

## Hyperledger Fabric

### 1. 기능

#### A. Events

- i. Validating peers와 Chaincode는 블록체인 네트워크에서 이벤트를 발생시킬 수 있다.
- ii. 이벤트가 발생하게 되면 대기하고 있는 어플리케이션에 필요한 notice를 보낸다.
- iii. 이벤트는 미리 정의된 이벤트 혹은 커스텀 이벤트가 있으며, 이벤트 수신을 위해서는 하나 이상의 adapter가 필요하다.
- iv. Web hooks나 Kafka등을 이용한 이벤트 수신도 가능하다.

#### B. API

- i. 기본 인터페이스 : REST API
- ii. API 호출을 통해 사용자 등록, 블록체인에 대한 쿼리, 트랜잭션 발생이 가능하다.
- iii. 체인코드와 상호작용하기 위한 API를 통해서 트랜잭션을 관리 할 수 있다.

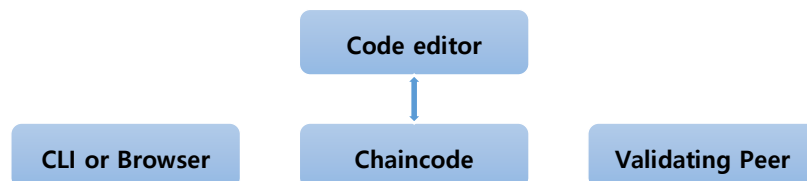
#### C. CLI

- i. REST API의 일부 기능을 지원하는 CLI 기능을 통해서 체인코드의 deploy 및 트랜잭션 처리 등을 빠르게 진행할 수 있도록 한다.
- ii. CLI는 Golang으로 제작이 되었으며 OS 다양성을 보장한다.

### 2. Network Topology

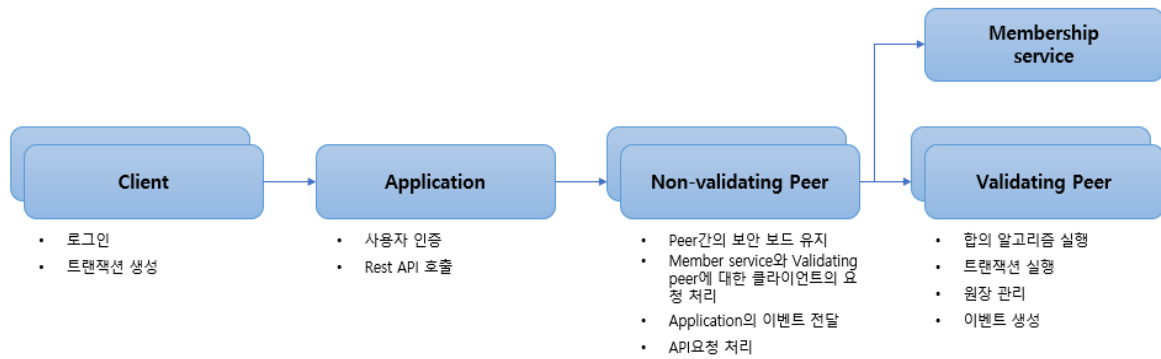
#### A. Single Validating Peers

- i. 기능적으로 non-validating peer는 validation peer의 서브셋이므로 모든 non-validating peer의 기능은 validating peer에서 사용 가능하다.
- ii. 가장 간단한 네트워크 구성은 하나의 validating peer로 구성된 블록체인 네트워크이다.
- iii. 이 토폴로지는 개발을 위한 환경으로서 주로 이용되며 하나의 peer로 구성되어 있기 때문에 합의 알고리즘을 사용할 수 있다. 그래서 기본적으로 설정되어 있는 합의 모듈인 noops 플러그인을 적용해서 사용한다.



## B. Multiple Validating Peers

- i. 실제 서비스 운영 환경이거나 테스트 환경일 경우 다양한 validating peer와 non-validating peer로 블록체인 네트워크를 구성해야 한다.
- ii. Non-validating peer는 이벤트 처리 및 REST API 서비스 관리 등의 역할을 한다.
- iii. Validating peers들은 발생하는 모든 이벤트, 트랜잭션 등의 데이터를 공유한다.



## iv. Multichain

