블록체인DApp설계

1. 이더리움이란 무엇인가?

경기대학교 AI컴퓨터공학부 이재흥
jhlee@kyonggi.ac.kr

KGU 경기대학교

CONTENTS
PRESENTATION

- 이더리움(Ethereum)
- 비트코인과의 비교
- 블록체인 구성요소
- 이더리움 개발의 4단계
- 이더리움 타임라인
- 이더리움 로드맵
- 이더리움: 범용 블록체인
- 이더리움의 구성요소
- 이더리움과 튜링 완전
- 탈중앙화 애플리케이션(DApp)
- 제3세대 인터넷

이더리움(Ethereum)

- 월드 컴퓨터(world computer)
  - 컴퓨터 과학의 관점
    - 결정론적(deterministic)이지만, 사실상 한정되지 않은 상태 머신(unbounded state machine)
    - 전역적으로(globally) 접근 가능한 싱글톤(singleton) 상태와 그 상태를 변화시킬 수 있는 가상 머신으로 구성
  - 좀 더 실용적인 관점
    - 스마트 컨트랙트(smart contract)라는 프로그램을 실행하는 오픈 소스에 기반을 둔, 전 세계에 걸쳐 탈중앙화된 컴퓨팅 인프라스트럭처
    - 블록체인을 사용하여 시스템의 상태 변화를 동기화하고 저장하며, 이더(ether)라고 하는 암호화폐를 이용하여 실행 자원 비용을 측정하고 제한

비트코인과의 비교

- 다른 개방형 블록체인과 공유하는 공통 요소
  - 참여자들을 연결하는 피어투피어(peer-to-peer) 네트워크
  - 상태 변경을 동기화하는 비잔틴 결함 허용 합의 알고리즘(ex. 작업증명, 지분증명)
  - 디지털 서명과 해시
  - 디지털 화폐(이더) 같은 암호학 기반 기술의 사용
- 다른 개방형 블록체인과 다른 점
  - 디지털 화폐 지급 네트워크가 주 목적이 아님
  - 디지털 화폐인 이더는 월드 컴퓨터로서의 이더리움 플랫폼 사용료를 지불하기 위한 유틸리티 화폐(utility currency)로 사용됨
  - 매우 제한된 스크립트 언어를 사용하는 비트코인과 달리, 임의성과 무한 복잡성을 가진 코드를 실행할 수 있는 가상 머신(virtual machine)을 운영하는 범용 프로그램이 가능한 블록체인으로 설계 ➔ 튜링 완전(Turing complete) 언어

블록체인 구성요소

- 일반적인 공개 블록체인의 구성요소
  - 피어투피어(P2P) 네트워크
    - 표준화된 가십(gossip) 프로토콜을 기반으로 참여자를 연결하고 트랜잭션 및 검증된 트랜잭션 블록을 연결
  - 트랜잭션
    - 상태 전이를 나타냄
  - 합의 규칙
    - 트랜잭션의 구성 요건과 트랜잭션의 유효성을 판단
  - 상태 머신
    - 합의 규칙에 따라 트랜잭션을 처리

블록체인 구성요소

- 일반적인 공개 블록체인의 구성요소
  - 데이터 구조
    - 검증되고 적용된 모든 상태 전이의 장부(journal) 역할을 해줄 수 있는, 암호학적으로 보호된 체인(chain)
  - 합의 알고리즘
    - 합의 규칙들을 적용하는 데 모든 참여자가 협력할 수 있도록 강제함으로써 블록체인의 통제 권한을 탈중앙화함
  - 인센티브 메커니즘
    - 공개된 환경에서 상태 머신에 경제적인 보안성을 제공할 수 있도록 게임 이론적으로 유효해야 함
    - 예) 작업증명 비용 + 블록 보상
  - 구현 소프트웨어 (오픈소스)

이더리움 개발의 4단계

- 2017년 이더리움 재단이 제시한 이더리움 로드맵 4단계

1단계 프론티어(Frontier)

- **암호화폐인 이더리움을 개발·채굴하고 네트워크를 형성하는 단계**
- 이더리움 제네시스 블록 생성
- 2015년 7월 30일 시작

2단계 홈스테드(Homestead)

- **이더리움이라는 신대륙에 가정집이 하나둘씩 생기면서 생태계 구축 단계**
- 이더리움 1,150,000블록 시행
- 2016년 3월 14일 시작

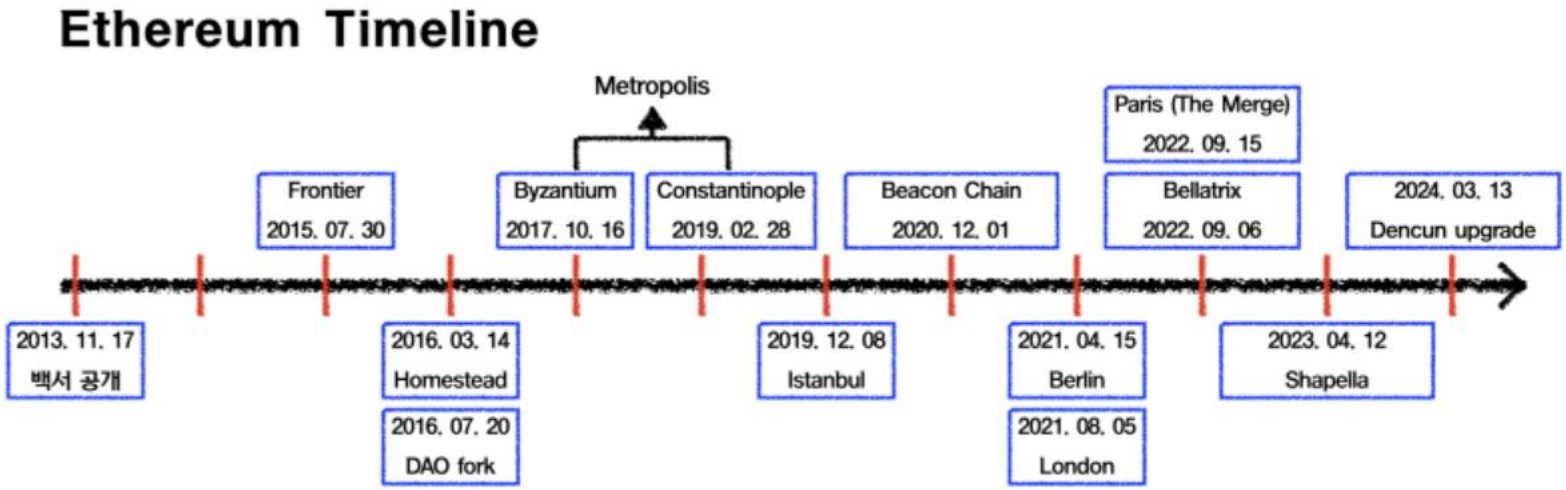
3단계 메트로폴리스(Metropolis)

- **대중화를 위한 사회적 인프라 형성단계**
- 3-1단계 비잔티움(Byzantium)
  채굴보상 5→3ETH 감소(17.10.16 성공)
- 3-2단계 콘스탄티노플(Constantinople)
  채굴보상 3→2ETH 감소(19.3.1 성공)

4단계 세레니티(Serenity)

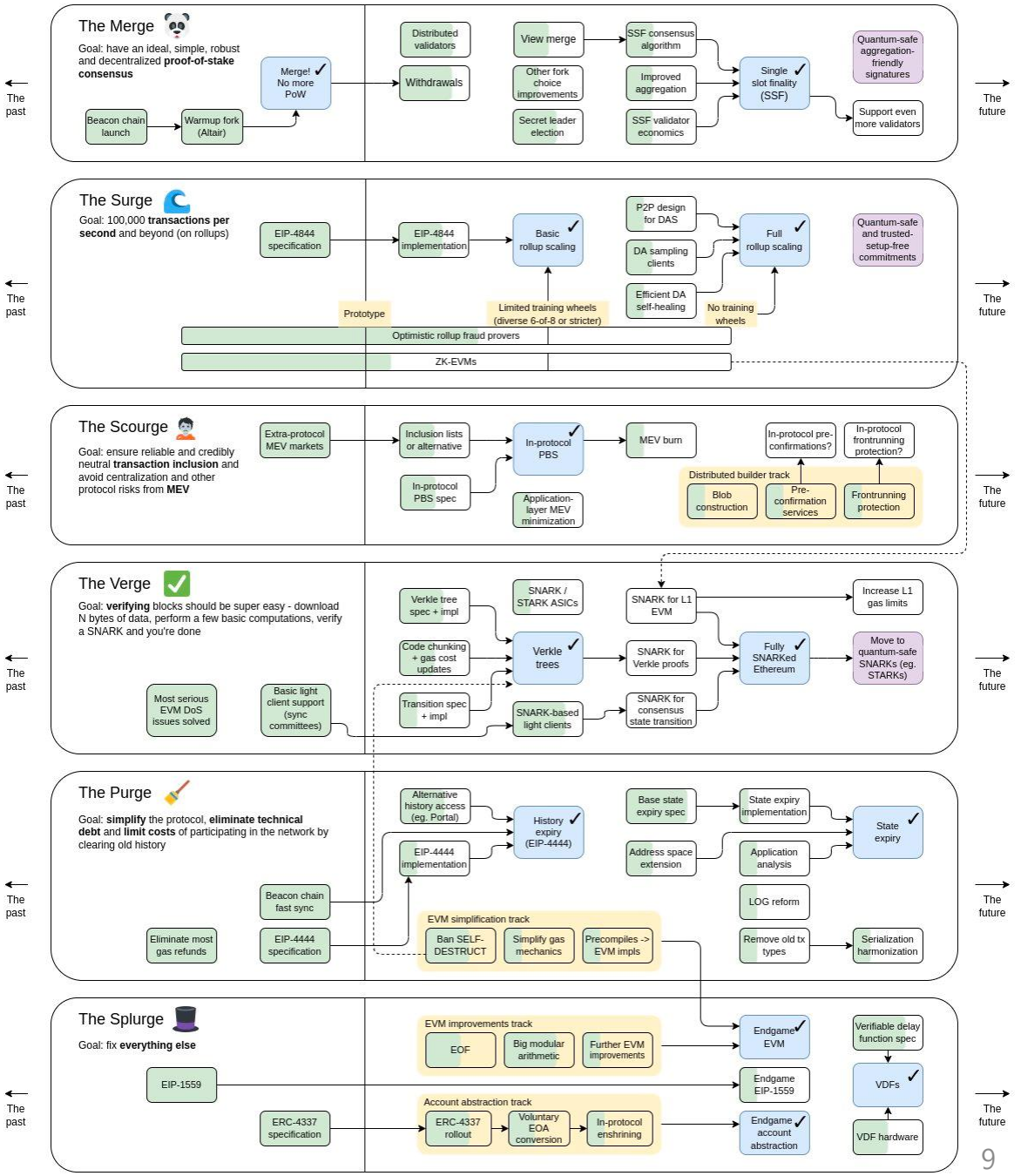
- ***3.5단계 이스탄불(Istanbul)***
  ProgPOW 실행 검토 외 (19.10월 예정)
- **모든 변화 후에 평온 또는 평정을 찾는 마지막 단계**
- Casper(POW→POS로의 완벽한 전환)
- 이더리움 2.0 의미, Shading(샤딩), POS, eWASM 등이 제공(20.1월 예정)

이더리움 타임라인



https://contents.premium.naver.com/yonseident/ysdent/contents/240311131216289pm

이더리움 로드맵


The past
The future
The Merge
Goal: have an ideal, simple, robust and decentralized proof-of-stake consensus
Beacon chain launch
Warmup fork (Altair)
Merge! ✓ No more PoW
Distributed validators
Withdrawals
View merge
Other fork choice improvements
Secret leader election
SSF consensus algorithm
Improved aggregation
SSF validator economics
Single slot finality (SSF) ✓
Quantum-safe aggregation-friendly signatures
Support even more validators
The Surge
Goal: 100,000 transactions per second and beyond (on rollups)
EIP-4844 specification
EIP-4844 implementation
Basic rollup scaling ✓
Prototype
Limited training wheels (diverse 6-of-8 or stricter)
P2P design for DAS
DA sampling clients
Efficient DA self-healing
No training wheels
Full rollup scaling ✓
Quantum-safe and trusted-setup-free commitments
Optimistic rollup fraud provers
ZK-EVMs
The Scourge
Goal: ensure reliable and credibly neutral transaction inclusion and avoid centralization and other protocol risks from MEV
Extra-protocol MEV markets
Inclusion lists or alternative
In-protocol PBS spec
In-protocol PBS ✓
Application-layer MEV minimization
MEV burn
In-protocol pre-confirmations?
In-protocol frontrunning protection?
Distributed builder track
Blob construction
Pre-confirmation services
Frontrunning protection
The Verge
Goal: verifying blocks should be super easy - download N bytes of data, perform a few basic computations, verify a SNARK and you're done
Most serious EVM DoS issues solved
Basic light client support (sync committees)
Verkle tree spec + impl
Code chunking + gas cost updates
Transition spec + impl
Verkle trees ✓
SNARK-based light clients
SNARK / STARK ASICs
SNARK for Verkle proofs
SNARK for consensus state transition
SNARK for L1 EVM
Fully SNARKed Ethereum ✓
Increase L1 gas limits
Move to quantum-safe SNARKs (eg. STARKs)
The Purge
Goal: simplify the protocol, eliminate technical debt and limit costs of participating in the network by clearing old history
Eliminate most gas refunds
Beacon chain fast sync
EIP-4444 specification
Alternative history access (eg. Portal)
EIP-4444 implementation
History expiry (EIP-4444) ✓
EVM simplification track
Ban SELF-DESTRUCT
Simplify gas mechanics
Precompiles -> EVM impls
Base state expiry spec
Address space extension
State expiry implementation
Application analysis
LOG reform
Remove old tx types
State expiry ✓
Serialization harmonization
The Splurge
Goal: fix everything else
EIP-1559
ERC-4337 specification
EVM improvements track
EOF
Big modular arithmetic
Further EVM improvements
Account abstraction track
ERC-4337 rollout
Voluntary EOA conversion
In-protocol enshrining
Endgame EVM ✓
Endgame EIP-1559
Endgame account abstraction ✓
Verifiable delay function spec
VDFs ✓
VDF hardware

이더리움: 범용 블록체인

- 비트코인 블록체인
  - 비트코인 단위 및 소유 상태를 추적
  - 트랜잭션이 상태 전이(state transition)을 일으켜 코인의 소유권을 변경하는 탈중앙화된 합의 상태 머신(state machine)으로 생각 가능
  - 상태 전이는 여러 블록이 채굴된 후 모든 참가자가 시스템의 공통(합의) 상태로 수렴할 수 있도록 합의 규칙에 의해 제한됨
- 이더리움 블록체인
  - 탈중앙화 상태 머신
  - 화폐 소유 상태만 추적하는 대신 범용 데이터 저장소, 즉 키-밸류 튜플(key-value tuple)로 표현할 수 있는 모든 데이터를 저장할 수 있는 저장소의 상태 전이를 추적
  - 특정 키는 키-밸류 데이터 저장소에서 임의의 값을 보유하고 그 값을 참고

이더리움의 구성요소

- 이더리움 블록체인 시스템의 구성요소
  - 피어투피어(P2P) 네트워크
    - TCP 포트 30303으로 접속 가능한 이더리움 메인 네트워크에서 실행
  - 트랜잭션
    - 보낸 사람, 받는 사람, 값 및 데이터 페이로드가 포함된 네트워크 메시지
  - 합의 규칙
    - 기준 사양인 황서(Yellow Paper)에 정의되어 있음
  - 상태 머신
    - 바이트코드(bytecode)를 실행하는 스택 기반 가상 머신인 EVM(Ethereum Virtual Machine, 이더리움 가상 머신)에 의해 상태 전이 처리
    - 스마트 컨트랙트라는 EVM 프로그램은 고수준 프로그래밍 언어(ex. 솔리디티)로 작성되고, EVM에서 실행되도록 바이트코드로 컴파일됨

이더리움의 구성요소

- 이더리움 블록체인 시스템의 구성요소
  - 데이터 구조
    - 이더리움의 상태는 트랙잭션 및 시스템 상태가 머클 패트리샤 트리라고 하는 시리얼라이즈된 해시 데이터 구조로, 각 노드의 데이터베이스에 저장
  - 합의 알고리즘
    - 비트코인의 합의 모델인 나카모토 합의(Nakamoto Consensus)를 사용
    - 순차 단일 서명 블록을 사용하여 작업증명(PoW)의 중요도 가중치가 가장 긴 체인(현재 상태)를 결정
    - 2022년 9월 15일 'Paris : The Merge'를 통해 지분증명(PoS)으로 전환
  - 인센티브 메커니즘
    - Ethash라는 작업증명(PoW) 알고리즘을 사용하다 지분증명(PoS)으로 전환
  - 구현 소프트웨어 (오픈소스)
    - 게스(Go-Ethereum, Geth)와 패리티(Parity)가 가장 유명

이더리움과 튜링 완전

- 튜링 머신(Turing machine)
  - 1936년 영국인 수학자 앨런 튜링(Alan Turing)이 제안
  - 순차적 메모리(무한 길이의 종이 테이프와 유사)에서 기호를 읽고 쓰는 방식으로 기호를 조작하는 상태 머신으로 구성된 컴퓨터의 수학적 모델
- 튜링 완전(Turing complete)
  - 시스템이 튜링 머신을 시뮬레이션하는 데 사용할 수 있음을 의미
- 이더리움과 튜링 완전
  - 이더리움은 한정된 메모리라는 제한 조건에서 모든 튜링 머신으로 계산될 수 있는 어떠한 알고리즘도 계산할 수 있음 (튜링 완전)

이더리움과 튜링 완전

- '기능'으로서의 튜링 완전
  - 튜링 완전 언어는 의외로 쉽게 만들 수 있음
    - 22개의 명령어 길이를 갖는 상태의 정의와 함께 4개의 상태를 가지고 6개의 기호만 사용하여 튜링 완전 상태 머신 제작 가능
  - 튜링 완전 언어는 정지 문제(halting problem, 임의로 주어진 튜링 머신이 주어진 입력 테이프에 대해 정지하는가 정지하지 않는가를 판정하는 알고리즘의 존재 여부를 묻는 문제)로 인해 심각한 문제를 야기할 수 있음
  - 이더리움은 튜링 완전하기 때문에 어떤 복잡도의 프로그램도 실행할 수 있는 유연성이 있어 이로 인해 자원 관리와 보안에 치명적 문제를 야기할 수 있음
  - 특히 이더리움은 공개 탈중앙화 블록체인이기 때문에 오류 발생시 "껐다 켤 수 없음"

이더리움과 튜링 완전

- 튜링 완전의 함축적 의미
  - 튜링 완전 시스템은 무한 루프가 발생할 수 있음
  - 모든 참여 노드(클라이언트)는 모든 트랜잭션을 검증하고 그 트랜잭션이 호출하는 스마트 컨트랙트를 실행해야 함
  - 정지 문제로 인해 이더리움은 스마트 컨트랙트가 종료될 지 혹은 실제로 스마트 컨트랙트를 실행하지 않고 얼마나 오랫동안 실행될지를 예측할 수 없음
  - 이를 악용하면 스마트 컨트랙트의 유효성 검사를 무한히 반복하도록 할 수 있음
    - 서비스 거부 공격(Denial of Service, DoS)
  - 이더리움은 이를 해결하기 위해 가스(gas)라는 과금 메커니즘을 도입
    - 스마트 컨트랙트를 실행하는 데 사용할 수 있는 가스의 최대 사용량을 미리 정해 놓는 방법으로 무한 루프 문제 해결

탈중앙화 애플리케이션(DApp)

- 이더리움의 발전
  - 다양한 용도로 프로그래밍을 할 수 있는 범용적인 블록체인을 만들기 위한 시도 → 댑(DApp) 프로그래밍을 위한 플랫폼
- 댑(DApp)
  - '스마트 컨트랙트'보다 넓은 의미
  - 공개되고 탈중앙화된 피어투피어 기반 서비스 위에 제공되는 웹 애플리케이션
  - 최소 구성
    - 블록체인 스마트 컨트랙트
    - 웹 프런트엔드 사용자 인터페이스
  - 추가 구성
    - 탈중앙화(P2P) 스토리지 프로토콜과 플랫폼
    - 탈중앙화(P2P) 메시지 프로토콜과 플랫폼

제3세대 인터넷

- 웹3.0(Web3.0)
  - 이더리움 재단의 개빈 우드 박사가 제안한 개념
  - 중앙 집중적으로 관리되는 애플리케이션이 아닌, 탈중앙화 프로토콜에 의해 구축된 애플리케이션(DApp)에 초점을 맞춤
  - web3.js 자바스크립트 라이브러리
    - 브라우저 안에서 실행되는 자바스크립트 애플리케이션과 이더리움 블록체인을 연결
    - 스웜(Swarm)이라는 P2P 스토리지 네트워크와 위스퍼(Whisper)라는 P2P 메시징 서비스를 포함

THANK YOU!