Introduction of Hyperledger CACTUS

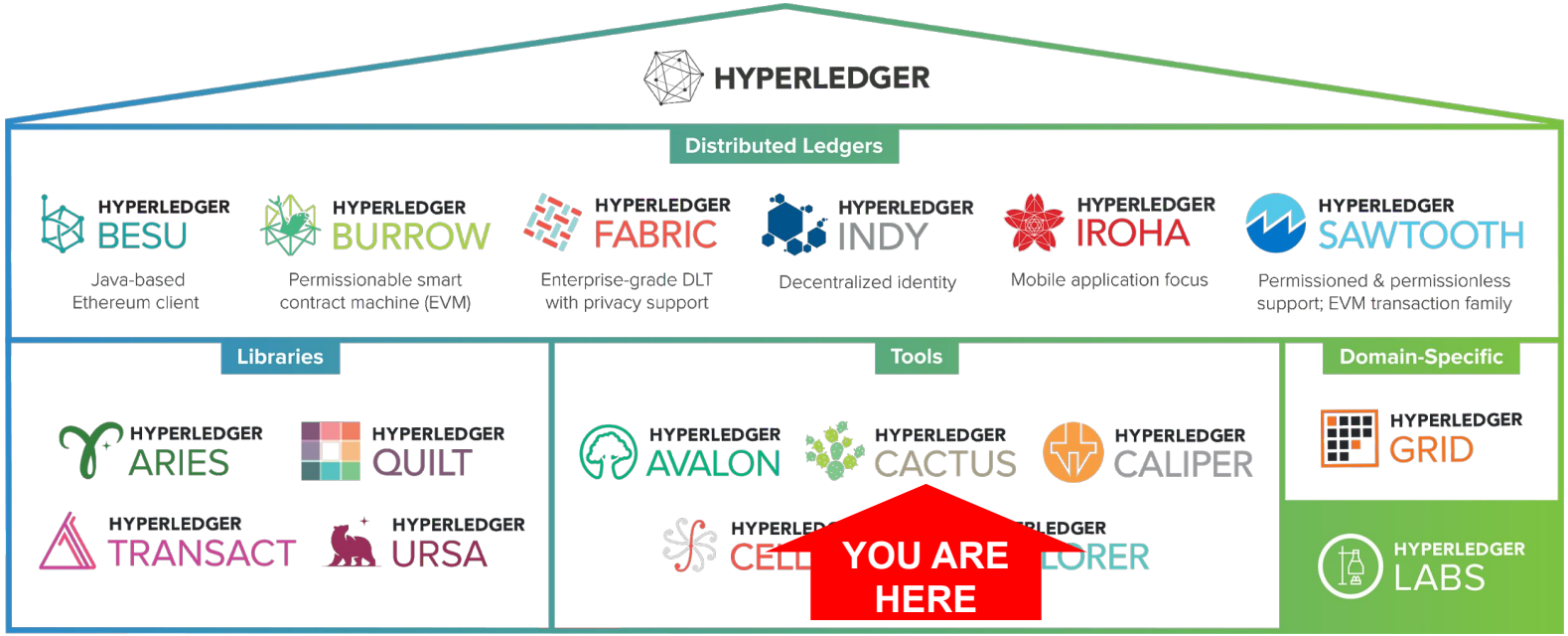
CACTUS core contributors

January 2021

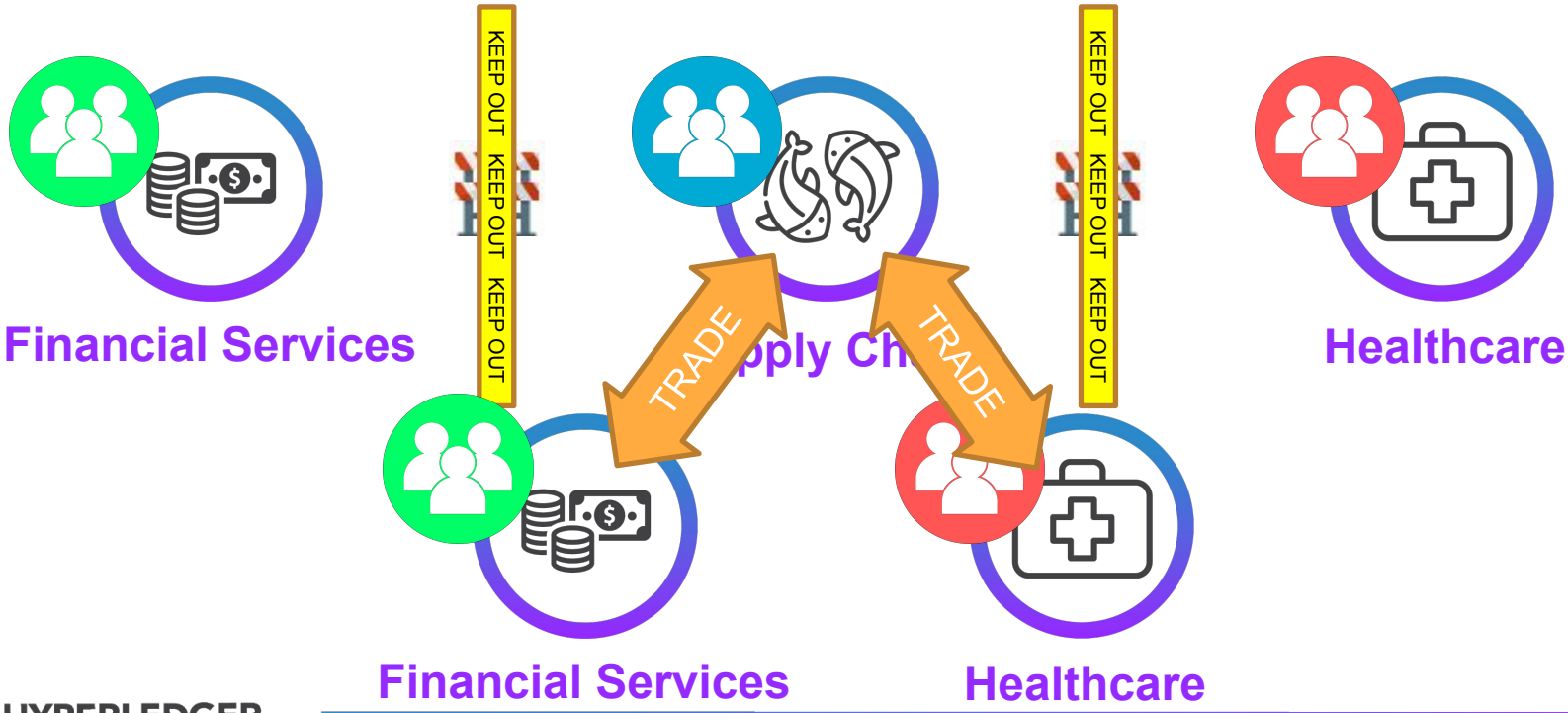
HYPERLEDGER

BLOCKCHAIN TECHNOLOGIES FOR BUSINESS

Hyperledger Modular Approach



Working with “A World of Many Networks”



Core principles for ‘interoperability’

Maximum Possible Pluggability and Generality	No Middlemen Whenever Possible	No Token Required	No Mandatory Toll Booth
we want to be able to plug and play components as much as possible.	we don’t want to have to go through (and trust!) an intermediate blockchain if we don’t have to do so.	we do not want users to have to use tokens for transactions.	we don’t want to require operators to make money by taking a cut of individual transactions.

CACTUS Features

Consistent transaction across multiple ledges

Support for transactions between permissioned and permission-less ledgers.

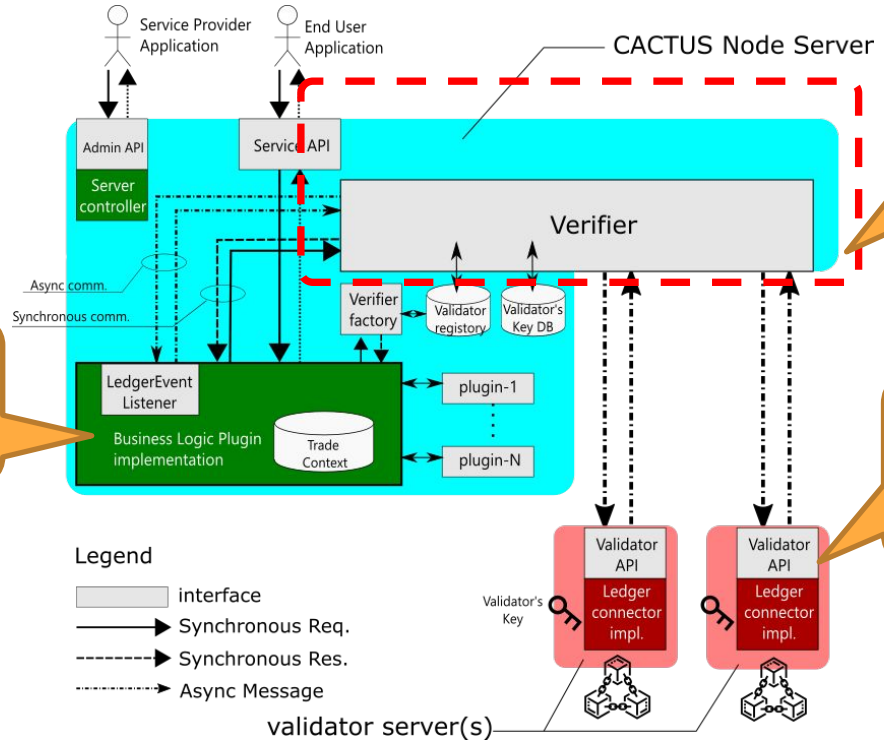
Propose, approve first, then executes

Protocol ensures prior consent from exchanging parties to avoid unexpected behaviors.

Validation of transactions

Validator node checks whether each transaction is acceptable, and taking part to local governance

CACTUS Architecture



Consortium management mechanism

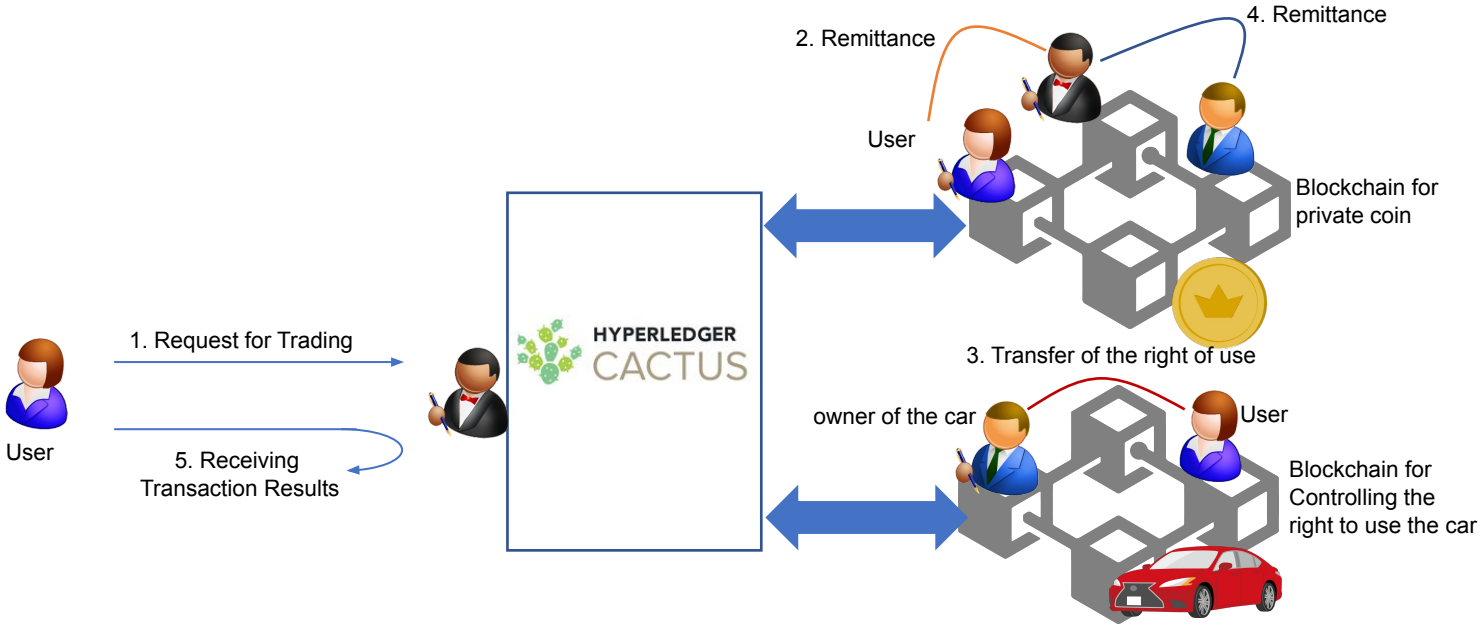
CACTUS application (examples are given)

Each Ledger is abstracted as Validator (for BC platforms)

Achievements (as of v0.3)

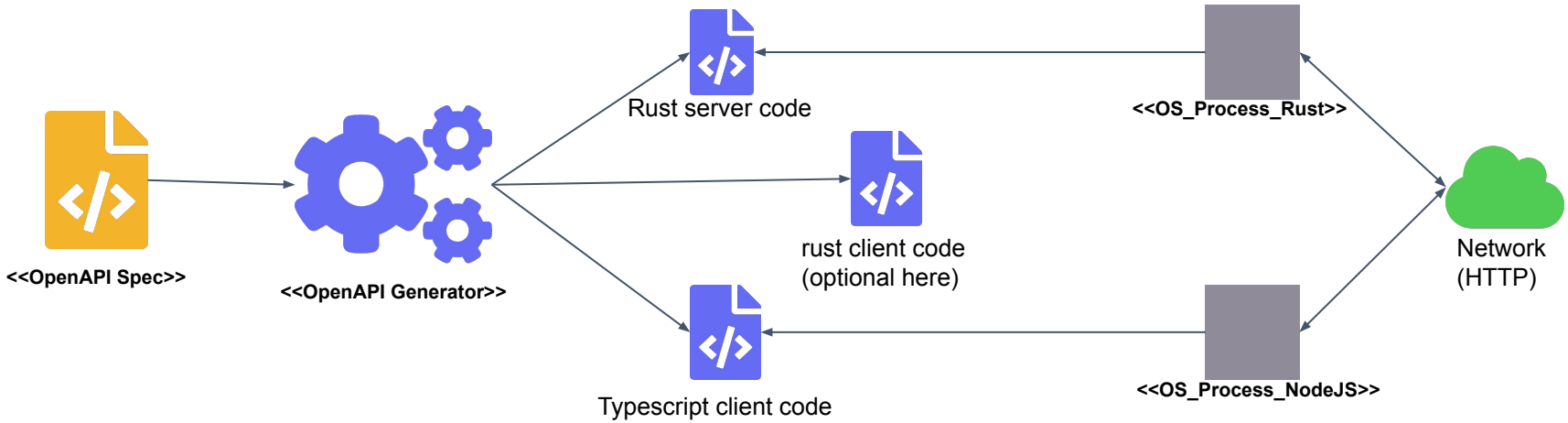
- Ledger Plugins (more DLTs are coming soon)
  - Fabic
  - Besu
  - Go-Ethereum
  - Quorum
- Example applications
  - Car Trade
  - Supply Chain Management

Car Trade example (escrow trade across DLTs)



Language Agnostic Plugin Development

- You can write Cactus plugins in any language!
  - Fine print: Pull request adding this is pending approval as of 2021-01-13



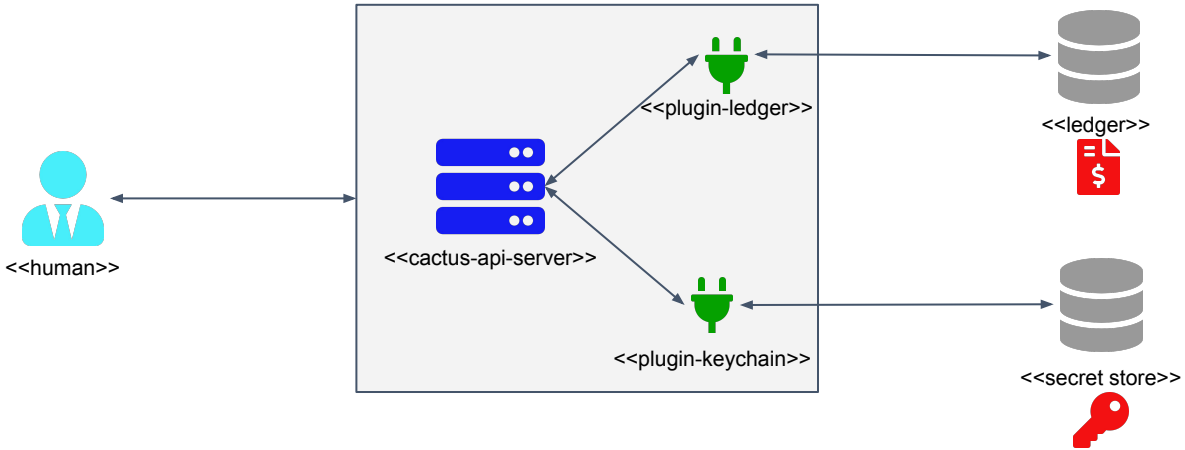
Plugin Example - Keychain

- The interface is meant to be really simple:

```
export interface IPluginKeychain extends ICactusPlugin {
    getKeychainId(): string;
    has(key: string): Promise<boolean>;
    get<T>(key: string): Promise<T>;
    set<T>(key: string, value: T): Promise<void>;
    delete<T>(key: string): Promise<void>;
}
```

Plugin Example - Keychain

- Store secrets that other plugins can retrieve
- Once we have identity/RBAC it will integrate with that too



Let’s get there faster together!

We want to realize the value of ledger interoperability in production ASAP, seeking contributors especially in the following areas:

1. Fabric v2/Corda Ledger plugins
2. HTLC contracts for cross-ledger atomic swaps
3. Transaction protocol implementation

Please join us!

- CACTUS project home page

https://wiki.hyperledger.org/display/cactus/Hyperledger+Cactus+Home

1620 lines (1140 sloc) | 128 KB

Raw Blame

HYPERLEDGER

Hyperledger Cactus Whitepaper

Version 0.1 (Early Draft)

: https://github.com/hyperledger/cactus/blob/master/whitepaper/whitepaper.md

Any Questions ?