RGB: Private and scalable smart contracts for Bitcoin and Lightning Network

RGB

Smart contract is

- A pre-arranged *agreement*
- of trade (i.e. mutual voluntary exchange of *goods*)
- automatically executed under certain *conditions*,
  where “automatic” means
  - ✴ anonymous: no KYC is done
  - ✴ trustless: no need to do KYC to protect from the failure to execute contract

Smart contract components

- Agreement: the code
- Goods: digital assets
- Conditions: contract parties or external actors able to call some code

Ownership & access: core properties

- **Ownership**: digital assets must be owned by a well-defined party
- **Access**: only well-defined parties should be able to call the contract execution

Pure blockchain/layer 1 approach is wrong:

- Mixing ***code***, ***ownership*** and ***access rights*** into a single layer (“blockchain”)
- which is inherently ***unscalable*** and well-trackable (***anti-privacy***) since VERIFICATION is needed by the whole world
- With Turing-complete ***code*** operating at the same level, ***compromising security***
- Running ***non-censorship-resistant*** consensus algorithms (PoS, PoW forks with small hashing power)

Challenges with digital assets today

- **Low scalability**
  - limited by blockchains, which are inherently unscalable
  - no layer 2 assets
- **Poor privacy**
  - Everyone in the world sees the transactions
  - Zero-knowledge is nearly absent for the assets even if it is present in blockchain (Monero, Grin, Beam, …)

Challenges with digital assets today

- **Inefficient smart contracts**
  - Asset ownership is mixed with contract business logic
  - Pseudo-decentralized (governance problem)
  - Not formally verified languages (security problem)

RGB was created to solve these issues

RGB history

- Originally proposed by **Giacomo Zucco** & **Peter Todd** in 2016
- Engineered and developed by **Dr. Maxim Orlovsky**
- Supported by **Tether Inc/Bitfinex, Pandora Core AG** & other sponsors in early 2019
- **Pandora Core AG** is the leading technological force behind the development
- Governed by non-profit **LNP/BP Standards Association**, Switzerland

What is RGB?

Client-validated state and smart contract system working at Layer 2/3 in Bitcoin and Lightning Network.

- Works with **Lightning Network**
- No on-chain usage nor trackable footprint: client-validated paradigm
- **Scales** independently from blockchain
- **Zero-knowledge** & privacy built on best research-based products
  - Mimblewhimble: Bulletproofs by Andrew Poelstra
  - Liquid: Confidential Assets by Blockstream

Problem 1: Blockchain does not scale

Problem 2: Blockchain is transparent

Solution: client-side validation

- Let’s not put data into blockchain!
  - Solves scalability problems of Ethereum, EOS and other systems…
  - Now the whole world does not see our transactions
- Use blockchain as a cryptographically commitment layer
  - Commit to some extra-blockchain data with elliptic curve homomorphic properties
- Data/history is maintained by asset owner
- Proposed by Peter Todd

Basic principles of client-side validation in RGB

1. There always must be an owner

- Smart contract state is not a “public good” (Ethereum/ “blockchain” approach); it must always have a well-defined ownership (private, multisig etc).
- RGB defines ownership by binding/assigning state to Bitcoin transaction outputs: whoever controls the output owns the associated state
- I.e. RGB leverages Bitcoin script security model and all its technologies (Schnorr/Taproot etc).

2. State ownership != state validation

- Ownership defines WHO can change the state
- Validation rules (client-side validation) define HOW it may change

2. State ownership != state validation

- Ownership controlled by Bitcoin script, at Bitcoin blockchain level (non-Turing complete)
- Validation rules controlled by RGB Schema with Simplicity script (Turing-complete)

This allows to avoid mistake done by “blockchain smart contracts” (Ethereum/EOS/Polkadot etc): mixing of layers & Turing completeness into non-scalable blockchain layer

Also it makes possible for smart contracts to operate on top of Layer 2 solutions (Lightning Network)

Smart contract language: Simplicity

- Proposed and developed by Russel O’Conner, Blockstream
- Planned to be included into Elements and Liquid
- Formal semantics
- Formally-verified language with proofs on execution
- Succinct (complete Schnorr signatures are just few kB)

Main components of RGB

- 1. Commitments in transactions, proving unique history
  - private
  - zero storage cost
  - work both with blockchain (layer 1) and Lightning Network (layer 2)
  - meaning extreme scalability
- 2. Off-chain data & code held by asset/contract owner
  - zero blockchain storage cost
  - assets are linked to transaction outputs, which define their ownership & prevent double-spending (single-use seals)
  - off-chain smart contract code defines asset evolution

RGB is:

- “Sharding made right”
- “DAG made right”
- “Digital assets made right”
- “Smart contracts made right”
- “Confidentiality made right”

What is possible to do with RGB?

- Fungible assets & securities
  - Centrally or federation-issued
  - Issued anonymously or publicly
  - With possible secondary issuance, demurrage, inflation,
- Different forms of bearer rights
- Non-fungible assets (collectibles, game skins, art tokenization)
- Decentralized digital identity & roaming profiles
- Complex accounting systems & utility tokens

And it’s all:

- Scalable
- Confidential
- Working over Lighting Network
- With DEX functionality
- Operating as a bearer instrument

RGB vs existing alternatives

RGB compared to Liquid Confidential Assets:

- Works with Lightning Network
- Replaces Large Borromean signatures range proofs with modern Bulletproofs
- No blockchain space consumption!
- Universal smart contract system
- Works on Bitcoin mainnet, does not require federation

RGB compared to OMNI/Colored coins/ Counterparty:

- No blockchain consumption
- Much higher privacy
- Works with LN without its modifications

RGB compared to Ethereum/EOS/other "corporate blockchains”:

- RGB is *NOT* a blockchain!
- Works on and with Bitcoin: the only censorship-resistant unconfiscatable hard money

Omni BOLT compared to RGB LN:

- Breaks BOLT message compatibility
- Breaks BOLT tx structure compatibility
- No backports from LND
- No TLV extensions
- Requires separate nodes for OMNI Bolt and Bitcoin LN
- Requires Omni Core backend, can't work with just Bitcoin Core

RGB & other Bitcoin tech

- Interoperable with Liquid
- Does not require changes to LN layer
- Leverages Taproot & Schnorr on the base layer
- Unified invoicing for RGB, Bitcoin and LN
- Can work with existing node implementations, but also has Rust-implementations (BP Node, LNP Node, RGB Node)