

EVM(Ethereum virtual machine)

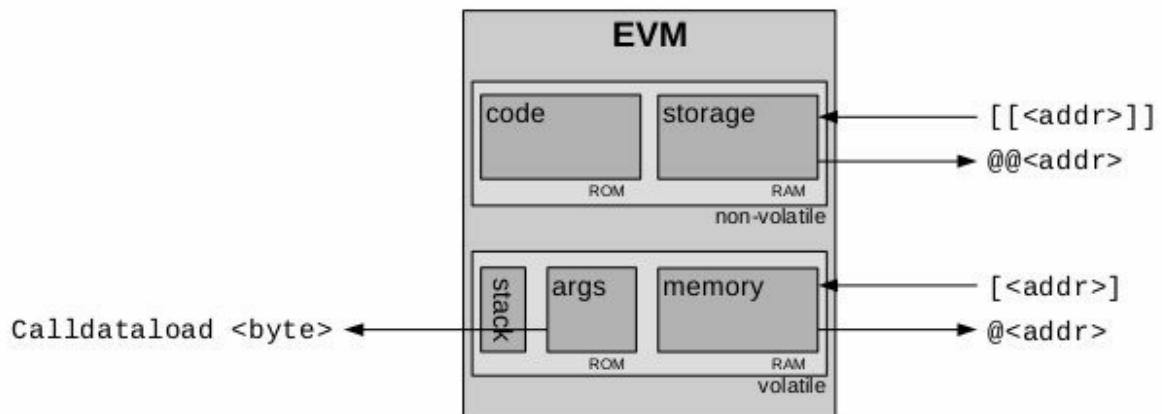
이더리움에서 스마트 컨트랙트를 실행하기 위한 실행환경 제공

The EVM is a stack-based virtual machine with a memory byte-array and key-value storage

EVM 구조



Ethereum coding recap:



instruction set (program counter)

모든 명령어는 기본 데이터 유형인 256비트 워드에서 작동합니다.

산술, 비트, 논리 및 비교 연산이 있고 조건적 및 무조건적 점프가 가능합니다.

contract는 block의 속성에 접근할 수 있습니다.

프로그램 영역에는 EVM이 실행할 스마트 컨트랙트의 EVM 명령어 목록을 보관

stack

연산에 필요한 데이터 저장,
256비트 크기로 값들 저장,
스택의 최대 크기는 1024개,
내부 함수에서 사용
(다른 컨트랙트 메소드를 호출, 메시지 콜을 위한 스택 사용)

message call (Call Data)

메시지 콜은 source, target, data payload, ether, gas, return data 가 있습니다.
호출된 contract는 메모리의 객체를 받아 payload에 접근합니다.
payload 는 calldata 라 불리는 영역에 있습니다.
실행이 끝나면 호출자의 메모리 위치에 데이터를 반환할 수 있습니다.
메시지 콜은 1024 로 depth 제한이 있다.

log

블록 레벨까지 모든 것을 맵핑하는 특별히 인덱스 된 데이터 구조에 데이터를 저장할 수 있습니다.
이 기능은 이벤트를 구현하기 위해 Solidity에서 사용합니다. contract 에서 데이터를 만든 후에 접근할 수 없지만
블록 체인 외부에서 효율적으로 액세스 할 수 있다.
로그 데이터의 일부는 블룸 필터에 저장되기 때문에 효율적이고 암호학적으로 안전한 방법으로이 데이터를 검색 할 수 있으므로
전체 블록 체인 ("라이트 클라이언트")을 다운로드하지 않는 네트워크 피어에서도 로그를 찾을 수 있습니다

저장영역 storage

블록체인에 영구적으로 기록, 모든 어카운트는 별도의 스토리지를 보유

- 각각의 계정은 storage 를 가지고 있고 256비트 단어와 256비트 단어를 매핑한다.
- memory는 32byte chunk단위로 읽거나 쓰여진다. 메모리는 크기가 커질수록 비용이 커진다.
- EVS는 stack 머신으로 stack 이라 불리는 연산 영역에 저장된다.
stack영역은 1024개의 요소를 지닐 수 있고, 각각의 요소는 256비트의 단어를 포함한다.

메모리

함수 호출, 메모리 연산을 수행할 때 임시로 사용하는 영역
메시지 호출이 발생할 때마다 초기화된 메모리영역이 컨트랙트에 제공

데이터 위치별 가스비용

- 스토리지 : 32바이트당 20000가스
- 로그 : 32바이트당 96가스
- 콜 데이터 : zero 바이트 4가스, non-zero 바이트 68 가스
- 메모리 : 32바이트당 1가스

delegatecall / call code, library

호출은 대상 주소의 코드가 호출 계약의 컨텍스트에서 실행되고
msg.sender와 msg.value가 변경되지 않는다는 점을 제외하고는 메시지 호출과 동일합니다
(solidity library 기능)

내부 함수 호출

EVM의 JUMP, 매개변수 EVM 스택

외부 함수 호출

EVM의 CALL 데이터 영역에 기록

public

EOA가 함수를 호출할 때 매개변수를 메모리로 복사하기 때문에
메모리에 대한 비용을 지급해야 한다
external 함수라면 콜 데이터에 대한 비용만 지급하면 된다

참고

[이더리움 가상머신](#)
[ethereum internal](#)
[evm jello paper](#)