B769AD3E7f633E4fc68cb2Cc1E33686</u>	OP 스탠다드 브릿지를 사용하여 이더리움 메인넷에서 베이스로 SYND 발행 토큰을 브릿지하고 L2 릴레이어를 트리거하기 위한 크로스체인 메시지를 전송합니다
L2Relayer	베이스	<u>0x5c1aD8136FF7C1bEF7fAc1AD09ccCdc40488119E</u>	Arbitrum 브릿지를 사용하여 베이스(L2)에서 Commons Chain(L3)으로 SYND 발행 토큰을 브릿지하고 작업을 릴레이합니다

## 발행

계약	네트워크	주소	설명
EmissionsCalculator	이더리움	<u>0x7CC604b2e117693fE214b8253504eC29BE9EcF0a</u>	48개 에포크 각각에 대한 발행량을 계산하는 공식을 구현하여 8천만 SYND 한도를 적용합니다
EmissionsScheduler	이더리움	<u>0xcD3602332fA70191A0e1A1b49aC9873aD4D87E0e</u>	적절한 시간에 민팅을 트리거하고 이더리움에서 베이스를 통해 Commons Chain으로의 브릿지 시퀀스를 시작하여 발행 프로세스를 조정합니다

< 브리징

체인 간 SYND 브리징

라이트페이퍼 >

라이트페이퍼 읽기

Q Search



K

SYND and emissions



기본 정보

토큰 이코노믹스

스테이킹 &amp; 배출

브리징

계약

라이트페이퍼

MiCAR 백서

# 라이트페이퍼

라이트페이퍼 읽기

## 개요

SYND 토큰 라이트페이퍼는 신디케이트 네트워크의 경제 모델, 토큰 분배, 그리고 원자적으로 구성 가능하고 프로그래밍 가능한 애플체인 네트워크에 대한 장기적인 비전에 관한 포괄적인 세부 정보를 제공합니다.

## 전체 라이트페이퍼 다운로드

**SYND 토큰 라이트페이퍼 v2.0**

SYND 토큰 라이트페이퍼 v2.0

전체 라이트페이퍼에는 신디케이트 네트워크 경제의 심층적인 메커니즘에 관심이 있는 개발자와 커뮤니티 구성원을 위한 상세한 공식, 발행 일정, 거버넌스 구조 및 기술 사양이 포함되어 있습니다.

Looking for the previous version? Download the **SYND Token Litepaper v1.0**.

&lt; 계약

SYND 토큰, 스테이킹 및 배출 시스템을 구동하는 핵심 계약

MiCAR 백서 &gt;

MiCAR 백서 읽기





Q Search



K

🔍 SYND and emissions



기본 정보

토큰 이코노믹스

스테이킹 &amp; 배출

브리징

계약

라이트페이퍼

MiCAR 백서

# MiCAR 백서

MiCAR 백서 읽기

## 개요

Syndicate Network MiCAR 백서는 암호화폐 자산 시장 규제(Markets in Crypto-Assets Regulation, MiCAR)에 따른 Syndicate Network의 규제 프레임워크, 준수 메커니즘 및 법적 구조에 대한 포괄적인 세부 정보를 제공합니다.

## 전체 백서 다운로드

**Syndicate Network MiCAR 백서**

Syndicate Network MiCAR 백서

*MiCAR 백서에는 규제 준수, 토큰 분류, 소비자 보호 조치 및 Syndicate Network 생태계 참여자를 위한 법적 프레임워크에 대한 자세한 정보가 포함되어 있습니다.*

&lt; 라이트페이퍼

라이트페이퍼 읽기