1. 하이퍼레저 패브릭 모델

* Assets: 자산 정의는 네트워크를 통해 화폐가치가 있는 거의 모든 것을 교환할 수 있게 한다. 전체 음식에서 골동품 자동차로, 통화 선물로. 체인코드 트랜잭션을 사용해 자산을 수정할 수 있는 기능을 제공한다. 자산은 키-값 쌍의 집합으로 패브릭에 표시되며 상태 변경은 채널 원장에 트랜잭션으로 기록된다. 자산은 이진 및/또는 JSON 형태로 나타낸다.
* Chaincode: 체인코드 실행은 거래 순서로부터 분할되어 노드 유형에 걸쳐 필요한 신뢰와 검증의 수준을 제한하고, 네트워크 확장성과 성능을 최적화한다. 체인코드는 자산이나 자산을 정의하는 소프트웨어와 자산을 사정하기 위한 거래 지침이다. 체인코드는 키-값 쌍 또는 기타 상태 데이터베이스 정보를 읽거나 변경하기 위한 규칙을 시행한다. 체인코드 기능은 원장의 현재 상태 데이터베이스에 대해 실행되며, 거래 제안을 통해 개시된다. 체인코드를 실행하면 네트워크에 제출되어 모든 피어의 원장에 적용할 수 있는 일련의 키 값 세트 (write set) 가 발생한다.
* Ledger Features: 상호작용하고, 공유된 원장은 각 채널 별 전체 트랜잭션 히스토리로 인코드 되고, SQL과 같은 효율적인 감사 및 논쟁 쿼리 가용성을 포함한다. 모든 상태 전환에 대한 서열화되고 변조 방지 기능이 있는 기록이다. 상태 전환은 참여자들이 제출한 체인코드 호출 (‘트랜잭션’)의 결과물이다. 각 거래는 생성, 업데이트 또는 삭제 시 장부에 커밋되는 일련의 자산 키-값 쌍을 낳는다. 원장에는 불변하고 서열화된 레코드를 저장하는 블록체인(‘체인’) 과 현재의 패브릭 상태를 유지하기 위한 상태 데이터베이스로 구성되어 있다. 한 채널에 한 개의 대장이 있다. 각 피어는 자신이 회원인 각 채널에 대해 원장 사본을 유지한다.
  + 키 기반 조회, 범위 쿼리 및 복합 키 쿼리를 사용한 조회 및 업데이트 원장
  + 리치 쿼리 언어를 사용하는 읽기 전용 쿼리(CouchDB를 상태 데이터베이스로 사용하는 경우)
  + 읽기 전용 기록 쿼리 - 키에 대한 원장 기록 쿼리, 데이터 증명 시나리오 지원
  + 트랜잭션은 체인 코드(읽기 세트)에서 읽은 키/값과 체인 코드(쓰기 세트)로 작성된 키/값 버전으로 구성된다.
  + 트랜잭션은 모든 승인 피어의 서명을 포함하고 주문 서비스에 제출됨
  + 트랜잭션이 블록으로 주문되고 주문 서비스에서 채널의 피어로 "배달"됨
  + 또래는 승인 정책에 대해 트랜잭션을 검증하고 정책을 시행함
  + 블록을 추가하기 전에 체인코드 실행 시간 이후 읽은 자산의 상태가 변경되지 않았는지 버전 확인을 수행한다.
  + 트랜잭션이 검증되고 커밋되면 불변성이 있음
  + 채널의 장부는 정책, 액세스 제어 목록 및 기타 관련 정보를 정의하는 구성 블록을 포함한다.
  + 채널에 다른 인증 기관에서 암호 자료를 추출할 수 있는 Membership Service Provider 인스턴스가 포함됨
  + 데이터베이스, 스토리지 구조 및 "쿼리 기능"에 대한 자세한 내용은 Leder 항목을 참조하십시오.
* Privacy: 채널과 private 데이터 모음은 개인적이고 기밀적인 멀티-측면의 트랜잭션을 가능하게 한다. 주로 같은 네트워크에서 자산을 교환하는 비즈니스 경쟁이나 규제 산업에서 요구되는 트랜잭션
* Security & Membership Services – 허가형 멤버십은 신뢰적인 블록체인 네트워크를 제공한다. 참가자들이 권한 있는 규제자와 감사자들에게 발견되고 추적되는 모든 트랜잭션을 알 수 있는 네트워크
* Consensus: 합의를 위한 유일한 접근은 기업에게 필요한 융통성과 가용성을 제공한다. 분산원장 기술에서, 컨센서스는 최근 단일 기능 내에서 특정 알고리즘과 동의어가 되었다. 그러나 합의는 단순히 거래 순서에 합의하는 것 이상을 포괄하며, 제안과 승인에서 주문, 검증 및 약속에 이르는 전체 거래 흐름에서 그것의 근본적인 역할을 통해 하이퍼레저 패브릭에서 이러한 차별화가 강조된다. 간단히 말해서, 합의는 블록을 구성하는 일련의 거래의 정확성에 대한 전체 순환 검증으로 정의된다. 블록의 거래 순서와 결과가 명시적인 정책 기준 점검을 충족했을 때 궁극적으로 합의가 이루어진다.

1. 애플리케이션 개발

Org1: Doctors (Read/Write)

Org2: Patients (Only Read), Peer 하나 (서비스 위한 노드)

Ordering node

두 Org가 한 채널 공유, 체인코드 존재

트랜잭션 구조 (Key-value): 이미지 해시 테이블, 타임스탬프, 소유권자