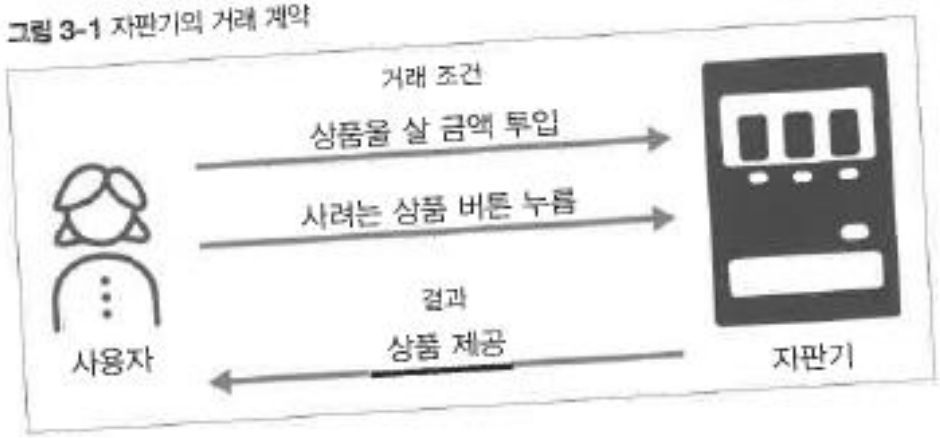
3장 스마트 계약과 이더리움 이해하기

출처 : 처음 배우는 블록체인 (한빛미디어)

1. 스마트 계약

- 현실의 권리 증명이나 이동 등을 포함한 계약을 자동으로 실행하는 구조를 스마트 계약이라고 함.
- 1997년 비트골드의 스폰서이며, 암호학자 닉 스자보가 처음 제안



- 블록체인의 스마트 계약 : 블록체인 안에서 동작하는 자동 계약 프로그램을 의미
- 블록체인의 스마트 계약의 장점
  ① 상대를 신뢰하지 않아도 거래에 문제가 발생하지 않음.
  ② 중개자가 필요 없으므로 비용을 절감

- 스마트 계약의 실제 예
  - ① 법을 자동으로 이행
  - ② 콘텐츠 수익을 자동으로 지급
  - ③ 보험금 자동 지급
  - ④ 카 렌트
  - ⑤ 자동기부 시스템
  - ⑥ 고용 계약 시스템
  - ⑦ 전자 투표

그림 3-2 재산 분할의 스마트 계약

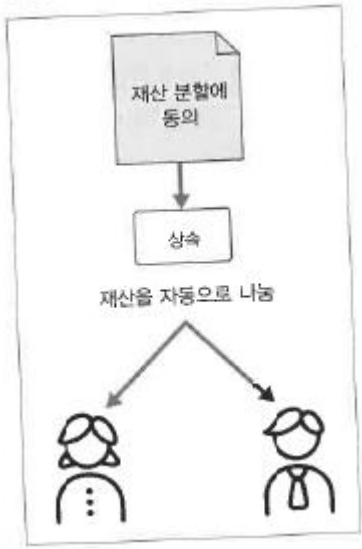
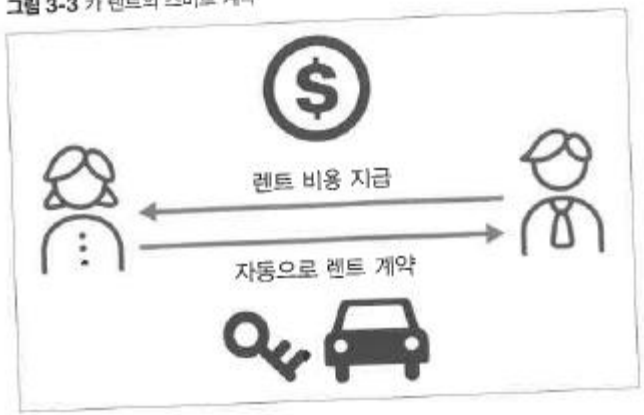


그림 3-3 카 렌트의 스마트 계약



2. 이더리움

- 초기 블록체인 애플리케이션의 개발 및 운영이 어려워서 비탈릭 부테린이 블록체인 플랫폼을 개발함.
- 이더리움은 프로토콜을 규정해 범용성을 갖음.
- 이더리움의 역사/로드맵
  ① 프런티어 : 2015년 7월 30일 실시, 명령 줄 인터페이스 기반, 기초 테스트 버전
  ② 홈스테드 : 2016년 3월 14일 실시, 노드가 생기면서 생태계를 구축
  ③ 메트로폴리스 : 2017년 10월 17일 실시, 대중화를 위한 인프라의 형성
  ④ 세레니티 : 진행일 임정, 지분 증명 알고리즘의 안정화를 위한 마지막 단계

3. 비트코인과 이더리움의 차이

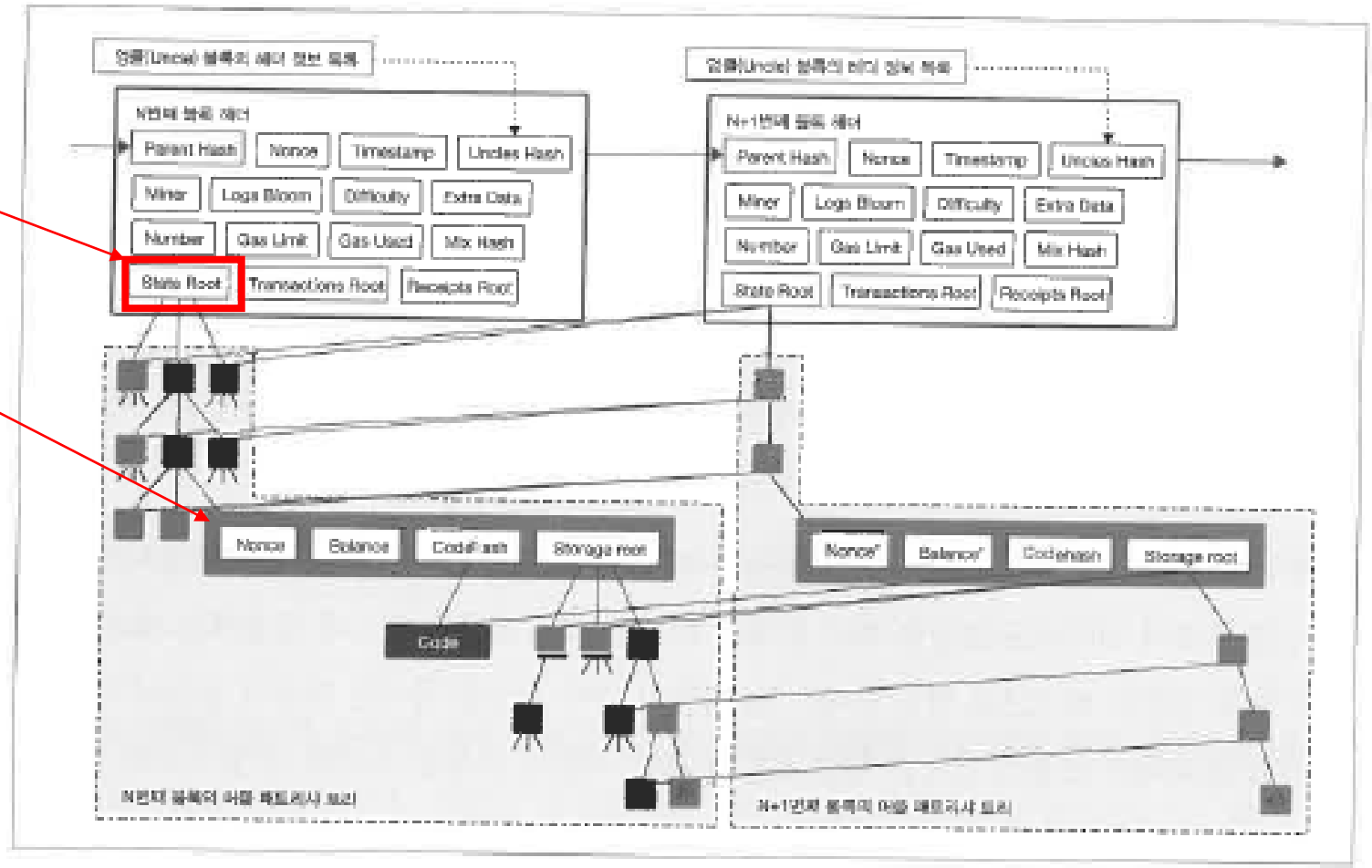
- 이더리움의 화폐 단위
- 거래 수수료 "가스"
  - ✓프로그램 연산을 위한 연료라는 뜻임.
  - ✓가스 가격 x 가스 제한 => 가스 상한
  - ✓가스 상한을 두어 무한 루프를 방지

명칭	WEI 기준	Ether 기준
WEI	1	0.000000000000000001
Ada	1000	0.000000000000001
Fentoether	1000	0.000000000000001
Kwei	1000	0.000000000000001
Mwei	1000000	0.000000000001
Babbage	1000000	0.000000000001
Pictoether	1000000	0.000000000001
Shannon	1000000000	0.000000001
Gwei	1000000000	0.000000001
Nano	1000000000	0.000000001
Szabo	1000000000000	0.000001
Micro	1000000000000	0.000001
Microether	1000000000000	0.000001
Finney	1000000000000000	0.001
Milli	1000000000000000	0.001
Milliether	1000000000000000	0.001
Ether	1000000000000000000	1
Einstein	10000000000000000000	1000
Kether	10000000000000000000	1000
Grand	10000000000000000000	1000
Mether	1000000000000000000000	1000000
Gether	1000000000000000000000000	1000000000
Tether	1000000000000000000000000000	1000000000000

- 계정 구조
  - ✓ 외부계정 : 이더리움 사용자를 위한 계정, 주소와 연결해 잔액 정보를 갖음.
  - ✓ 계약계정 : 계약 증명이 있는 계정

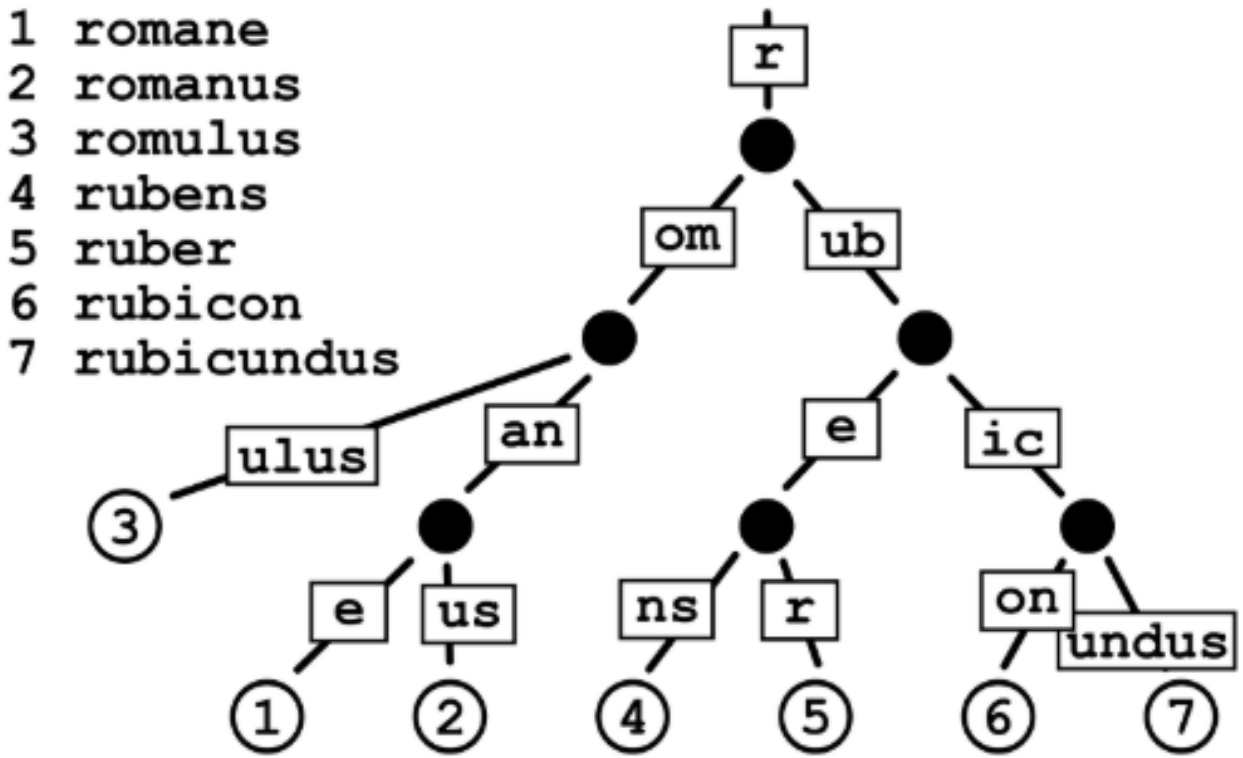
- 이더리움의 블록의 구조
- 상태 트리의 루트
  - ✓ 각각의 노드에 있는 계정주소들로 해시 값을 계산하여 데이터 조작을 방지
- 상태 트리는 해시 트리와 패트리샤 트리 결합 구조
- 상태 트리에 저장하는 계정 정보
  - ✓ Nonce : 계정에서 보낸 거래 횟수
  - ✓ Balance : 계정의 잔액 정보, wei단위
  - ✓ storageRoot : 계정과 연결된 저장소 트리의 루트 노드
  - ✓ CodeHash : Ethereum Virtual Machine 코드의 해시값, 계약 계정은 실행코드를 저장, 외부계정은 공백문자를 해시값

그림 3-4 이더리움 블록 구조[4]



4. "패트리샤 트리"

이제 패트리샤 트리에 대한 설명을 시작합니다. 이것도 역시 공간을 절약하기 위함입니다.

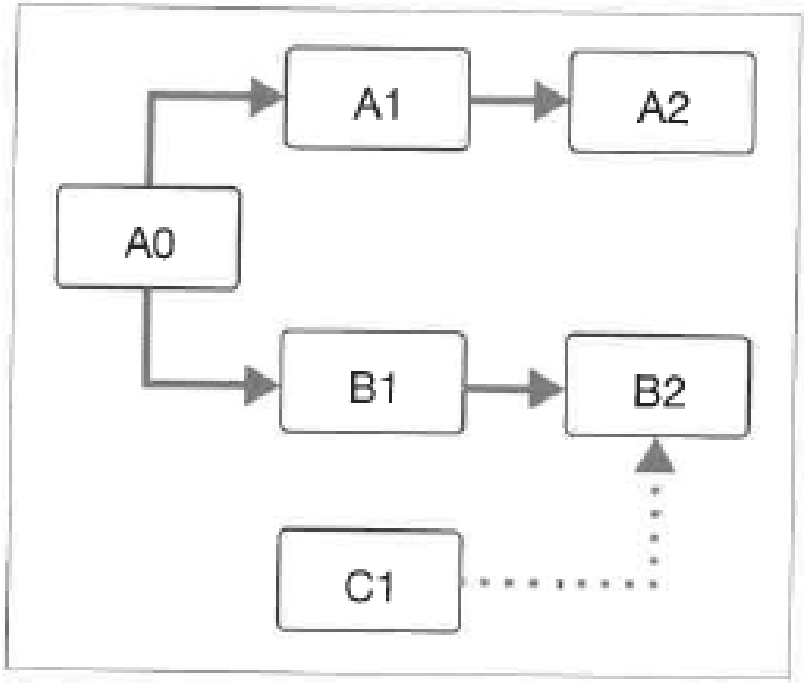


- 잔액 확인 방식의 차이
  - 비트 코인 : 사용하지 않은 금액을 담은 UTXO의 합으로 잔액을 표현
    - ✓장점 : 병렬로 거래 가능
    - ✓단점 : 잔액 확인 구조가 복잡
  - 이더리움 : 송금내역을 블록에 거래로 저장한 후, 계정 정보를 담은 상태 트리를 만들어 잔액을 표현
    - ✓장점 : 계정정보를 별도의 자료구조로 분리해 빠른 검색 가능
    - ✓단점 : 계정정보를 변경시 선입선출 방식으로 작업으로 실행 => 병렬처리에 문제

- 블록생성 속도의 차이
  - ✓비트코인 : 10분에 1회
  - ✓이더리움 : 15초에 1회

- 블록선택 방식의 차이
  - ✓비트코인 : 나카모토 합의에 따라 여러 블록중 작업 증명 알고리즘을 푸는 난이도가 높은 블록
  - ✓이더리움 : 각 체인에 묶인 블록들의 생성 난이도 전체를 더한 결과가 가장 높은 체인을 메인 체인 => 수정 고스트 프로토콜

그림 3-5 블록 개수로 메인 체인 선택



- 작업 증명 알고리즘의 차이
  - ✓비트코인 : SHA-256해시함수 연산 반복
    - 단점 : 전용 연산 하드웨어를 도입한 일부 사용자가 채굴 보상 독식
  - ✓이더리움 : 이더해시
    1. 블록의 헤더를 스캔해 블록을 대상으로 계산할 수 있는 값인 시드를 추출
    2. 시드에서 16MB 단위의 의사랜덤 캐시를 만들고 GB크기를 갖는 비순환 방향성 그래프 기반의 데이터셋을 만듬
    3. 1GB 이상의 데이터셋에서 일부를 추출해 특정 조건이 될때까지 임의의 값을 변경하면서 채굴
    4. 데이터셋은 3만 블록 단위로 바꾸며 선형으로 커짐. 이를 이용해 일정 패턴으로 메모리 읽기 연산을 못하게 막음. 데이터셋 저장 공간도 일정하지 않도록 함.
    - 장점 : 특정 주체가 채굴 보상을 독식을 막음.
- 이더리움 가상 머신
  - ✓애플리케이션을 실행하는 기반으로 튜링 완전한 가상 머신을 제공하며 이를 Ethereum Virtual Machine (EVM) 함.
  - ✓스마트 계약은 코드 작성후 EVM에서 실행