Hyperledger Burrow

Hyperledger Burrow Documentation

<table>
<thead>
<tr>
<th>Project</th>
<th><img src="https://hyperledger.github.io/burrow/#/tutorials/5-bonding-validators" alt="Hyperledger Burrow Logo" /></th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>INCUBATION</td>
</tr>
<tr>
<td>CII Badge</td>
<td><img src="https://hyperledger.github.io/burrow/#/tutorials/5-bonding-validators" alt="CII Badge" /></td>
</tr>
<tr>
<td>Description</td>
<td>Permissioned Ethereum smart-contract blockchain</td>
</tr>
</tbody>
</table>

Burrow is a permissively licensed (Apache 2.0) EVM smart contract machine and Byzantine Fault Tolerant permissioned ledger that uses Tendermint consensus and implements some novel extensions to the EVM whilst remaining EVM-compliant. It provides EVM execution within the Ethereum account model and an internal token to meter computation in the permissioned setting with transactions finality. Burrow is named after the trans-dimensional intergalactic tubules used by marmots to communicate.

Key Characteristics

Burrow has three primary aims:

- To be a good compliant and simple EVM library friendly towards integrators (see for example [https://github.com/hyperledger/sawtooth-seth](https://github.com/hyperledger/sawtooth-seth) and [https://github.com/hyperledger/fabric-chaincode-evm](https://github.com/hyperledger/fabric-chaincode-evm))
- To be a fast, light, lean single-process full Tendermint/EVM permissioned ledger with transaction finality
- To provide a practical base for EVM extensions in a many-chain world

Burrow is not trying to be:

- Highly pluggable (see Sawtooth or Fabric)
- Hard to deploy

Notable features:

- Single pure go binary including all tooling
- GRPC API interfaces (see: [https://github.com/hyperledger/burrow/tree/develop/protobuf](https://github.com/hyperledger/burrow/tree/develop/protobuf) - use from any supported GRPC [https://grpc.io/docs/quickstart/](https://grpc.io/docs/quickstart/))
- Javascript client library: [https://github.com/hyperledger/burrow/tree/develop/js](https://github.com/hyperledger/burrow/tree/develop/js) with smart contract function mapping layer
- Permissioned EVM (see: [https://github.com/hyperledger/burrow/blob/develop/permission/perm_flag.go](https://github.com/hyperledger/burrow/blob/develop/permission/perm_flag.go))
- On-chain EVM ABIs (function and contract definitions) and contract metadata
- Experimental WASM contract support
- Governance mechanism capable of atomically upgrading systems of contracts based on quorum voting
- Validator bonding for proof-of-stake networks ([https://hyperledger.github.io/burrow/#/tutorials/5-bonding-validators](https://hyperledger.github.io/burrow/#/tutorials/5-bonding-validators))
- Solidity compilation, contract deployment, and testing tool `burrow deploy`
- Scriptable transaction tool `burrow tx`
- Forensics tool `burrow examine`
- Chain generation and genesis tool `burrow spec` and `burrow configure`
- Dump/restore functionality `burrow dump` and `burrow restore` (ship state between versions or chains) - also allows state serialisation
- Keys signing service command-line wallet and server `burrow keys`
Hyperledger Burrow Documentation

- **Kubernetes Helm charts:** [https://hyperledger.github.io/burrow/#/tutorials/5-bonding-validators](https://hyperledger.github.io/burrow/#/tutorials/5-bonding-validators)
- **Godoc:** [https://godoc.org/github.com/hyperledger/burrow](https://godoc.org/github.com/hyperledger/burrow)
- See also: [https://github.com/hyperledger/burrow/tree/develop/docs](https://github.com/hyperledger/burrow/tree/develop/docs) for documentation source, architecture decision records, and further historical and design documentation

## Project Management

Burrow is being heavily tested as the core of the [Agreements Network](https://hyperledger.github.io/burrow/#/tutorials/5-bonding-validators). It sits at the intersections of a number of emerging technologies:

- EVM contracts and host-native code contracts
- Public permissioned networks
- Permissioned Ethereum and public Tendermint/Cosmos
- Layer 2 scaling
- Acting as a side-chain or state channel

We aim to provide a robust blockchain node for running multiple interconnected chains in a many-chain world. As well as blurring the public/private chain divide.

## Roadmaps

- **Q1 2019**
- **Q4 2018** - post-mortem
- **Q3 2018** - post-mortem
- **Q2 2018**
- **Q1 2018** - post-mortem

## Repositories

Burrow's repository is on github here: [https://github.com/hyperledger/burrow](https://github.com/hyperledger/burrow)

The Burrow binary contains everything you need to specify, configure, run, and deploy smart contracts to a chain.

- `burrow spec` - for describing template genesis state
- `burrow configure` - for realising a specific configuration (including key generation)
- `burrow keys` - both a standalone key signing daemon and key generation tool
- `burrow deploy` - a declarative Solidity compilation, chain management, testing, and smart contract deployment tool
- `burrow dump` - a forensics, auditing, and data extraction tool
- `burrow snatives` - a tool for interacting with Burrow’s ‘secure natives’ - host code that is callable as if it were an EVM contract
- `burrow start` - for starting a blockchain node

For deploying contracts you will need a local installation of [Solidity](https://github.com/hyperledger/burrow).

For previous versions of standalone Burrow you can find:

- **binaries**: [https://github.com/hyperledger/burrow/releases](https://github.com/hyperledger/burrow/releases)
- **docker images**: [https://hub.docker.com/r/hyperledger/burrow](https://hub.docker.com/r/hyperledger/burrow)

## Deployment

Burrow can be deployed in any environment but we have focussed on deploying related sets of validators (or validator pools) using Kubernetes/Helm and you can find helm charts here: [https://github.com/helm/charts/tree/master/incubator/burrow](https://github.com/helm/charts/tree/master/incubator/burrow).

## Contributing

Please fork, branch, and make pull requests to the [develop](https://github.com/hyperledger/burrow/tree/develop/docs) branch.

Our build, CI, and testing process is executed via our [Makefile](https://github.com/hyperledger/burrow/tree/develop/docs), see the comments there for details.

## Communication

### Mailing List

- **burrow**

### Chat (for questions and ephemeral discussions)
Meeting

Quarterly updates

2019

- Q3 2019 Update
- Q2 2019 Update
- Q1 2019 Update

2018

- Q4 2018 Update
- Q3 2018 Update
- Q2 2018 Update
- Q1 2018 Update

2017

- Q4 2017 Update

History

Burrow was approved for incubation on the 6th of April 2017 by the Hyperledger TSC.

Recent space activity

Silas Davis
- Burrow - The Boring Blockchain updated Feb 29, 2020 · view change

Kelly Cooper
- Burrow - The Boring Blockchain updated Oct 08, 2019 · view change
- Burrow Contributor or Volunteer Welcome updated Sep 06, 2019 · view change
- Meeting Notes updated Sep 05, 2019 · view change
- Meeting Agendas updated Sep 05, 2019 · view change

Space contributors

- Silas Davis (103 days ago)
- Kelly Cooper (247 days ago)
- David Huseby (384 days ago)
- Tracy Kuhr (498 days ago)
- Silona Bonewald (513 days ago)