BLOCKCHAINS

ARCHITECTURE, DESIGN AND USE CASES

SANDIP CHAKRABORTY
COMPUTER SCIENCE AND ENGINEERING,
IIT KHARAGPUR

PRAVEEN JAYACHANDRAN
IBM RESEARCH,
INDIA

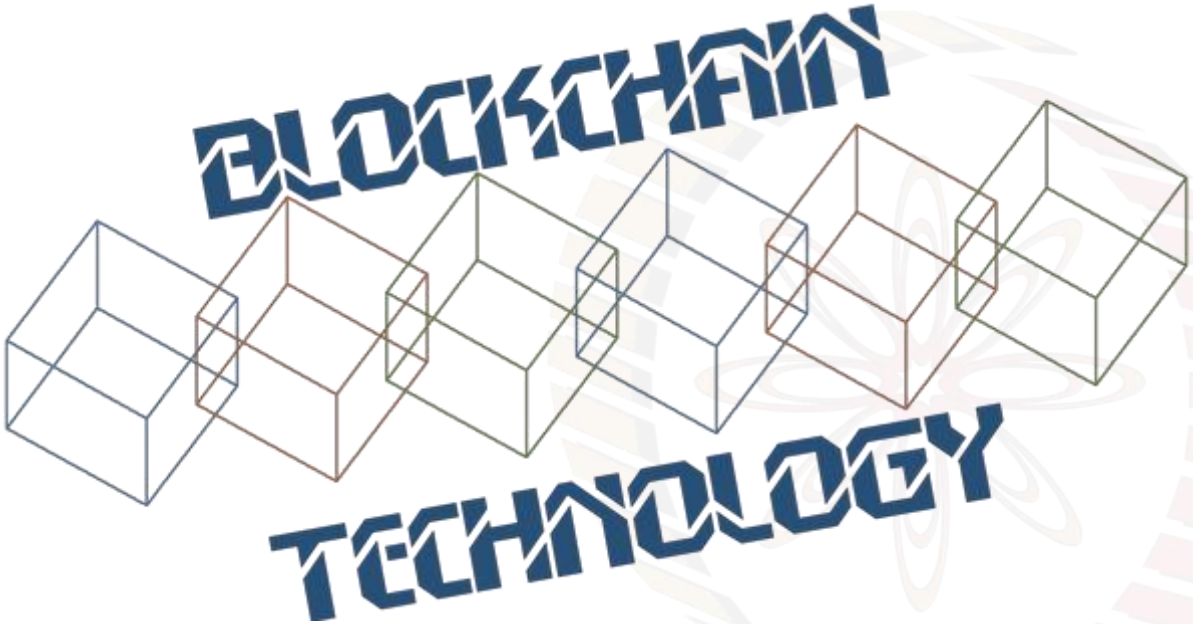


*Image courtesy: http://beetfusion.com/

ETHEREUM TOOLS AND QUORUM

Smart Contract Development and Deployment

Development Environment

- Built by Consensys
- Scaffolding of project structure (contract, migration scripts, tests…)
- Built-in smart contract compilation, linking, deployment and binary management.
- Automated contract testing in JS (using Mocha and Chai)
- Configurable build pipeline with support for custom build processes.
- Scriptable deployment & migrations framework.
- Network management for deploying to many public & private networks.
- Interactive console for direct contract communication.

Development & Testing

TestRPC

- Node.js based Ethereum client for testing and development.
- Simulates full-client behavior with most of the Ethereum RPC calls
- Tuned for speed of development and testing
- No more than one transaction per-block
- Good substitute for Geth and other vanilla Ethereum clients during development, but not for Enterprise Ethereum related capabilities

Deployment

Geth

(or other vanilla Eth Client)

Quorum

(or other Enterprise Eth Client)

Developer Framework – Application Layer

- Complete implementation of Ethereum's JSON-RPC client API over HTTP and IPC
- Reactive-functional API for working with filters
- **Auto-generation of Java smart contract wrappers to create, deploy, transact with and call smart contracts from native Java code**
- Android compatible
- Support for JP Morgan's Quorum via web3j-quorum

```
// Auto generated Java Class
public final class CustomerContract extends Contract {
  public CompletableFuture<TransactionReceipt>
   createNewCustomer(Bytes32 customerId, Bytes32 name){
    …
  }
}
```

Enterprise Ethereum Alliance

Mission: “Learn from and build upon the only smart contract supporting blockchain currently running in real-world production and to define enterprise-grade software capable of handling the most complex, highly demanding applications at the speed of business.”

Focus: To support enterprise use cases

- Enterprise security and data privacy
- Permissioning
- High throughput
- Pluggable architecture

Launch Members

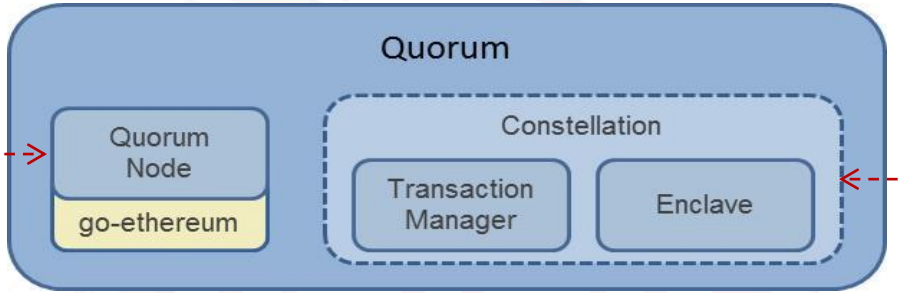
http://entethalliance.org/

Quorum

- A permissioned implementation of Ethereum that supports transaction and contract privacy.
- The primary features of Quorum, and therefore extensions over public Ethereum, are:
  - Transaction and contract privacy
  - Voting-based consensus mechanism
  - Network/Peer permissions management
  - Higher throughput (~300 tps for txn submission rate)
- A fork of the Go Ethereum client (a.k.a geth)

Quorum - Architecture

Modified version of the Go Ethereum client



Constellation: a general-purpose peer-to-peer encrypted message exchange system

Logical Architecture

- 'QuorumChain' consensus – a vote based consensus
- The P2P layer allows connections to/from permissioned nodes.
- The State Patricia trie split into two: a public state trie and a private state trie.
- Transaction creation has been modified to allow for Transaction data to be replaced by encrypted hashes in order to preserve private information
- The pricing of Gas has been removed, although Gas itself remains

- Transaction Manager is responsible for Transaction privacy.
- Cryptographic work including symmetric key generation and data encryption/decryption is delegated to the Enclave.

Quorum – Transaction Processing

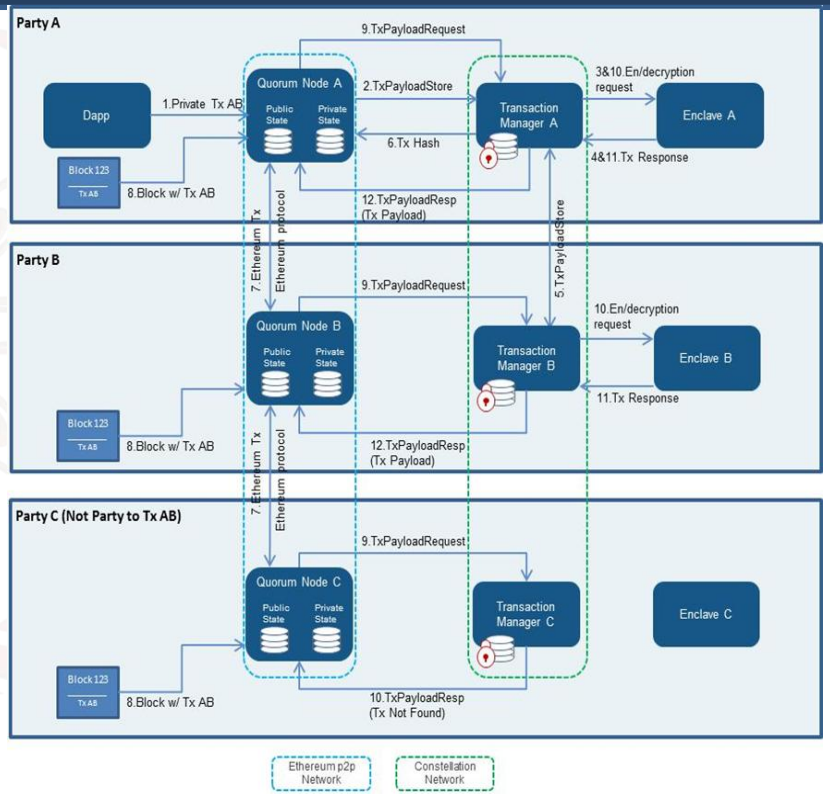
Quorum supports public and private Transactions

Public Transactions

- When a transaction is public, each participant will execute the same contract code and their underlying public StateDBs will be updated accordingly.

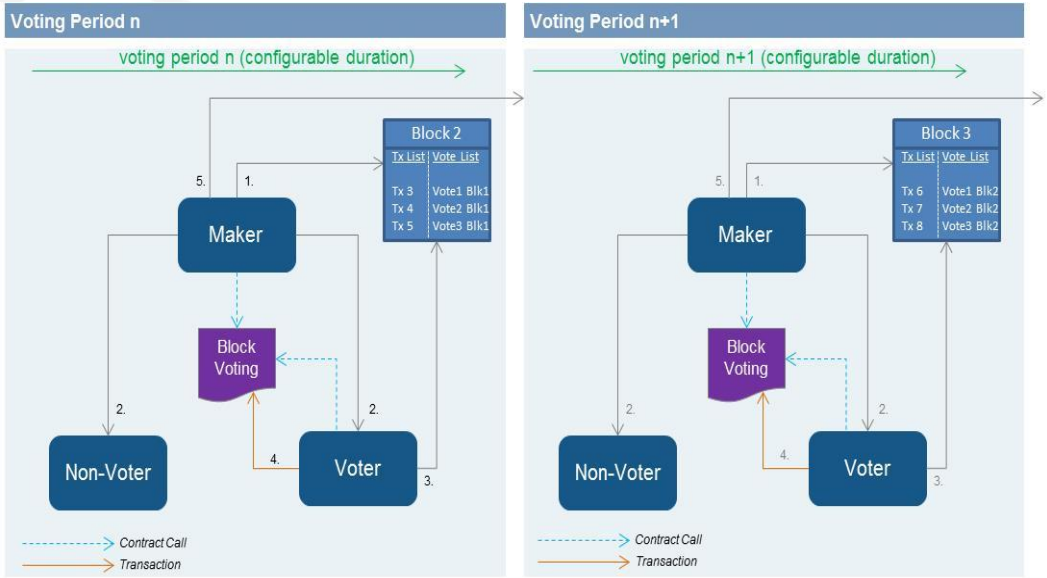
Transaction Privacy

- Private transactions are only visible to participants whose public keys are specified in the `privateFor` field
- The Quorum Node propagates the transaction to the rest of the network, after replacing payload with a hash of the encrypted payload it receives from Constellation.
- Participants party to the private transaction will replace the hash with the original payload before calling EVM for execution, and their private StateDBs is updated accordingly.



Quorum - Consensus

- Pluggable Consensus
- Current implementations: QuorumChain, Raft and Istanbul BFT
- QuorumChain:
- Nodes have roles and can be dynamically added/removed
  - Maker – responsible for making blocks
  - Voter – responsible for voting on the validity of blocks
  - Observer – only receives and validates blocks
- Most recent block with most votes is considered canonical head
- Leverages a smart contract called 'BlockVoting'



QuorumChain Consensus

Fun Reading

- Ethereum developer tools: https://github.com/ethereum/homestead-guide/blob/master/source/contracts-and-transactions/developer-tools.rst
- Quorum Whitepaper: https://github.com/jpmorganchase/quorum-docs/raw/master/Quorum Whitepaper v0.1.pdf
- Quorum overview documentation: https://github.com/jpmorganchase/quorum/wiki/Quorum-Overview
- RAFT Consensus: In Search of an Understandable Consensus Algorithm, https://ramcloud.stanford.edu/wiki/download/attachments/11370504/raft.pdf


thank you!