

Task-2

Comparison table

Blockchain Name	Type	Consensus Mechanism	Permission Model	Speed / Throughput	Smart Contract Support	Token Support	Typical Use Case
Ethereum	Public	Proof of Stake (Casper FFG)	Open	~30 TPS (base layer)	Yes – Solidity, Vyper	Native ETH + ERC	Decentralize apps, DeFi
Hyperledger Fabric	Private	Pluggable (Raft, Kafka, etc.)	Permissioned	1,000–3,500 TPS	Yes – Chaincode in Go/Java/Node.js	No native token	Private asset tracking, supply chain
R3 Corda	Consortium	Notary-based (RAFT or BFT)	Permissioned	~2,000 TPS	Yes – JVM contracts (Java/Kotlin)	Token SDK support	Interbank settlements, trade finance

Short Report

The three platforms cater to distinct technical needs. Ethereum excels as an open, public network with a Turing-complete EVM, broad token standards, and mature tooling, making it ideal for consumer-facing decentralized apps and DeFi protocols. Its Proof-of-Stake consensus, while secure and energy-efficient, delivers modest throughput (~30 TPS) at base layer but scales via Layer 2.

Hyperledger Fabric, by contrast, is a fully permissioned, modular framework where consensus (e.g., Raft) and membership are pluggable. With private “channels” and data collections, it achieves much higher throughput (1,000–3,500 TPS) and strict access controls, rendering it well-suited for supply-chain networks among known partners requiring confidentiality and performance.

R3 Corda's notary-based architecture offers point-to-point privacy and a UTXO-style ledger, with pluggable consensus and a Token SDK for asset issuance. Its design targets regulated financial institutions, making it the natural choice for inter-bank applications demanding high privacy, legal-level contracts, and compliance.

- **Decentralized app?** Ethereum, for its open ecosystem and EVM support.
- **Supply-chain network?** Hyperledger Fabric, for permissioned privacy and high TPS.
- **Inter-bank financial app?** R3 Corda, for its granular privacy and legal-grade contract mode