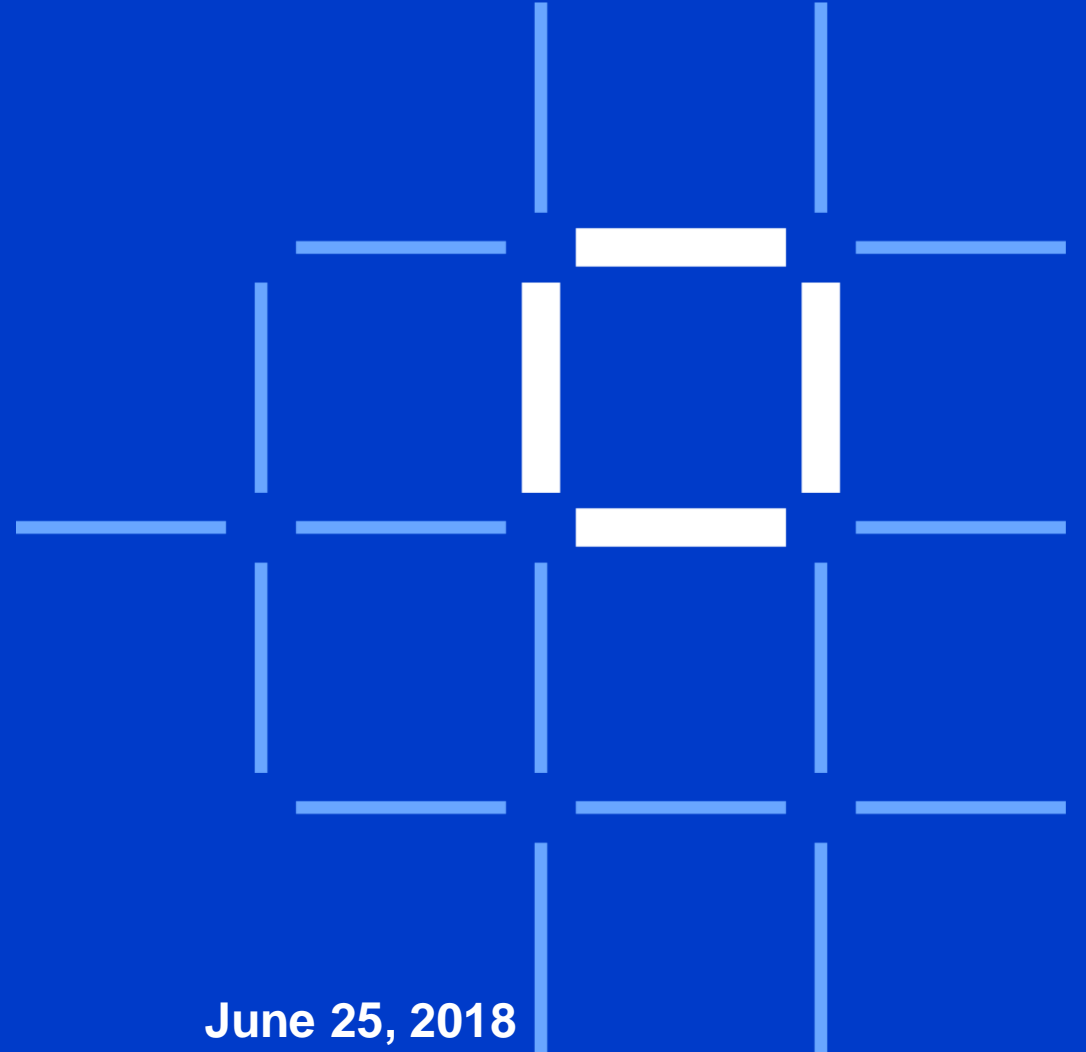


KRnet2018

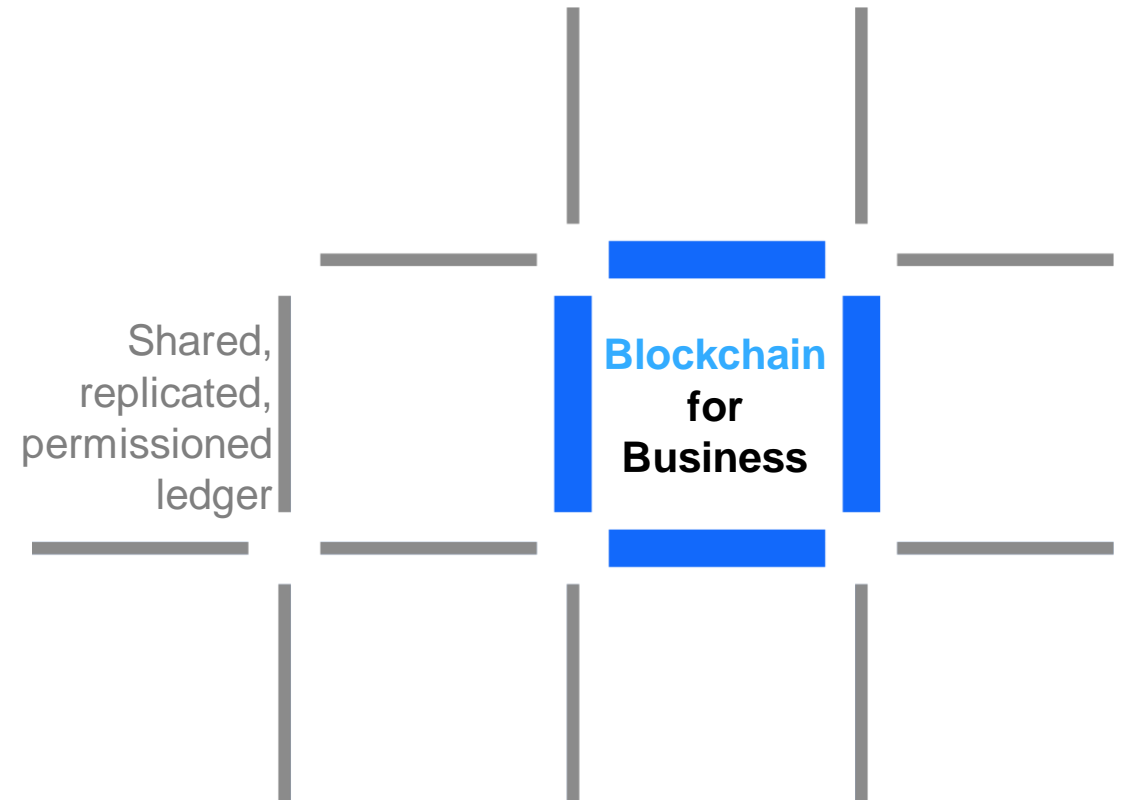
Hyperledger Fabric Deep dive



June 25, 2018














Agenda

1. Hyperledger Fabric V1
2. 글로벌 구축사례 및 시사점





블록체인 기술은 비즈니스 목적에 따라 서로 다른 기술들로 구현됨으로써 퍼블릭 블록체인과 허가형 블록체인 영역으로 나누어짐

	퍼블릭(Public) 블록체인	허가형(Permissioned) 블록체인
암호화폐 기반	<p>암호화폐 가치가 동력</p> <div> ETHEREUM</div>	<p>비즈니스 사용을 위한 암호화폐</p> <div> </div>
암호화폐 기반이 아님		<p>비즈니스를 위한 블록체인 기술</p> <div>  Quorum™</div> <div><p>산업표준단체</p> </div>



- 리눅스 재단의 Hyperledger 프로젝트는 2015년 12월 17일에 17개 회원사로 시작되었으며, 현재 200개 이상의 회원사가 참여하고 있음
- Hyperledger 프로젝트는 전세계적으로 비즈니스 거래가 수행되는 방식을 변혁할 수 있는 분산 원장에 대한 산업 표준에 중요한 기능들을 확인하고 적용하여 블록체인을 발전시키기 위한 협력 프로젝트임
- 오픈 소스, 오픈 표준, 오픈 거버넌스 기반
- 1개의 Active 프레임워크("Fabric")과 4개의 인큐베이터

**Enable adoption of shared ledger technology at
a pace and depth not achievable by any one
company or industry**



Brian Behlendorf
Executive Director



Blythe Masters
Board Chair



Chris Ferris
TSC Chair

www.hyperledger.org

1

Hyperledger Fabric V1

Hyperledger 회원

Premier



Associate



General





Infrastructure
Technical, Legal, Marketing,
Organizational

Ecosystems that accelerate
open development and
commercial adoption

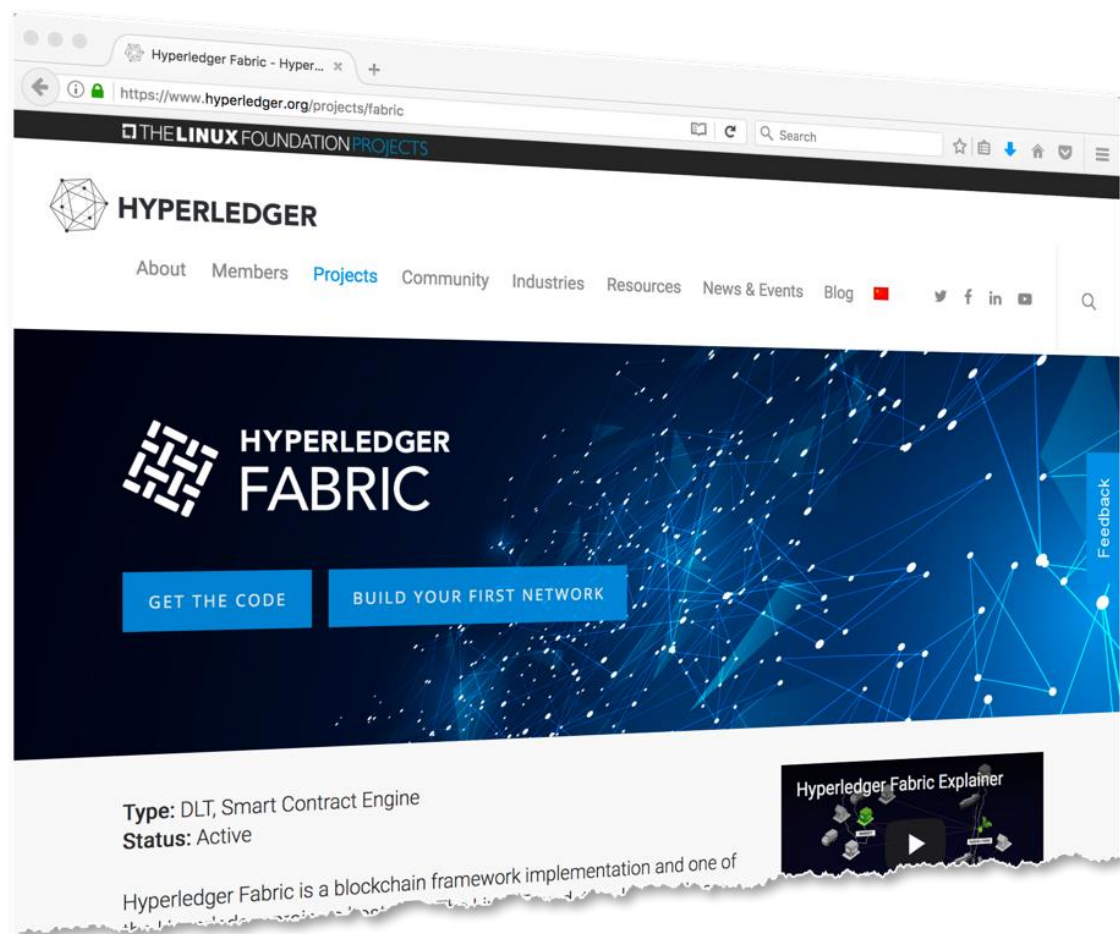


Frameworks
Meaningfully differentiated approaches to
business blockchain frameworks
developed by a growing community of
communities from the entire industry



Tools
Typically built for one framework, and through
common license and community of
communities approach, ported to other
frameworks





- An implementation of blockchain technology that is a foundation for developing blockchain applications
- Emphasis on ledger, smart contracts, consensus, confidentiality, resiliency and scalability.
- V1.1 released March 2018
 - Includes significant performance, security, migration and smart contract improvements
- IBM is one of the many contributing organizations

<http://hyperledger-fabric.readthedocs.io/>

Hyperledger Fabric Roadmap



V1 Alpha

- Docker images
- Tooling to bootstrap network
- Fabric CA or bring your own
- Java and Node.js SDKs
- Ordering Services - Solo and Kafka
- Endorsement policy
- Level DB and Couch DB
- Block dissemination across peers via Gossip

V1 GA

- Hardening, usability, serviceability, load, operability and stress test
- Chaincode ACL
- Chaincode packaging & LCI
- Pluggable crypto
- HSM support
- Consumability of configuration
- Next gen bootstrap tool (config update)
- Config transaction lifecycle
- Eventing security
- Cross Channel Query
- Peer management APIs
- Documentation

V1.1

- Node.js smart contracts
- Node.js connection profile
- Smart Contract APIs:
 - Encryption library
 - Txn submitter identity
 - Access control (using above)
- Performance & Scale
 - More orderers at scale
 - Parallel txn validation
 - CouchDB indexes
- Events
 - Per channel vs global
 - Block info minimal events
- CSR for more secure certs
- Serviceability
 - Upgrade from 1.0
- **Technical Preview features**
 - Private channel data
 - Finer grained access control on channels (beyond orgs)
 - ZKP features (ID Mixer)
 - Java for Smart contracts

V1.2

- V1.1 Technical Preview features
- Chaincode lifecycle improvements
- Usability Features
 - e.g. Service discovery
- Technical Debt/Hygiene
 - e.g. testing frameworks
 - Parallel testing
 - More modular code
- (Other candidates)

Based on <https://wiki.hyperledger.org/projects/fabric/roadmap>

March 2017

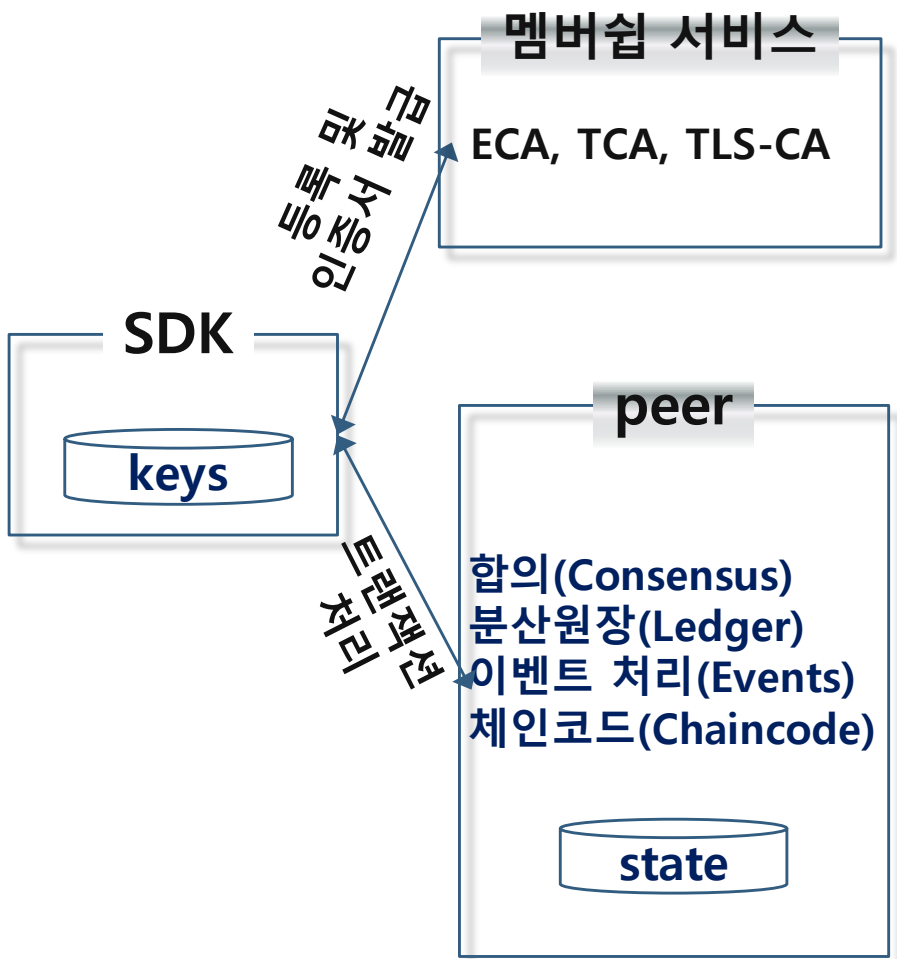
July 2017

March 2018

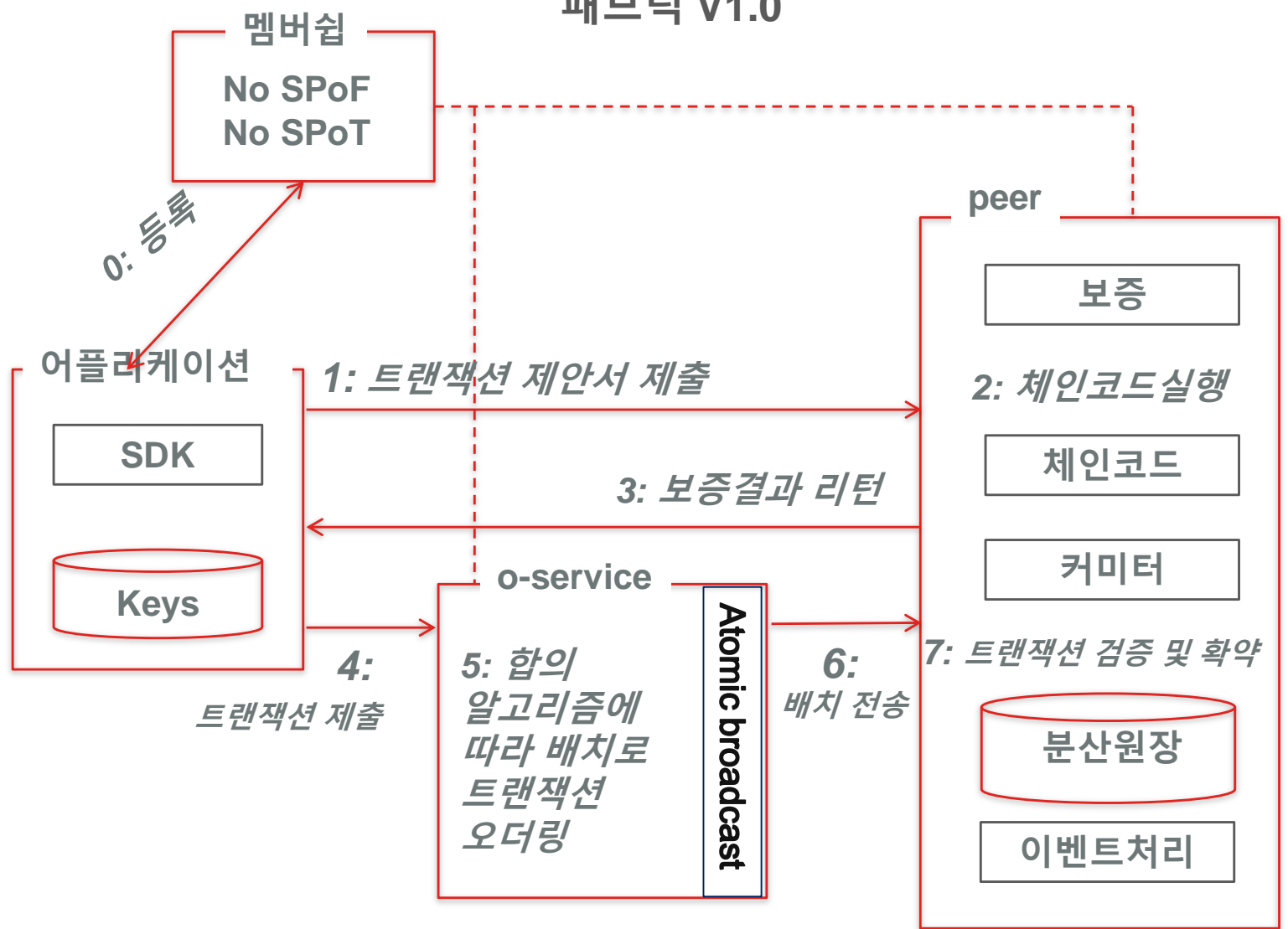
June 2018 (quarterly)

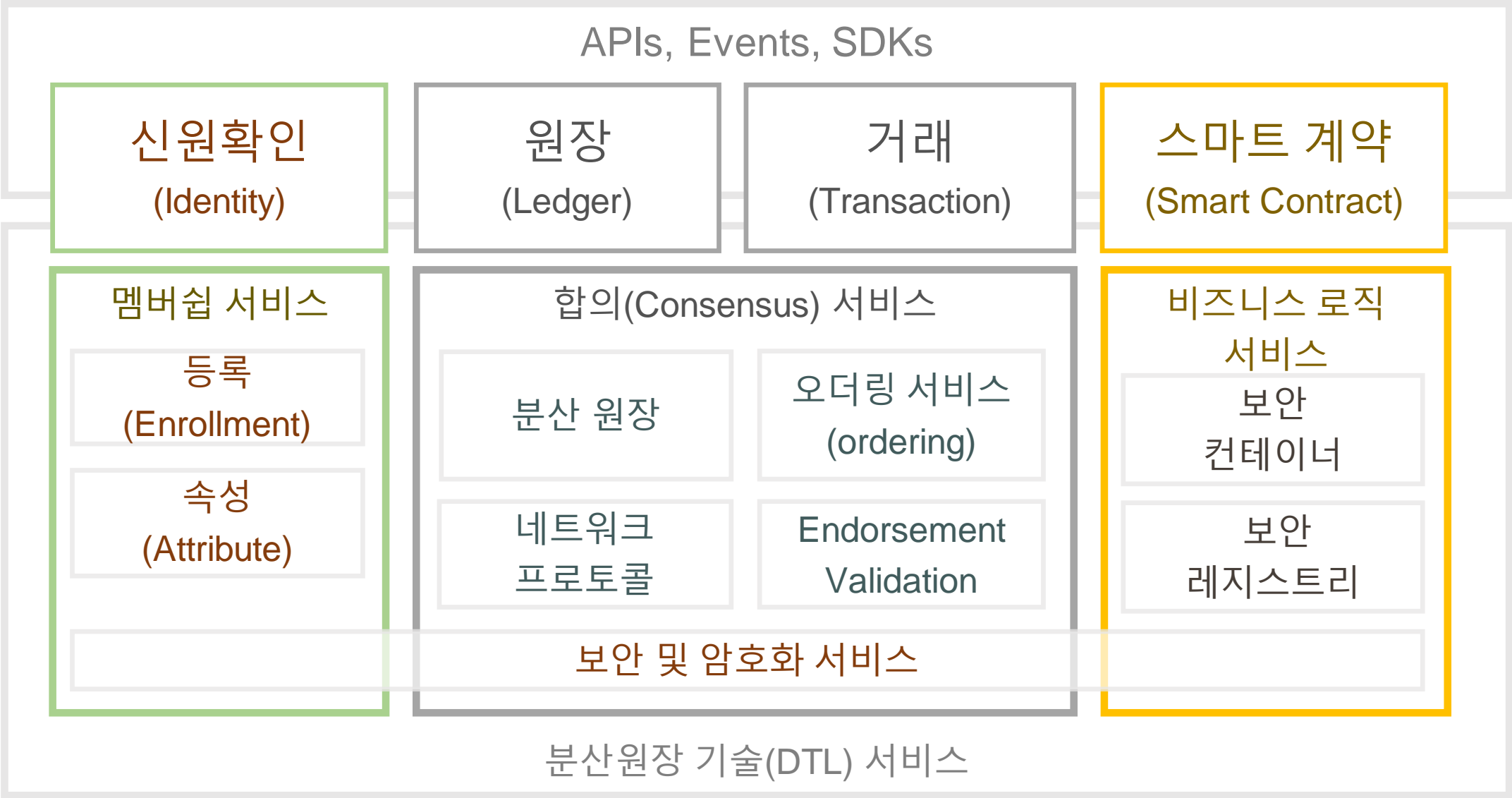
Fabric 아키텍처 비교

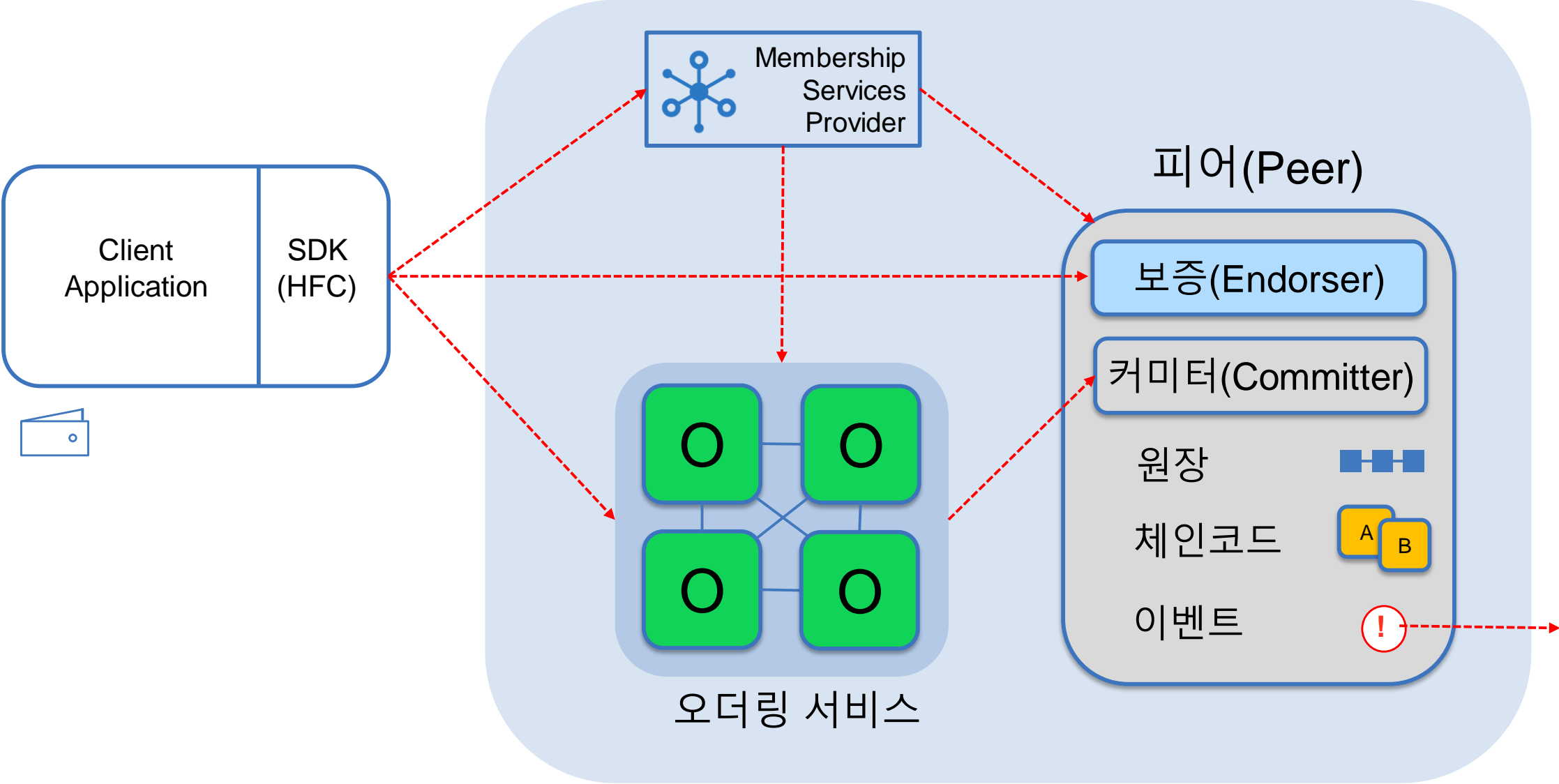
패브릭 V0.6



패브릭 V1.0



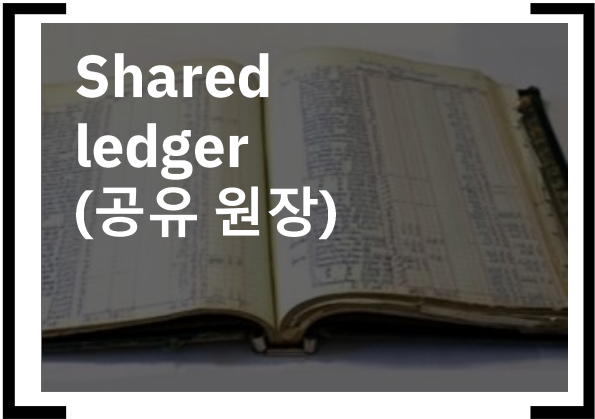






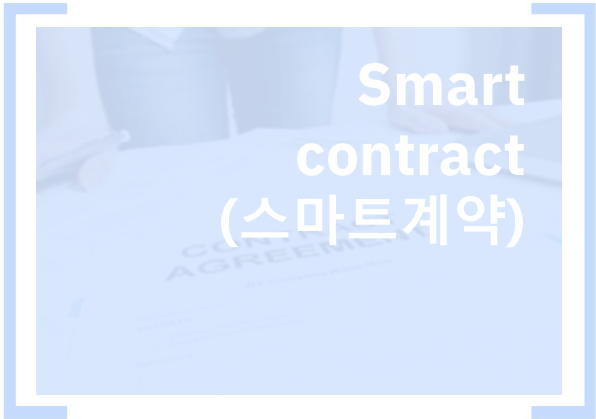
1

비즈니스
네트워크내에 모든
거래가 기록되고
공유됨



2

비즈니스 규칙 및
로직은 계약에
함축되어 트랜잭션
수행시 실행됨



원장은 공유되지만,
참여자의 개인정보는
암호화 기술을
통해서 보호되어야 함

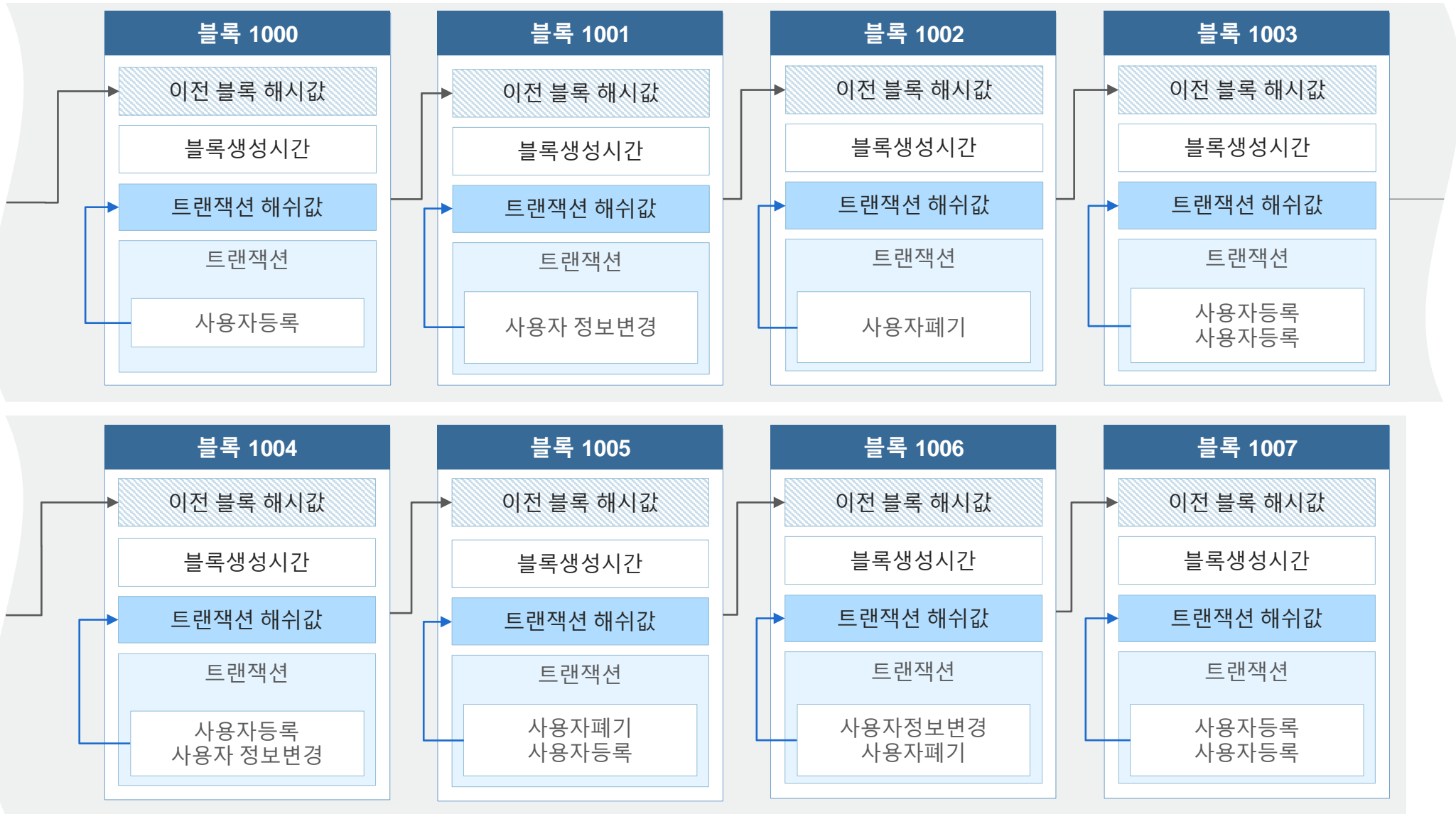


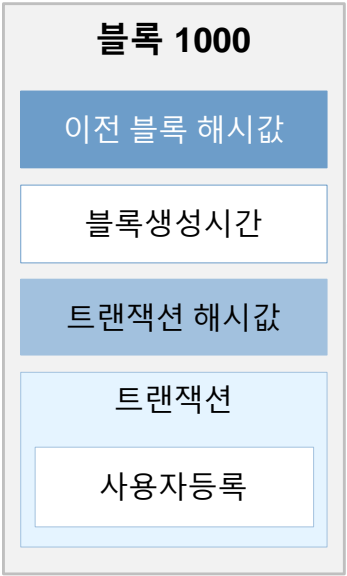
3

검증된 트랜잭션에
대한 네트워크에
참여한 참여자의
동의가 필요함



4





```
{
  "transactions": [
    {
      "type": 3,
      "chaincodeID": "EoABNmU5MGVkJNjQzZTg4OGM3OTFIZT . . .",
      "payload": "CqsBCAESgwESgAE2ZTkWZWQ2NDNIODg4Yzc5MWVIMGE1MjgzND . . .",
      "uuid": "5edbcc1c-41f1-47bd-a21f-fa0b6912c668",
      "timestamp": {
        "seconds": 1457340858,
        "nanos": 829120056
      },
      "cert": "MIIB/zCCAYSgAwIBAgIBATAKBggqhkJOPQQDAzApMQswC . . .",
      "signature": "MGUCMDOaV0uwwNO0xZM+ . . ."
    }
  ],
  "stateHash": "FXFL5whKH8tsJIXffybWYn2enDmcCeP2OMWBywwlKn5 . . .",
  "previousBlockHash": "cb5TSolNEAGiRjpxAvYXFIQ0O2MBAD8hl9zIV3 . . .",
  "nonHashData": {
    "localLedgerCommitTimestamp": {
      "seconds": 1457340858,
      "nanos": 936184231
    }
  }
}
```

사용자등록

트랜잭션

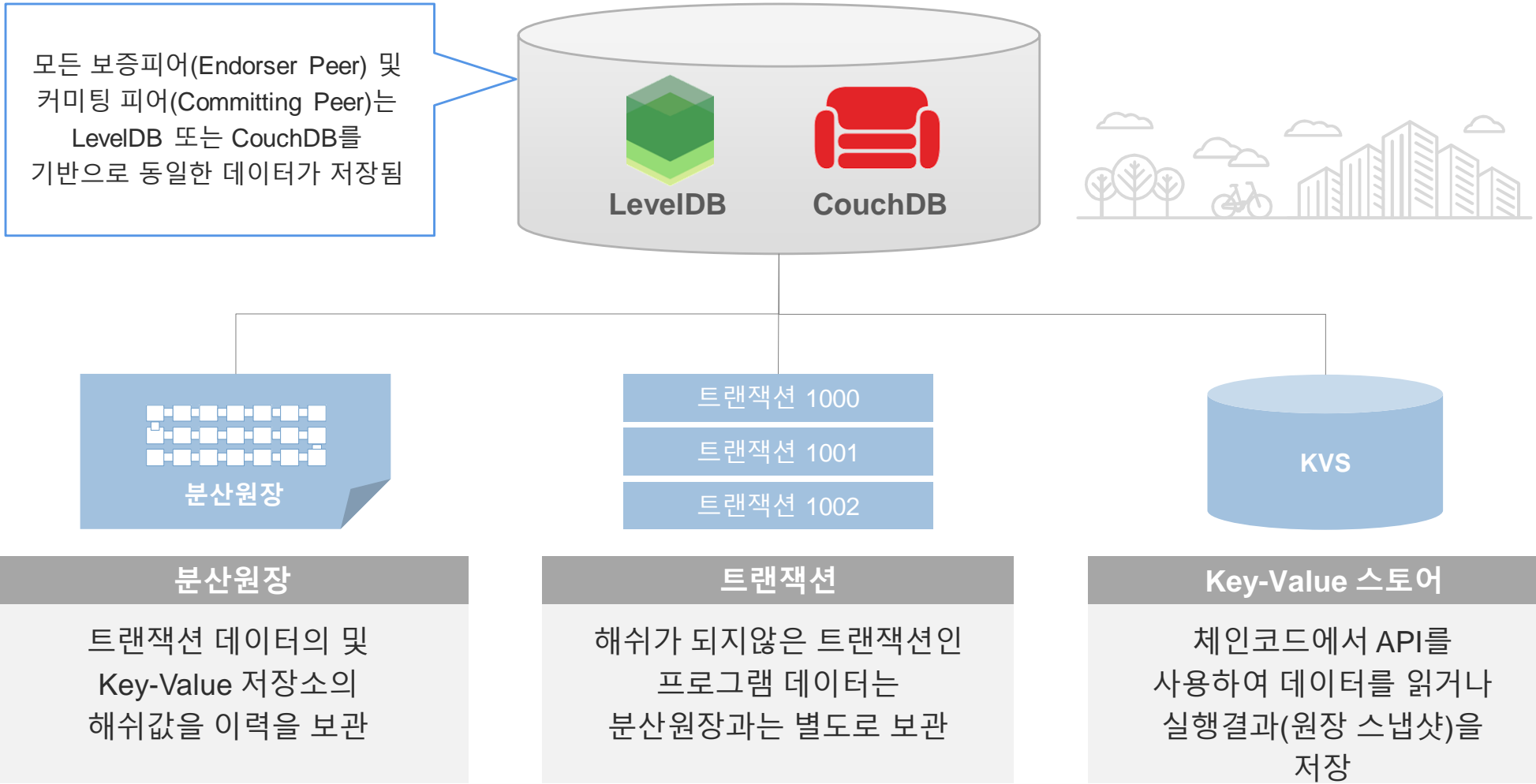
블록 1000

트랜잭션 해시값

이전 블록 해시값

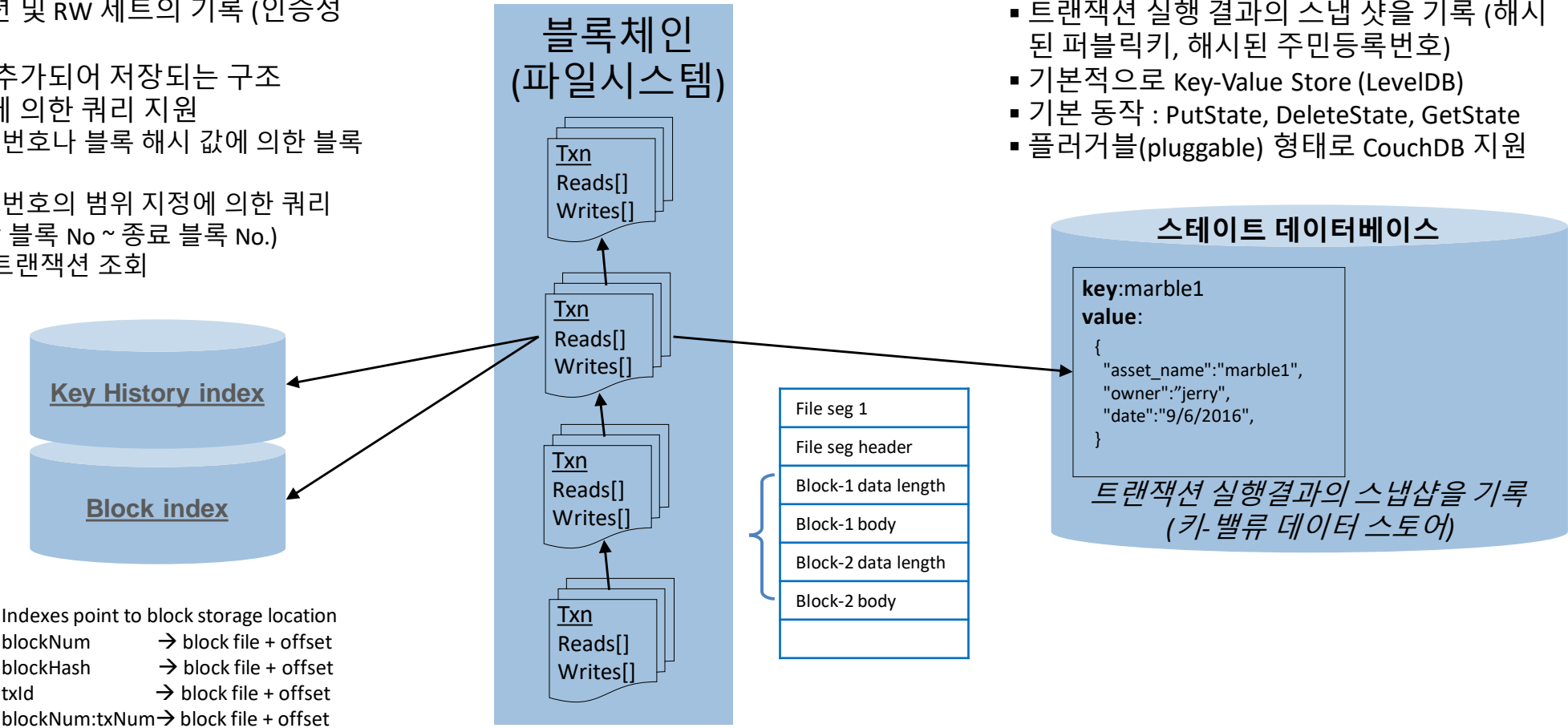
블록생성시간

검증노드(Validating Peer) 노드는 NoSQL 데이터베이스인 LevelDB나 CouchDB로 key-value 형식의 데이터베이스가 설치되어 있으며, “분산 원장”, “트랜잭션”, “체인 코드(옵션)” 데이터가 저장됨



블록 데이터 관리 (파일 시스템)

- 트랜잭션 및 RW 세트의 기록 (인증정보 등록)
- 파일에 추가되어 저장되는 구조
- 인덱스에 의한 쿼리 지원
 - ✓ 블록 번호나 블록 해시 값에 의한 블록 검색
 - ✓ 블록 번호의 범위 지정에 의한 쿼리 (시작 블록 No ~ 종료 블록 No.)
 - ✓ txId 트랜잭션 조회



스테이트 데이터베이스(State DB)

- 트랜잭션 실행 결과의 스냅 샷을 기록 (해시된 퍼블릭키, 해시된 주민등록번호)
- 기본적으로 Key-Value Store (LevelDB)
- 기본 동작 : PutState, DeleteState, GetState
- 플러그블(pluggable) 형태로 CouchDB 지원



1

비즈니스
네트워크내에 모든
거래가 기록되고
공유됨



Shared
ledger
(공유 원장)



Smart
contract
(스마트계약)

2

비즈니스 규칙 및
로직은 계약에
함축되어 트랜잭션
수행시 실행됨

원장은 공유되지만,
참여자의 개인정보는
암호화 기술을
통해서 보호되어야 함



Privacy & Security
(프라이버시 및 보안)



Consensus
(합의)

3

검증된 트랜잭션에
대한 네트워크에
참여한 참여자의
동의가 필요함

4



현대화된 암호화 원장(Ethereum, Hyperledger)들은
스마트계약이나 체인코드를 지원하는 것으로 목표로 하고 있음

A smart contract is an **event driven program**, with state, which runs on a replicated, shared ledger and which can take custody over assets on that ledger. [Swanson2015]

“Smart contract” → (replicated) state machine



참여기관
(노드의 구성)

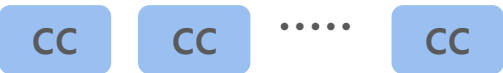


분산원장

블록체인

분산원장

- ✓블록체인 자체
- ✓트랜잭션 (스마트 계약의 처리 호출)가 로그처럼 기록됨



체인코드
(스마트 계약)

체인코드(스마트계약)

- ✓트랜잭션을 계기로 실행되는 프로그램
- ✓똑같은 체인코드가 모든 검증노드에 배포되어 샌드박스(Docker 컨테이너)에서 실행됨
- ✓사용 목적별로 여러 체인코드가 만들어짐



KVS

Key-Value Store

("chaincode state",
"world state"
이라고도 명명됨)

Key-Value Store

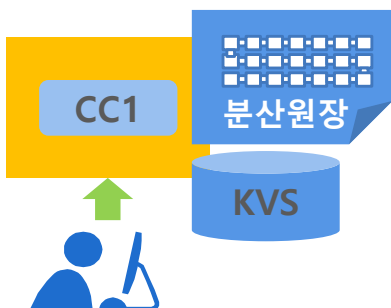
- ✓트랜잭션을 실행 한 결과 얻어지는 "최신 상태"를 기록함
 - Key = chaincode ID + cKey
 - Value = 임의의 데이터
- ✓모든 검증노드에서 동일한 내용을 갖고, 그 해쉬값이 블록 체인에 기록됨
- ✓변경 상태(자산의 이전등)가 관리됨

◆ 주요 체인코드 관련 처리 (가상머신에서 수행)



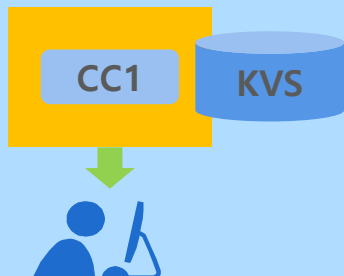
배포(Deploy)

- ✓소스코드에 따라 체인코드를 등록
- ✓원장에 기록됨(새로 배치했다는 정보가 기록됨)



호출(Invoke)

- ✓체인코드를 실행함
- ✓KVS에 데이터를 읽고 저장함
- ✓원장에 기록됨

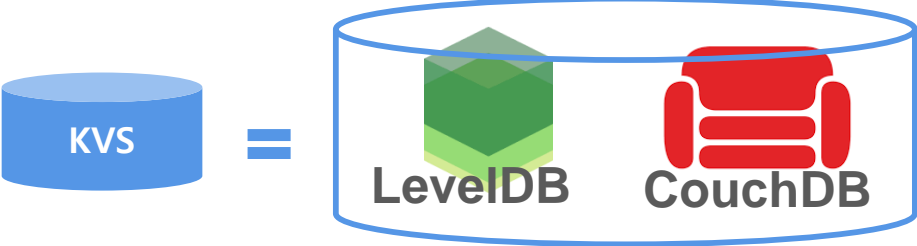


조회(Query)

- ✓체인코드에 데이터 조회함
- ✓KVS에서 데이터 읽기만 수행함
- ✓원장에 기록하지 않음



Hyperledger의 체인코드는 데이터베이스로 키(Key)와 값(Value)의 쌍으로 데이터를 등록하는 KVS 방식의 LevelDB또는 CouchDB를 채용하고 있으며, ORACLE, MySQL과 같은 관계형 데이터베이스와 비교하여 데이터베이스 스키마를 유연하게 변경할 수 있는것이 특징임.



전체 검증노드(Validating Peer)에 존재하고 동일한 데이터가 저장됨



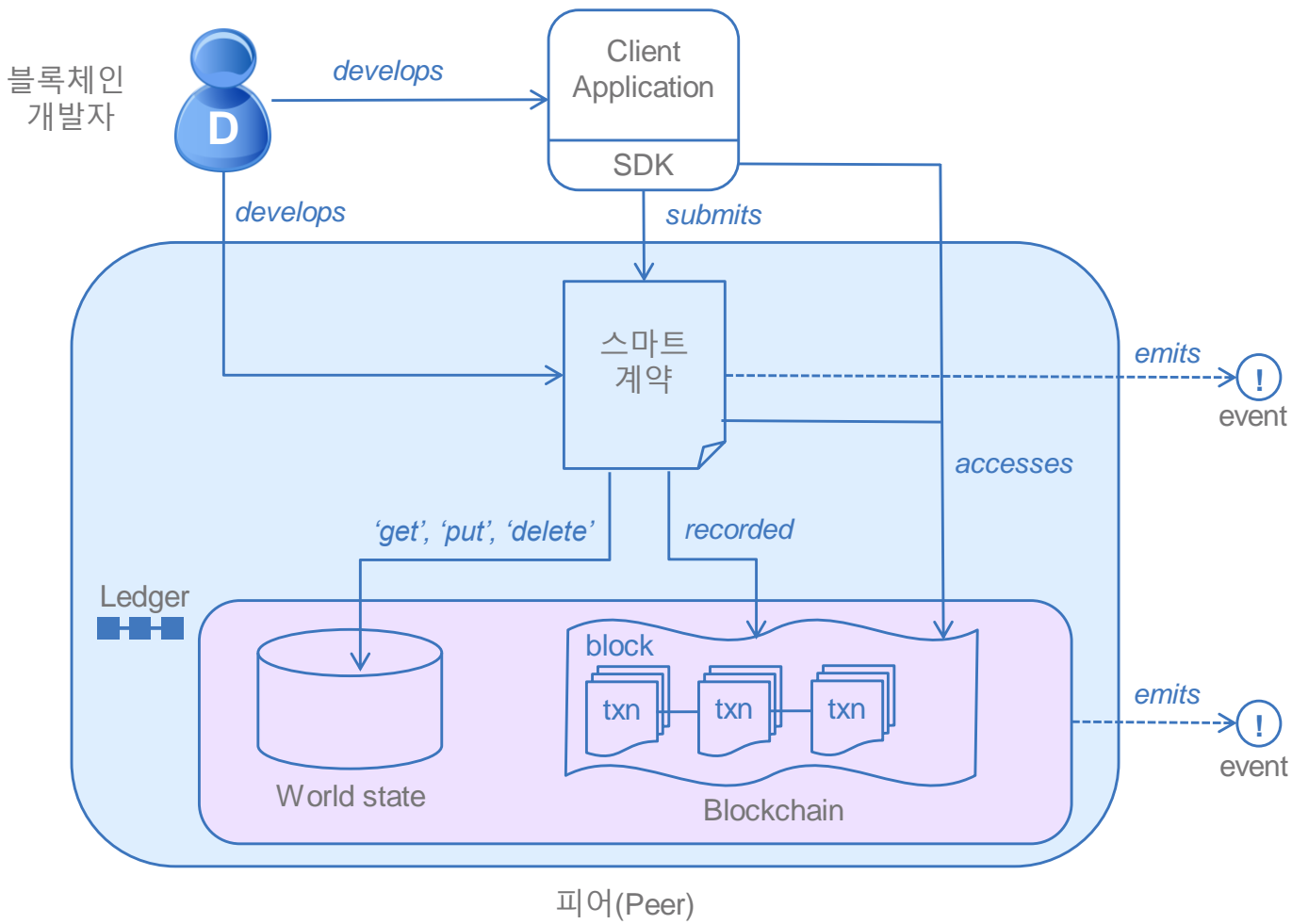
PutState ("Asset-1-Holder",
"Manufacturer")

GetState ("Asset-2-Holder")

“Lease”

키(Key)	값(Value)
CC1:Asset-1-Manufacturer	Hyundai
CC1:Asset-1-Holder	Manufacturer
CC1:Asset-1-Price	60,000,000 Won
CC1:Asset-2-Manufacturer	Toyota
CC1:Asset-2-Holder	Lease
CC1:Asset-2-Price	45,000,000 Won
.....

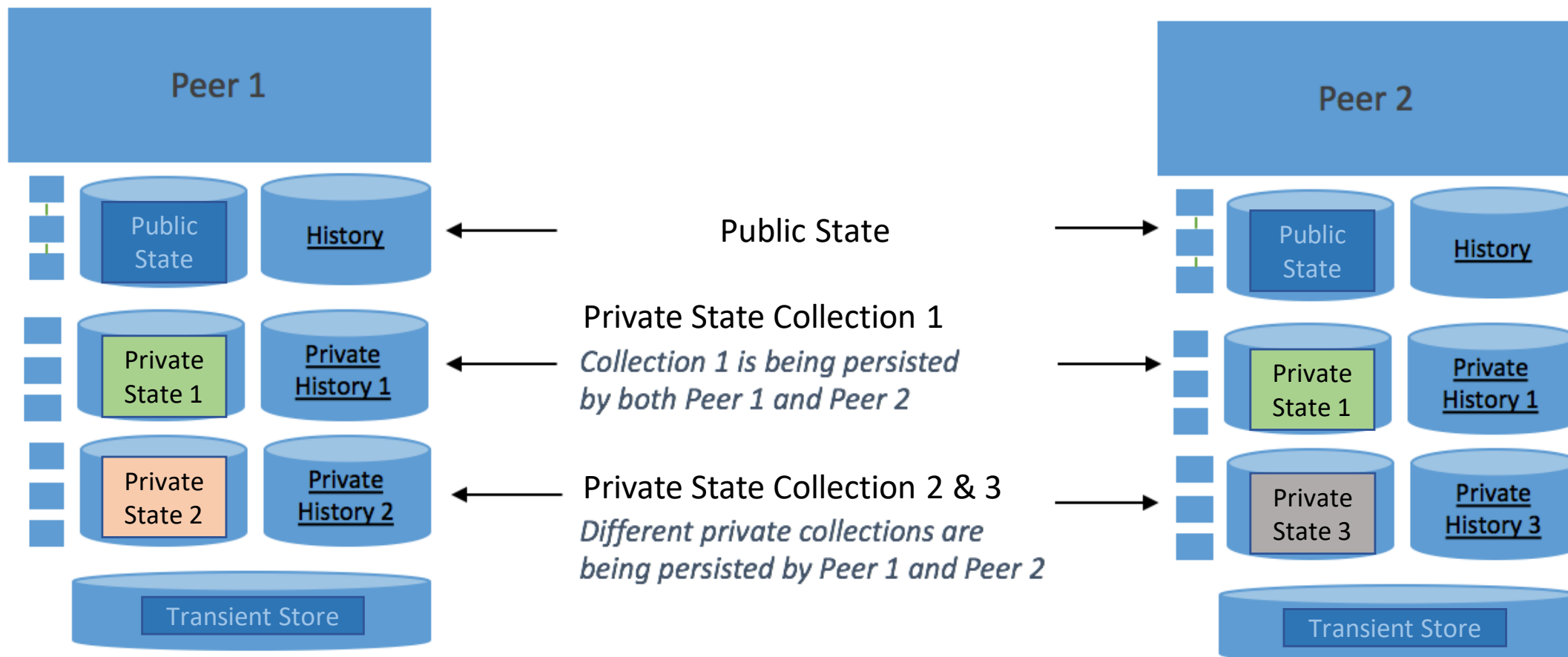
키의 체인코드의 ID가 접두어로 지정되기 때문에 다른 체인코드의 데이터끼리 충돌하는 것은 아님





- Private data is stored in a 'collection'
- Each collection has a policy which specifies which organization's peers can persist the collection data

`GetPrivateData(collection, key)`
`PutPrivateData(collection, key, value)`
`DelPrivateData(collection, key)`





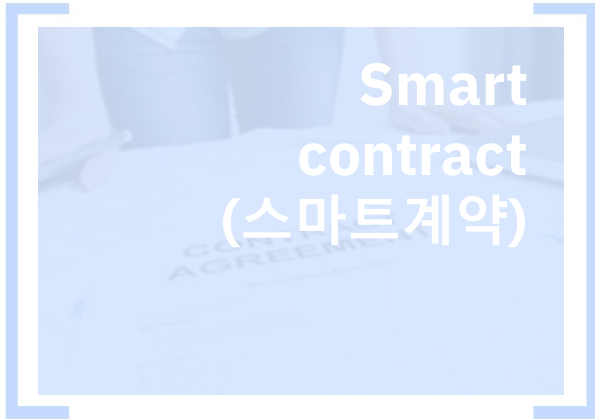
1

비즈니스
네트워크내에 모든
거래가 기록되고
공유됨



2

비즈니스 규칙 및
로직은 계약에
함축되어 트랜잭션
수행시 실행됨

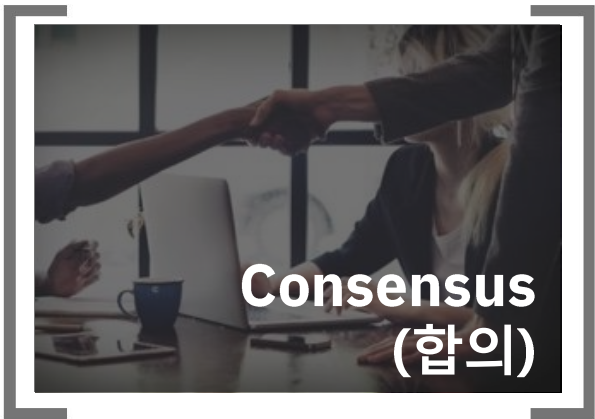


원장은 공유되지만,
참여자의 개인정보는
암호화 기술을
통해서 보호되어야 함



3

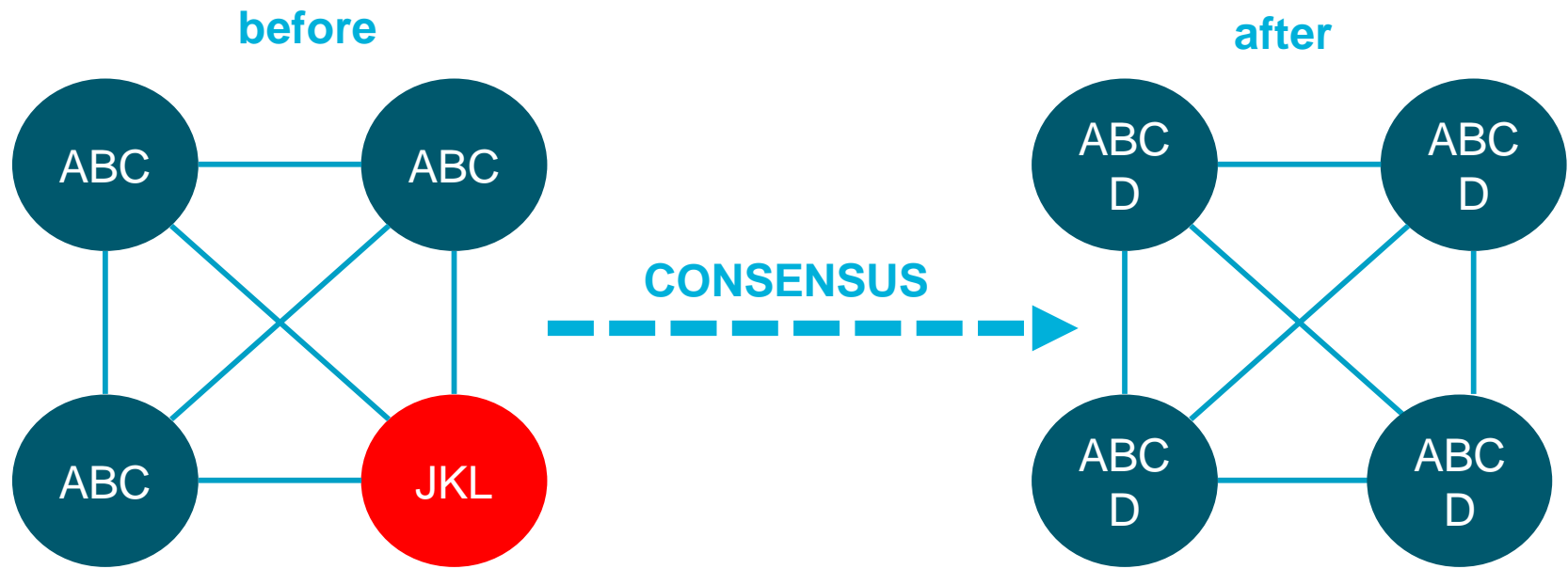
검증된 트랜잭션에
대한 네트워크에
참여한 참여자의
동의가 필요함



4

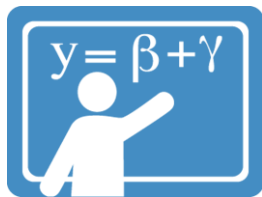


합의는 블록체인 네트워크에있는 모든 참여자의 원장(블록 및 상태)이 일관성이 있는지 확인하는 메커니즘



- 거래 및 거래 실행 순서에 대한 동의
- 동일한 원장을 유지하기 위하여 검증 참여자들의 상태를 동기화
- 거래원장이 일치하지 않는 참여자 노드의 상태 수정
- 악의적인 참여자 노드들은 격리





Proof of work



Proof of stake



Solo /
No-ops



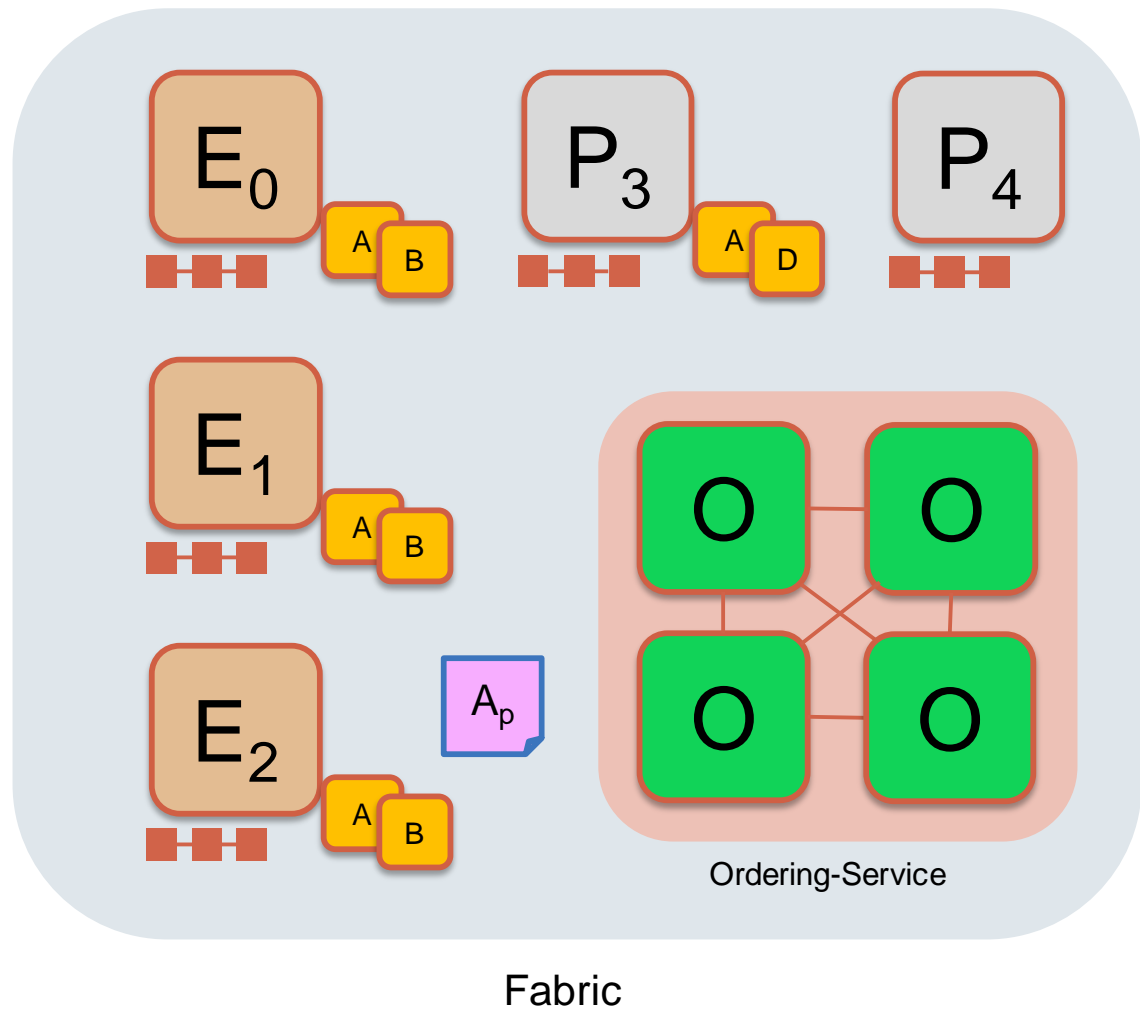
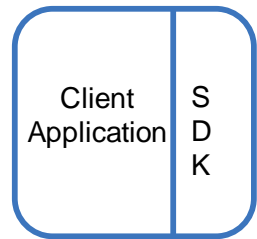
Kafka /
Zookeeper



Proof of
Elapsed Time



PBFT
based



클라이언트의 거래 제안

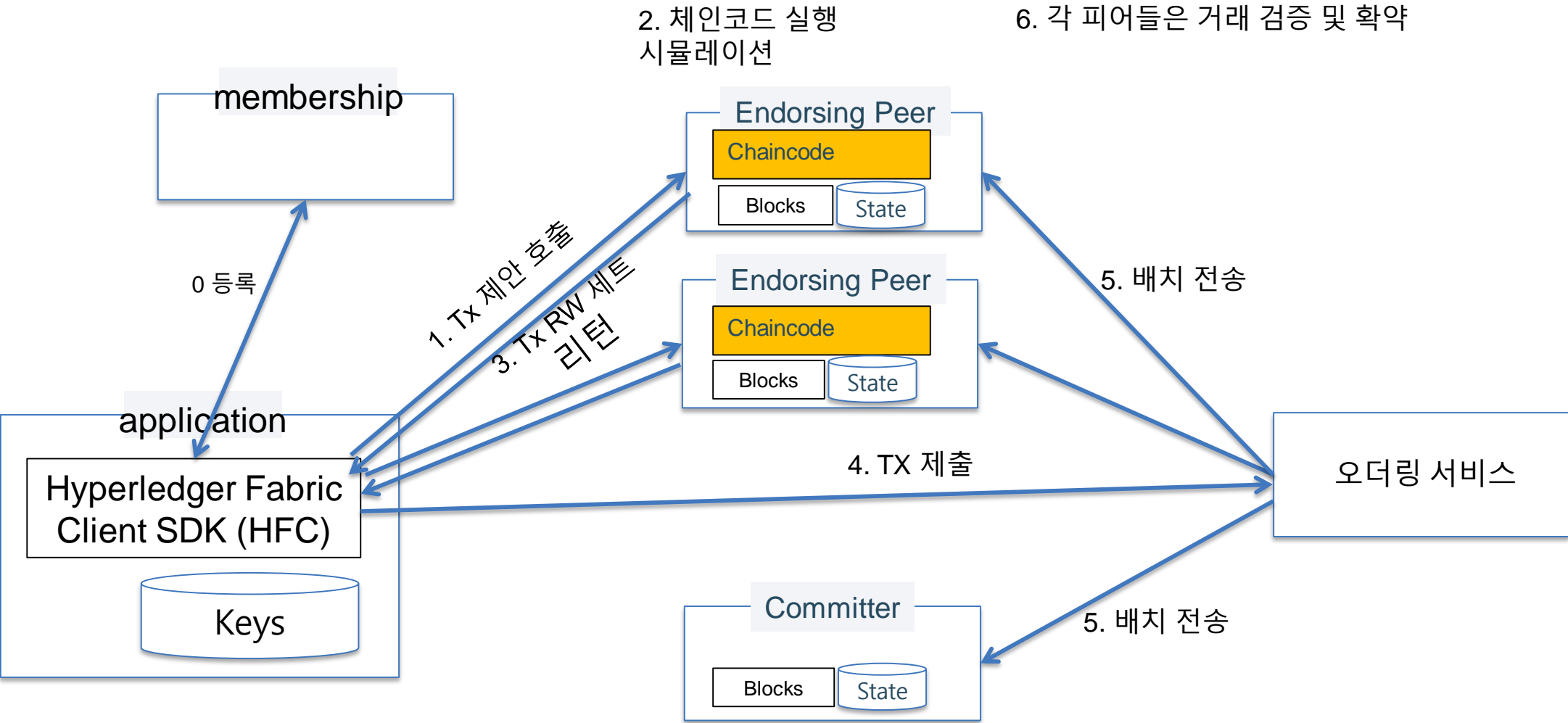
보증 정책:

- “E₀, E₁ 및 E₂ 는 반드시 전자서명”
- (P₃, P₄ 노드는 정책의 대상이 아님)

클라이언트 어플리케이션은 체인코드 A 실행을 위한 거래 제안을 하게되며, 반드시 {E₀, E₁, E₂} 노드들이 대상이 됨

Key:

Endorser			Ledger
Committer			Application
Orderer			
Smart Contract (Chain code)			Endorsement Policy





1. Tx proposal { send(Bob, Alice, 30) }

3. Endorsement {
 Read("Alice", 3)
 Read("Bob", 2)
 Write("Alice", 130)
 Write("Bob", 170)
} Sig_peer0

2. 체인코드 실행
시뮬레이션

거래수행후

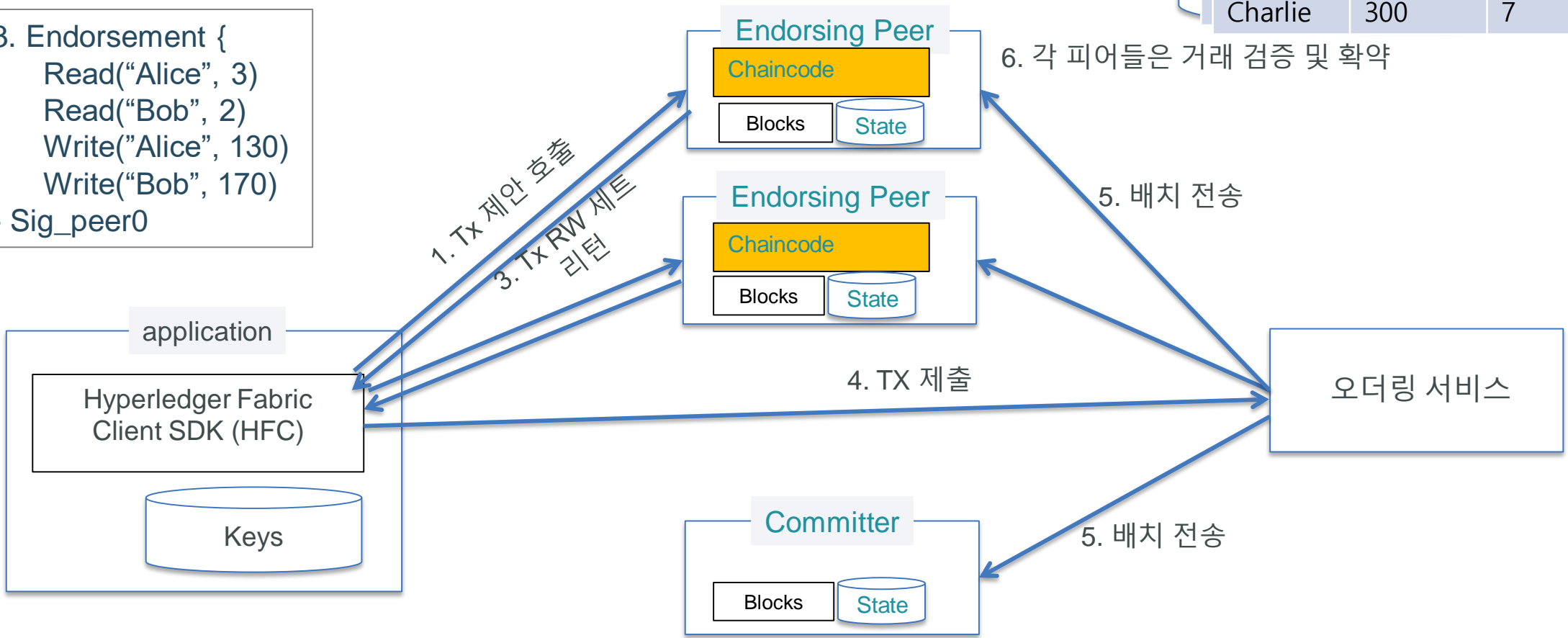
Key	Value	version
Alice	130	4
Bob	170	3
Charlie	300	7

6. 각 피어들은 거래 검증 및 확약

5. 배치 전송

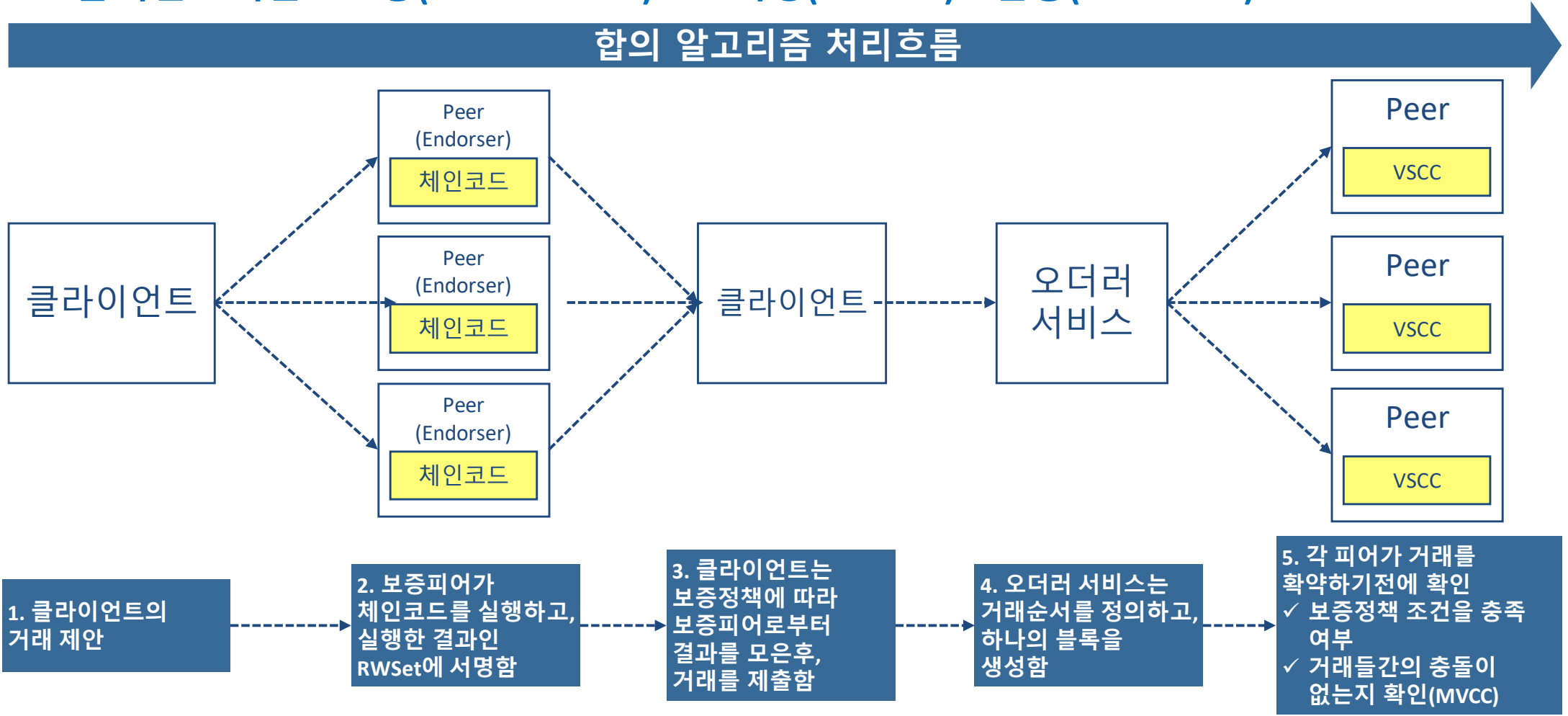
4. TX 제출

5. 배치 전송



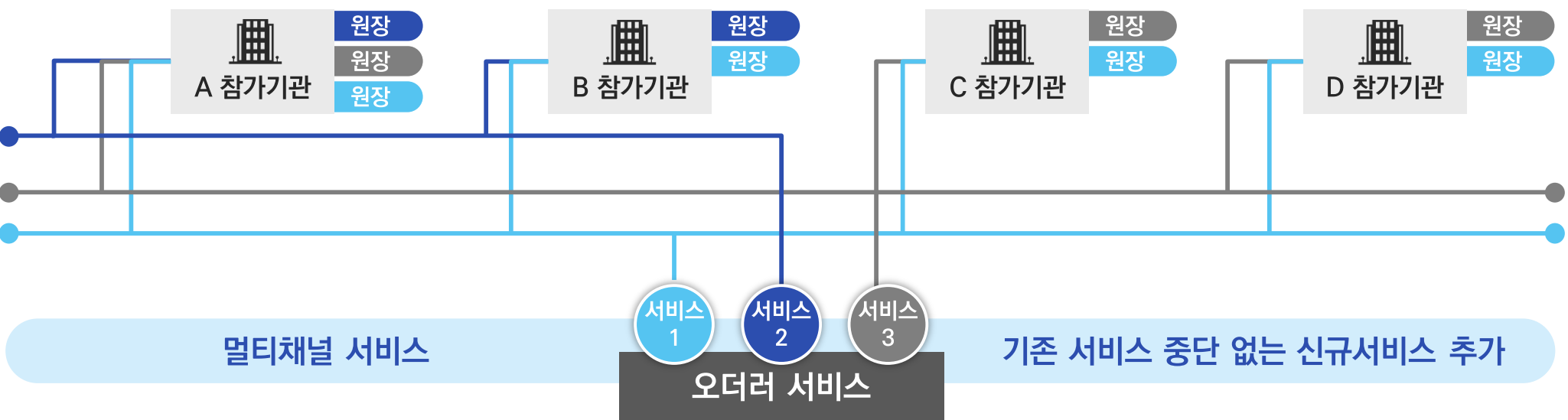


합의 알고리즘 = 보증(Endorsement) + 오더링(Orderer) + 검증(Validation)





⚙ 멀티채널 서비스 : 특정 업무와 관련이 있는 노드만 합의 과정 참여, 참여자별 서비스, 원장 구분 관리





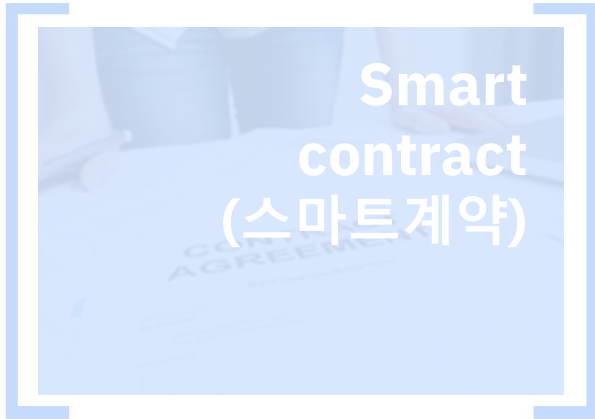
1

비즈니스
네트워크내에 모든
거래가 기록되고
공유됨



2

비즈니스 규칙 및
로직은 계약에
함축되어 트랜잭션
수행시 실행됨



원장은 공유되지만,
참여자의 개인정보는
암호화 기술을
통해서 보호됨



3

검증된 트랜잭션에
대한 네트워크에
참여한 참여자의
동의가 필요함



4



신원 관리(Identity)

- 업무 사용을 위해서는 참가자의 신원 확인이 필요
 - ✓ 책임 스푸핑(Spoofing) 방지
- 참가자의 신원인증 (참가자의 정체성과 특성의 확인)

트랜잭션의 기밀성(Confidentiality)

- 트랜잭션이 암호화되어 일반 사용자에게 보이지 않도록 함

재생(Replay) 공격 대책

- 과거의 트랜잭션을 복사하여 재전송하는 공격을 방지함

개인정보 보호

- 트랜잭션의 발행자의 익명화
- 동일한 사용자가 발행한 여러 트랜잭션들은 연관성이 없도록 함

액세스 제어

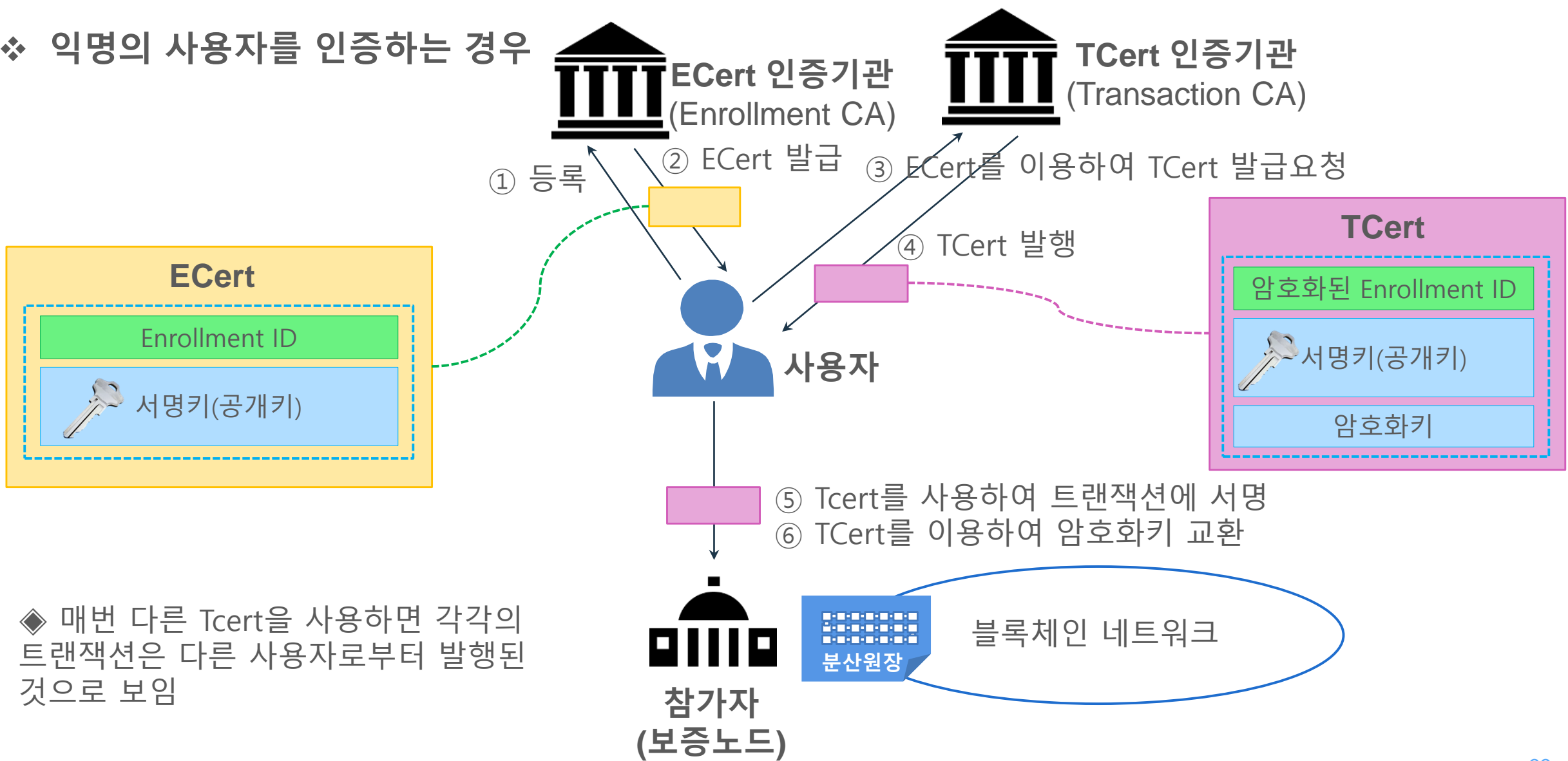
- 스마트계약을 실행할 수 있는 사람을 제한 (초기화, 함수 실행, 데이터 참조 등)

PKI 인증서와 전자서명

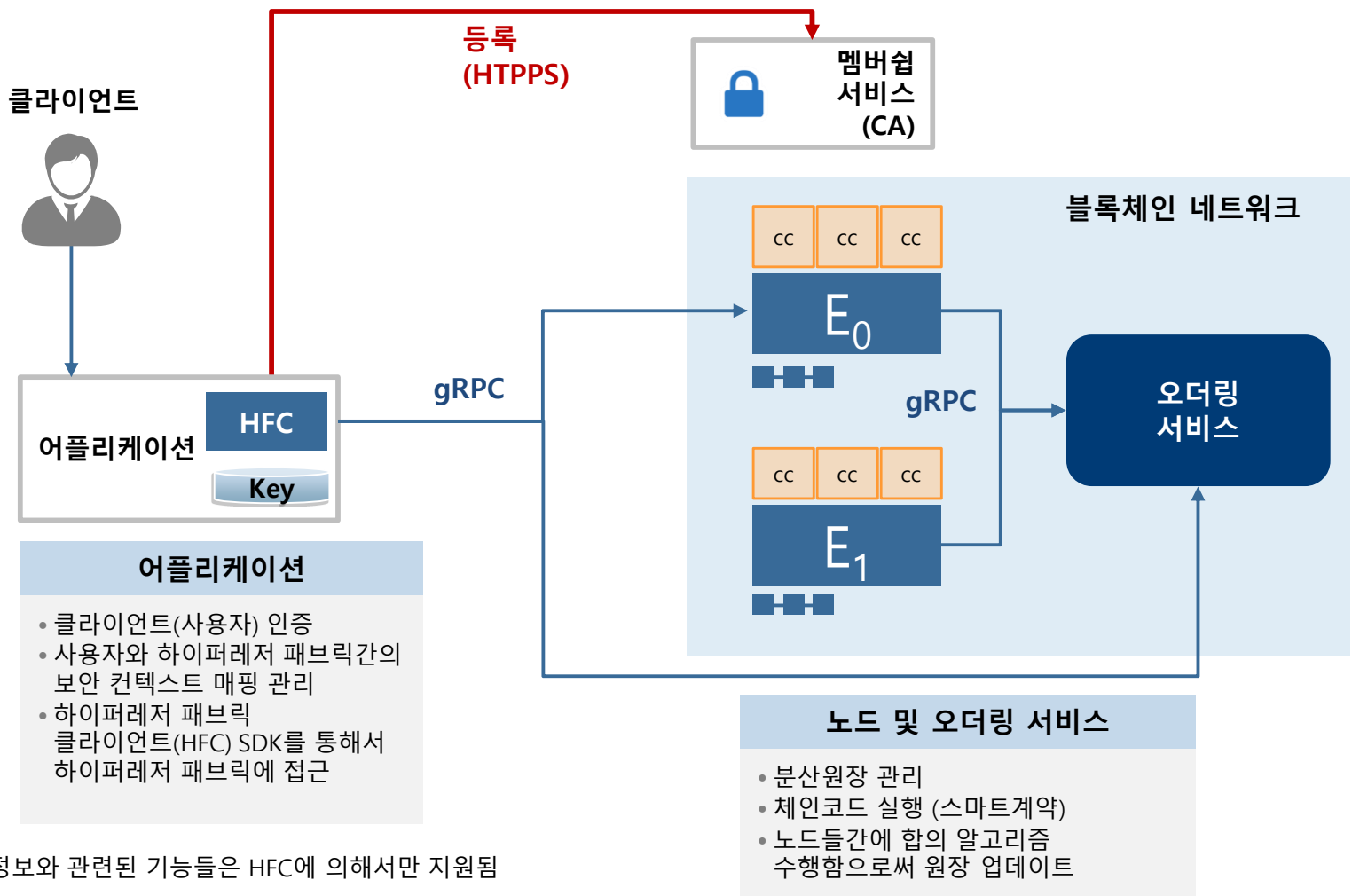
- 사용자의 신원을 확인함
- 개별 트랜잭션은 익명의 인증서를 통해 신원을 숨긴 채 인증 할 수 있음

IBM의 연구보고에 따르면, 기업용 블록체인은 허가형 블록체인 (Permissioned Blockchain)이 유일한 방법이라는 결론

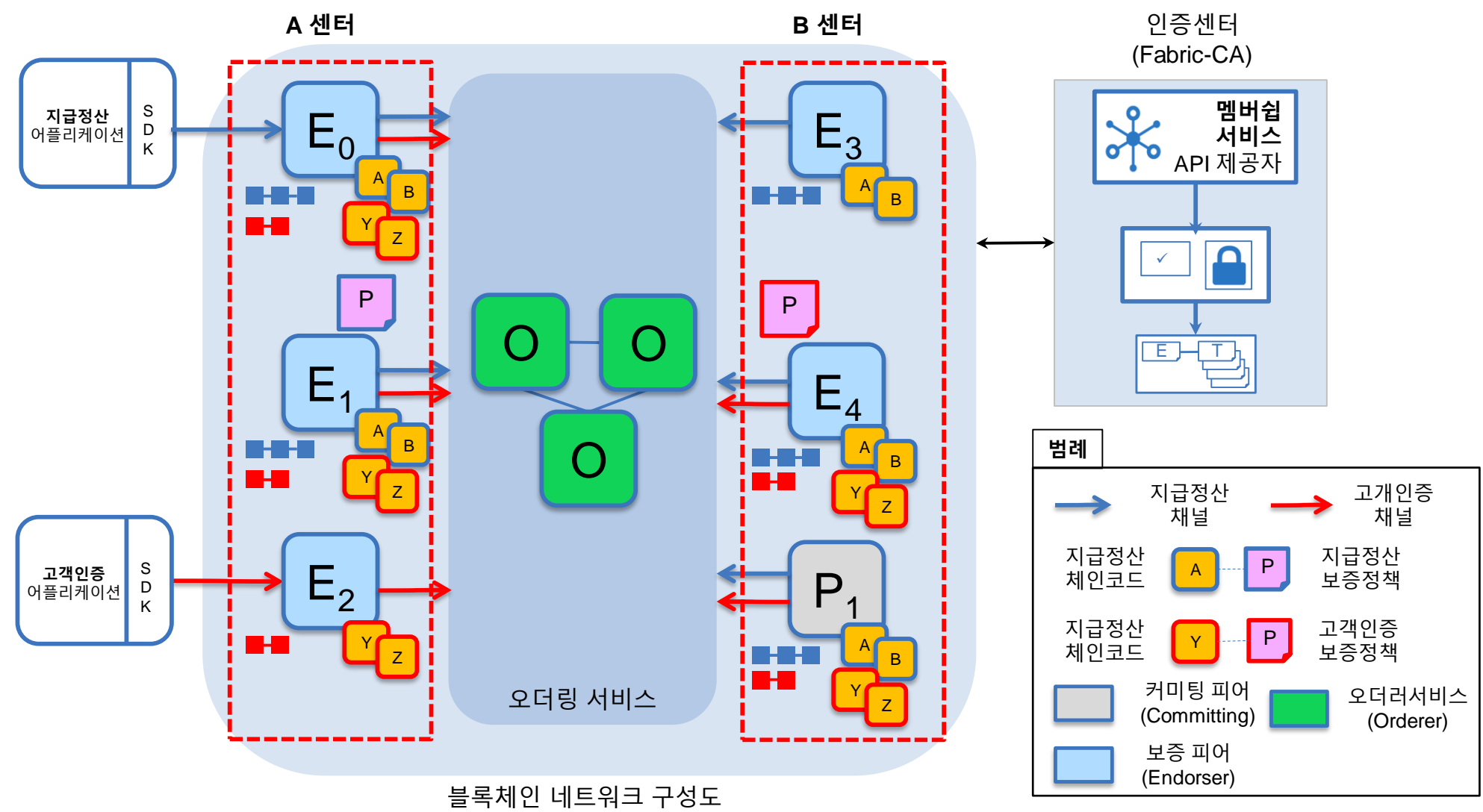
❖ 익명의 사용자를 인증하는 경우

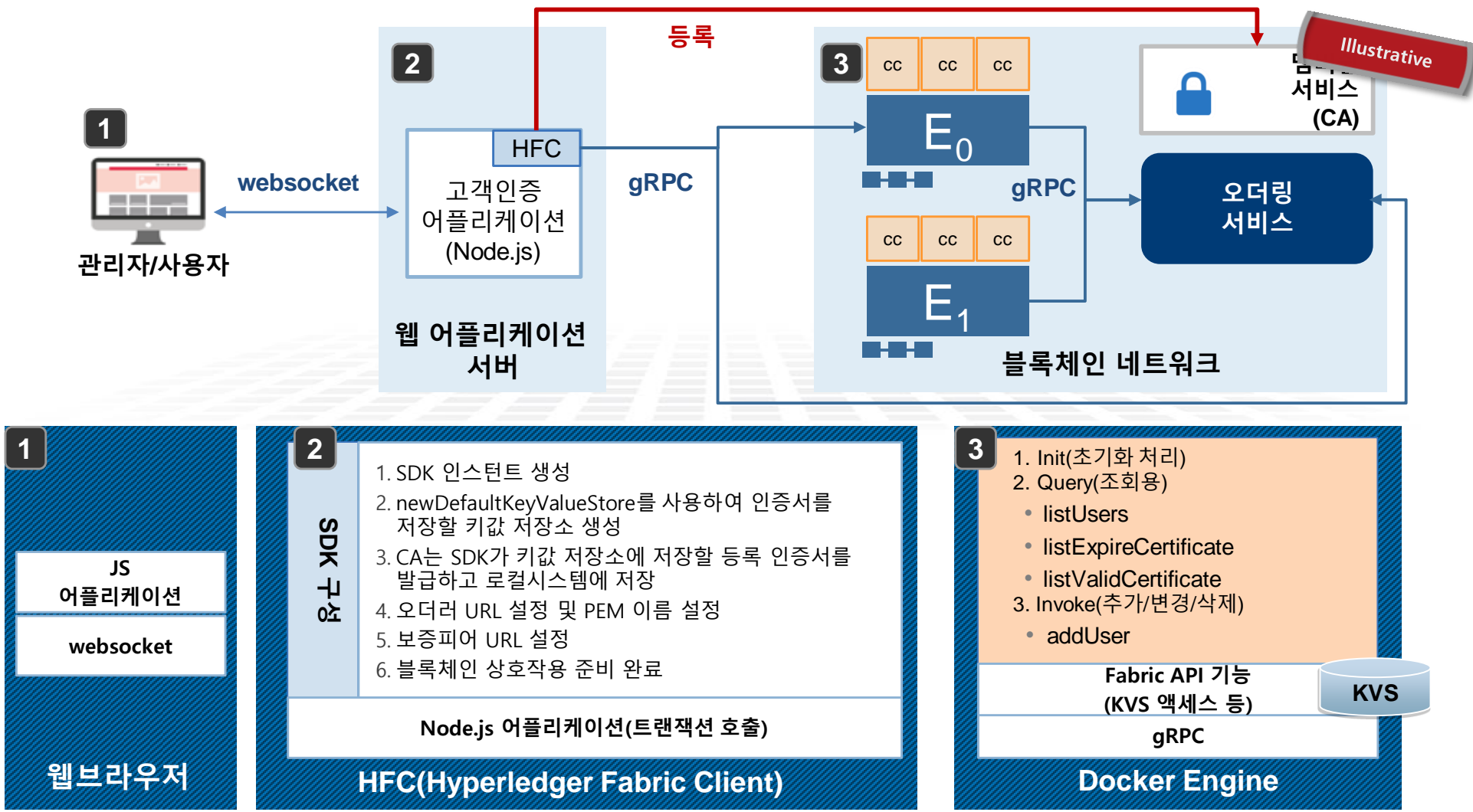


◆ 매번 다른 Tcert을 사용하면 각각의 트랜잭션은 다른 사용자로부터 발행된 것으로 보임



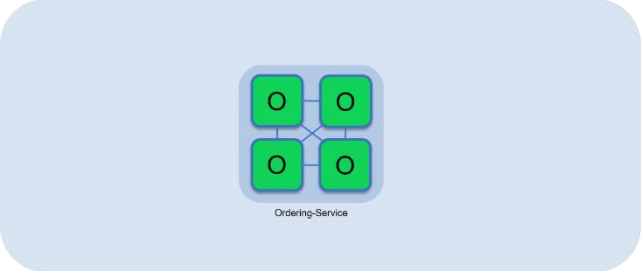
※ 이벤트나 인증서 속성정보와 관련된 기능들은 HFC에 의해서만 지원됨







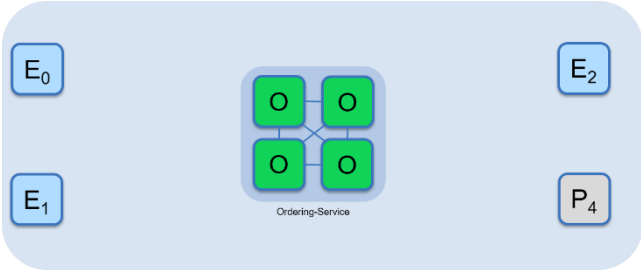
1 오더링서비스 구성 및 시작



```
$ docker-compose [-f orderer.yml] ...
```

✓ 오더링 서비스를 구성하고 시작

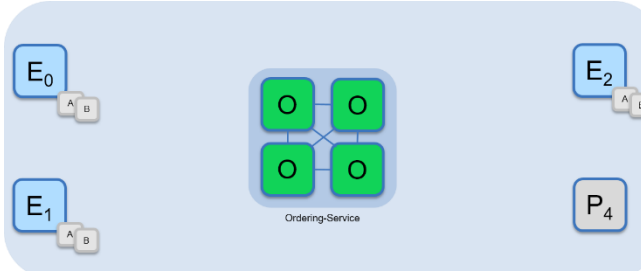
2 피어구성 및 시작



```
$ peer node start ...
```

✓ 보증피어(E0,E1,E2) 및 커미팅피어(P4) 구성

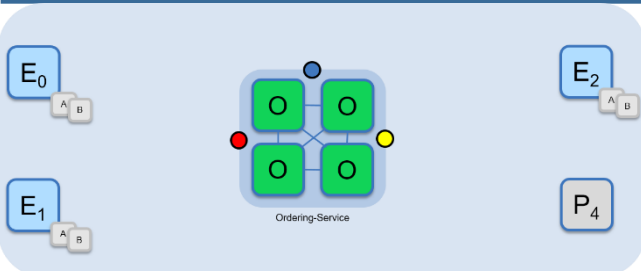
3 체인코드 설치



```
$ peer chaincode install ...
```

✓ 보증피어(E0,E1,E2)에 체인코드 설치

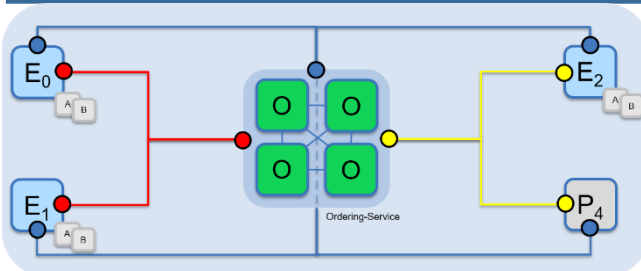
4 채널 생성



```
$ peer channel create -o [orderer] ...
```

✓ 오더링 서비스에 채널 생성

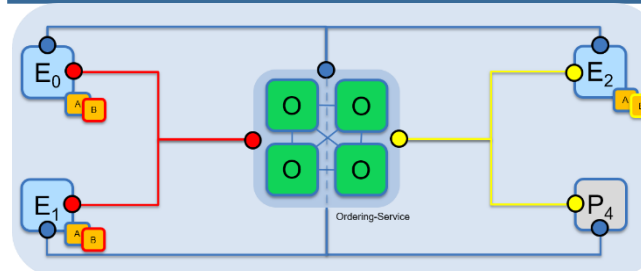
5 채널 가입



```
$ peer channel join ...
```

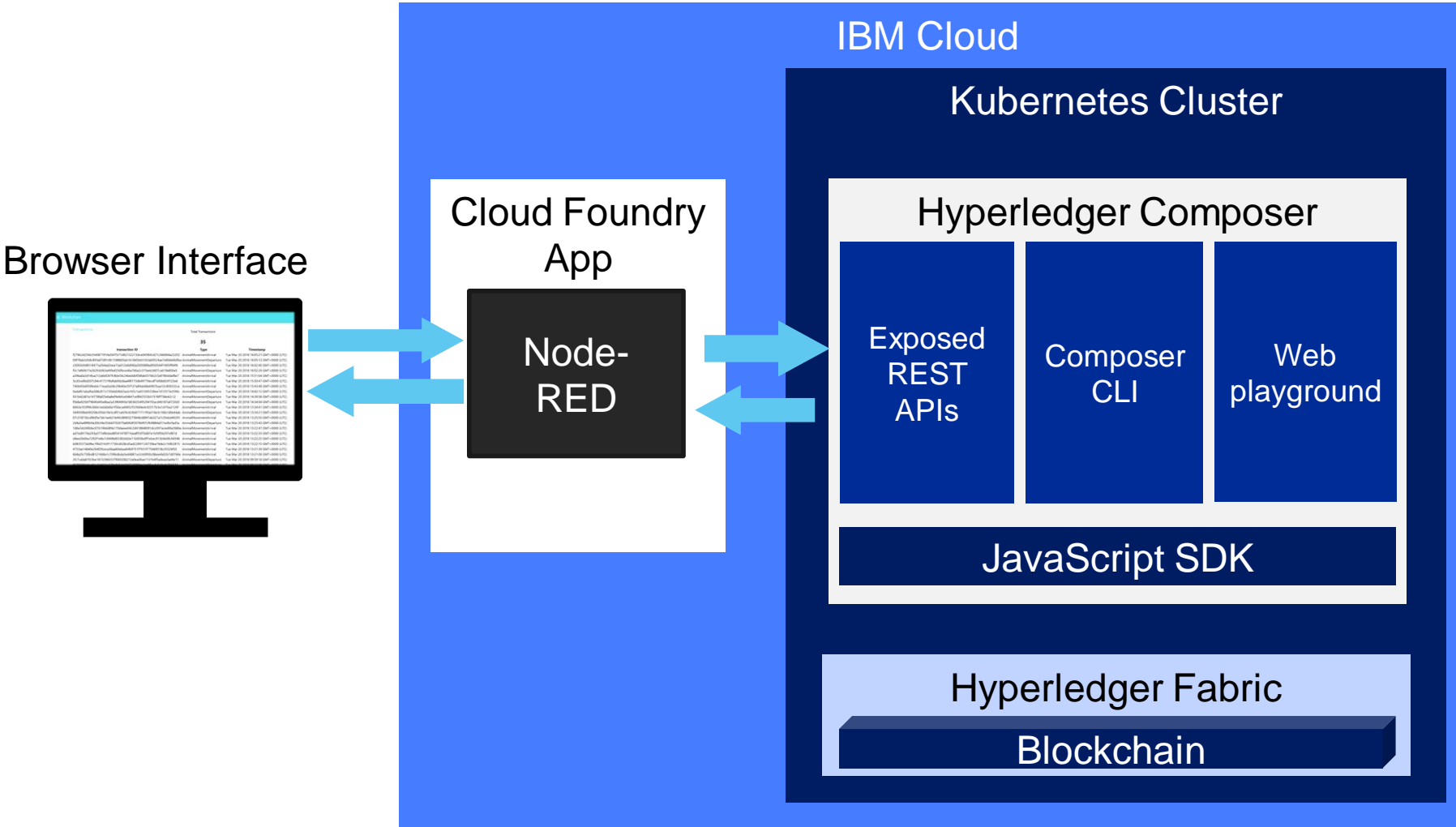
✓ 허가된 피어들이 채널에 가입

6 채널서비스 활성화



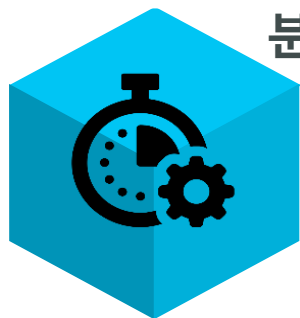
```
$ peer channel instantiate ... -P 'policy'
```

✓ 피어들은 채널에 있는 체인코드 활성화





기밀성



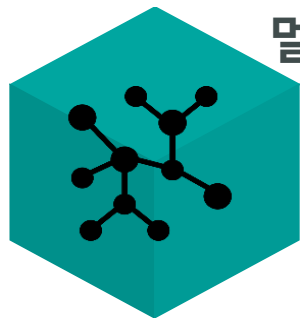
분할 실행

체인코드 실행과 트랜잭션 오더링을 분리함으로써 네트워크 성능을 최적화 시킴



허가형 멤버십

알려진 참여자 및 규제감독을 기반으로 신뢰된 블록체인 네트워크 운영



멀티 채널

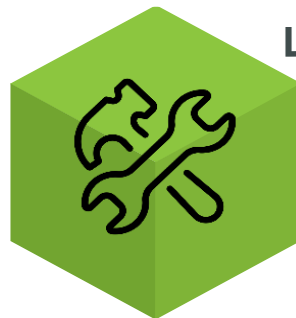
규제대상 산업에 필요한 개인정보 보호 및 기밀 유지기능등을 통해서 다자간 거래에 필요한 멀티 서비스 제공

운영 워크로드



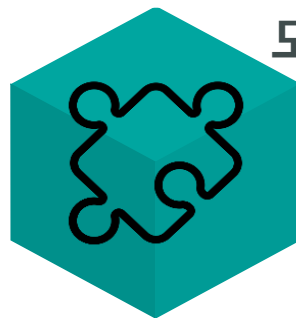
거래 내역

효율적인 감사 및 분쟁해결을 위한 검색 가능한 거래내역 조회



네트워크 도구

IBM은 모니터링, 로깅 및 컴플라이언스를 위한 백업/복원 도구를 제공

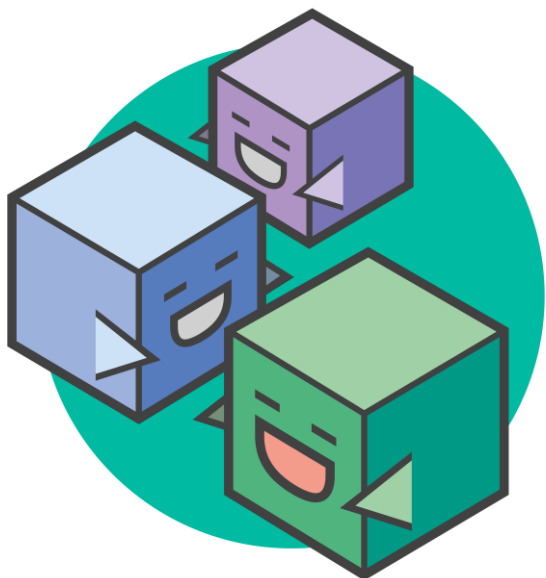


모듈러 아키텍처

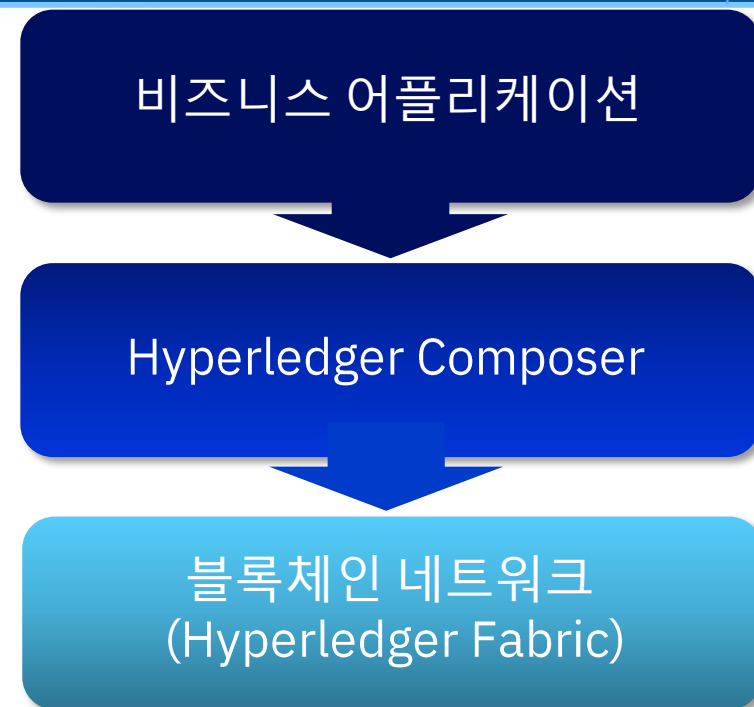
비즈니스 네트워크를 동적으로 확장 용이한 노드 수, 합의 알고리즘, ID 관리, 암호화에 대한 환경 설정

블록체인 비즈니스 네트워크 모델링 도구
















- 블록체인 비즈니스 네트워크를 위한 하이-레벨 어플리케이션 추상화 스위트(suite)
- 신속하게 솔루션 생성을 위한 비즈니스 중심의 용어 집중
- 리스크를 줄이고, 이해와 유연성을 높임

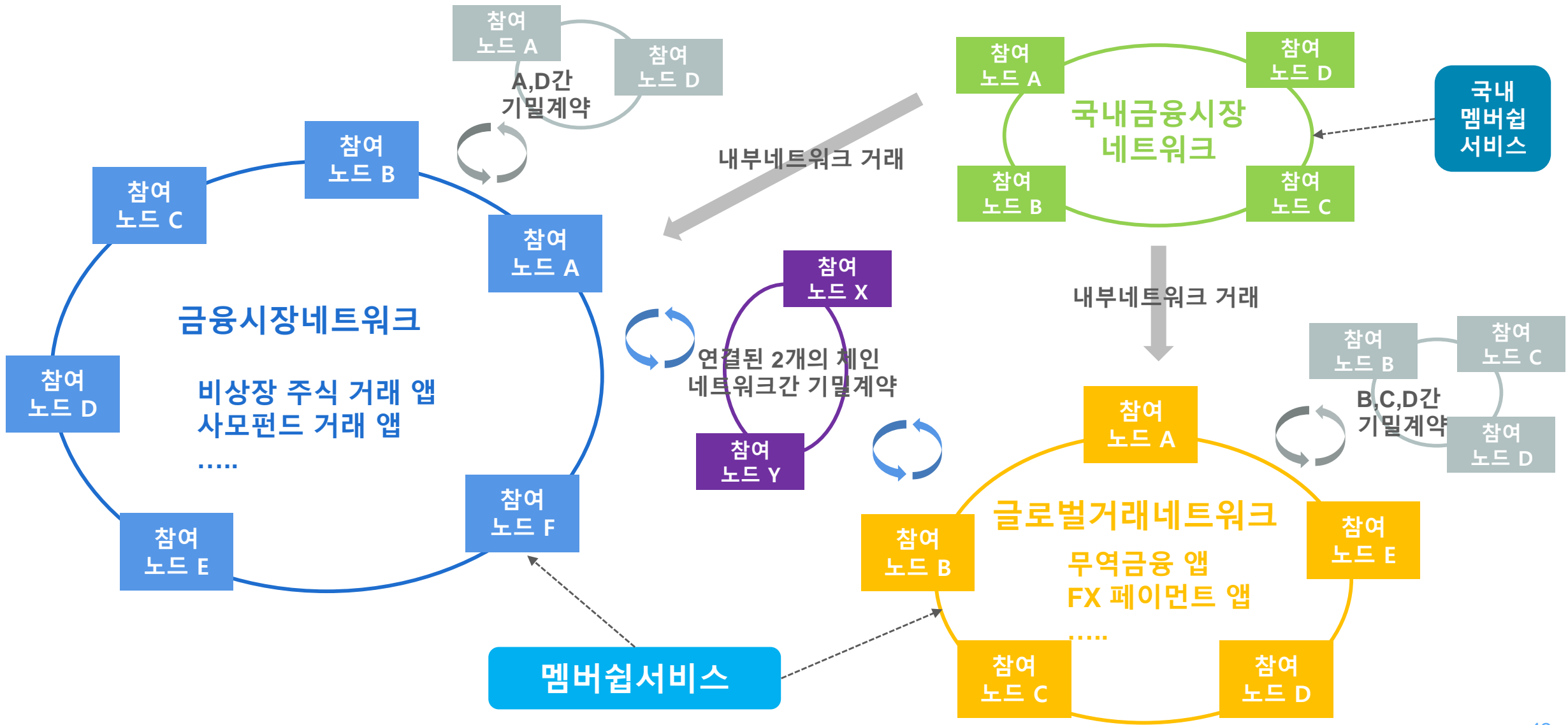


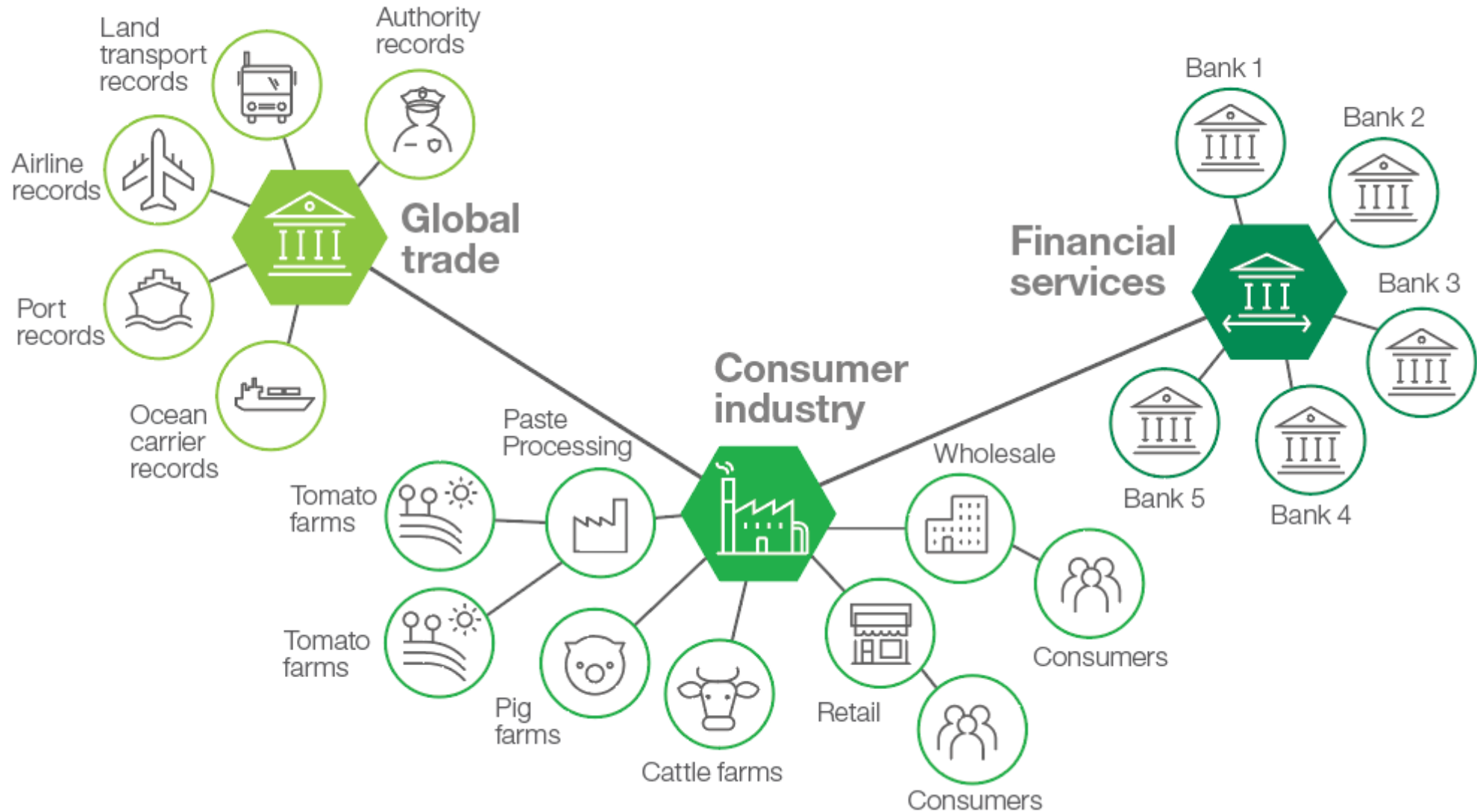
- 기능
 - 비즈니스 네트워크를 모델링하고, 테스트 및 APIs를 공개
 - 응용 어플리케이션은 비즈니스 네트워크와 통신하기 위해 APIs 트랜잭션을 호출
 - Loopback/REST를 이용하여 기존시스템과 통합
- 오픈 소스로 <http://fabric-composer.org> 사이트에서 활용 가능
- <http://composer-playground.mybluemix.net/> 사이트에서 활용 가능





Trade Finance	Pre and Post Trade	Complex Risk Coverage
   	   	 
Identity/ Know your customer (KYC)	Unlisted Securities/ Private Equity Funds	Incentive Program
  	  	 
Medicated Health Data Exchange	Government	Distributed Energy/ Carbon Credit
		 
Supply Chain	Food Trust	Provenance/ Traceability
 	         	







산업표준 기술

- 블록체인의 산업 표준 방향은 오픈소스(Open Source), 오픈기술(Open Technology), 오픈 거버넌스(Open Governance) 기반으로 발전하고 있음
- 암호화폐를 넘어 다양한 산업(금융, 식품, 물류, 에너지, 인증, 콘텐츠 등)에 디지털 혁명을 주도하고 있음

산업생태계 혁신

- 거래처리 시간절약, 중개자 비용절감, 사이버 리스크 감소 및 위변조 불가능한 신뢰 확산을 통해서 기존 산업생태계를 빠르게 변화시킬 핵심 기술로 변화
- 식품산업의 식품 안전망(Food Safety) 구축, 실시간 운송정보 및 페이퍼리스 기반의 물류무역 구축, 디지털 신원인증 및 부가서비스(자전거 서비스/공공서류 등)등의 새로운 사용자 경험을 통해 정부주도하의 블록체인 생태계 혁신 필요

블록체인 잠재력

- 제2의 차세대 인터넷 기술인 블록체인이 글로벌 블록체인 생태계를 빠르게 변화시키고 있음
- 지금도 블록체인 기술은 발전하고 있는 이머징 기술(Emerging Technology)로 미래의 가능성과 로드맵을 통해서 준비 필요



박 세열
블록체인 기술리더/
금융총괄 아키텍트

Blockchain Technical Leader/
Client Technical Leader
S&D Architect Team
Technical Sales
Banking & Financial Markets



IBM Korea
Three IFC, 10 Gukjegeumyung-ro,
Yeongdeungpo-gu, Seoul, Korea

Mobile: +82-10-4995-7163
Phone: +82-2-3781-7163

E-mail: sypark@kr.ibm.com

