Hi everyone this is Tracy Salander. By making this video, I would like to share something about Substrate I learned. Every expert was a beginner. You don't need any background to get the point of the video. Now, let's start!

what is Substrate?

Substrate is a scalable, modular, and open-source blockchain framework written in the programming language Rust. It implements most of the commonly used functions encountered in blockchain development, such as the p2p network, consensus algorithms, data storage, transaction management, etc. Here you may have a question about why Substrate is a framework instead of being a library? The reason is that the framework will actively call our code so when we use substrate, the substrate framework will control when to establish a p2p network, when to add transactions to the transaction pool and when to generate blocks., but if we use some libraries which is a collection of some functions and needs to be actively called by others to do this, it takes us a lot of efforts rather than focusing on the specific functions.

Why substrate is scalable?

The scalability we say in the Blockchain industry is that the number of transactions we can deal with. So how we can increase our scalability? One way is that we can increase the size of each block but we have bandwidth limitations therefore we can't increase the size too much. One way is to change the blockchain into a block graph so we don't need to go from A to B to C, instead of that we can go from A to C but the problem is that we may have transaction collisions. Now, from the block side and the chain side, we can't increase the scalability a lot without having other problems. How about only putting the transaction results on the chain and leaving the transactions off the chain? You may notice

It may have a centralization problem. The same problem may be caused by the agent mechanism as well which is that we only find some representatives to run the nodes to deal with the data. Now we can use sharding or cross-chain technology to increase the scalability. The idea of sharding or cross-chain technology is that we randomly separate computers to calculate the answers to the questions. For example, we have 100 questions and 10 people. We don't need to let 10 people deal with 100 questions one by one. Instead, we can randomly let them become small groups to deal with several questions together. In this way, we can accelerate our speed to deal with transactions.

Now let's talk about the modular part. Substrate is completely modular mixing and matching components and building core business logic, while the rest is left to the framework. Modules such as consensus allow developers to focus on specific areas, thereby saving a lot of time and effort in the development process.

Why we should learn Substrate?

The benefits are low-cost development of a chain running in a production environment, finding a well-paying job, deeply participating in the open-source community, upgrade on the chain without forks and substrate can directly connect to commonly used databases through off-chain workers although there is no guarantee that the data in MySQL is true, it can store MySQL operations on the chain. 423

Now I will introduce a little bit about blockchain since we need to know something about it before we start to use Substrate. The reason why we will put our money into banks is that the fame and reputation of banks, but to be honest banks may go bankrupt. In the world of blockchain, everyone has a ledger that is not seized by banks. If anyone would like to make any changes to the ledger, all the others will know. Plus, if one's computer crashes, the other ledgers won't be affected. Blockchain nodes need databases, peer-to-peer networks, consensus algorithms, transaction processing, runtime, and other special functions. Substrate can provide scalability, modular, open-source to the blockchain.

Now let's start to run a substrate node. To run the node, we need to install Rust. The official website is https://www.rust-lang.org/ then we click install or get started. The website will give us the instruction based on our systems. If we want to update rust then we can use 'rustup update'. For the development tool, we can use Visual Studio Code and Rust plugin.

Usually, for big projects, we will use Cargo. Cargo is Rust's build system and package management tool. When we install Rust, Cargo will be installed automatically. Now we can create a project called 'sample' by using 'Cargo new sample'. Then we can use 'Cargo build --release' to get the files that can be run. Using the word 'release' can compile the files faster.

Lets 'git clone https://github.com/paritytech/substrate.git'

'cd substrate'

'cargo build --release'

After we compile it, we will get a directory called target.

Then we can start one node by using './target/release/substrate --dev'

If we start this chain before, we can delete it firstly './target/release/substrate purge-chain --dev'

After starting it, let's open polkadot.js.org/apps/, and then for the setting part, we choose LocalNode and Substrate, then we will see the condition of the blockchain.

Now, let's see how to start two nodes. './target/release/substrate --alice --chain local --base-path /tmp/alice'. Here /tmp/alice is to set the database directory. Initializing the height is 0. If we want to generate the first block, we need to start another node './target/release/substrate --bob --chain local --base-path /tmp/bob'. If you ran them before which may cause some problems. So before run the two nodes, you can remove their databases by using 'rm -rf /tmp/alice' and 'rm -rf/tmp/bob'.

Thank you for watching.