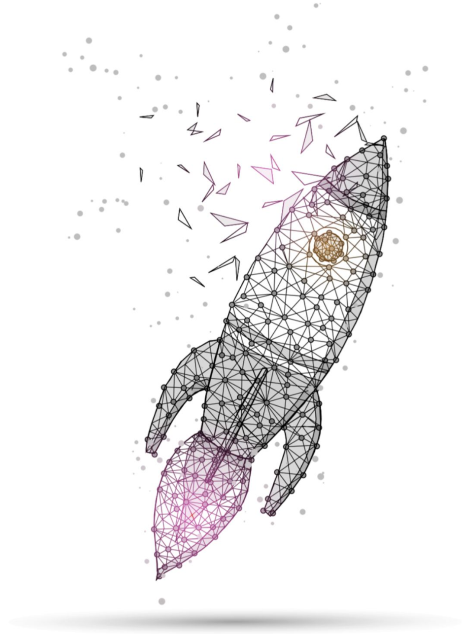
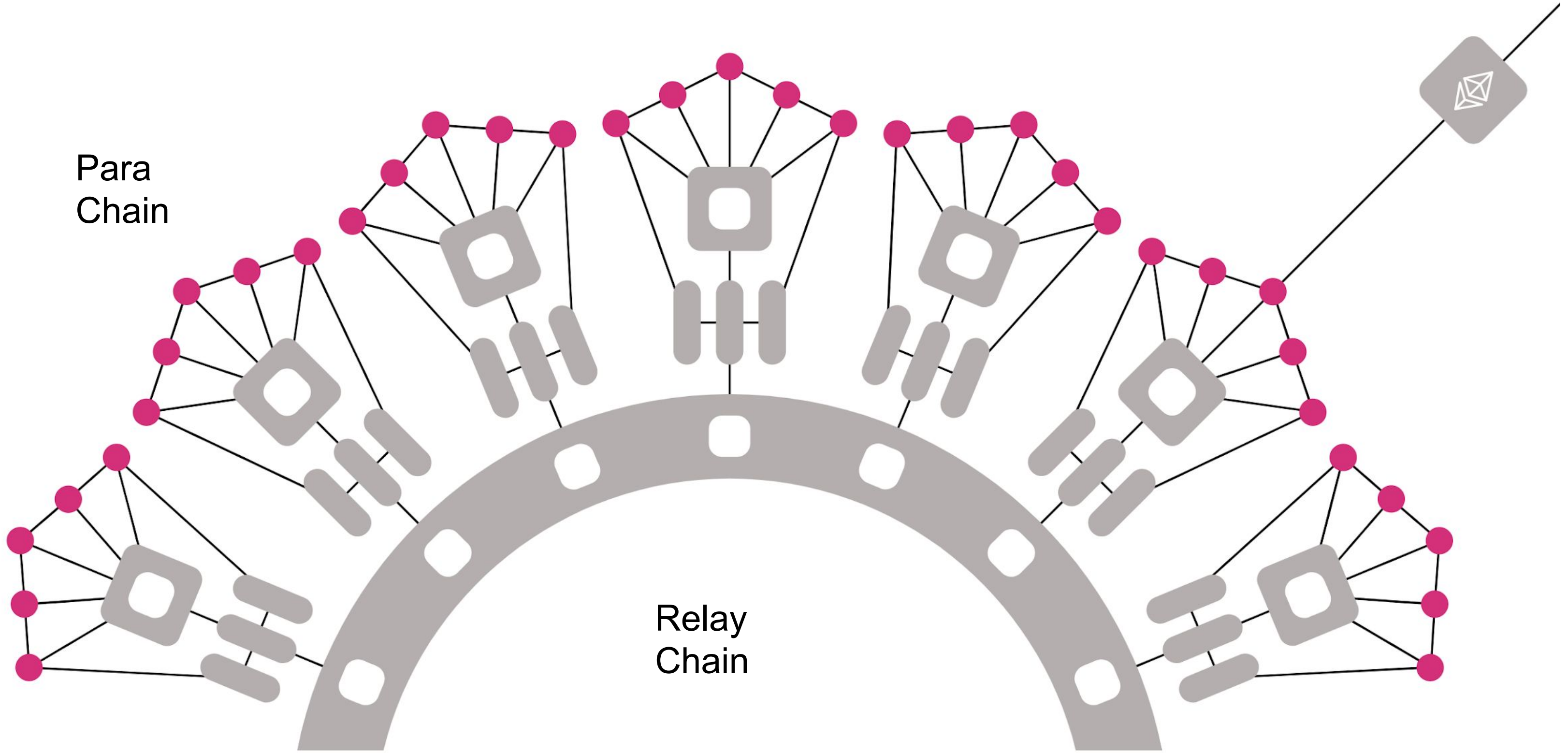
Intro To

Substrate

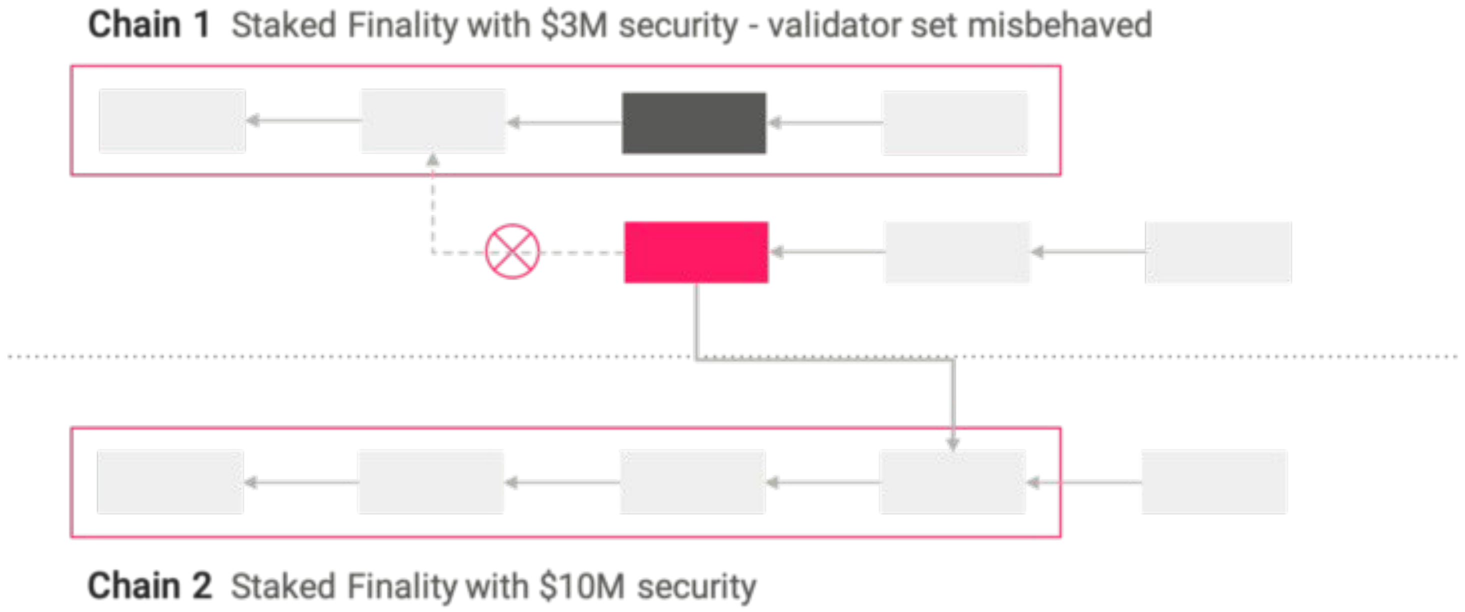
Hammer Wang

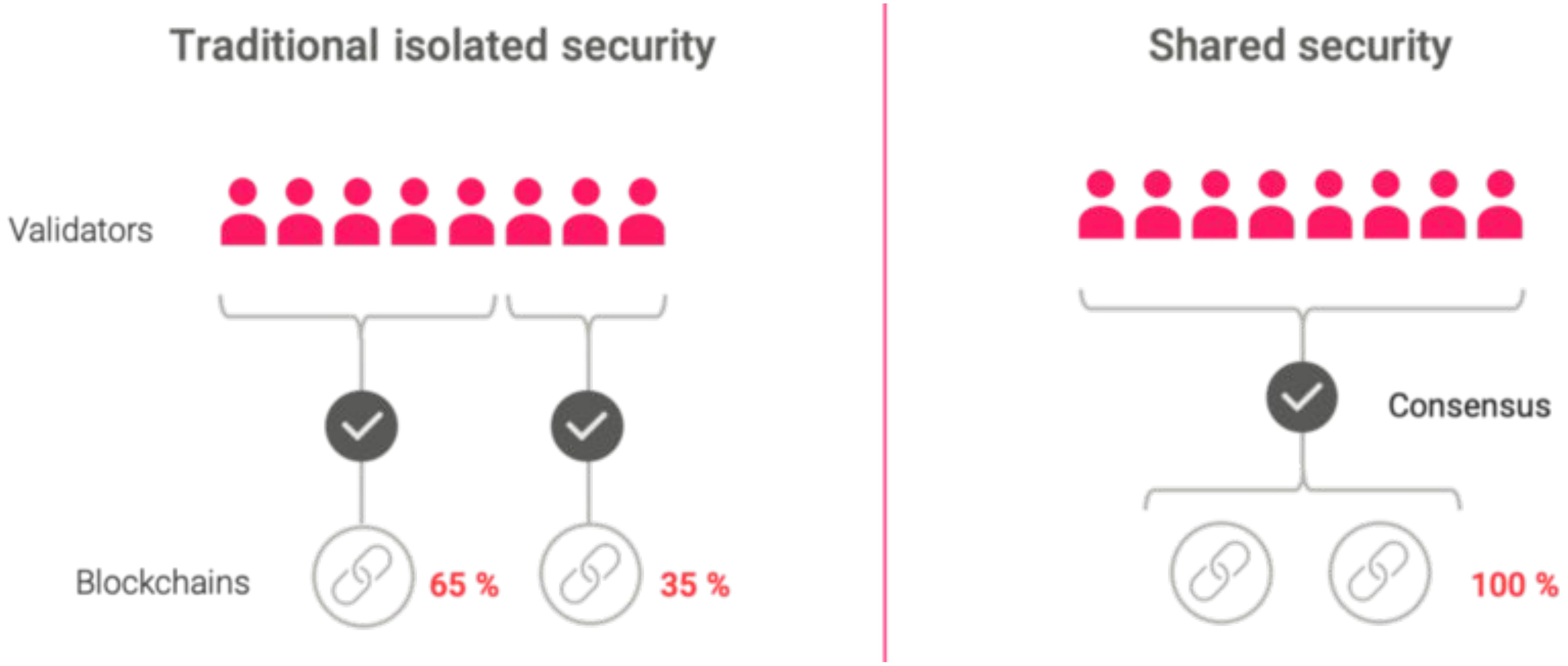


What is POLKADOT.


Para
Chain
Relay
Chain

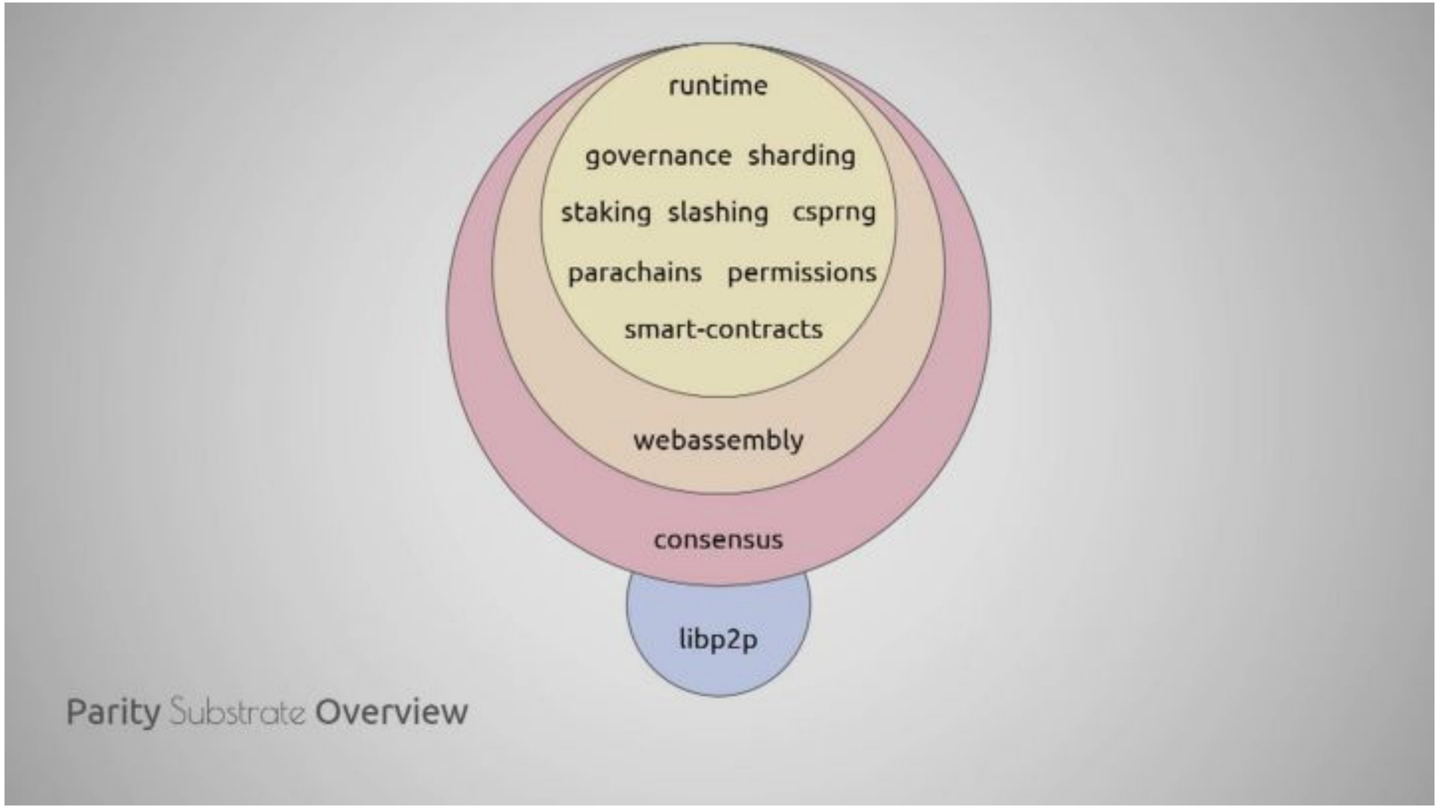
合并安全性/共享安全池


Chain 1 Staked Finality with $3M security - validator set misbehaved
Chain 2 Staked Finality with $10M security


Traditional isolated security
Validators
Blockchains
65 %
35 %
Shared security
Consensus
100 %

What is Substrate?

Substrate Architecture



Layers

NODE	构建节点全套组件。CORE + SRML
SRML	构建链的逻辑组件。包括通证经济系统、链上治理等
CORE	构建链的工具库。包括同步、网络、数据库、客户端、RPC等

加密算法

- **Blake2-256 (Hash）**
- **Ed25519**
- **Sr25519 (阈值签名）**

可扩展：crypto::Pair & Hasher trait

std
vs
no_std

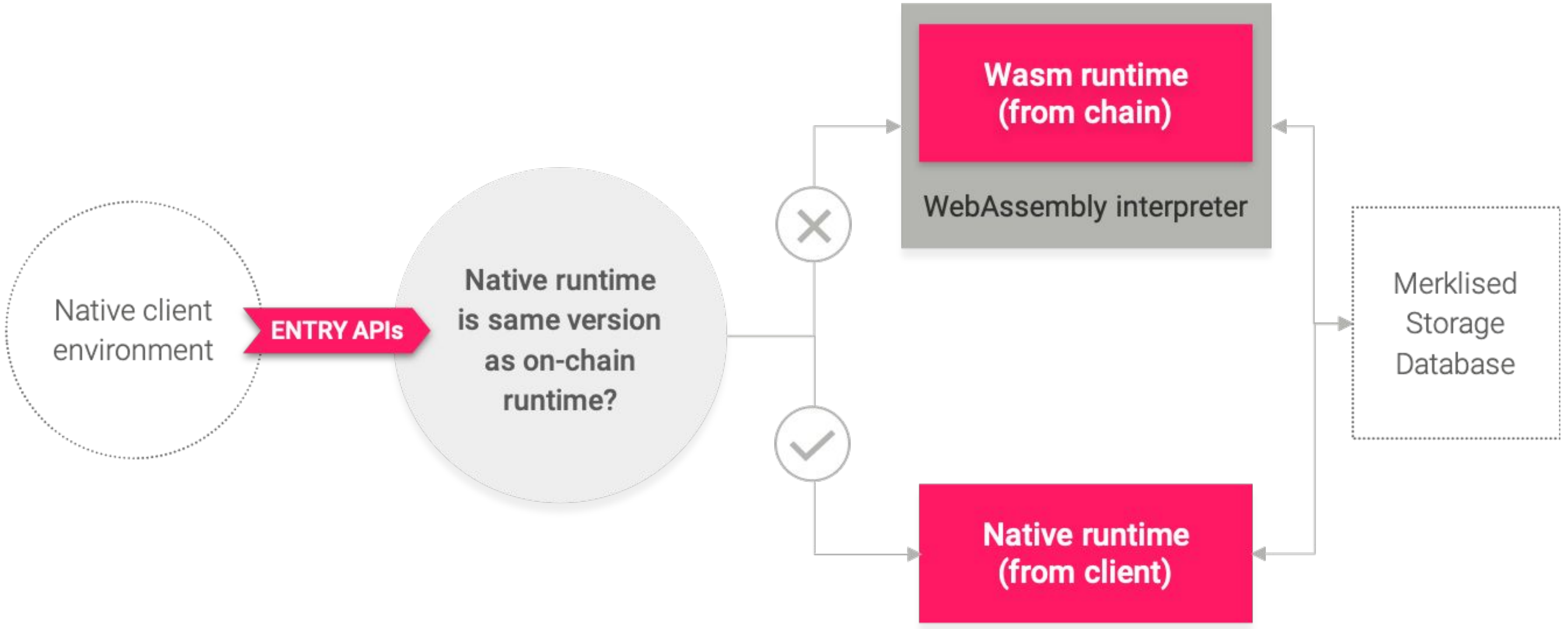
- std = native

  提供了更加高效的运行方式

- no_std = wasm

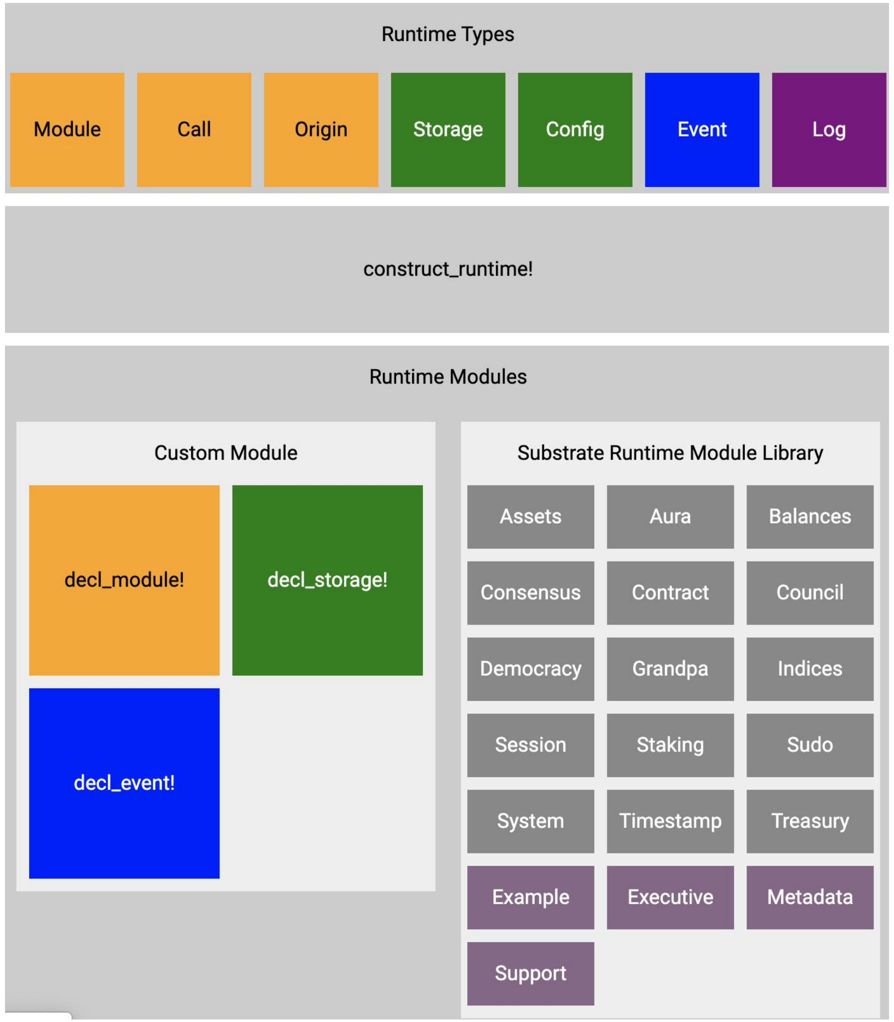
  key to the forkless upgrade

substrate提供了比较精细的
runtime version来判断运行
native code 或者 wasm code


Native client environment
ENTRY APIs
Native runtime is same version as on-chain runtime?
Wasm runtime (from chain)
WebAssembly interpreter
Native runtime (from client)
Merklised Storage Database

code here

RUNTIME Architecture



RUNTIME Types

- Module
- Call
- Origin
- Storage
- Config
- Event
- Log

模块化的
RUNTIME

类型/模块都在定义在trait中，高度抽象的定义使得模块松耦合，容易交互

(code here)

多实例的 RUNTIME MODULE

同一个module可以被实例化多次。每个实例拥有自己独立的存储和设置

(code here)

RUNTIME METADATA

like ABI in ethereum

Substrate Module LIst

- Staking (NPoS)
- Sessions (管理session key)
- Balances (currency & fees)
- Contracts (智能合约)
- Timestamp (设置链上时间)
- Indices (账户的“简写”)
- Treasury (金库)
- ...

Key Concepts In SRML

- Header & Block
- (Un)checkedExtrinsic
- Inherent
- Account & Index (nonce)
- Origin
- Call
- Hash

Origin

Origin 用来追溯方法的调用者

可自定义扩展，目前有三种基本的origin:

- Root（super user)
- Signed (owned by AccountId)
- Nobody (common sense)

Extrinsics & Inherent

General Extrinsics:

- Transactions(extrinsics)
- Inherent

Call

a *Call* is a runtime function object. Also known as *Proposal* or a *Dispatch*.

可序列化，包括一个额外的参数：Origin

Balances

A module in SRML:

- Balance
- Imbalance: NegativeImbalance & PositiveImbalance & OnUnbalanced
- ExistentialDeposit & Dust
- Currency

Macros In SRML

How macros help building modules:

- decl_module!
- decl_storage!
- decl_event!

how macros help constructing runtime:

- construct_runtime!

THANKS.

Substrate Workshop 南京现场群

该二维码7天内(8月10日前)有效，重新进入将更新