Harmony

Open Consensus for 10 Billion

Hungry & Foolish in Cupertino

Stephen

Protocol PhD

Nicolas

VR Startup Founder

Alok

Apple Siri

Rongjian

Google Search

Minh

Google Voice AI

Nick

Stanford AI

Sahil

Harvard Business

Eugene

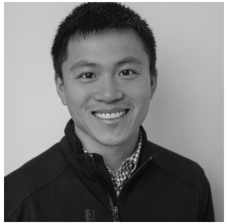
AWS Networking

Leo

Amazon Lab126

Kunal

Samsung Security



Li

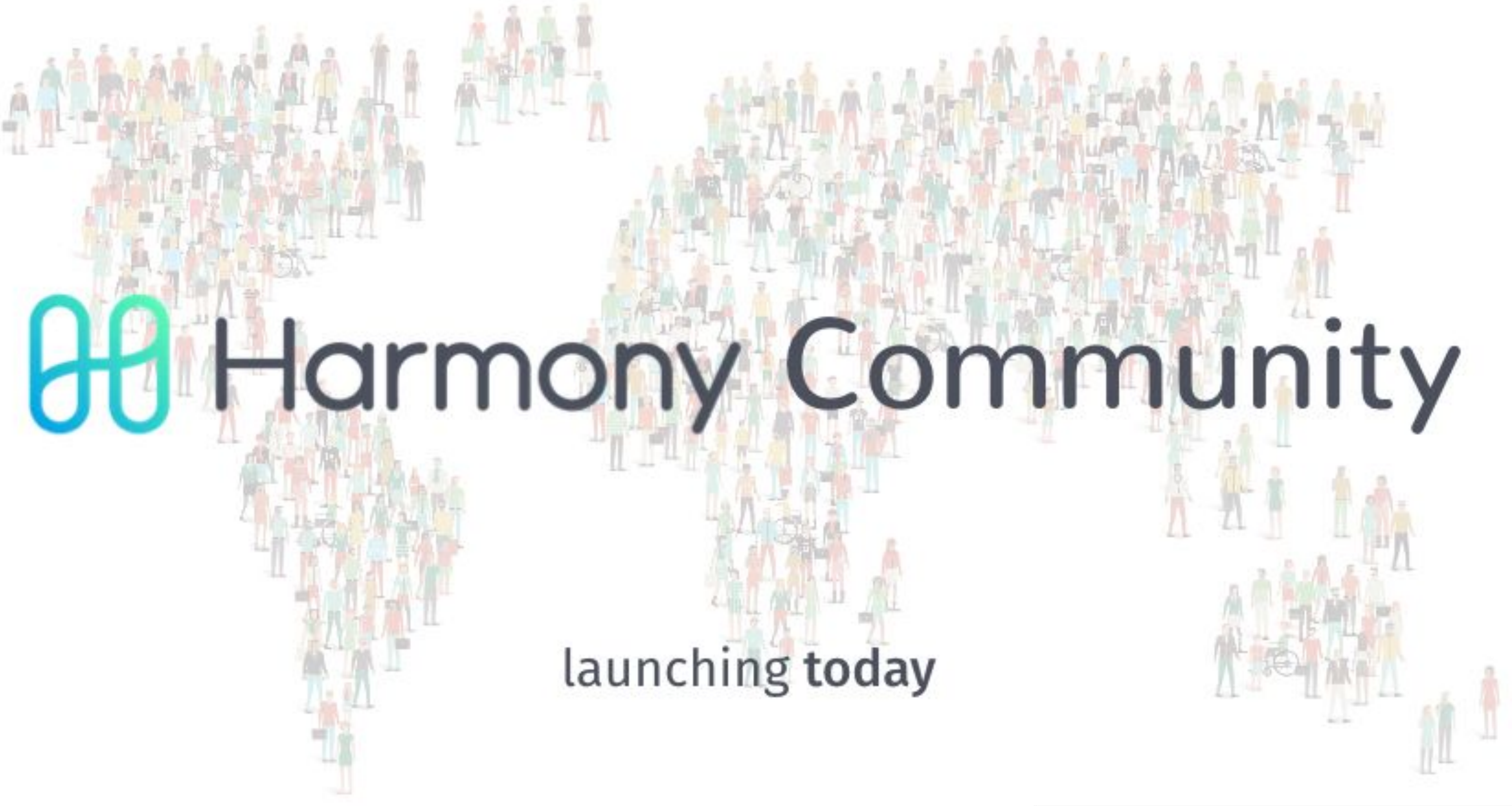
GSV Capital

Chao

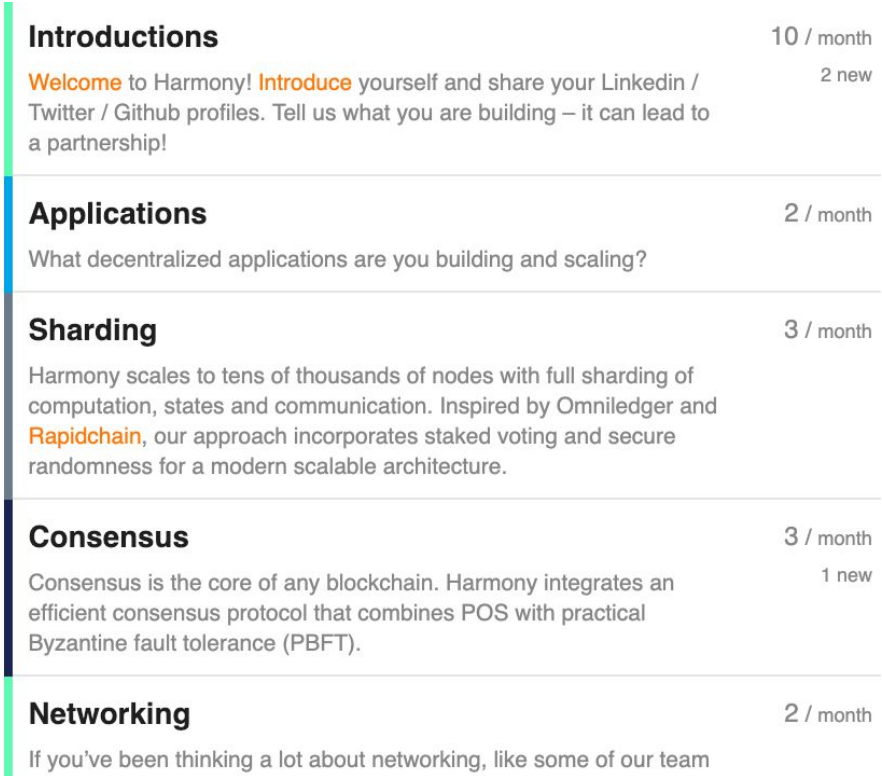
Math PhD

Google Microsoft amazon Penn University of Pennsylvania Carnegie Mellon University HARVARD UNIVERSITY Stanford University

Bring 10X Research & **Open Consensus**



Join Our Conversations at talk.harmony.one



Introductions 10 / month 2 new

Welcome to Harmony! Introduce yourself and share your Linkedin / Twitter / Github profiles. Tell us what you are building – it can lead to a partnership!

Applications 2 / month

What decentralized applications are you building and scaling?

Sharding 3 / month

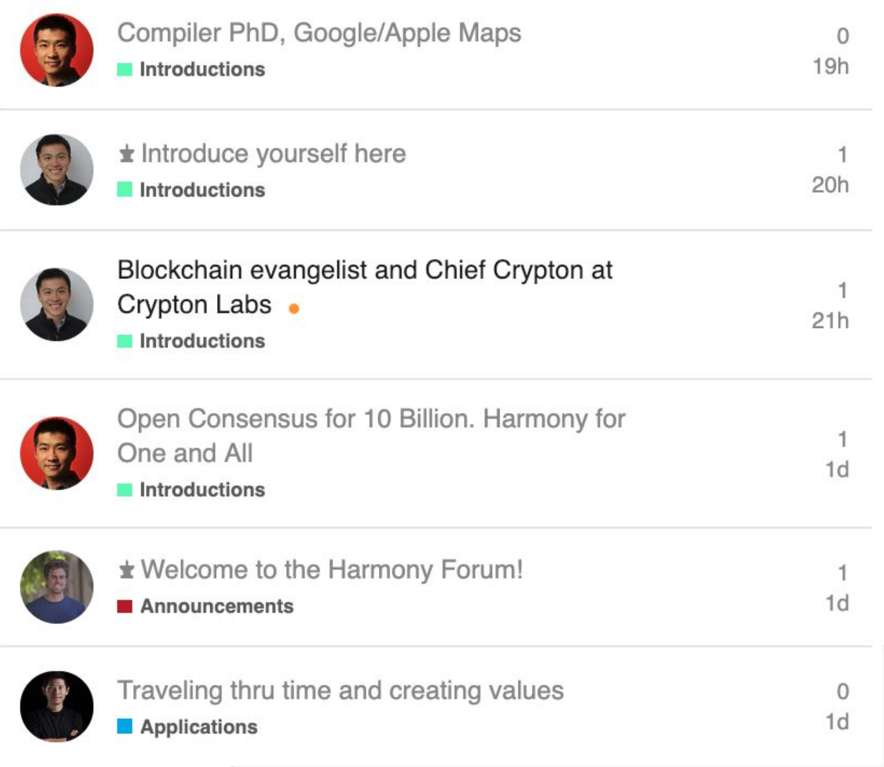
Harmony scales to tens of thousands of nodes with full sharding of computation, states and communication. Inspired by Omniledger and Rapidchain, our approach incorporates staked voting and secure randomness for a modern scalable architecture.

Consensus 3 / month 1 new

Consensus is the core of any blockchain. Harmony integrates an efficient consensus protocol that combines POS with practical Byzantine fault tolerance (PBFT).

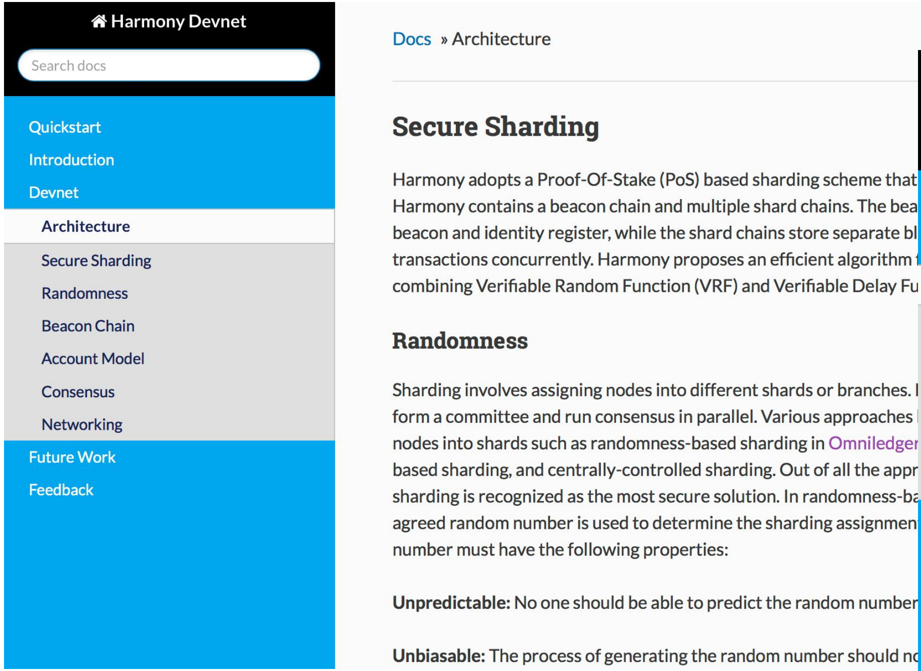
Networking 2 / month

If you've been thinking a lot about networking, like some of our team



Compiler PhD, Google/Apple Maps 🟩 Introductions	0	19h
📌 Introduce yourself here 🟩 Introductions	1	20h
Blockchain evangelist and Chief Crypton at Crypton Labs ● 🟩 Introductions	1	21h
Open Consensus for 10 Billion. Harmony for One and All 🟩 Introductions	1	1d
📌 Welcome to the Harmony Forum! 🟥 Announcements	1	1d
Traveling thru time and creating values 🟦 Applications	0	1d

Launching devnet.harmony.one!



Harmony Devnet

Search docs

- Quickstart
- Introduction
- Devnet
  - Architecture
    - Secure Sharding
    - Randomness
    - Beacon Chain
    - Account Model
    - Consensus
    - Networking
- Future Work
- Feedback

Docs » Architecture

Secure Sharding

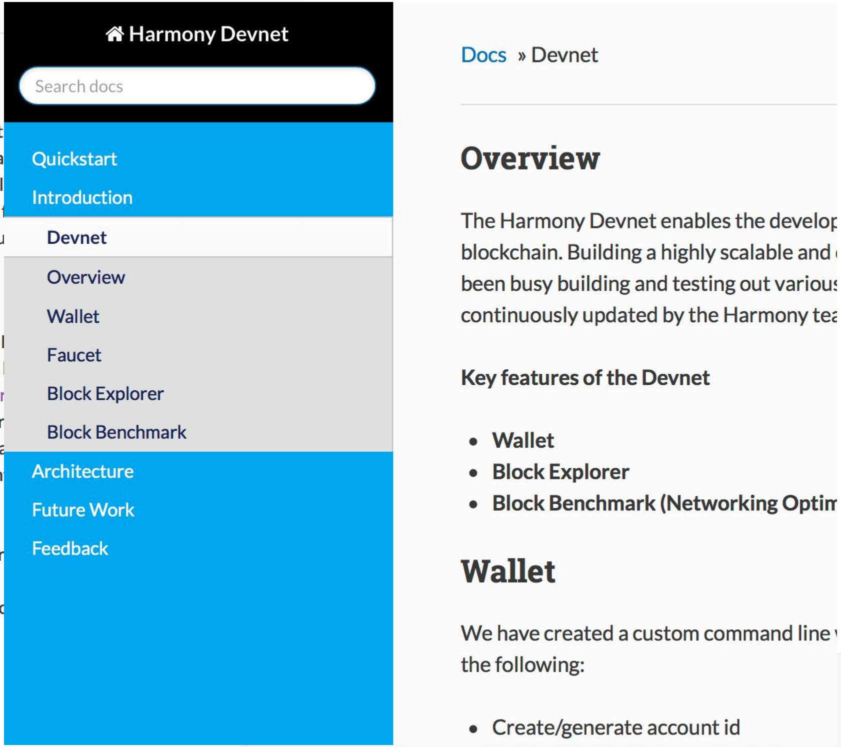
Harmony adopts a Proof-Of-Stake (PoS) based sharding scheme that
Harmony contains a beacon chain and multiple shard chains. The bea
beacon and identity register, while the shard chains store separate bl
transactions concurrently. Harmony proposes an efficient algorithm
combining Verifiable Random Function (VRF) and Verifiable Delay Fu

Randomness

Sharding involves assigning nodes into different shards or branches.
form a committee and run consensus in parallel. Various approaches
nodes into shards such as randomness-based sharding in Omniledger
based sharding, and centrally-controlled sharding. Out of all the appr
sharding is recognized as the most secure solution. In randomness-ba
agreed random number is used to determine the sharding assignmen
number must have the following properties:

Unpredictable: No one should be able to predict the random number

Unbiasable: The process of generating the random number should no



Harmony Devnet

Search docs

- Quickstart
- Introduction
- Devnet
  - Overview
  - Wallet
  - Faucet
  - Block Explorer
  - Block Benchmark
- Architecture
- Future Work
- Feedback

Docs » Devnet

Overview

The Harmony Devnet enables the develop
blockchain. Building a highly scalable and
been busy building and testing out various
continuously updated by the Harmony tea

Key features of the Devnet

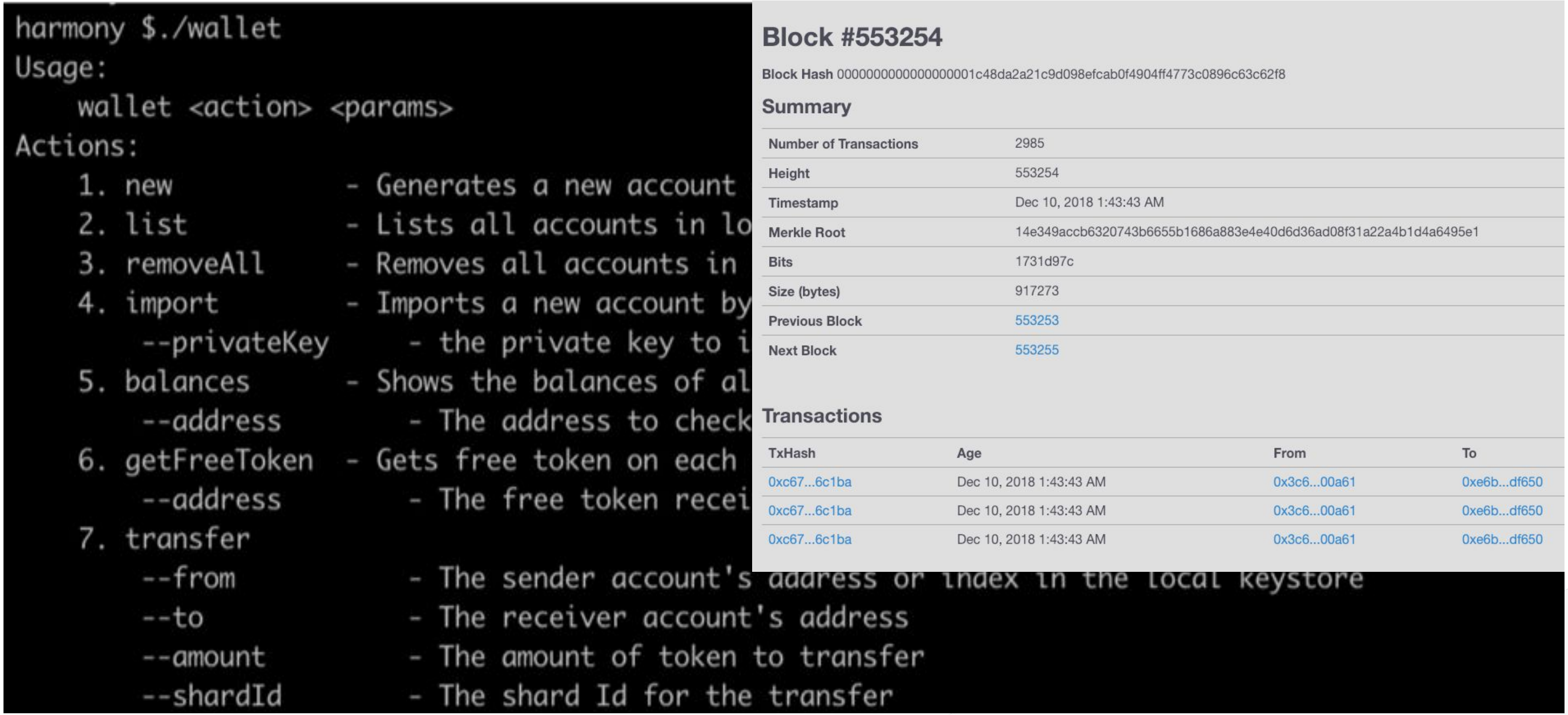
- **Wallet**
- **Block Explorer**
- **Block Benchmark (Networking Optim**

Wallet

We have created a custom command line
the following:

- Create/generate account id

Wallet, explorer, **sharding** ready



```
harmony $./wallet
Usage:
    wallet <action> <params>
Actions:
    1. new            - Generates a new account
    2. list           - Lists all accounts in lo
    3. removeAll      - Removes all accounts in
    4. import         - Imports a new account by
        --privateKey      - the private key to i
    5. balances       - Shows the balances of al
        --address         - The address to check
    6. getFreeToken   - Gets free token on each
        --address         - The free token recei
    7. transfer
        --from            - The sender account's address or index in the local keystore
        --to              - The receiver account's address
        --amount          - The amount of token to transfer
        --shardId         - The shard Id for the transfer
```

Block #553254

Block Hash 0000000000000000001c48da2a21c9d098efcab0f4904ff4773c0896c63c62f8

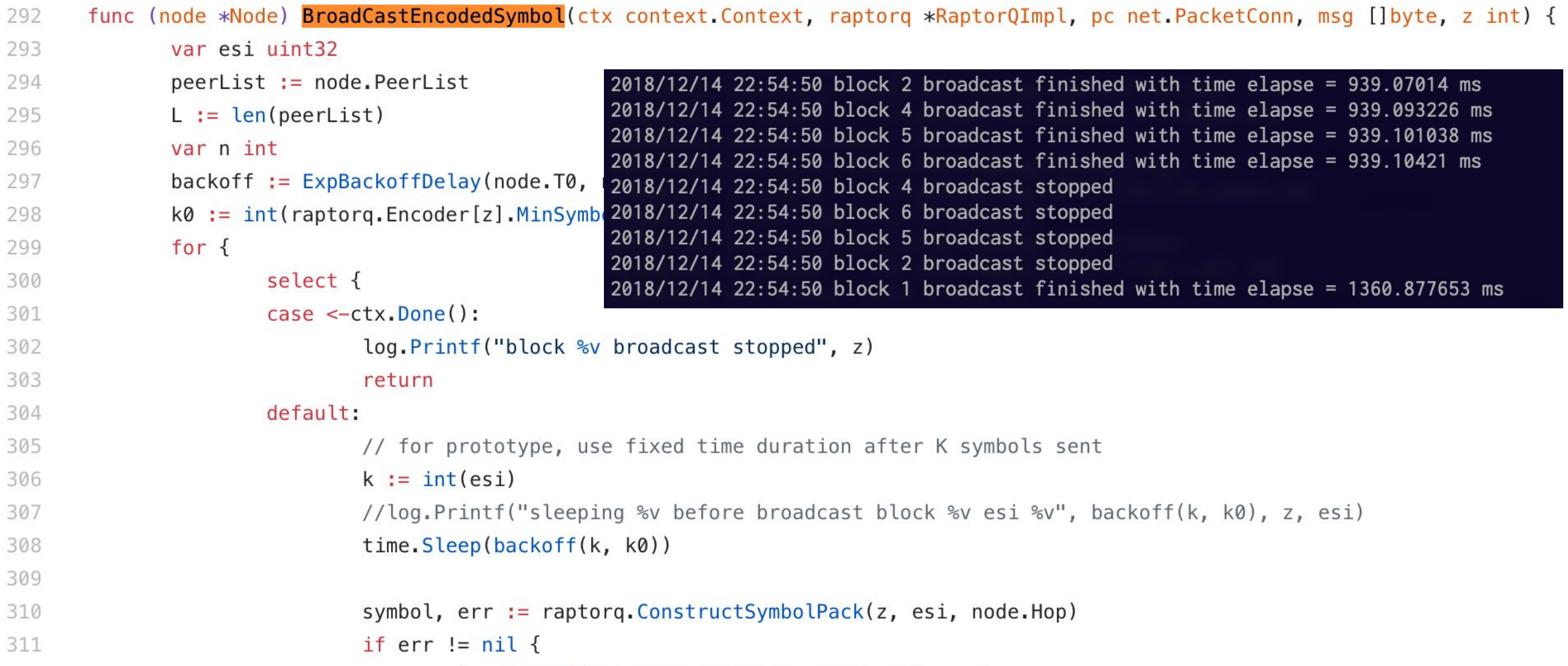
Summary

Number of Transactions	2985
Height	553254
Timestamp	Dec 10, 2018 1:43:43 AM
Merkle Root	14e349accb6320743b6655b1686a883e4e40d6d36ad08f31a22a4b1d4a6495e1
Bits	1731d97c
Size (bytes)	917273
Previous Block	553253
Next Block	553255

Transactions

TxHash	Age	From	To
0xc67...6c1ba	Dec 10, 2018 1:43:43 AM	0x3c6...00a61	0xe6b...df650
0xc67...6c1ba	Dec 10, 2018 1:43:43 AM	0x3c6...00a61	0xe6b...df650
0xc67...6c1ba	Dec 10, 2018 1:43:43 AM	0x3c6...00a61	0xe6b...df650

1.36s Speed w/ Adaptive RaptorQ Encoding



```go
func (node *Node) BroadCastEncodedSymbol(ctx context.Context, raptorq *RaptorQImpl, pc net.PacketConn, msg []byte, z int) {
    var esi uint32
    peerList := node.PeerList
    L := len(peerList)
    var n int
    backoff := ExpBackoffDelay(node.T0,
    k0 := int(raptorq.Encoder[z].MinSymb
    for {
        select {
        case <-ctx.Done():
            log.Printf("block %v broadcast stopped", z)
            return
        default:
            // for prototype, use fixed time duration after K symbols sent
            k := int(esi)
            //log.Printf("sleeping %v before broadcast block %v esi %v", backoff(k, k0), z, esi)
            time.Sleep(backoff(k, k0))

            symbol, err := raptorq.ConstructSymbolPack(z, esi, node.Hop)
            if err != nil {
```

```
2018/12/14 22:54:50 block 2 broadcast finished with time elapse = 939.07014 ms
2018/12/14 22:54:50 block 4 broadcast finished with time elapse = 939.093226 ms
2018/12/14 22:54:50 block 5 broadcast finished with time elapse = 939.101038 ms
2018/12/14 22:54:50 block 6 broadcast finished with time elapse = 939.10421 ms
2018/12/14 22:54:50 block 4 broadcast stopped
2018/12/14 22:54:50 block 6 broadcast stopped
2018/12/14 22:54:50 block 5 broadcast stopped
2018/12/14 22:54:50 block 2 broadcast stopped
2018/12/14 22:54:50 block 1 broadcast finished with time elapse = 1360.877653 ms
```

A SUPERMASSIVE STORY
POPCORN BRAIN PRESENTS
HARMONY ONE
"THE NEW HEROES OF SPEED"
STEPHEN TSE
NICOLAS BURTEY
ALOK KOTHARI
LEO CHEN
RONGJIAN LAN
SAHIL DEWAN
MINH DOAN
KUNAL PATEL
NICK WHITE
EUGENE KIM

和谐号
Meta

rtant now?
General Data Protection Regulation

568

Bring Research to Production w/ Community

https://talk.harmony.one

Build (Peer) Apps w/ Byzantine Gossips

harmony.one/libunison

Bring 10M Users & 100K Nodes

harmony.one/partners

Mainnet Launch in 2019 Q2

harmony.one/newsletter

harmony.one