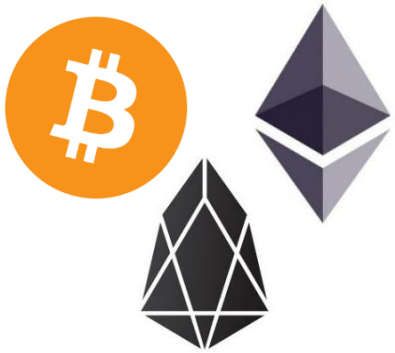
# 하이퍼레저 페브릭 개요

# 하이퍼레저 페브릭



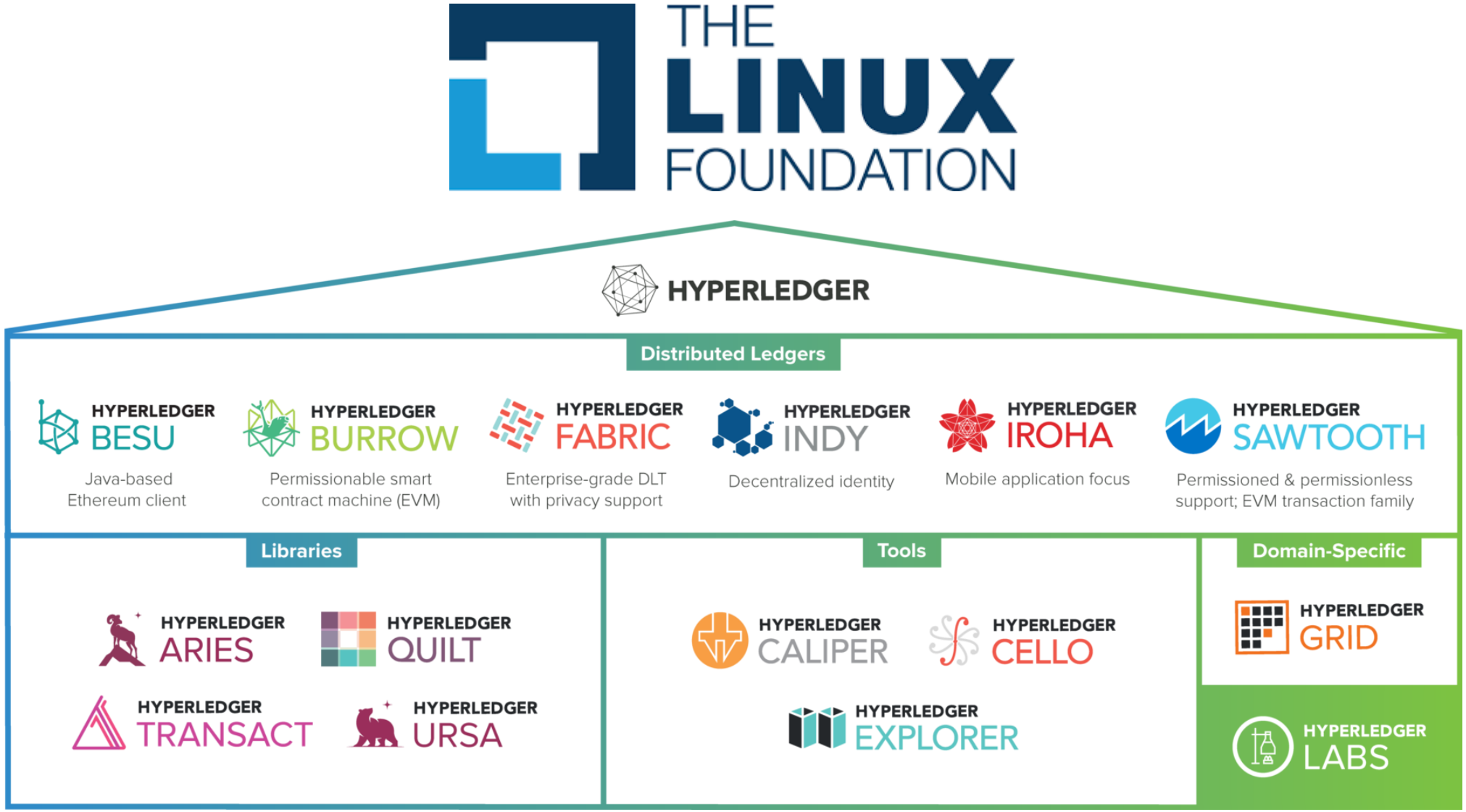
## 퍼블릭 블록체인

- 네트워크 누구나 참여 가능

## 프라이빗 블록체인

- 허가받은 사용자만 참여 가능

# Hyperledger Besu 소개



# Hyperledger Besu 소개

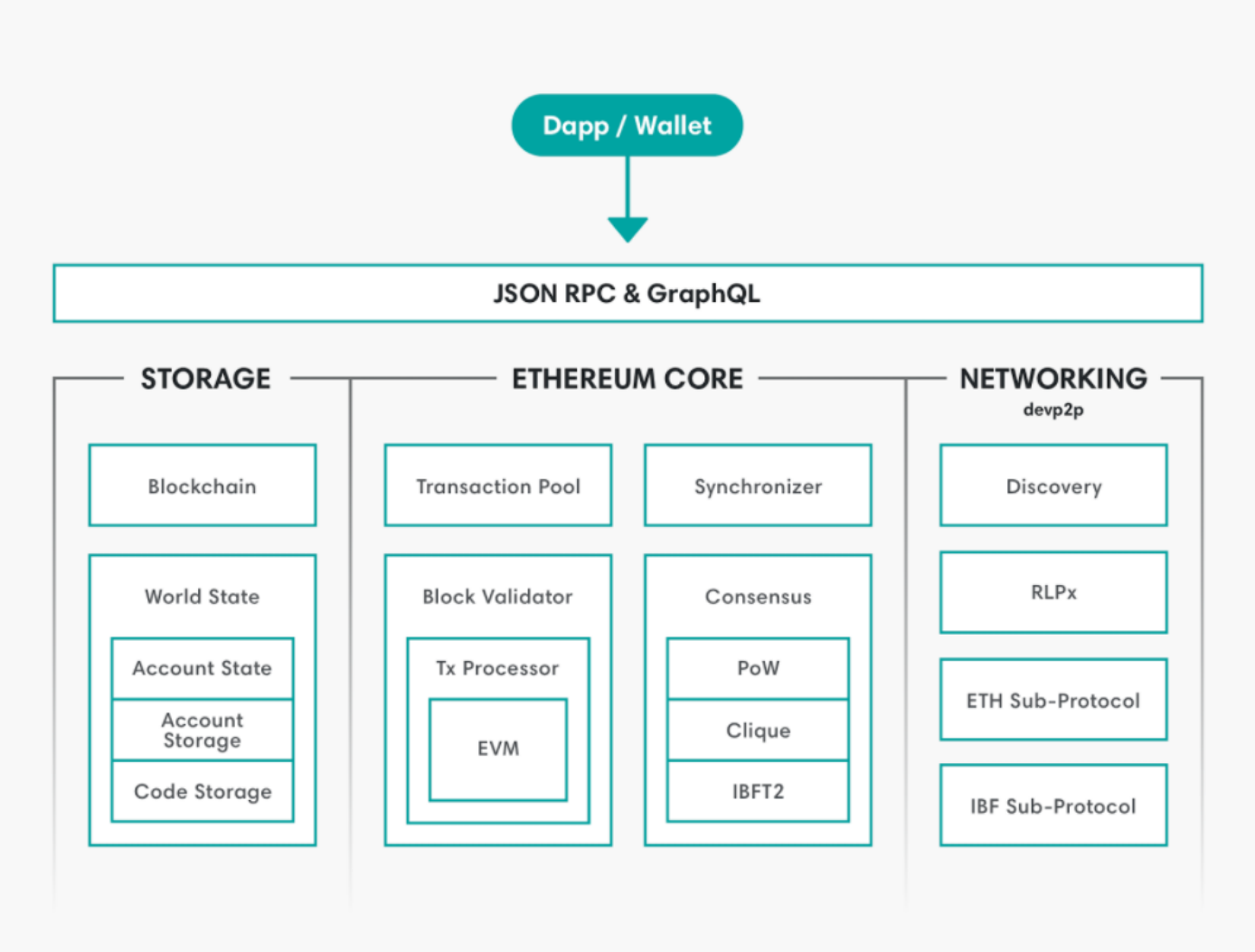

- IBM이 주도하는 프로젝트
- 프라이빗 블록체인에서 가장 유명함
- 접근 제어 기능을 제공


- DID 제공 플랫폼


- Intel이 주도하는 프로젝트
- IoT 환경에서의 블록체인


- 이더리움 프라이빗 네트워크

# Hyperledger Besu 소개



# Hyperledger Besu 소개

## What is Hyperledger Besu?

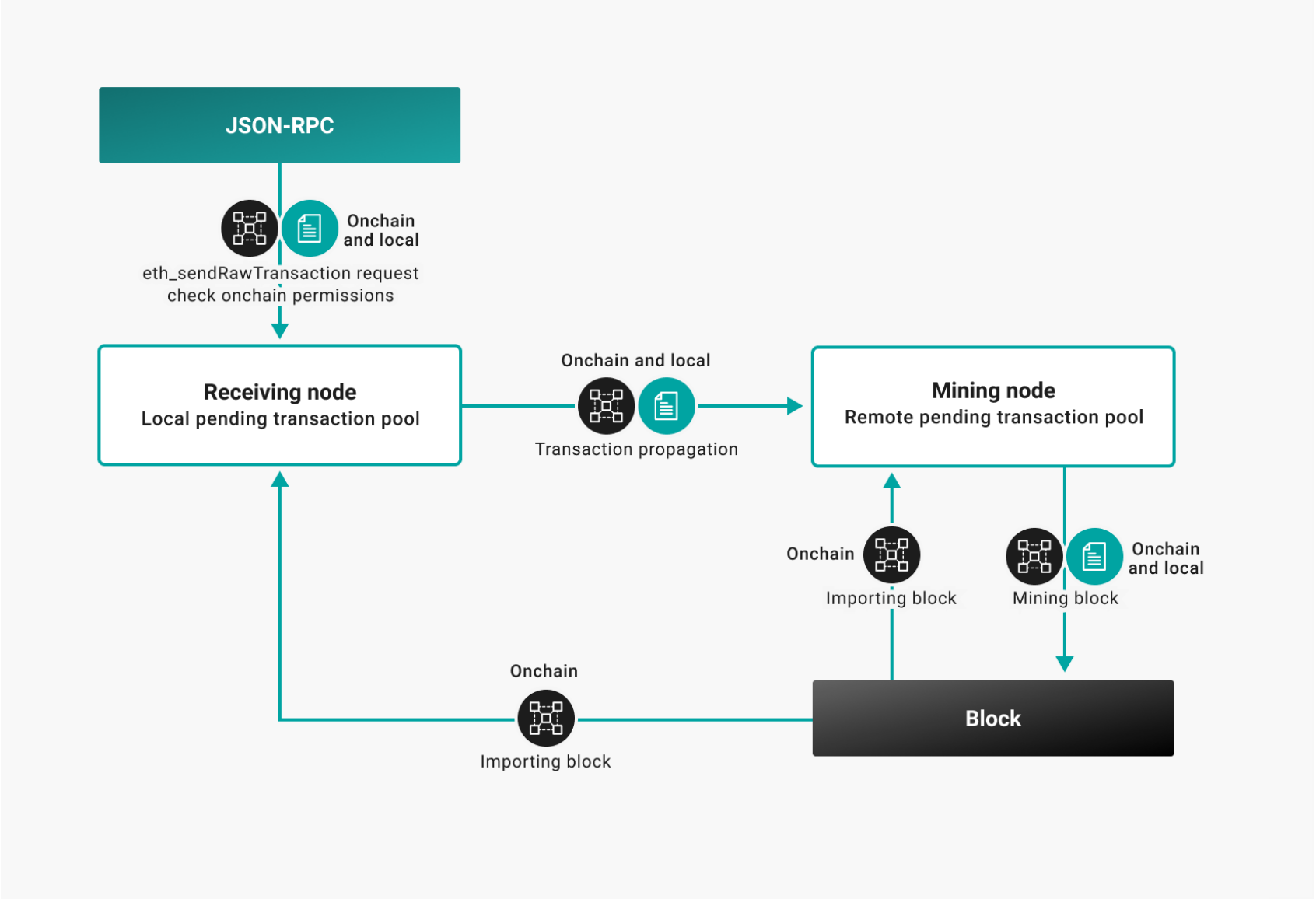
Hyperledger Besu is an open-source Ethereum client developed under the Apache 2.0 license and written in Java. It runs on the Ethereum public network, private networks, and test networks such as Rinkeby, Ropsten, and Görli. Besu implements proof of work (Ethash) and proof of authority (IBFT 2.0, Clique, and QBFT) consensus mechanisms.

You can use Besu to develop enterprise applications requiring secure, high-performance transaction processing in a private network.

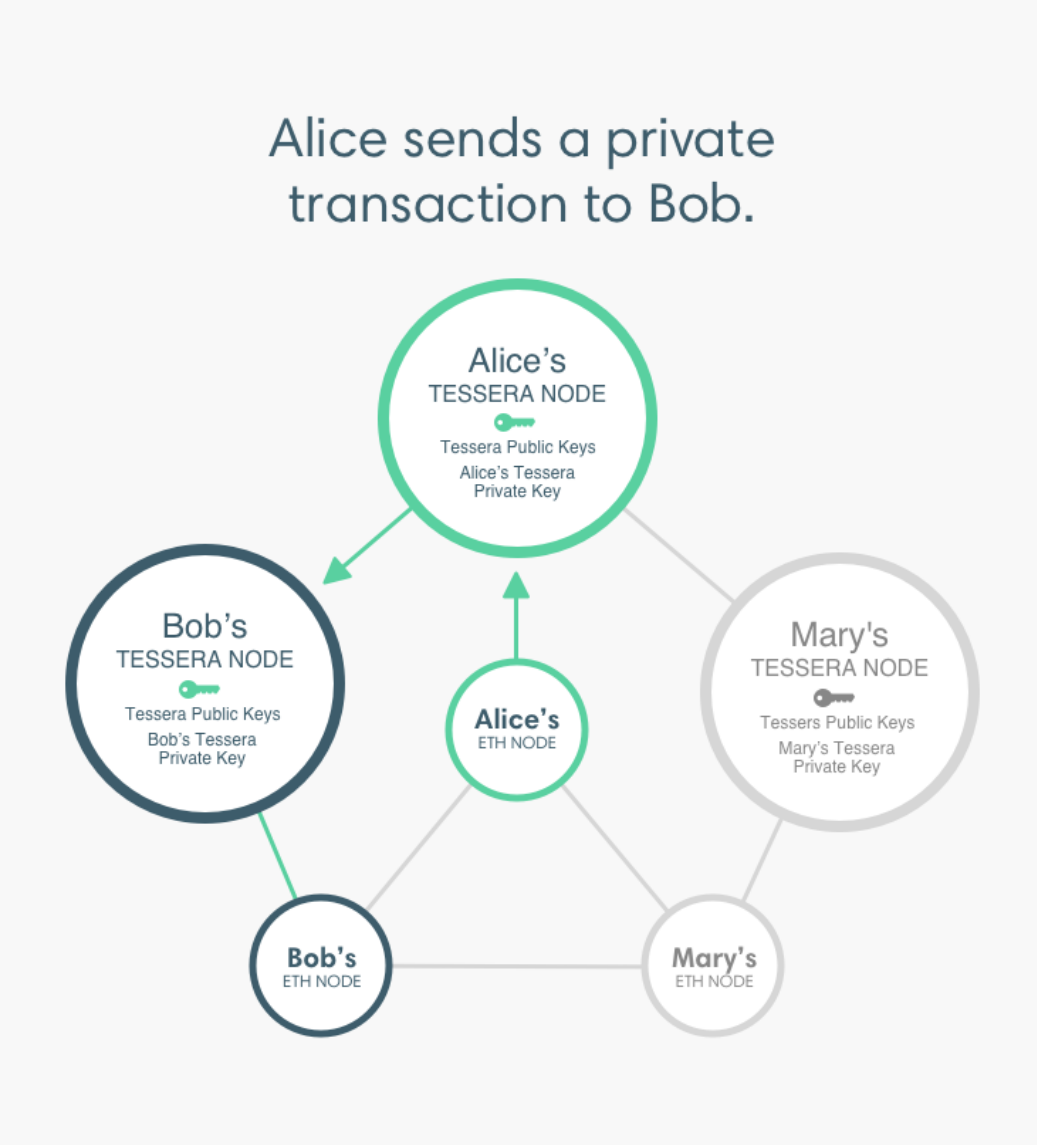
Besu supports enterprise features including privacy and permissioning.

- 자바 언어로 작성된 이더리움 클라이언트
- 이더리움과 동일하게 동작하며, 이더리움의 테스트 넷에서도 동작
- PoW와 QBFT, IBFT를 사용
- 엔터프라이즈용

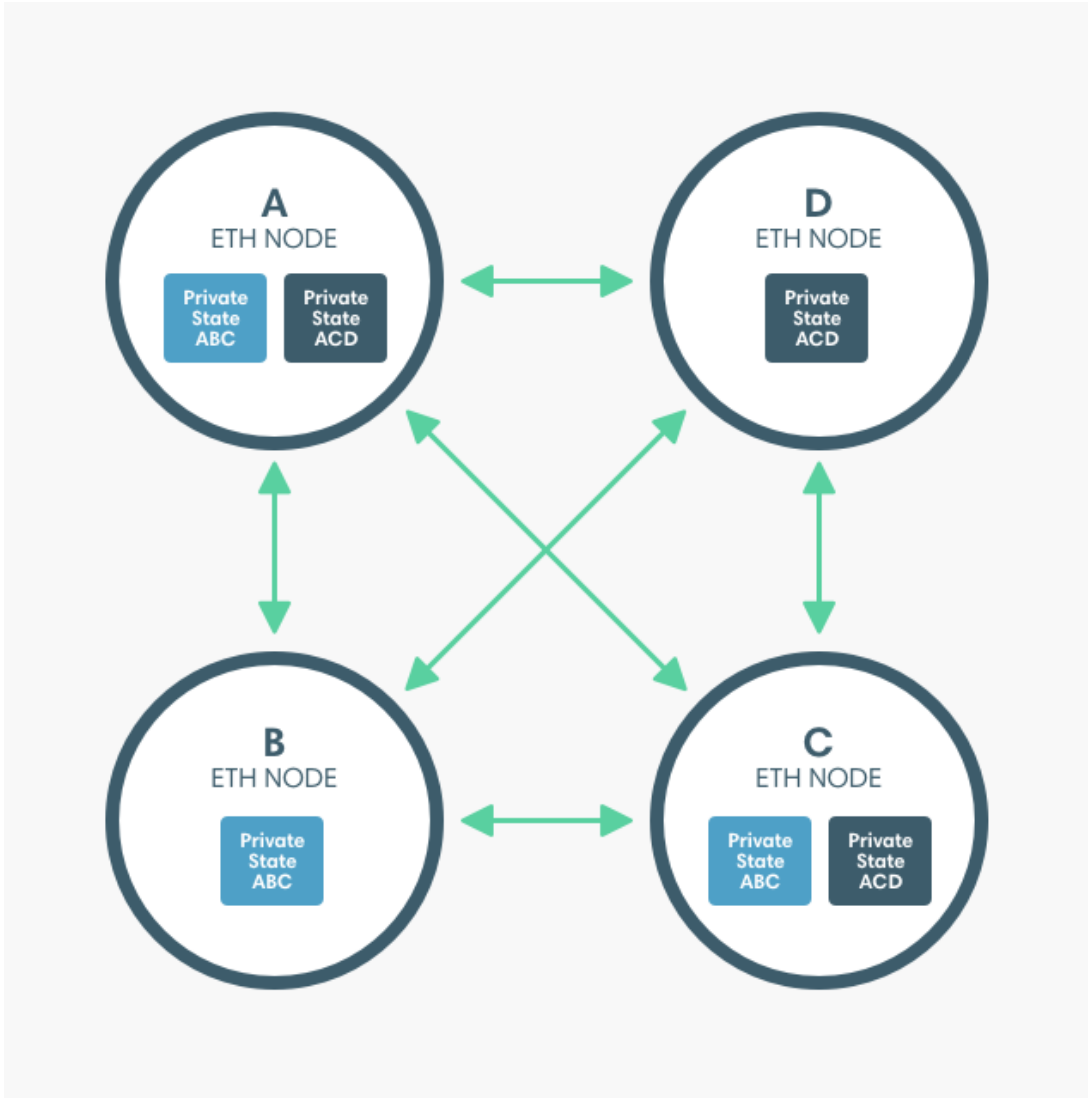
# Hyperledger Besu 소개



# Hyperledger Besu 소개



# Hyperledger Besu 소개



# Hyperledger Besu 소개

Besu implements the following consensus protocols:

- Ethash (proof of work)
- Clique (proof of authority)
- IBFT 2.0 (proof of authority)
- Quorum IBFT 1.0 (proof of authority).
- QBFT (proof of authority).

Ethash
- 작업증명

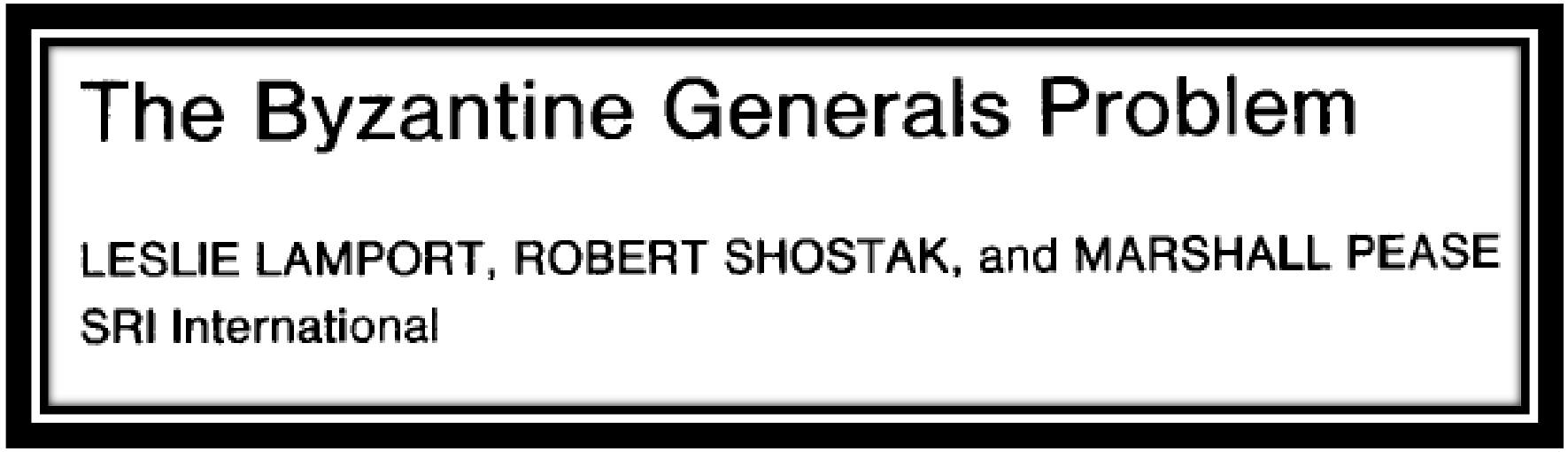
Clique
- 선정된 인원만 작업증명

IBFT 2.0
- 선정된 인원만 PBFT 진행

Quorum IBFT 1.0
- 베수노드 전원 PBFT 진행

QBFT
- 합의에 참여하는 인원을
  Block Header validator, Contract
  validator로 구분해서 따로 검증을 진행

# BFT (PBFT, IBFT, XBFT…)


The Byzantine Generals Problem

LESLIE LAMPORT, ROBERT SHOSTAK, and MARSHALL PEASE
SRI International

# BFT (PBFT, IBFT, XBFT…)

A reliable computer system must be able to cope with the failure of one or more of its components. A failed component may exhibit a type of behavior that is often overlooked—namely, sending conflicting information to different parts of the system. The problem of coping with this type of failure is expressed abstractly as the Byzantine Generals Problem. We devote the major part of the paper to a discussion of this abstract problem and conclude by indicating how our solutions can be used in implementing a reliable computer system.

Reliable Computer System은 오류(Failure)에 대처할수 있어함

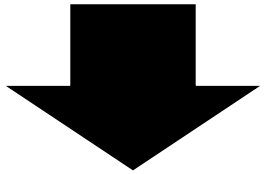
# BFT (PBFT, IBFT, XBFT…)

## [Byzantine General Problem]

전체 노드의 수가 N 이라고 할때
A라는 노드가 본인을 제외한 다른 노드에게 메시지를 어떻게 전달할 것인가

# BFT (PBFT, IBFT, XBFT…)

3분의 2 이상의 노드가 정상적인
노드가 아니라면 해결할 방법이 없다

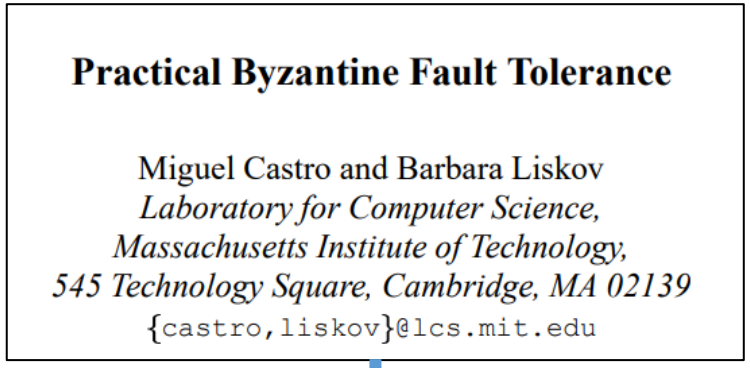


$$N=3f+1$$

[ 해결책 ]

1. 투표 (다수결)
2. 서명
3. 네트워크 토폴로지 조정

# BFT (PBFT, IBFT, XBFT…)




## 합의에 참여하는 노드수 줄이기

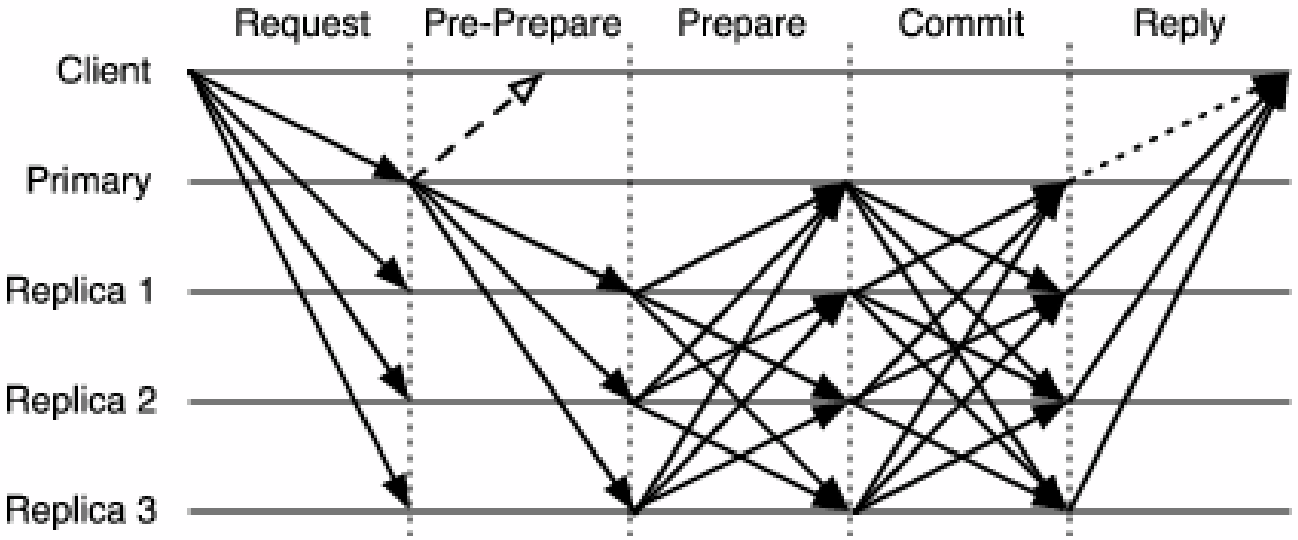
- Scalable Dynamic Multi-Agent Practical Byzantine Fault-Tolerant Consensus in Permissioned Blockchain
- SCP: A computationally-Scalable Byzantine Consensus Protocol For Blockchain

## 합의에 발생하는 통신량 줄이기

- Zyzzyva: Speculative Byzantine Fault Tolerance
- SBFT: a Scalable and Decentralized Trust Infrastructure
- State Machine Replication for the Masses with BFT-SMART
- Resource-Efficient Byzantine Fault Tolerance
- Efficient Byzantine Fault-Tolerance
- Scalable Byzantine Consensus via Hardware-Assisted Secret Sharing

# BFT (PBFT, IBFT, XBFT…)

**실용적 비잔틴 장애 허용 (PBFT; Practical Byzantine Fault Tolerance)**




- 클라이언트가 블록을 제시 하고 해당 블록에 대해서 매번 투표를 진행 하는 방식
- 데이터의 안전성을 보장
- 노드 수가 많아질수록 합의에 도달하는 시간이 오래 걸림

# BFT (PBFT, IBFT, XBFT…)

**Libra Blockchain 사용**

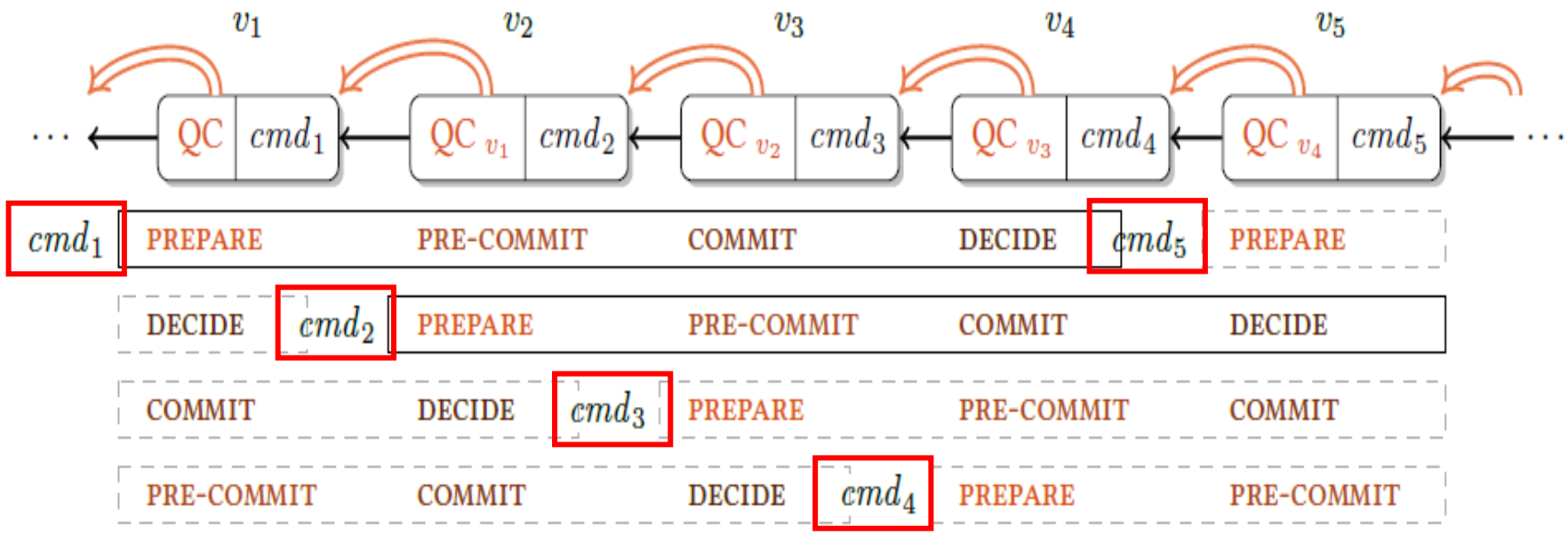



Figure 1: Chained HotStuff is a pipelined Basic HotStuff where a QC can serve in different phases simultaneously.

**Multi-Thread 방식의 합의 진행**

# Q & A