

# 2019 Q3 Hyperledger Burrow

## Project

<https://github.com/hyperledger/burrow/>

## Project Health

Burrow has seen an uptick in issues on GitHub and questions on Hyperledger chat, and also the return of a prodigal contributor. The last quarter has been a bumper one for new features, see below. Burrow desperately needs to document and publicise its existing features better.

## Issues

Issues on GitHub highlight the difficulty that new users have getting started with the project. Much of this comes down to a lack of documentation or stale documentation. On that latter, a cluster of new features meant to help spinning up single-machine bare metal networks made some invalidated some of the instructions in our getting started guides. In another instance two incompatible flags were used in a code snippet which made the example throw an error. None of this was very hard to fix but it's a dumb way to lose contributors and users. I have no measurement but I'm sure this hurts our 'contributor funnel'.

We can fix this by:

- Using a markdown pre-processor to include snippets and make them part of our integration testing.
- Write basic docs across our entire feature set.

Neither of these are that hard but realistically we need help/more resources to get this done. I would like to make it an aim for this quarter to get Burrow to '100% tested docs coverage' - meaning we at *least* talk about every main feature of Burrow even if the docs are rubbish/incomplete. And where we have example code and scripts we run them with our CI so they don't go stale.

To try and drum up support I plan to do the following:

- Write a summary Burrow newsletter (I'm going to call it `Burrata`, the first one will be `Burrata 1`). Given our mailing list subscription numbers I think this would be best primarily published as a blog post on Hyperledger (who should I talk to?). I'll then syndicate from Monax social media and blog channels. In the post I will sum up Burrow features, talk about new and experimental features, and beg for documentation help.

My aim is to exclusively focus on documentation contribution - which would include writing some example code. If anyone wants to get up to speed on Burrow this would be a good way to do it. We are not looking for perfect docs at all. The markdown pre-processor testing harness might be moderately interesting for other project/in itself if you are in to that sort of thing.

Any advice/help from TSC appreciated.

## Releases

- **v0.27.0** - Support for WASM contracts compiled with solang
- **v0.26.2** - Fix non-deterministic block time on restart
- **v0.26.1** - Remote burrow dump, no empty blocks (massive space saving)
- **v0.26.0** - Tendermint upgrade, vent store chain\_id for restore chains

## Overall Activity in the Past Quarter

There has been plenty of development, the vast majority from Monax employees, including:

- Experimental WASM support
- Vent supports append only SQL log mode - supporting SQL-based event queues from EVM events
- Elective validator bonding - validators may now bond and unbond themselves as they come online (on develop)
- On-chain Ethereum ABI storage and contract metadata - EVM contracts now self-describe names, functions, signatures, types (develop)
- Massive performance improvement in merkle tree by increasing node size
- Command line tooling for forensics (why did this production validator break with quorum?)
- Command line tooling for sending transaction (without a deploy.yaml script)

## Current Plans

As above we need to have a push on explaining Burrow and documenting.

Technical challenges include:

- Improving our state mechanism - we use <https://github.com/tendermint/iavl> but it has scaling issues with large state trees - we have a experimental replacement in the works: <https://github.com/monax/trieste>
- Expanding our WASM syscalls - we would like to collaborate with Transact and will consider what eWASM is defining
- State channels via ad-hoc Tendermint sub-networks (for privacy, for scaling)

- Scheduled transactions - ability to schedule transactions for a future block height and/or time - including ProposalTx

## Maintainer Diversity

No new maintainers. We have 4 from Monax, 2 non-Monax

## Contributor Diversity

We have 3 new non-Monax contributors.

## Reviewed by

- Arnaud Le Hors
- Baohua Yang
- Binh Nguyen
- Christopher Ferris
- Dan Middleton
- Hart Montgomery
- Kelly Olson
- Mark Wagner
- Mic Bowman
- Nathan George
- Silas Davis