

如何实现以太坊

秘猿科技工程师 - 蒋金洋

自我介绍



Jiang Jinyang

jjyr

do nothing and know nothing

Edit bio

📍 Shanghai, China

✉ jjyruby@gmail.com

🌐 <https://justjjy.com>

Overview

Repositories 72

Stars 2.5k

Followers 139

Following 143

Pinned repositories

Customize your pinned repositories

≡ [ciri-ethereum/ciri](#)

Ciri project intent to implement a full feature set Ethereum client.

● Ruby ★ 25

≡ [socketry/lightio](#)

LightIO is a userland implemented green thread library for ruby

● Ruby ★ 156 🍴 3

≡ [bootgo](#)

A barebones OS kernel written in go

● Go ★ 253 🍴 20

≡ [minidown](#)

Minidown is a lightweight & fast markdown parser, with complete GFM support.

● Ruby ★ 140 🍴 9

≡ [sponge](#)

Cryptocurrency exchange arbitrage

≡ [zerotest](#)

Lazy guy's testing tool. Capture HTTP traffic and generate python integration test for your API server.

以太坊价格走势



第一款火了的游戏



猫咪 528614 · 3代 · 急速
♡ 26035



猫咪 529341 · 4代 · 急速
♡ 26020



猫咪 273346 · 6代 · 迟缓
♡ 25814



猫咪 451610 · 13代 · 慢速
♡ 18955



猫咪 403172 · 5代 · 快速



猫咪 503678 · 21代 · 迟缓
♡ 18116

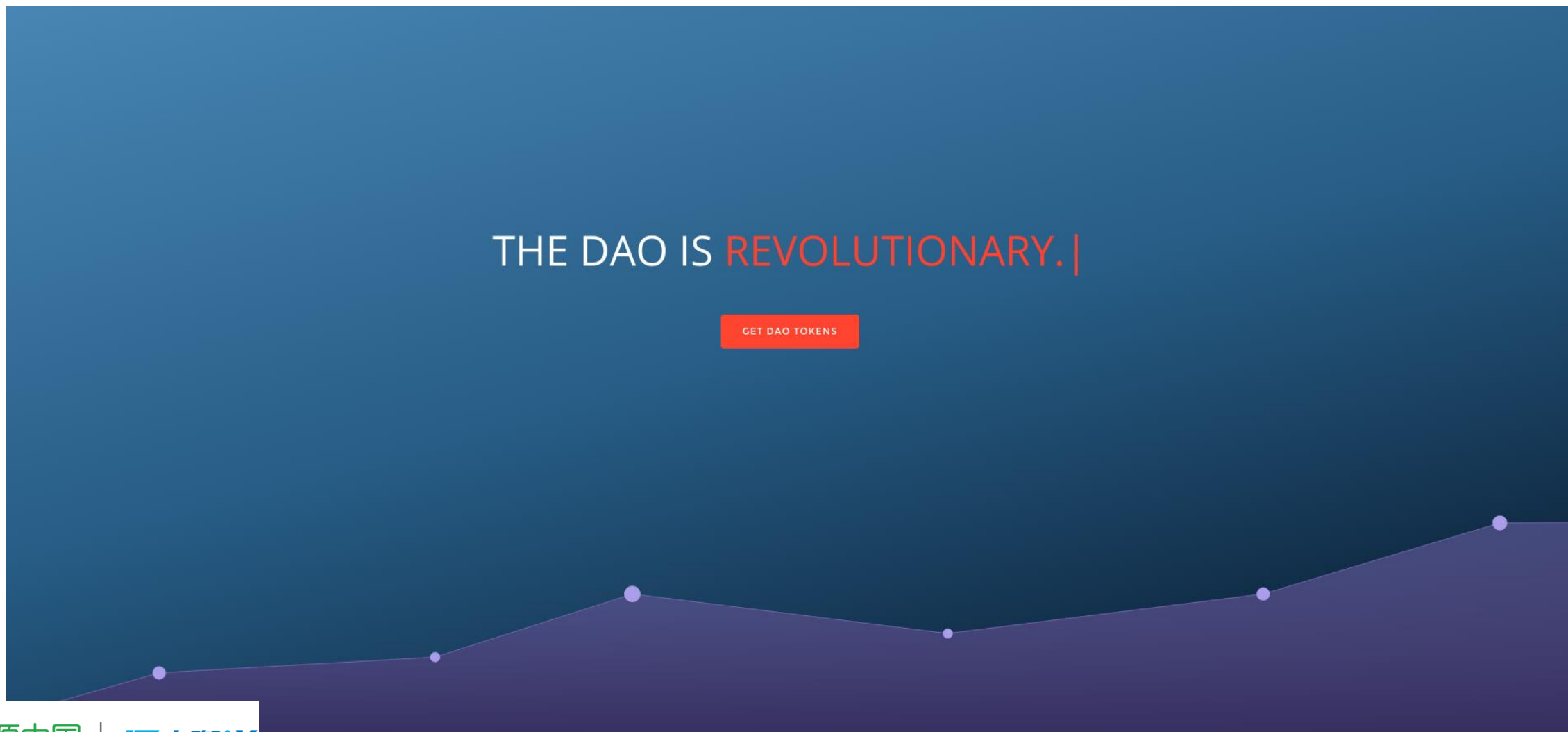


猫咪 504772 · 5代 · 急速
♡ 18114



猫咪 345185 · 10代 · 迟缓
♡ 18107

第一款“火了”的应用



去中心化交易所



去中心化应用的特点

- 可以没有中心化的操作者
- 你的钱是你的，应用只提供服务
- 应用逻辑是透明的

传统 Web 中心化应用的困境

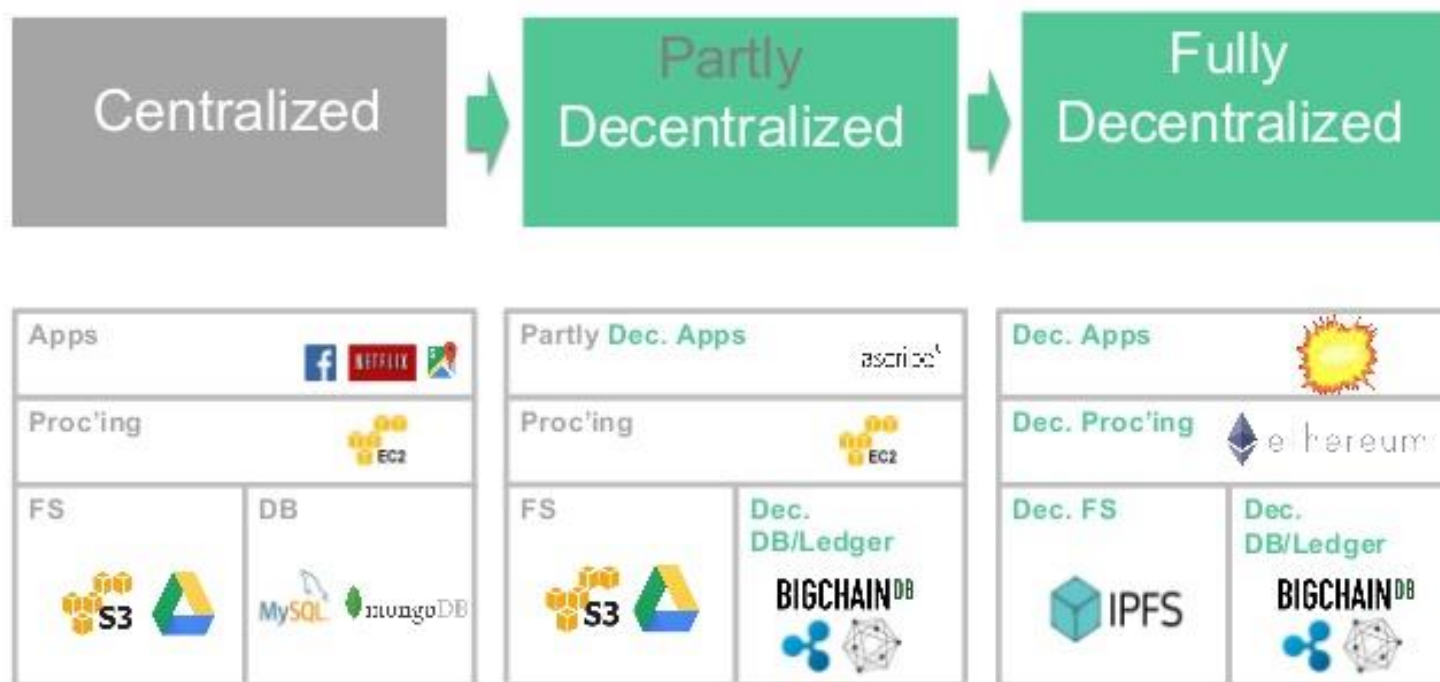
- 你的钱不是你的
- 你的数据不是你的
- 服务商是爸爸（某某云）

Web -> Blockchain

Web + Blockchain

去中心化应用 - Web 3

Decentralization of the Cloud



如何成为区块链工程师

如何学习区块链？

- 听区块链讲座
- 上区块链培训班
- 看书

如何学习区块链？

- 听区块链讲座
- 上区块链培训班
- 看书
- 加入秘猿科技
- 撸链

秘猿科技



Cryptape

We build trust.

Hangzhou, China <http://cryptape.com/> contact@cryptape.com

Repositories 69

People 30

Teams 2

Projects 0

Pinned repositories

cita

A fast and scalable blockchain for enterprise users.

Rust ★ 446 🍴 76

cita-whitepaper

CITA white-paper

★ 50 🍴 14

cita-bft

CITA BFT

Rust ★ 3 🍴 5

cita-cli

Forked from driftluo/cita-cli

A easy-to-use CITA command line tool

Rust ★ 6 🍴 1

Neuron-Android

Welcome to Neuron's open source Android app!
<https://cryptape.github.io/Neuron-Android/#/>

Java ★ 5

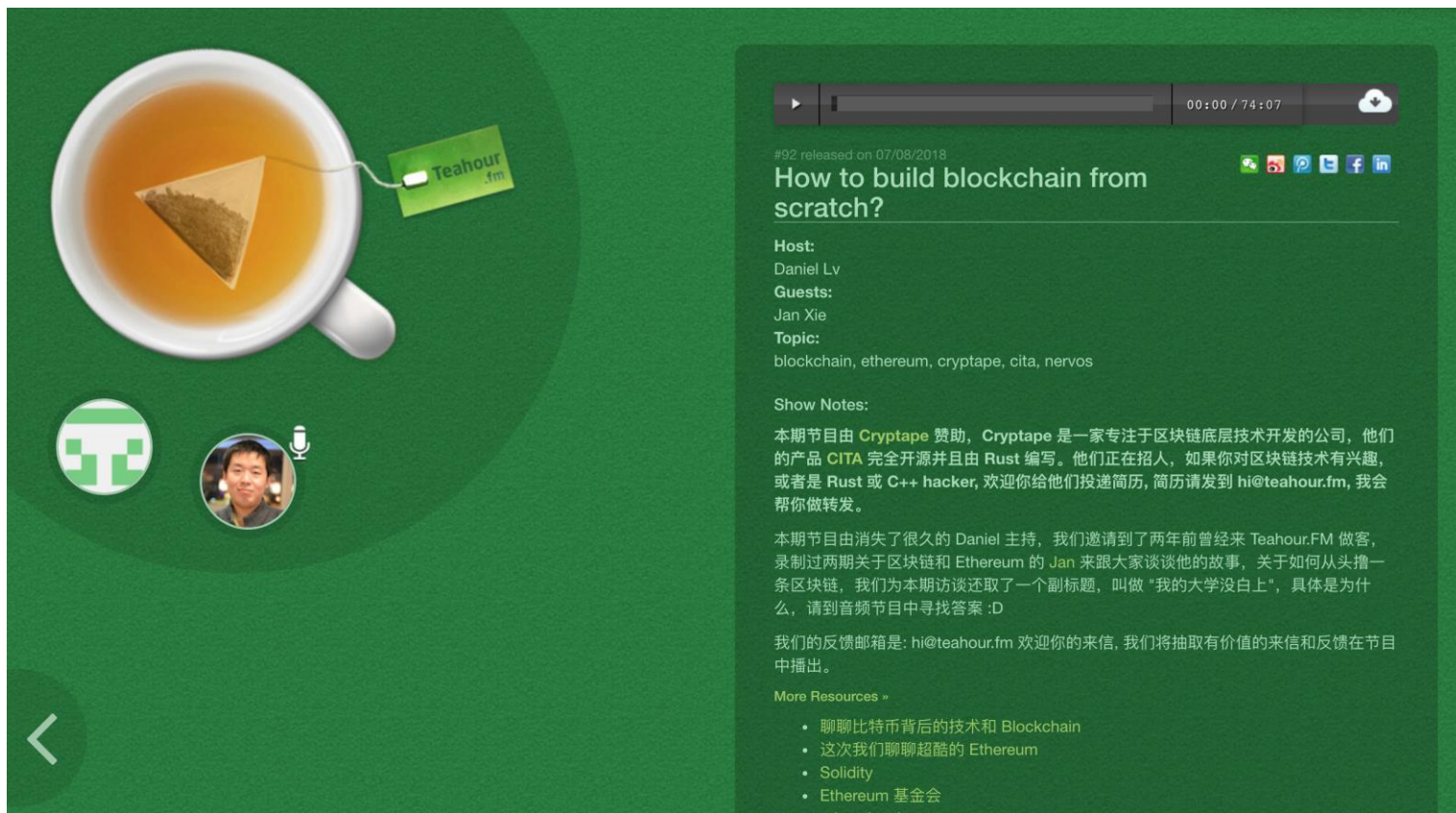
Neuron-iOS

The CITA wallet app on iOS operation system

Swift ★ 2

撸链？

撸链文化



The image shows a podcast player interface for Teahour.fm. On the left, there is a green background with a white teacup containing a tea bag labeled 'Teahour.fm'. Below the cup are two circular icons: one with a green 'T' and another with a portrait of a man. On the right, the player controls show a progress bar at 00:00 / 74:07. The episode title is '#92 released on 07/08/2018 How to build blockchain from scratch?'. The host is Daniel Lv and the guest is Jan Xie. The topic is 'blockchain, ethereum, cryptape, cita, nervos'. The 'Show Notes' section contains text about Cryptape's funding and the episode's content. At the bottom, there are 'More Resources' listed: '聊聊比特币背后的技术和 Blockchain', '这次我们聊聊超酷的 Ethereum', 'Solidity', and 'Ethereum 基金会'.

#92 released on 07/08/2018

How to build blockchain from scratch?

Host:
Daniel Lv

Guests:
Jan Xie

Topic:
blockchain, ethereum, cryptape, cita, nervos

Show Notes:

本期节目由 **Cryptape** 赞助, **Cryptape** 是一家专注于区块链底层技术开发的公司, 他们的产品 **CITA** 完全开源并且由 **Rust** 编写。他们正在招人, 如果你对区块链技术有兴趣, 或者是 **Rust** 或 **C++** hacker, 欢迎你给他们投递简历, 简历请发到 hi@teahour.fm, 我会帮你做转发。

本期节目由消失了很多年的 Daniel 主持, 我们邀请到了两年前曾经来 Teahour.FM 做客, 录制过两期关于区块链和 Ethereum 的 Jan 来跟大家谈谈他的故事, 关于如何从头撸一条区块链, 我们为本期访谈还取了一个副标题, 叫做 "我的大学没白上", 具体是为什么, 请到音频节目中寻找答案 :D

我们的反馈邮箱是: hi@teahour.fm 欢迎你的来信, 我们将抽取有价值的来信和反馈在节目中播出。

More Resources »

- 聊聊比特币背后的技术和 Blockchain
- 这次我们聊聊超酷的 Ethereum
- Solidity
- Ethereum 基金会



学习区块链最好最快的方法

- 加入秘猿科技
- 撸链

三个月学会以太坊

Ciri Ethereum

<https://github.com/ciri-ethereum/ciri>

ciri-ethereum / ciri

View Repository

Unwatch 7

Unstar 39

Fork 4

<> Code

Issues 8

Pull requests 0

Projects 1

Wiki

Insights

Settings

Ciri is an ongoing Ethereum implementation.

Edit

ethereum

ruby

eth

blockchain

geth

ciri

parity

rlp

trie

Manage topics

207 commits

1 branch

5 releases

3 contributors

Apache-2.0

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

 jjyr

fix NameError typo in lib/ciri/key.rb

Latest commit 15d4026 4 days ago

bin

Bump version to v0.0.3, Ciri pass all Frontier Ethereum tests

10 days ago

docker

move ciri-rlp, ciri-utils, ciri-crypto to separate repos

4 days ago

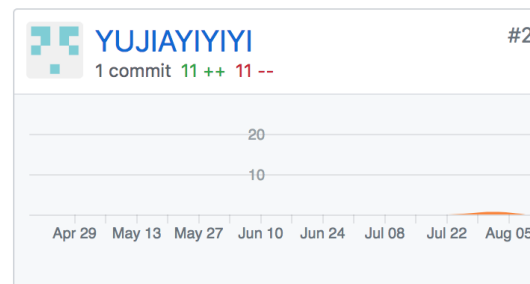
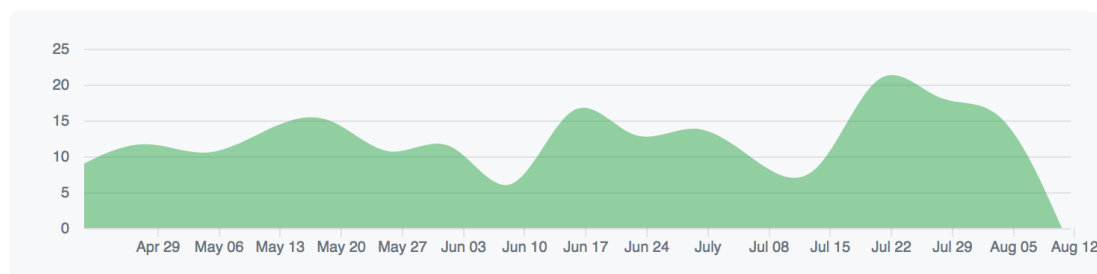
Ciri Ethereum

<https://github.com/ciri-ethereum/ciri>

Apr 22, 2018 – Aug 13, 2018

Contributions: Commits ▾

Contributions to master, excluding merge commits



Ethereum Tests

<https://github.com/ethereum/tests>

 **ethereum / tests**

 [Configure Sourcegraph](#)

 [Watch](#) 72

 [Unstar](#) 132

 [Fork](#) 125

[Code](#) [Issues 44](#) [Pull requests 4](#) [Projects 0](#) [Wiki](#) [Insights](#)

Common tests for all Ethereum implementations

 **1,915** commits

 **12** branches

 **2** releases

 **40** contributors

 MIT

Branch: **develop** [New pull request](#) [Create new file](#) [Upload files](#) [Find file](#) [Clone or download](#)

 **winsvega** Merge pull request [#485](#) from ethereum/fixrlptest [...](#) Latest commit `ad2184a` 3 days ago

 ABITests	Changed one test	a year ago
 BasicTests	Document difficulty tests	7 months ago
 BlockchainTests	fix mining reward in sar test	5 days ago
 GeneralStateTests	fix mining reward in sar test	5 days ago

Ethereum Tests

<https://github.com/ethereum/tests>

Branch: **develop** ▾ **New pull request** **Create new file** **Upload**

 **winsvega** Merge pull request #475 from ethereum/create ...

 ABITests	Changed one test
 BasicTests	Document difficulty tests
 BlockchainTests	check returndatasize after failing create
 GeneralStateTests	check returndatasize after failing create
 GenesisTests	Unify the cases of "mixHash"; there were some "mixhash"
 JSONSchema	test.py, VMTests, src/VMTestsFiller, JSONSchema/: output newline at e...
 KeyStoreTests	Updated one test and added another for keys
 PoWTests	Unify the cases of "mixHash"; there were some "mixhash"
 RLPTests	Update invalidRLPTest.json
 RPCTests	Mention the GitHub issue where the errors are tracked
 TransactionTests	complete general transaction tests
 TrieTests	added a missing test - it was possible to write Tree code that would ...
 VMTests	VMTests/: refill with new spaces, only source hash changes

通过了所有测试 -

```
1516 .when decoding #<StringIO:0x00005584112141a8> into Ciri::Types::Address
1517 .....
1518
1519 Pending: (Failures listed here are expected and do not affect your suite's status)
1520
1521 1) Ciri::Chain fixtures/BlockchainTests/bcStateTests blockhashNonConstArg_Constantinople
1522    # No reason given
1523    # ./spec/ciri/fixtures_tests/block_chain_spec.rb:147
1524
1525 Finished in 15 minutes 51 seconds (files took 3.02 seconds to load)
1526 8272 examples, 0 failures, 1 pending
1527
1528
1529
```

Ethereum Tests

- 链的初始状态

```
    "pre" : {  
      "0xa94f5374fce5edbc8e2a8697c15331677e6ebf0b" : {  
        "balance" : "0x1d14a0219e54822428000000",  
        "code" : "",  
        "nonce" : "0x00",  
        "storage" : {  
        }  
      },  
      "0xaaaf5374fce5edbc8e2a8697c15331677e6ebf0b" : {  
        "balance" : "0x02540be400",  
        "code" : "0x60003551",  
        "nonce" : "0x00",  
        "storage" : {  
        }  
      }  
    }  
  }
```

Ethereum Tests

- 输入(块、交易)

[illegible]

Ethereum Tests

- 对比执行后的状态

```
..... ,
"postState" : {
  "0x095e7baea6a6c7c4c2dfef977efac326af552d87" : {
    "balance" : "0x6e",
    "code" : "",
    "nonce" : "0x00",
    "storage" : {
    }
  },
  "0x8888f1f195afa192cfef860698584c030f4c9db1" : {
    "balance" : "0x033ca3ae5d37d40000",
    "code" : "",
    "nonce" : "0x00",
    "storage" : {
    }
  },
  "0xa94f5374fce5edbc8e2a8697c15331677e6ebf0b" : {
    "balance" : "0x1d14a0215cf8145fe6a7ff92",
    "code" : "",
    "nonce" : "0x0b",
    "storage" : {
    }
  },
  "0xaaaf5374fce5edbc8e2a8697c15331677e6ebf0b" : {
    "balance" : "0x02540be400",
    "code" : "0x60003551",
    "nonce" : "0x00",
    "storage" : {
    }
  }
},
"pre" : {
```

Ciri

- 用在了公司的项目
- 用在了 Hackathon 项目

```
# block header
class Header
  include Ciri::RLP::Serializable

  schema [
    :parent_hash,
    :ommers_hash,
    :beneficiary,
    :state_root,
    :transactions_root,
    :receipts_root,
    :logs_bloom,
    {difficulty: Integer},
    {number: Integer},
    {gas_limit: Integer},
    {gas_used: Integer},
    {timestamp: Integer},
    :extra_data,
    :mix_hash,
    :nonce,
  ]

# header hash
def get_hash
  Utils.keccak(rlp_encode)
end
```

如何实现以太坊

思考和建议

- 如何了解以太坊的规范
- 如何规划实现路径
- 如何发展开源项目

如何了解以太坊的规范

Ethereum Wiki <https://github.com/ethereum/wiki/wiki>

Welcome to the Ethereum Wiki

gitter Docs chat

Ethereum wiki covering all things related to Ethereum

This is the community maintained wiki covering all sorts of information on the next-generation peer-to-peer technology platform built by the Ethereum Foundation, including **Ethereum**, *the generalized blockchain for smart contract development*, as well as sister protocols like **Whisper**, *the private low-level datagram communication platform*, and **Swarm**, a distributed storage platform and content distribution service.

Wiki 上对以太坊技术的介绍

ontained economy for transactional computation and
[r/@djrtwo/ethereum-in-a-sentence-ba2db11c6bec](#)

in platform for "building unstoppable applications", while
his platform. Ethereum can be used for virtually any kind of
other way, any kind of activity that has an economic or
ran conventional alternatives, such as debit card payments,
t is trustless (no intermediary like a bank is needed, Bob can
safe and live (with some assumptions like an honest
p-resistant. Ethereum has been described in several ways.

ereum is to know what it does. Just like how it is not
icate workings of how a car works in order to appreciate it's
y arrives in your light bulb when you turn it on,
ial utility of Ethereum is a good start. Ethereum can be used
agreement (to put it another way, any kind of activity that
ct), at a lower cost than conventional alternatives, such as
trustless, secure, safe and censorship-resistant, all with high
list of uses is available [here](#).

let points are more general introductions, while the second is
tion of the Ethereum Virtual Machine. Another introduction is

["Contract and Decentralized Application Platform"—Ethereum](#)

Ethereum Technologies

- [Patricia Tree](#)
- [Wire protocol](#)
- [Light client protocol](#)
- [Subtleties](#)
- [Solidity Documentation](#)
- [NatSpec Format](#)
- [Contract ABI](#)
- [Bad Block Reporting](#)
- [Bad Chain Canary](#)

Ethash/Dashimoto

- [Ethash](#)
- [Ethash Yellow Paper appendix](#)
- [Ethash C API](#)
- [Ethash DAG](#)

Whisper

- [Whisper Proposal](#)
- [Whisper Overview](#)
- [PoC-1 Wire protocol](#)
- [PoC-2 Wire protocol](#)
- [PoC-2 Whitepaper](#)

Wiki 上对以太坊协议的介绍

[with Proof of work vs proof of stake](#)

[acea](#)

[of Ethereum?](#)

[tural Machine \(EVM\) intro](#)

[talk.](#)

Infrastructure

- [Chain Spec Format](#)
- [Inter-exchange Client Address Protocol](#)
- [URL Hint Protocol](#)
- [NatSpec Determination](#)
- [Network Status](#)
- [Raspberry Pi](#)
- [Exchange Integration](#)
- [Mining](#)
- [Licensing](#)
- [Consortium Chain Development](#)

DEV Technologies

- [RLP Encoding](#)
- [Node Discovery Protocol](#)
- [DEVp2p Wire Protocol](#)
- [DEVp2p Whitepaper \(WiP\)](#)
- [Web3 Secret Storage](#)
- [libp2p](#)

Yellow paper (EVM 规范)

ETHEREUM: A SECURE DECENTRALISED GENERALISED TRANSACTION LEDGER

BYZANTIUM VERSION e94ebda - 2018-06-05

DR. GAVIN WOOD
FOUNDER, ETHEREUM & PARITY
GAVIN@PARITY.IO

ABSTRACT. The blockchain paradigm when coupled with cryptographically-secured transactions has demonstrated its utility through a number of projects, with Bitcoin being one of the most notable ones. Each such project can be seen as a simple application on a decentralised, but singleton, compute resource. We can call this paradigm a transactional singleton machine with shared-state.

Ethereum implements this paradigm in a generalised manner. Furthermore it provides a plurality of such resources, each with a distinct state and operating code but able to interact through a message-passing framework with others. We discuss its design, implementation issues, the opportunities it provides and the future hurdles we envisage.

EIPs

<https://github.com/ethereum/EIPs>

EIPs [gitter](#) [join chat](#)

Ethereum Improvement Proposals (EIPs) describe standards for the Ethereum platform, including core protocol specifications, client APIs, and contract standards.

A browsable version of all current and draft EIPs can be found on [the official EIP site](#).

EIP - 1

<https://github.com/ethereum/EIPs>

eip	title	status	type	author	created
1	EIP Purpose and Guidelines	Active	Meta	Martin Becze <mb@ethereum.org>, Hudson Jameson <hudson@ethereum.org>, and others https://github.com/ethereum/EIPs/blob/master/EIPS/eip-1.md	2015-10-27, 2017-02-01

What is an EIP?

EIP stands for Ethereum Improvement Proposal. An EIP is a design document providing information to the Ethereum community, or describing a new feature for Ethereum or its processes or environment. The EIP should provide a concise technical specification of the feature and a rationale for the feature. The EIP author is responsible for building consensus within the community and documenting dissenting opinions.

EIP Rationale

We intend EIPs to be the primary mechanisms for proposing new features, for collecting community technical input on an issue, and for documenting the design decisions that have gone into Ethereum. Because the EIPs are maintained as text files in a versioned repository, their revision history is the historical record of the feature proposal.

For Ethereum implementers, EIPs are a convenient way to track the progress of their implementation. Ideally each implementation maintainer would list the EIPs that they have implemented. This will give end users a convenient way to current status of a given implementation or library.

如何了解以太坊规范

- 以太坊 Wiki
- Yellow paper
- EIPs

例子: DevP2P

Wiki - DevP2P

libp2p Whitepaper

James Ray edited this page on Jun 6 · 12 revisions

Contents

- [What it is](#)
- [Basic Design](#)

DEVp2p Wire Protocol

James Ray edited this page on May 25 · 20 revisions

Contents

- [Low-Level](#)
- [Payload Contents](#)
- [P2P](#)
- [Node identity and reputation](#)
- [Session Management](#)

DevP2P induction

Introduction

RLPx is a cryptographic peer-to-peer network and protocol suite which provides a general-purpose transport and interface for applications to communicate via a p2p network. RLPx is designed to meet the requirements of decentralized applications and is used by Ethereum.

The current version of RLPx provides a network layer for Ethereum. Roadmap:

- Completed:
 - UDP Node Discovery for single protocol
 - ECDSA Signed UDP
 - Encrypted Handshake/Authentication
 - Peer Persistence
 - Encrypted/Authenticated TCP
 - TCP Framing

参考源码

参考现有实现

EVM Implementations

Live on Main Network

- [go-ethereum](#)
 - A popular Ethereum client with its own EVM implementation ([core/vm](#) directory)
- [Parity](#) in Rust
 - Another popular Ethereum client with its own EVM implementation ([ethcore](#) directory)
- [cpp-ethereum](#)
 - An Ethereum client that generates the consensus test suite ([libevm/VM.cpp](#))
- [Pyethereum](#) in Python
 - A mostly deprecated client ([ethereum/vm.py](#))
- [Py-EVM](#) in Python
 - A Python implementation designed to be highly configurable and modular and compliant with the Ethereum test suite, work is in progress on it to run a full node and develop sharding.
- [EthereumJ](#) in Java
 - A client with its own EVM implementation
- For more, see [here](#).

TRACING Geth

```
geth --nodiscover --verbosity=7 console
```

Tracing Geth

```
diff --git a/eth/handler.go b/eth/handler.go
index 4069359c9..9bd3b3659 100644
--- a/eth/handler.go
+++ b/eth/handler.go
@@ -263,6 +263,7 @@ func (pm *ProtocolManager) handle(p *peer) error {
    number = head.Number.Uint64()
    td      = pm.blockchain.GetTd(hash, number)
    )
+   fmt.Printf("network %d, td %d, hash %s, genesis %s \n", pm.networkId, td, hash.Hex(), genesis.Hash().Hex())
    if err := p.Handshake(pm.networkId, td, hash, genesis.Hash()); err != nil {
        p.Log().Debug("Ethereum handshake failed", "err", err)
        return err
@@ -288,6 +289,7 @@ func (pm *ProtocolManager) handle(p *peer) error {
    // If we're DAO hard-fork aware, validate any remote peer with regard to the hard-fork
    if daoBlock := pm.chainconfig.DAOForkBlock; daoBlock != nil {
        // Request the peer's DAO fork header for extra-data validation
+   fmt.Printf("start dao check...%d\n", daoBlock.Uint64())
    if err := p.RequestHeadersByNumber(daoBlock.Uint64(), 1, 0, false); err != nil {
        return err
    }
@@ -417,6 +419,7 @@ func (pm *ProtocolManager) handleMessage(p *peer) error {
    if err := msg.Decode(&headers); err != nil {
        return errResp(ErrDecode, "msg %v: %v", msg, err)
    }
+   fmt.Printf("headers %d: %v\n", len(headers), headers)
    // If no headers were received, but we're expending a DAO fork check, maybe it's that
    if len(headers) == 0 && p.forkDrop != nil {
        // Possibly an empty reply to the fork header checks, sanity check TDs
diff --git a/eth/peer.go b/eth/peer.go
index 42ead5396..9df2646e0 100644
--- a/eth/peer.go
+++ b/eth/peer.go
@@ -235,7 +235,7 @@ func (p *peer) Handshake(network uint64, td *big.Int, head common.Hash, genesis
    errc := make(chan error, 2)
```

Tracing Geth

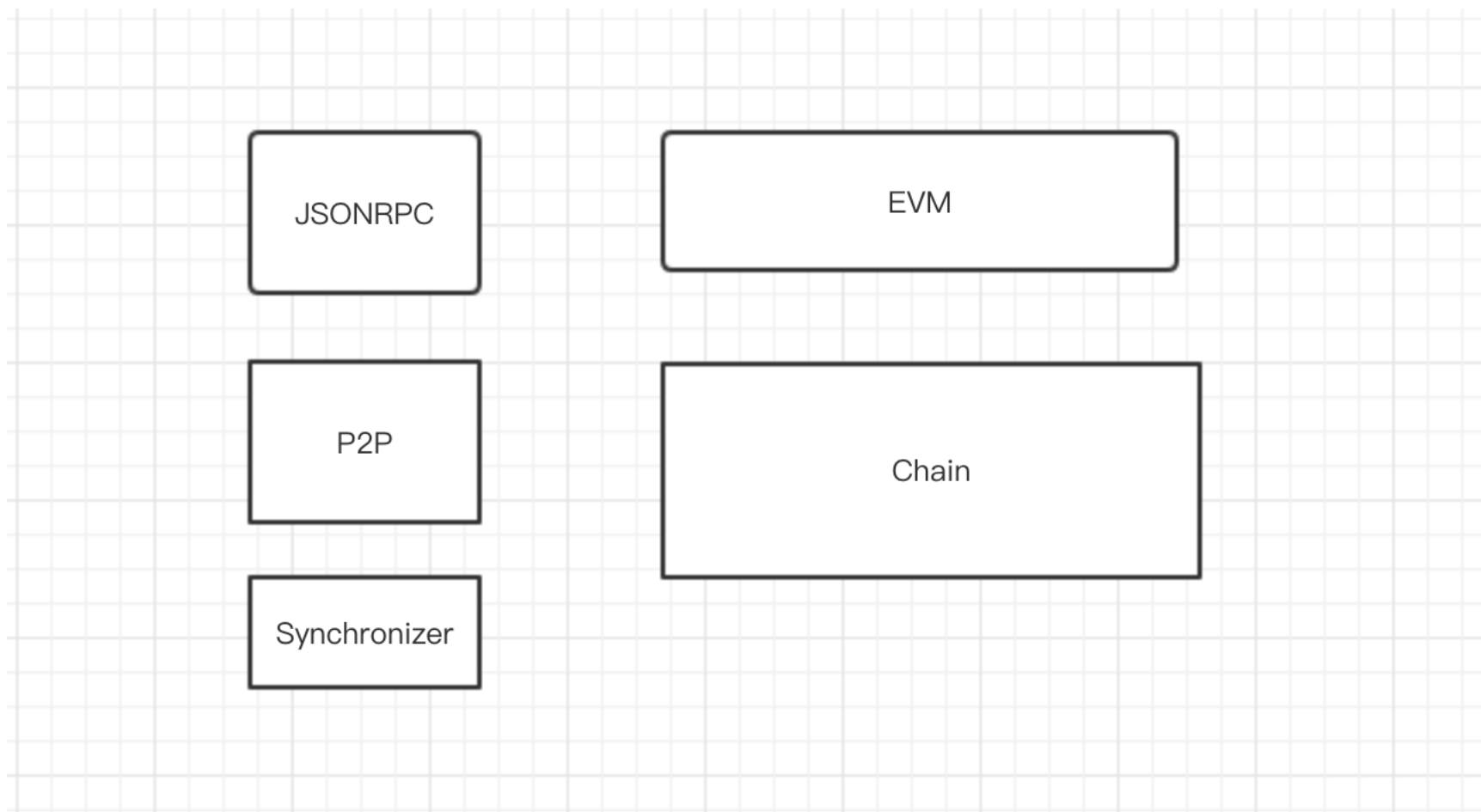
```
diff --git a/eth/handler.go b/eth/handler.go
index 4069359c9..9bd3b3659 100644
--- a/eth/handler.go
+++ b/eth/handler.go
@@ -263,6 +263,7 @@ func (pm *ProtocolManager) handle(p *peer) error {
    number = head.Number.Uint64()
    td      = pm.blockchain.GetTd(hash, number)
+   fmt.Printf("network %d, td %d, hash %s, genesis %s \n", pm.networkId, td, hash.Hex(), genesis.Hash().Hex())
    if err := p.Handshake(pm.networkId, td, hash, genesis.Hash()); err != nil {
        p.Log().Debug("Ethereum handshake failed", "err", err)
        return err
@@ -288,6 +289,7 @@ func (pm *ProtocolManager) handle(p *peer) error {
    // If we're DAO hard-fork aware, validate any remote peer with regard to the hard-fork
    if daoBlock := pm.chainconfig.DAOForkBlock; daoBlock != nil {
        // Request the peer's DAO fork header for extra data validation
+       fmt.Printf("start dao check...%d\n", daoBlock.Uint64())
        if err := p.RequestHeadersByNumber(daoBlock.Uint64(), 1, 0, false); err != nil {
            return err
        }
@@ -417,6 +419,7 @@ func (pm *ProtocolManager) handleMsg(p *peer) error {
    if err := msg.Decode(&headers); err != nil {
        return errResp(ErrDecode, "msg %v: %v", msg, err)
    }
+   fmt.Printf("headers %d: %v\n", len(headers), headers)
    // If no headers were received, but we're expending a DAO fork check, maybe it's that
    if len(headers) == 0 && p.forkDrop != nil {
        // Possibly an empty reply to the fork header checks, sanity check TDs
diff --git a/eth/peer.go b/eth/peer.go
index 42ead5396..9df2646e0 100644
--- a/eth/peer.go
+++ b/eth/peer.go
@@ -235,7 +235,7 @@ func (p *peer) Handshake(network uint64, td *big.Int, head common.Hash, genesis
    errc := make(chan error, 2)
```

如何规划实现路径

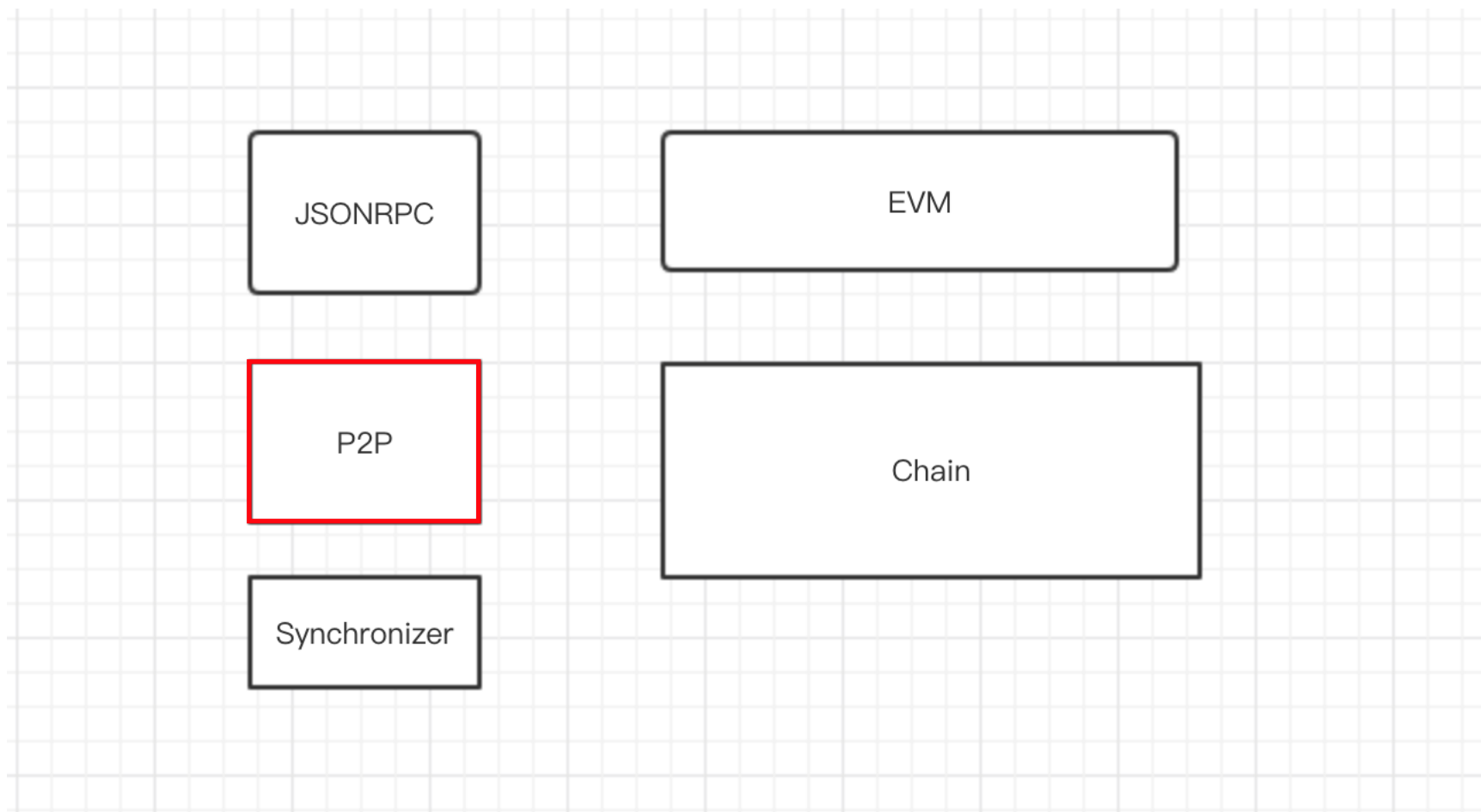
以太坊用到的基础库

- RLP 编码
- Secp256k1 签名
- Patricia Tree
- ...

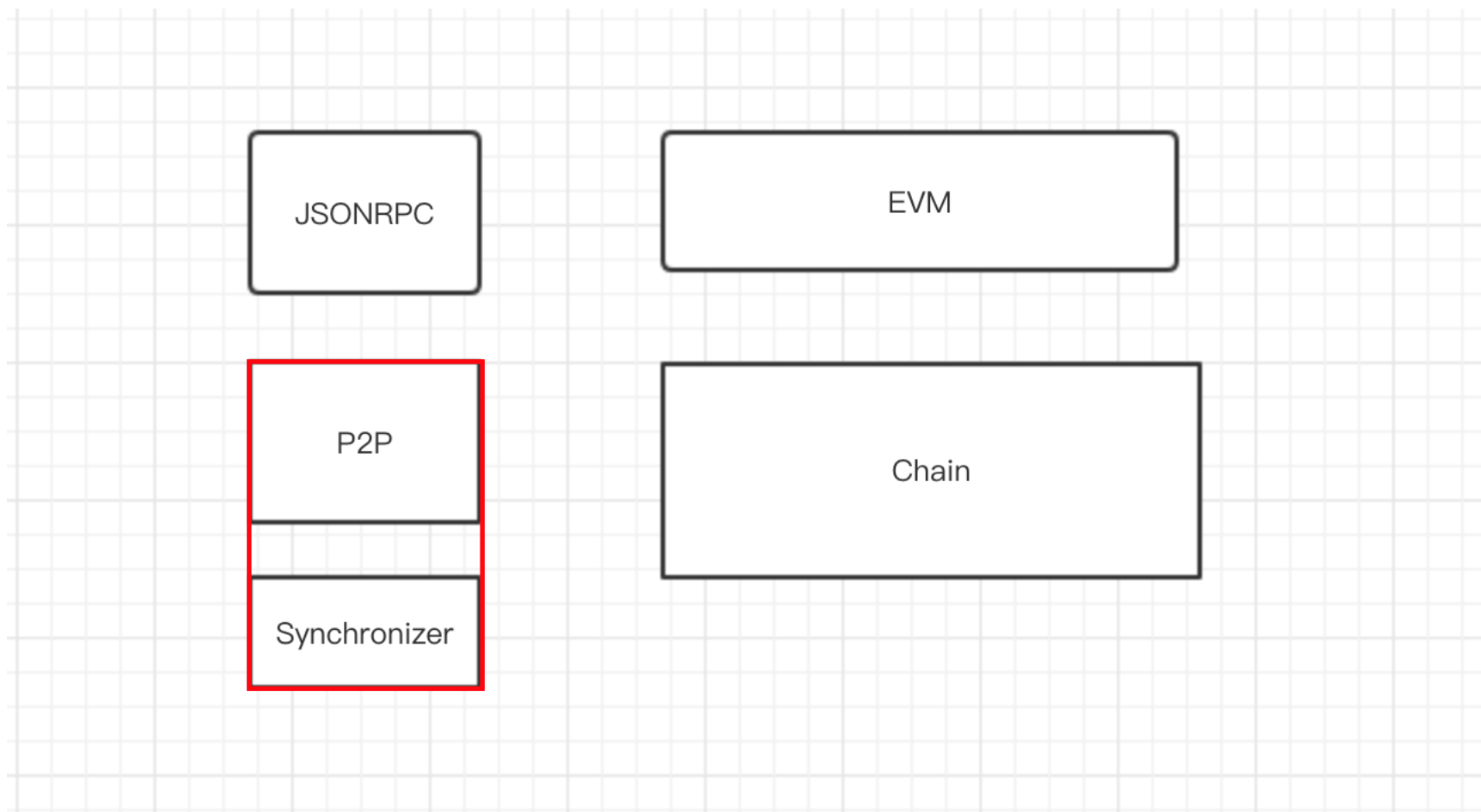
以太坊的组件



从可测试的组件开始实现

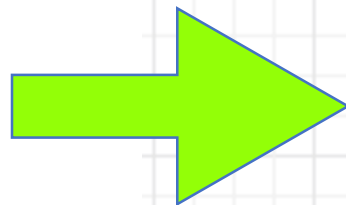


从可测试的组件开始实现



从可测试的组件开始实现

- 实现基本的握手/通信协议
- 实现下载机制
- 与 Geth 客户端链接
- 反复调试..增加单元测试



JSONRPC

EVM

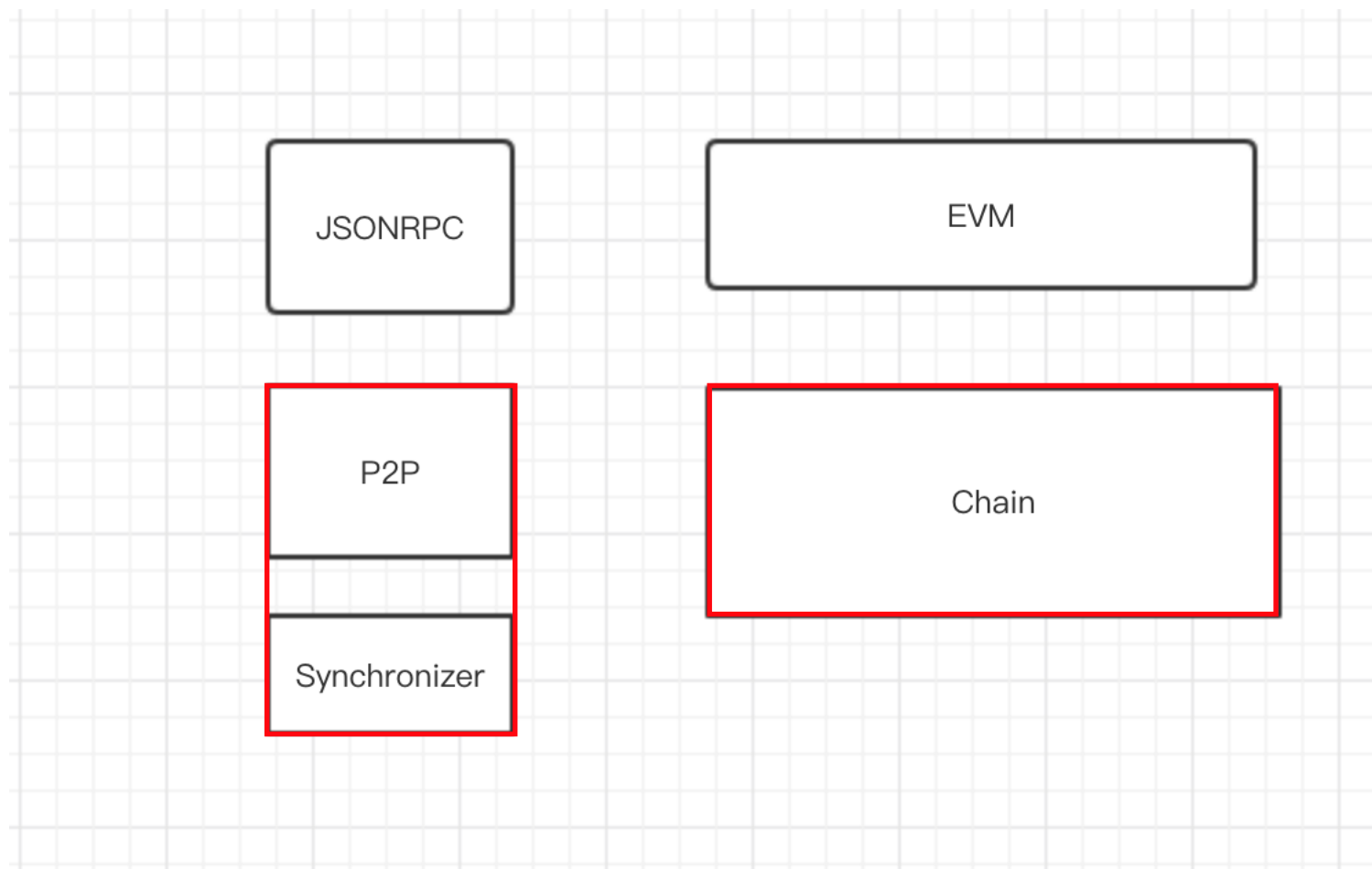
P2P

Chain

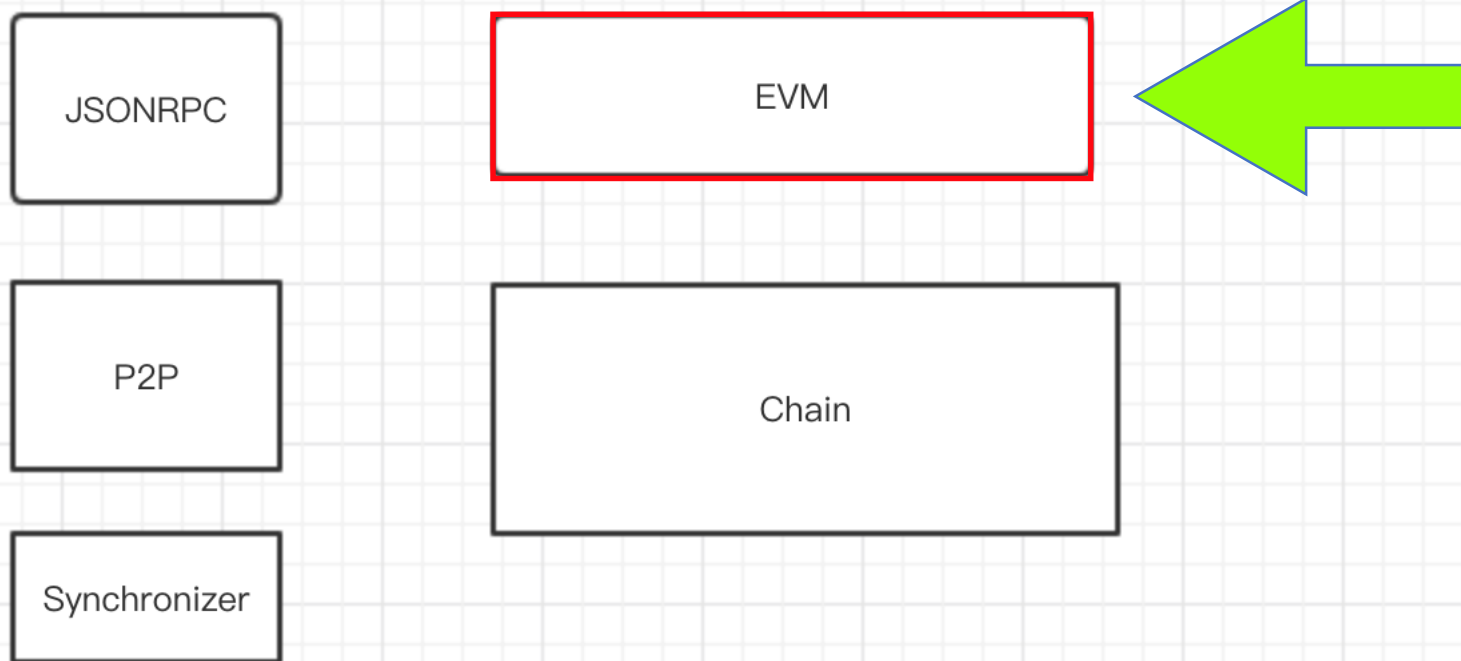
Synchronizer

从可测试的组件开始实现

- 实现验证区块头
- 实现 Chain 存储区块



从可测试的组件开始实现



- 黄皮书
- EIP(各种 fork 改动)
- Ethereum Tests
- 参考现有客户端

开源项目如何吸引开发者

开源的项目管理方式

EVM  Updated 6 days ago

Filter cards

1 TODO + ...

Implement Byzantium fork EIP-609 ...

0 of 3

#36 opened by jjyr

good first issue next release

0 Doing + ...

9 Done + ...

Blockchain tests ...

6 of 6

#18 opened by jjyr

Chain support validate block ...

3 of 3

#16 opened by jjyr

StateDB support Trie ...

Added by jjyr

Trie ...

2 of 2

#15 opened by jjyr

EVM support run with a block ...

3 of 3

#11 opened by jjyr

GeneralStateTests ...

1 of 1

#14 opened by iivr

开源的项目管理方式

ciri-ethereum / ciri

View Repository

Unwatch 7

Unstar 39

Fork 4

<> Code

Issues 8

Pull requests 0

Projects 1

Wiki

Insights

Settings

Filters

is:issue is:open

Labels

Milestones

New issue

☐ 8 Open ✓ 13 Closed

Author ▾ Labels ▾ Projects ▾ Milestones ▾ Assignee ▾ Sort ▾

☐ **Implement Byzantium fork EIP-609** **good first issue** **next release**

#36 opened 6 days ago by jjyr 0 of 3

☐ **Apply to Ethereum foundation grants program**

#35 opened 6 days ago by jjyr

☐ **Journal DB** **good first issue**

#17 opened on Jun 28 by jjyr 1 of 2

☐ **Support JRuby** **enhancement**

#6 opened on May 28 by jjyr 0 of 3

☐ **CLI** **good first issue**

#5 opened on May 28 by jjyr

☐ **Web3 JSONRPC server** **good first issue** **help wanted**

#4 opened on May 28 by jjyr

☐ **Light client**

#3 opened on May 28 by jjyr

☐ **LibP2P** **help wanted** **next release**

#1 opened on May 28 by jjyr 2

开源的项目管理方式

The screenshot shows the GitHub repository page for 'ciri-ethereum / ciri'. The repository has 7 views, 39 stars, and 4 forks. The 'Issues' tab is selected, showing 8 open issues. The filter is set to 'is:issue is:open'. The issues list includes:

- ☐ **Implement Byzantium fork EIP-609** good first issue next release
#36 opened 6 days ago by jjyr 0 of 3
- ☐ **Apply to Ethereum foundation grants program**
#35 opened 6 days ago by jjyr
- ☐ **Journal DB** good first issue
#17 opened on Jun 28 by jjyr 1 of 2
- ☐ **Support JRuby** enhancement
#6 opened on May 28 by jjyr 0 of 3
- ☐ **CLI** good first issue
#5 opened on May 28 by jjyr
- ☐ **Web3 JSONRPC server** good first issue help wanted
#4 opened on May 28 by jjyr
- ☐ **Light client**
#3 opened on May 28 by jjyr
- ☐ **LibP2P** help wanted next release
#1 opened on May 28 by jjyr

A yellow arrow points to the 'good first issue' label on the first issue, with the text '鼓励新的贡献者' (Encourage new contributors) next to it.

完善 README

<https://github.com/ciri-ethereum/ciri#ciri>

Ciri

build passing gitter join chat

Ciri is an ongoing Ethereum implementation.

It aims to be a feature complete Ethereum implementation and expects to achieve both research-friendly and high-performance. As you see, it's still under development.

See [How to learn Ethereum and contribute to Ciri](#).

See [projects](#) and [milestones](#) for current development status.

Roadmap

See [Roadmap](#) on Wiki

Development

Ciri depends on [rocksdb](#), [secp256k1](#), [ethash](#) and [snappy](#).

It's a recommendation to [setup with docker](#), it will help to save lots of time.

Setup with docker

Use docker command to pull image:

```
docker pull ciriethereum/ciri
```

完善 Roadmap - <https://github.com/ciri-ethereum/ciri/wiki#roadmap>

Roadmap

Ciri project aimed to implement a full feature set Ethereum client, so our first target is to bring ciri to Ethereum main-net as soon as possible, There are many hard problems need to work out.

Minimal Ethereum

Ethereum is mainly a composite of EVM, Chain and P2P networking.

Currently, we already have EVM and Chain.

There are some tests provide by [Ethereum tests project](#), we can consider our EVM and Chain is fit the Ethereum specs if we pass all the tests, and then we can step into P2P networking part implementation.

See <https://github.com/ciri-ethereum/ciri/projects/1> for EVM and Chain progress. (Currently, we have passed all Ethereum tests)

方便开发者 SETUP

Setup with docker

Use docker command to pull image:

```
docker pull ciriethereum/ciri
```

Or you can use our prepared rake tasks if you're not familiar with docker:

clone repo and submodules

```
git clone --recursive https://github.com/ciri-ethereum/ciri.git  
cd ciri
```

make sure we have installed docker, ruby and rake

```
# make sure we have installed docker, ruby and rake  
docker -v  
ll rake
```

开源项目如何吸引开发者

- 开源方式管理项目
- 鼓励新贡献者
- 完善 README
- 完善 Roadmap
- 方便开发者 SETUP

开源项目需要更多人参与

 **ethereum / py-evm**

 View Repository

 Watch ▾

86

 Unstar

607

 Fork

269

 Code

 Issues 117

 Pull requests 16

 Projects 3

 Insights

A Python implementation of the Ethereum Virtual Machine <https://trinity.ethereum.org/>

evm

ethereum

python

ethereum-virtual-machine

 1,992 commits

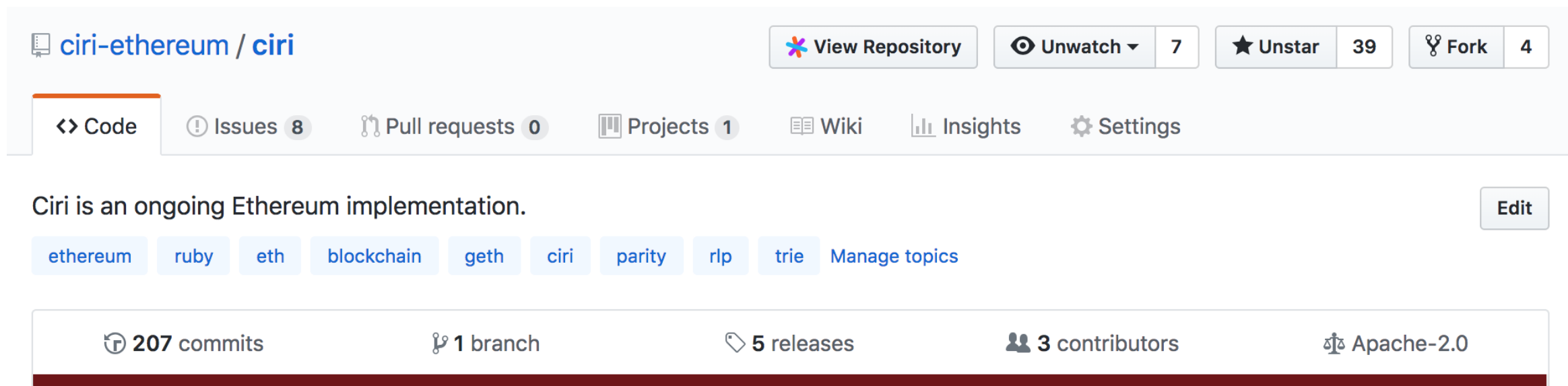
 13 branches

 42 releases

 43 contributors

 MIT

开源项目需要更多人参与



The screenshot shows the GitHub repository page for `ciri-ethereum / ciri`. The repository is described as "Ciri is an ongoing Ethereum implementation." It has 207 commits, 1 branch, 5 releases, 3 contributors, and is licensed under Apache-2.0. The page includes navigation tabs for Code, Issues (8), Pull requests (0), Projects (1), Wiki, Insights, and Settings. There are also buttons for View Repository, Unwatch (7), Unstar (39), and Fork (4). A list of topics (ethereum, ruby, eth, blockchain, geth, ciri, parity, rlp, trie) is shown, along with a Manage topics link.

`ciri-ethereum / ciri` [View Repository](#) [Unwatch](#) 7 [★ Unstar](#) 39 [Fork](#) 4

[↔ Code](#) [! Issues](#) 8 [🔗 Pull requests](#) 0 [📁 Projects](#) 1 [📖 Wiki](#) [📊 Insights](#) [⚙ Settings](#)

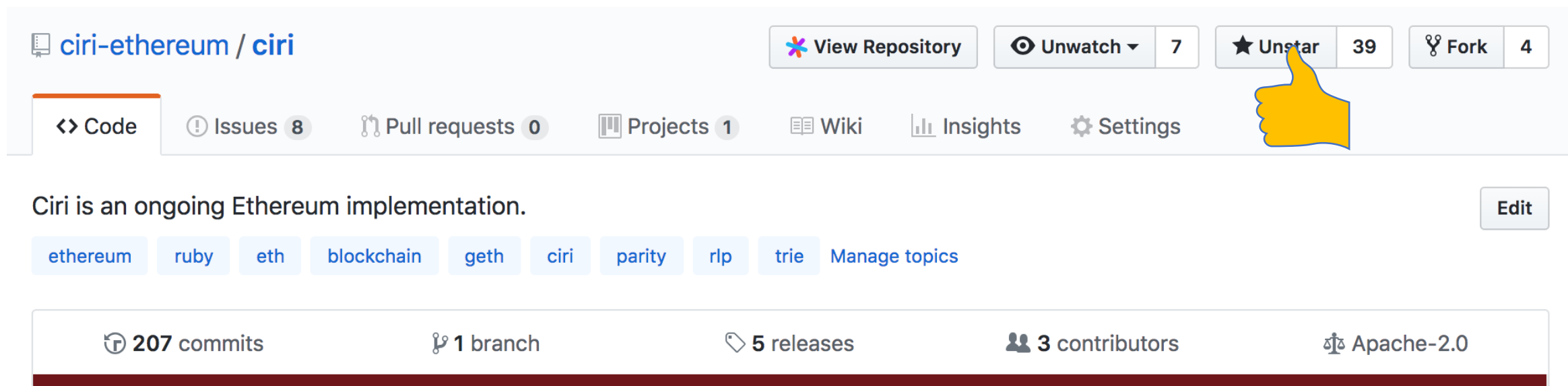
Ciri is an ongoing Ethereum implementation. [Edit](#)

[ethereum](#) [ruby](#) [eth](#) [blockchain](#) [geth](#) [ciri](#) [parity](#) [rlp](#) [trie](#) [Manage topics](#)

[📄 207 commits](#) [🔗 1 branch](#) [📦 5 releases](#) [👤 3 contributors](#) [📄 Apache-2.0](#)

<https://github.com/ciri-ethereum/ciri>

开源项目需要更多人参与



ciri-ethereum / ciri

View Repository Unwatch 7 ★ Unstar 39 Fork 4

<> Code Issues 8 Pull requests 0 Projects 1 Wiki Insights Settings

Ciri is an ongoing Ethereum implementation. Edit

ethereum ruby eth blockchain geth ciri parity rlp trie Manage topics

207 commits 1 branch 5 releases 3 contributors Apache-2.0

<https://github.com/ciri-ethereum/ciri>

FAQ

- 联系方式: jjy@cryptape.com
- 我的博客: <https://justjjy.com>
- Ciri Ethereum: <https://github.com/ciri-ethereum/ciri>
- 微信: fwweedm



CRYPTAPE第三季度招聘

岗位	工作内容	岗位	工作内容
区块链工程师	CITA底层开发	PR渠道	媒体渠道搭建及对接媒体关系
测试工程师	测试CITA及相关产品	专利流程专员	知识产权的流程管理、合同的收发归档
高级前端开发工程师	产品前端开发、Dapp开发	招聘专员(活动方向)	线下招聘渠道搭建、校园招聘及雇主品牌搭建
内容运营	输出有价值的平台/行业/公关类文章	IOS 开发工程师	区块链IOS端钱包APP及相关SDK开发

投递简历方式:

1. 邮箱直接投递: join@cryptape.com
邮件标题“应聘岗位+姓名”

2. 通过扫描官网的二维码加入我们
所有简历会在3个工作日之内反馈

