Grin Meetup
16.05.2019 // Moscow

Grin overview

@lehnberg

Contents

- Protocol
- Project
- Implementation
- Status
- Contributing
- Questions

Grin tl;dr

Better money ツ

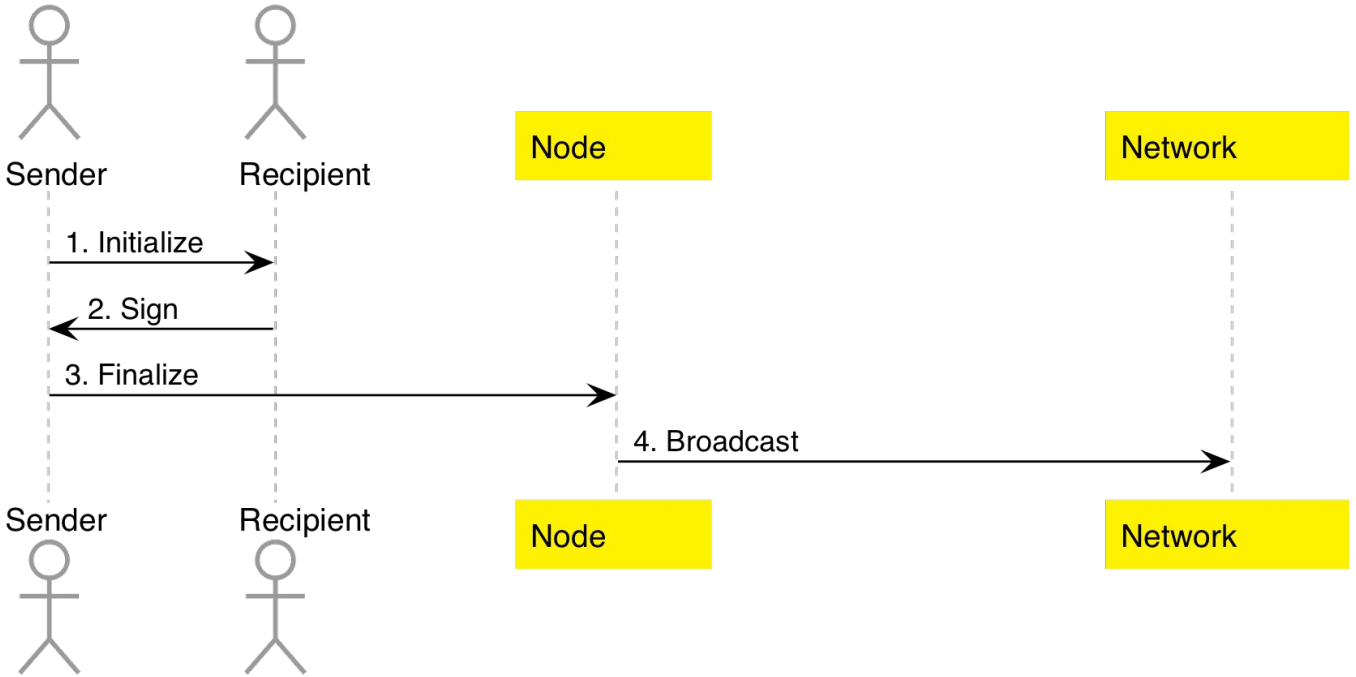
A lightweight implementation of a cryptocurrency that aims to be privacy preserving, scalable, and fair.

Mimblewimble

Mimblewimble

- Blockchain design proposed by Jedusor (2016), improved by Poelstra (2016).
- Relies on Confidential Transactions (Maxwell 2016), CoinJoin (Maxwell 2013), and OWAS (Mouton 2013)
- Scalable: Little data required for full sync.
- Fungible: No amounts, no scripts, no addresses, no non-confidentiality, in a simple protocol.
- Requires interaction to build transactions.

Interactive transaction format



Transaction Structure

Input

Existing UTXO

Output

New commitment

Kernel

Excess and signature

Transaction Structure

Input

Existing UTXO

Output

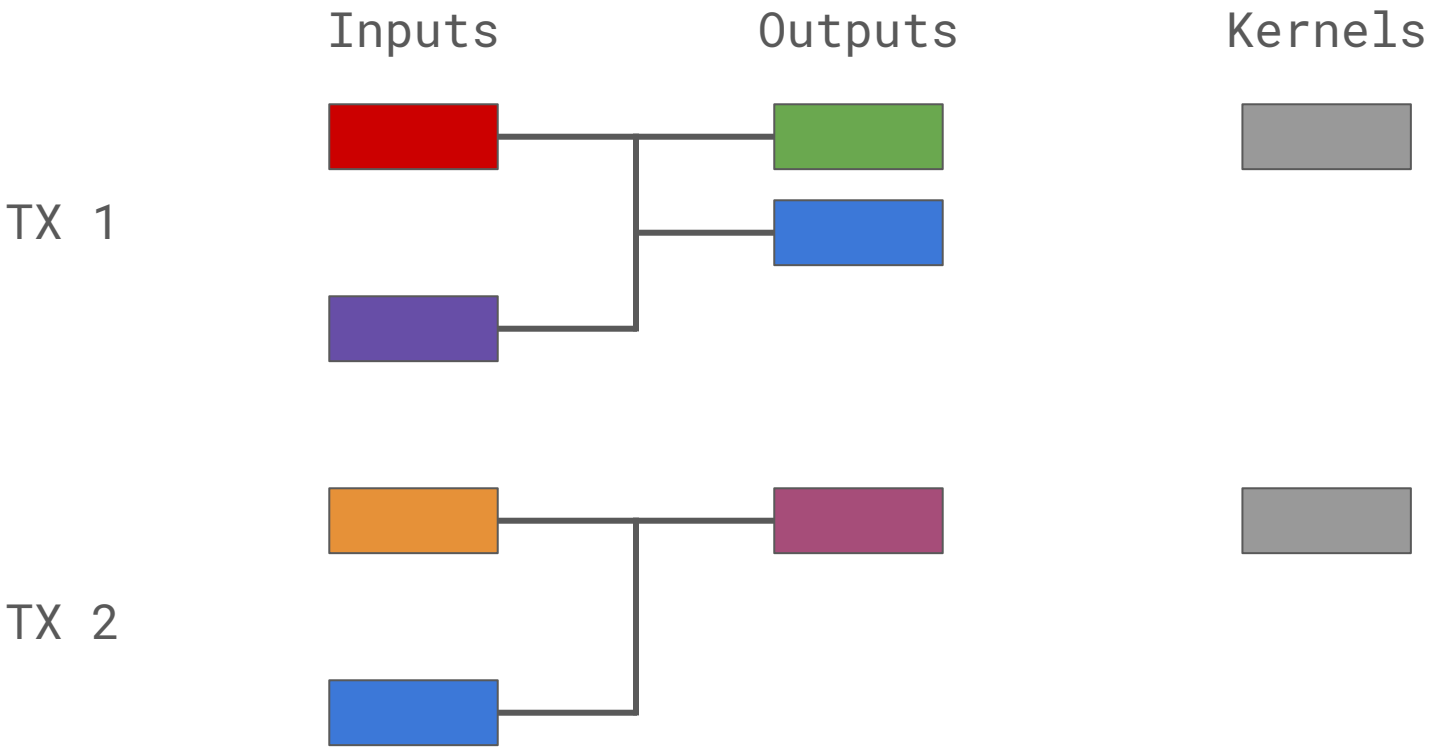
New commitment

Kernel

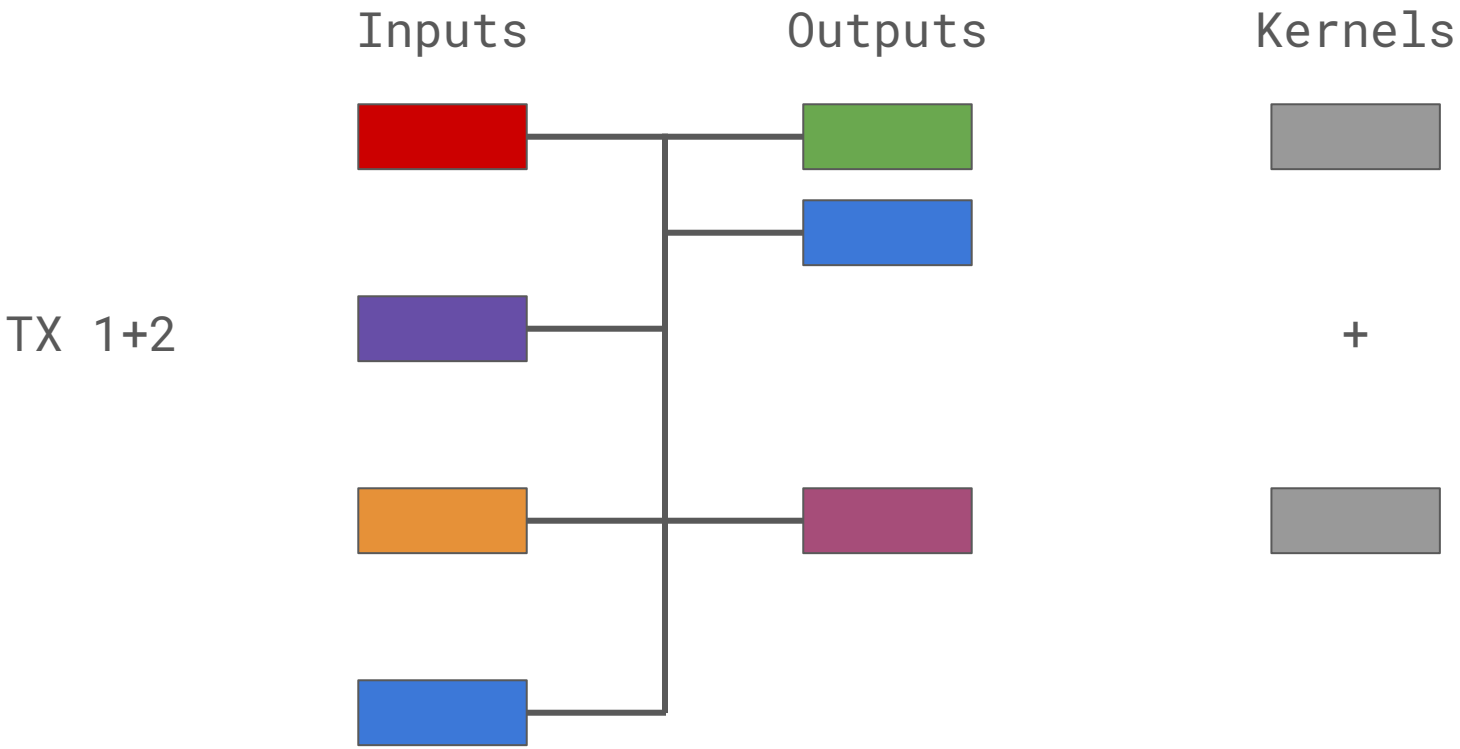
Excess and signature

TX validation:
sum outputs - sum inputs == sum excess

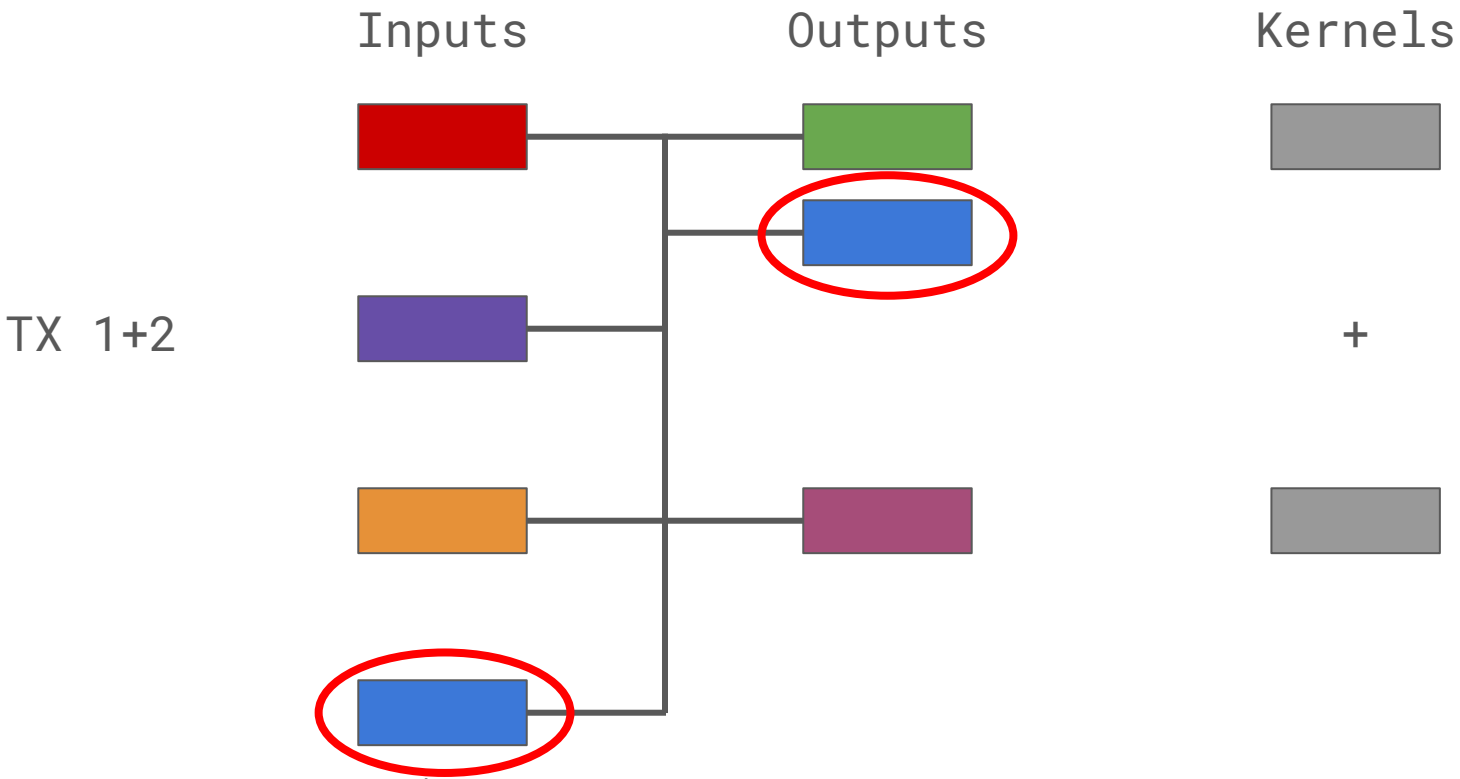
Transactions...



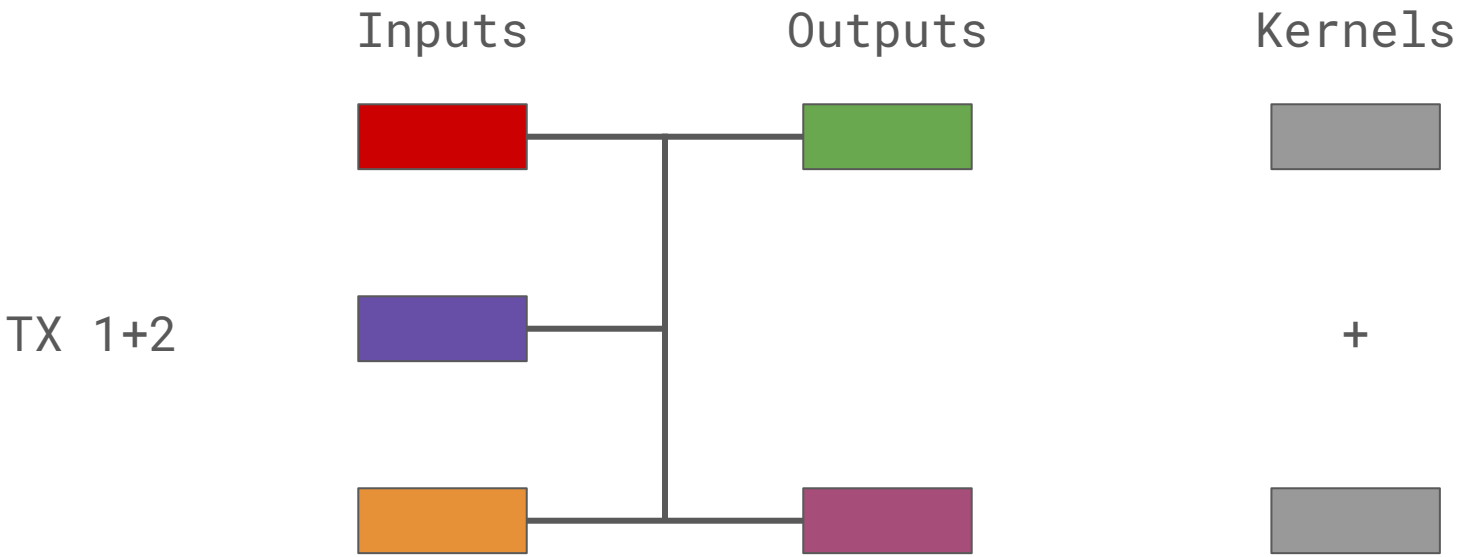
...can be joined together.



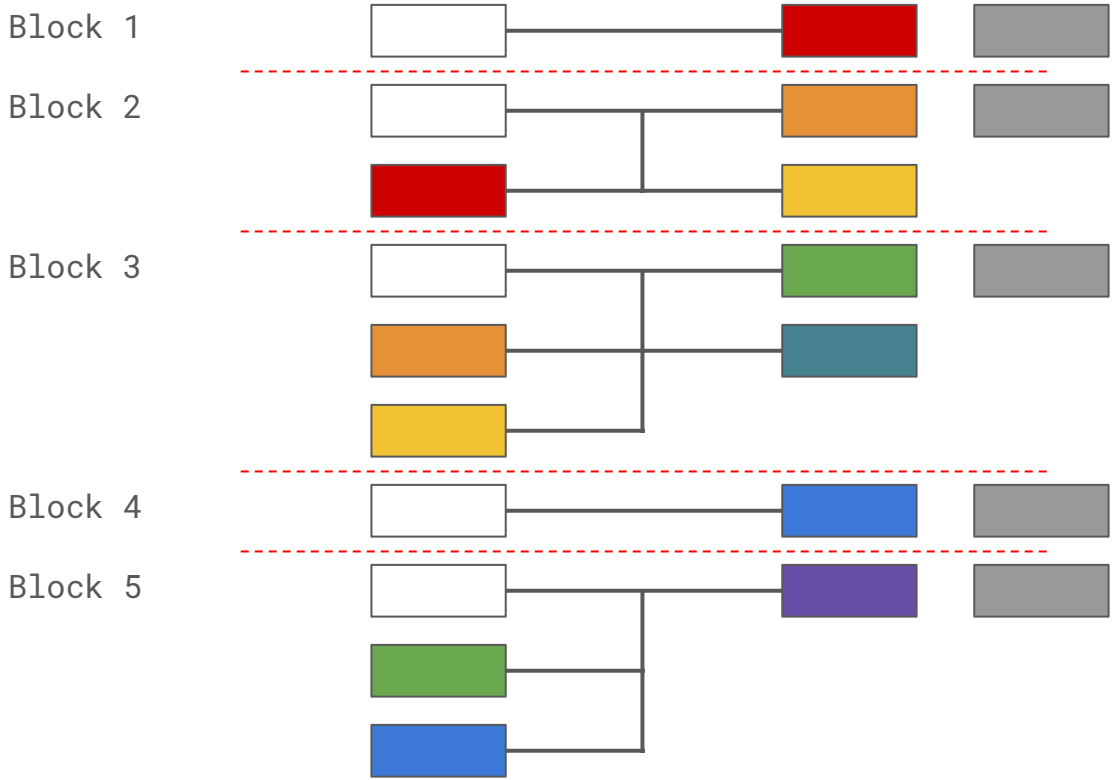
And outputs later used as inputs...



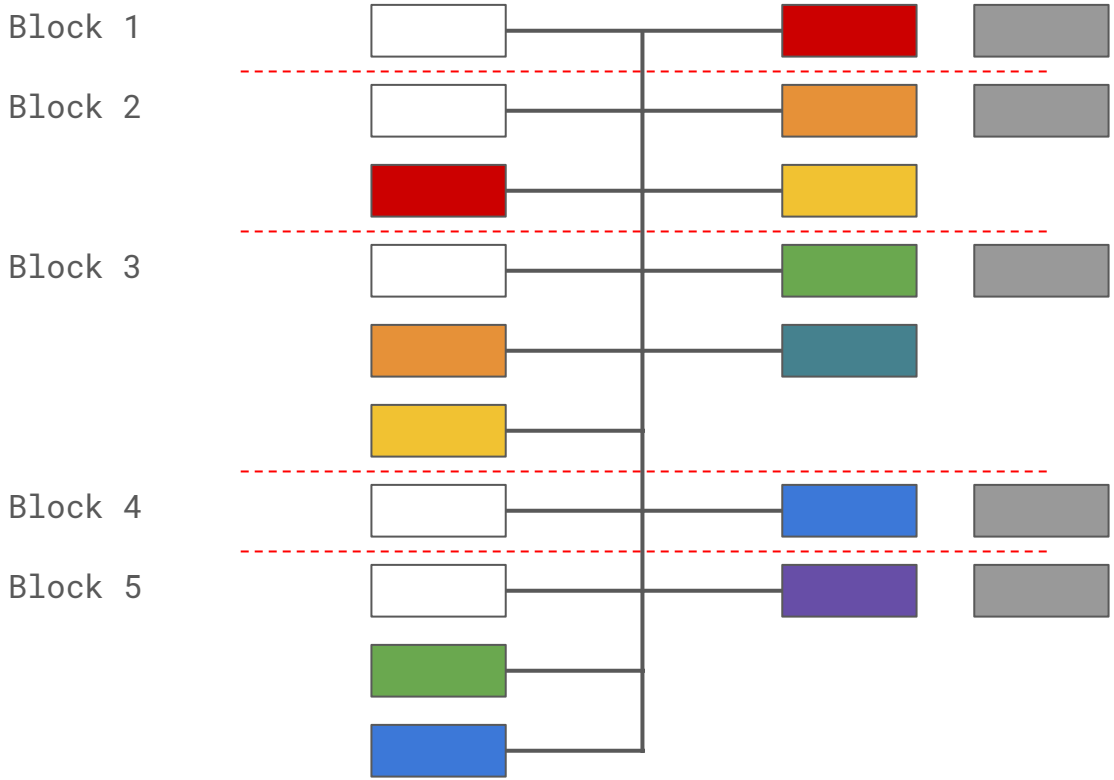
...can be discarded.



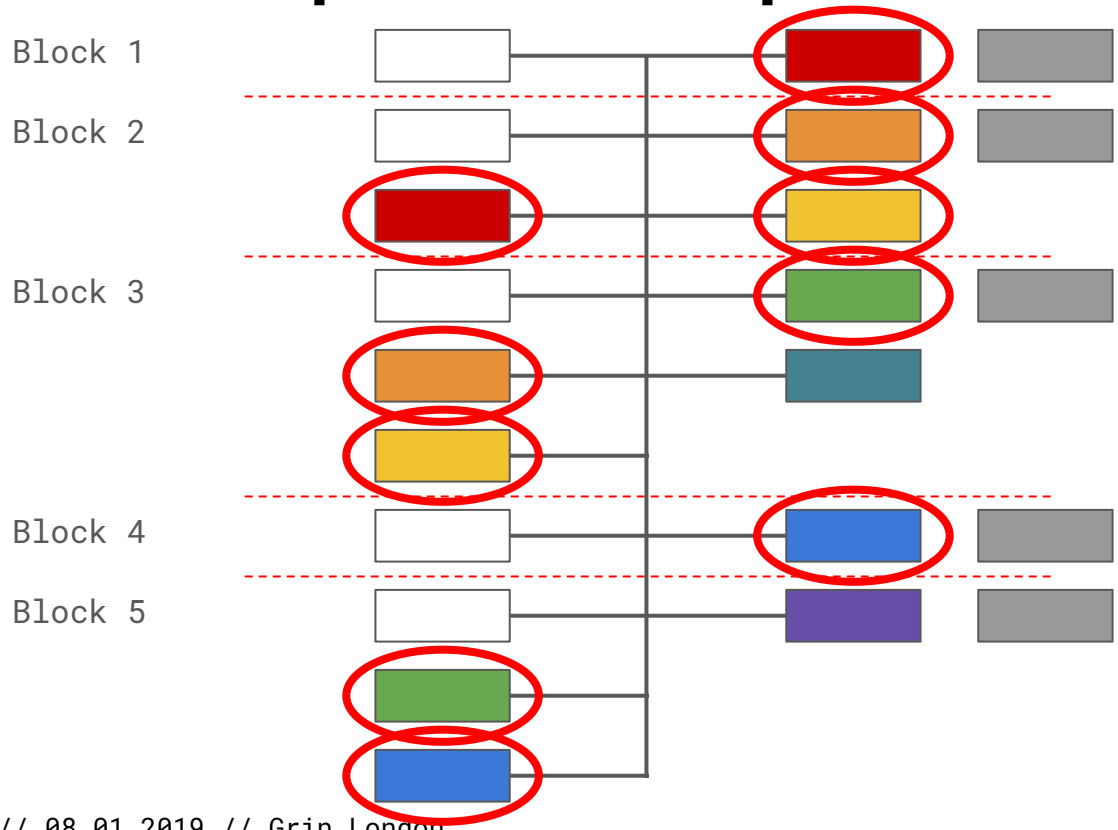
Similarly, the blockchain...



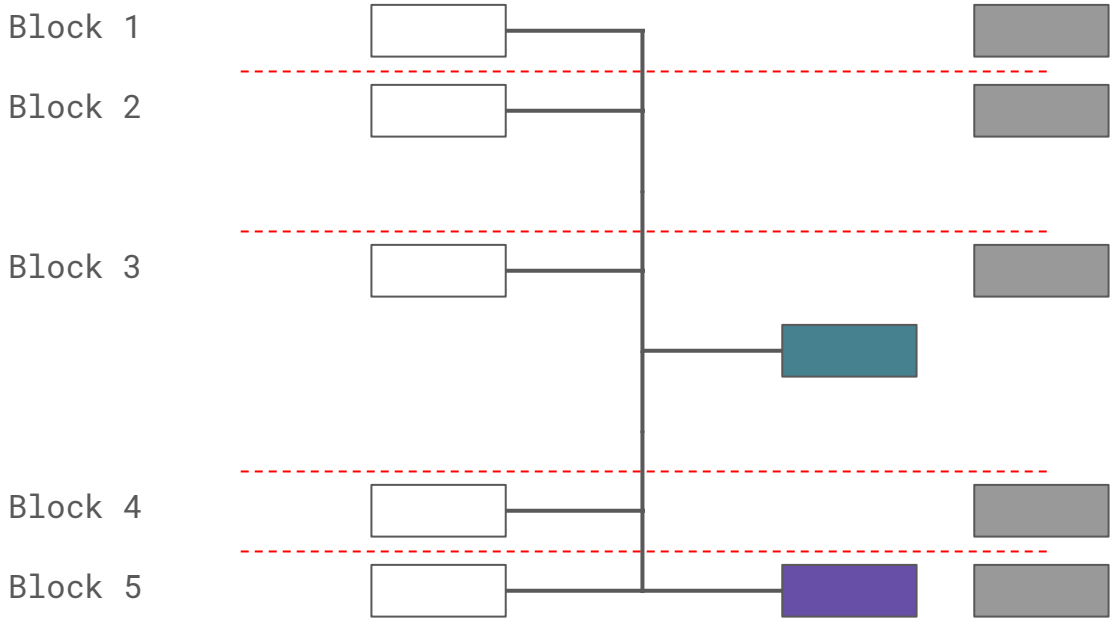
...can be joined.



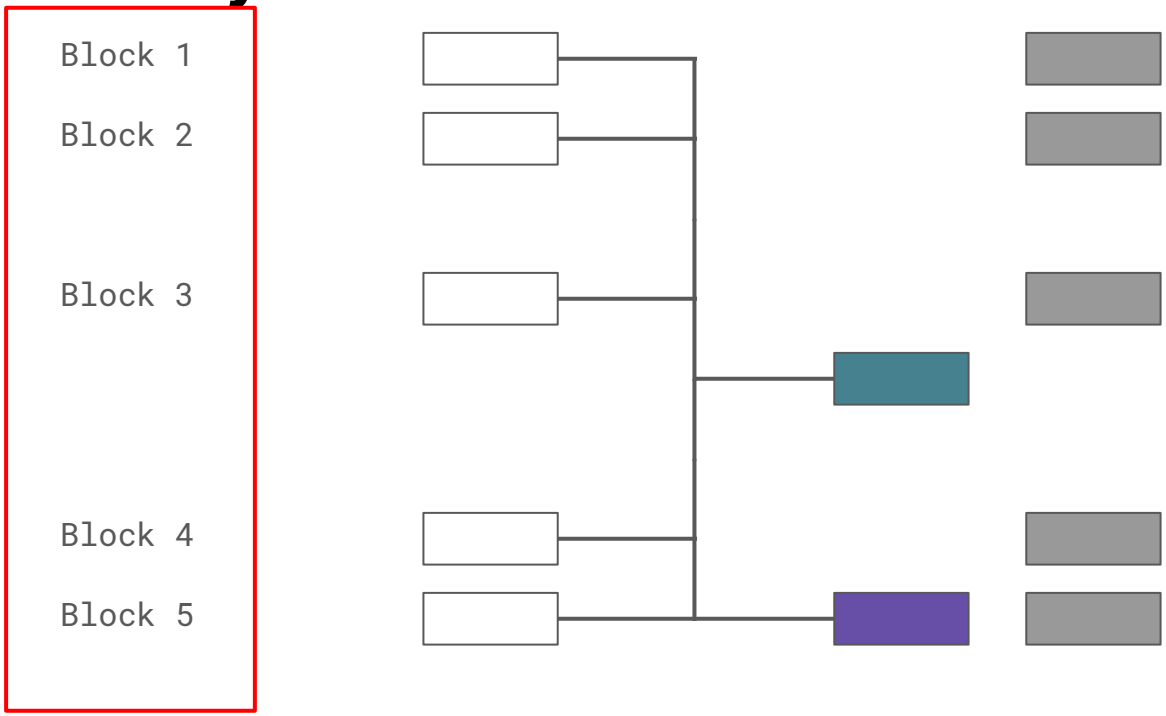
And when outputs are spent...



...they can be removed.

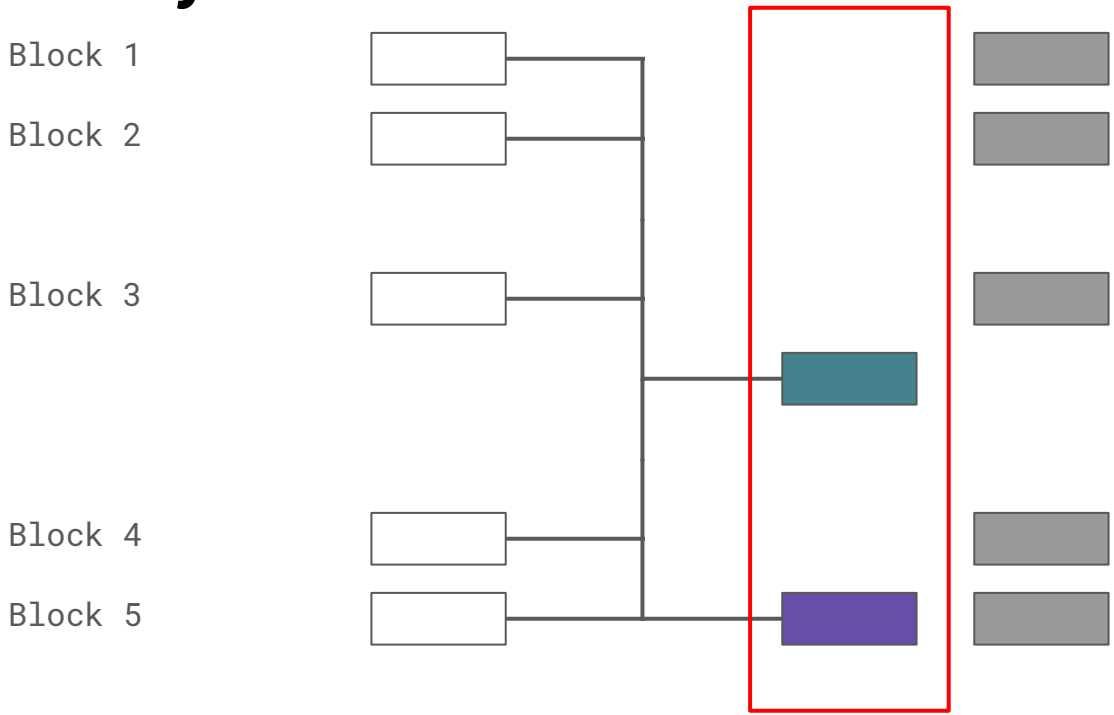


Initial sync

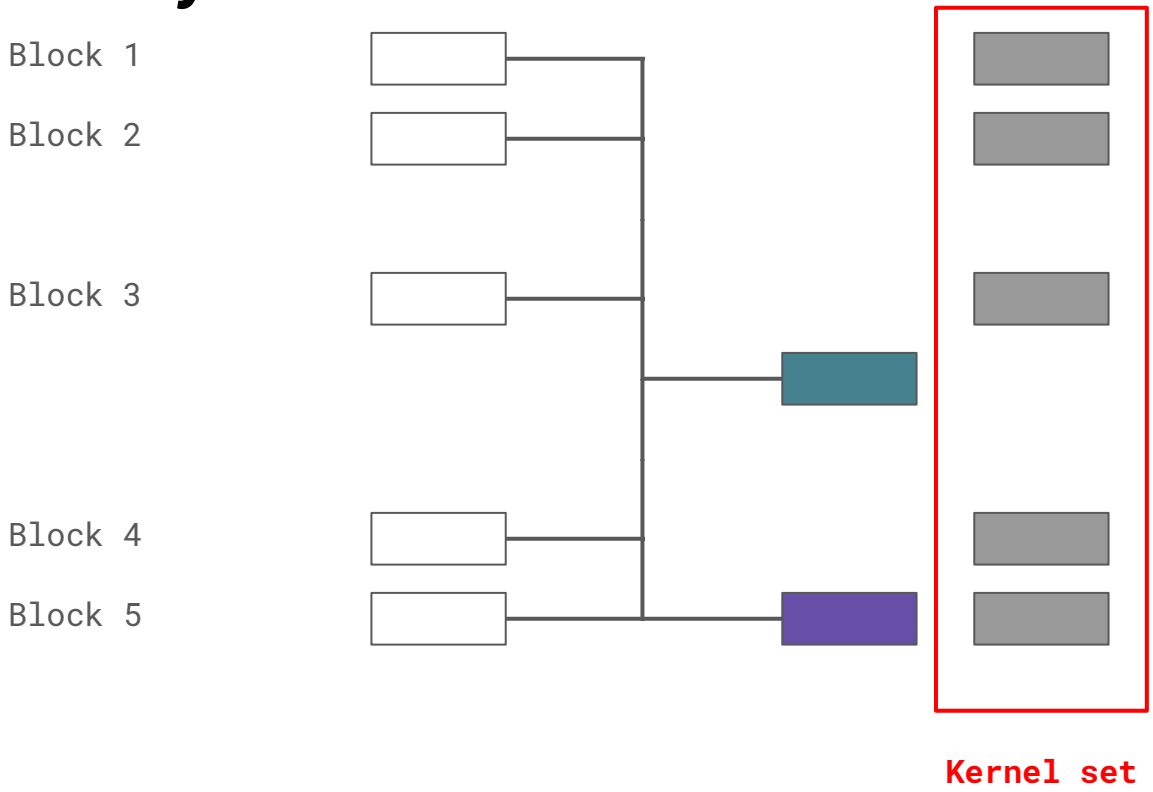


Block headers

Initial sync



Initial sync



Mimblewimble

Pros:

+ Elegant: Minimal data kept on chain
+ Simple: Relies on tried and tested cryptography
+ Performant: Improved syncing

Cons:

- An interactive protocol
- Some output linking is still possible
- No scripting (but scriptless scripts)

Project

Grin

Announced October 20th, 2016 by “Ignotus Peverell”

First Mimblewimble implementation

Written in Rust

Open source, 100% community driven

Funded by donations

No: ICO, CEO, DevCo, advisors, investors, founder rewards, premines, pre-allocation, pre...

Words I use to describe the project

- Open
- Fair
- Honest
- Minimal
- Rational
- Transparent

Governance

KISS

No foundation

Technocratic council

Constantly evolving work in progress

Decisions taken in the open in bi-weekly development and governance meetings where possible

Implementation

Technologies used (sub-set)

Schnorr signatures: Smaller, multi-sig, scriptless scripts, certifiable transactions

Bulletproofs: Smaller ZK range proof for CT

Dandelion: Privacy-preserving transaction propagation and aggregation

Scriptless scripts: Enables atomic swaps in Grin and some other scripting behavior.

Emission

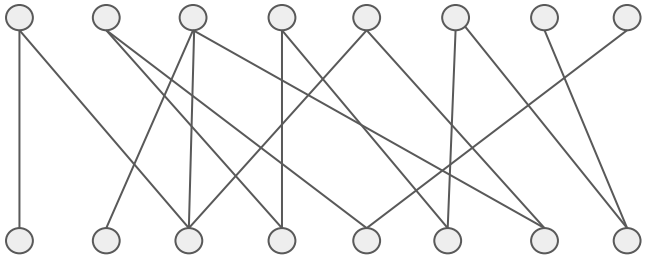
1 Grin/s forever.

Proof of work mined.
One minute block time.
60 grin constant coinbase reward.

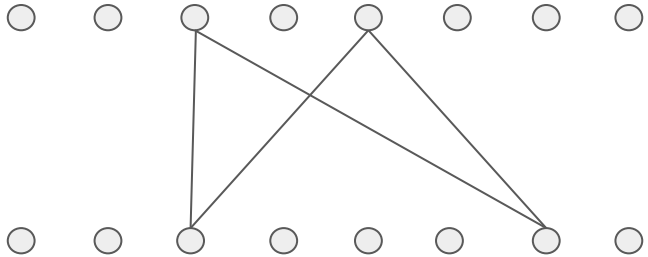
Simple. Incentivises spending. Discourages unfair advantage for early adopters for the benefit of improved longer term adoption.

Proof of Work

Cuckoo Cycle family. Finding 42-cycles in random bipartite graphs with billions of nodes. Creator: John Tromp



Begin with a mess



End up with a cycle

Status

Development

Join the next meeting:
May 28 @ 3PM UTC on Gitter

v1.1.0 (end of May 2019):
Windows support, Wallet API v2, standalone wallet, dandelion++, variable size MMRs

Active research:
I2P support, verifiable transaction, node API v2, new tx slate format

Governance

Join the next meeting:
May 21 @ 3PM UTC on Gitter

Decision on Security Auditing firm

Income and spending logs public on /grin-pm

Q12019 transparency report published

2019 budget in progress

Anonymous donation of BTC mined in 2010 received last week

Security

libsecp fork audited by JP Aumasson before launch

Coinspect audited our crates, report received last week

Critical vulnerability CVE-2019-9195 discovered, patch released within 72 hours

Additional findings from audit are being patched on ongoing basis

We are reviewing our vulnerability disclosure and security processes, please participate

Mainnet is 120 days old. Grin is *highly* experimental software.

Ecosystem

5+ mining pools, more to be announced

5+ different mining software providers

10+ exchanges added Grin voluntarily so far

3+ ASICs announced

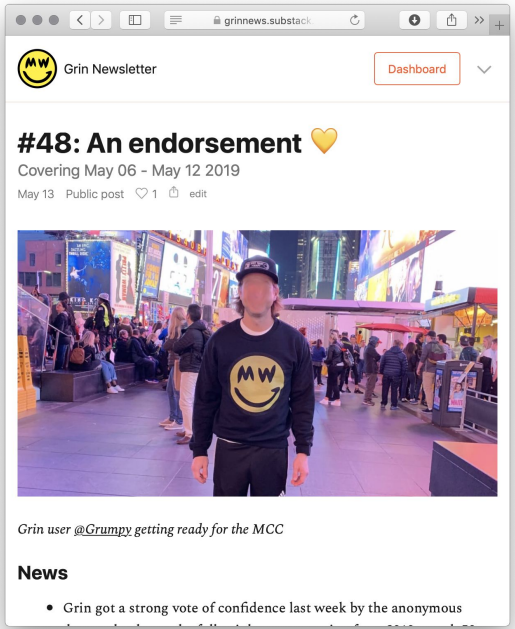
Grin++: a C++ implementation of node + wallet

cycle42: Building mobile wallet & merchant processor

vault713: Building wallet, transaction protocol, and atomic swaps

Stay up to date

https://grinnews.substack.com



Contributing

We need you!

Rust developers

Researchers

Frontend developers

UI/UX specialists

Graphic designers

Technical writers

Community members

“JUST DO IT.”
- @hashmap

Get involved

Don’t ask for permission, the project is open source.

https://github.com/mimblewimble/

Fund the project

A good way to protect your grins and the integrity of mankind.

https://grin-tech.org/funding

Take a technical crash course

https://github.com/mimblewimble/grin-pm#presentations

grincon0, Berlin
What is Grin
Contributing
Dandelion
Wallet
Atomic Swaps
Proof of Work
Panel

grincon.US, San Mateo CA
Grin value proposition
Security audit
Scriptless scripts
How to mine
Privacy & scalability
RSA Accumulators
...and more

Спасибо

telegram/gitter/keybase/twitter: @lehnberg