Parameter	Ethereum	Hyperledger Fabric	R3 Corda
Type	Public	Private	Consortium
Consensus Mechanism Used	Proof-of-Stake (Gasper)	Pluggable (e.g., Raft, Kafka)	Notary-based
Permission Model	Open	Permissioned	Permissioned
Speed / Throughput (TPS)	15–30 TPS	1,000+ TPS	~1,000 TPS per node
Smart Contract Support	Yes (Solidity, Vyper)	Yes (Chaincode: Go, Java, JavaScript, Solidity)	Yes (Java, Kotlin, DAML)
Token Support	Native (ETH)	FabToken	No native token
Typical Use Case	Decentralized apps, DeFi	Supply chain, enterprise applications	Financial services, inter-bank apps
Notable Technical Feature	EVM compatibility, modular execution	Channels, private data collections	Transaction privacy, no mining

Technical Capabilities Comparison

- Ethereum uses Proof-of-Stake (Gasper) for energy efficiency and supports Turing-complete smart contracts (Solidity/Vyper). Its public nature enables global participation but limits privacy.
- Hyperledger Fabric employs pluggable consensus (e.g., Raft) and channels for privacy, achieving high throughput (1,000+ TPS) in enterprise settings. Chaincode supports multiple languages (Go, Java).
- R3 Corda uses notary nodes for consensus, ensuring transaction privacy and immediate finality. Designed for financial institutions, it lacks native tokens but integrates DAML for cross-platform interoperability.

Platform Recommendations

1. Decentralized App: Choose Ethereum for its robust smart contract ecosystem, EVM compatibility, and public accessibility.
2. Supply Chain Network: Hyperledger Fabric is ideal due to channels for data segregation, scalability, and support for asset tokenization.
3. Inter-Bank Application: R3 Corda excels with its privacy-focused design, notary-based validation, and compliance-friendly architecture.

Justification

- Ethereum's public ledger and developer tools suit decentralized apps.
- Fabric's modularity and privacy features align with multi-partner supply chains.
- Corda's financial-grade security and regulatory compliance meet inter-bank needs.