



ABC Wallet



EOS Studio

搭建你的第一个区块链游戏

Build your first blockchain game on EOSIO software

为方便演示，请大家提前安装 EOS Studio IDE

To get started, please install EOS Studio and set up EOSIO dependencies

Windows: <https://download.eosstudio.io/win>

MacOS: <https://download.eosstudio.io/mac>

Linux: <https://download.eosstudio.io/linux>



清华大学
Tsinghua University

Decentralized Application Kickoff

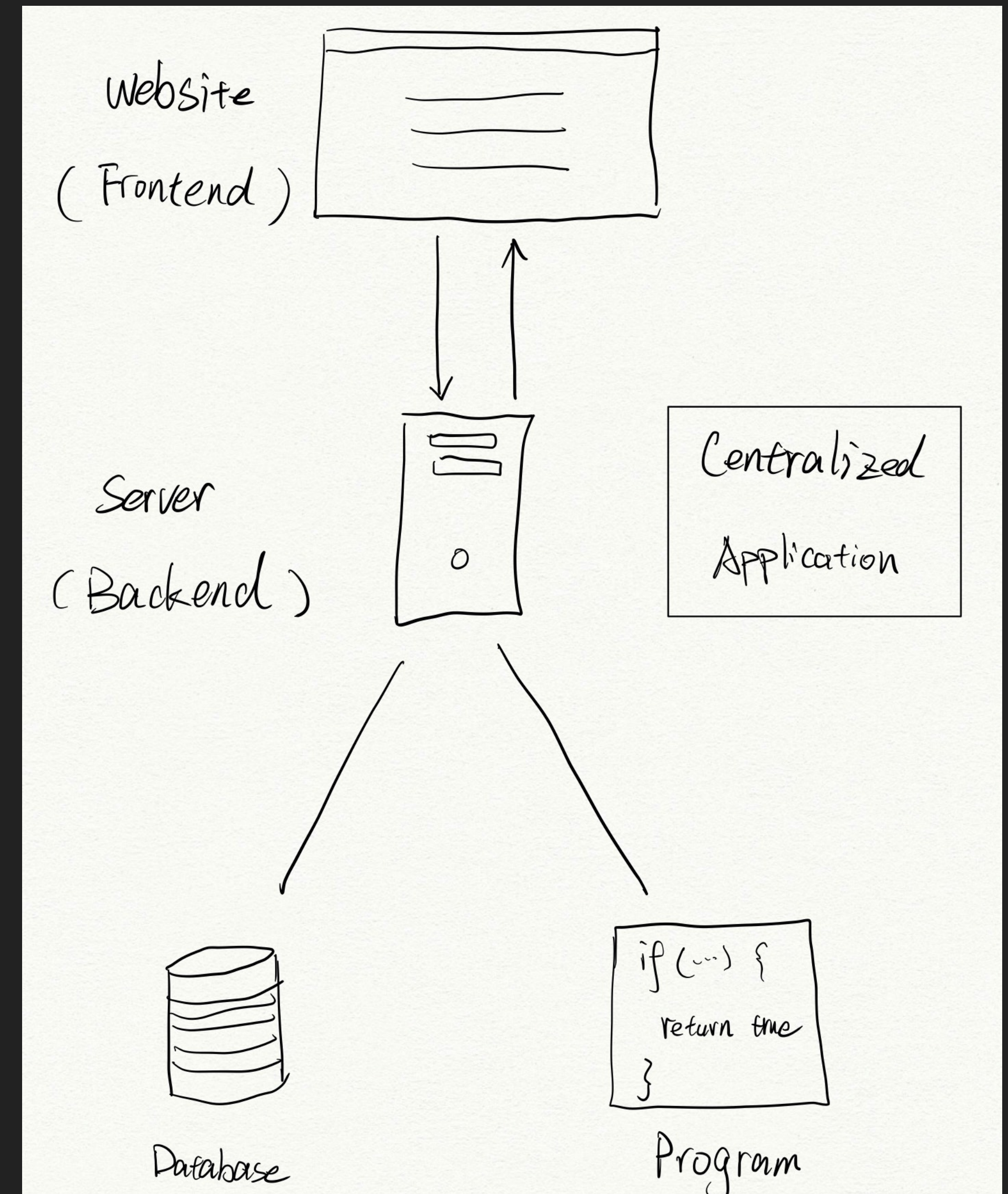
BLOCKCHAIN BLOG

Decentralized Application

Centralized VS Decentralized

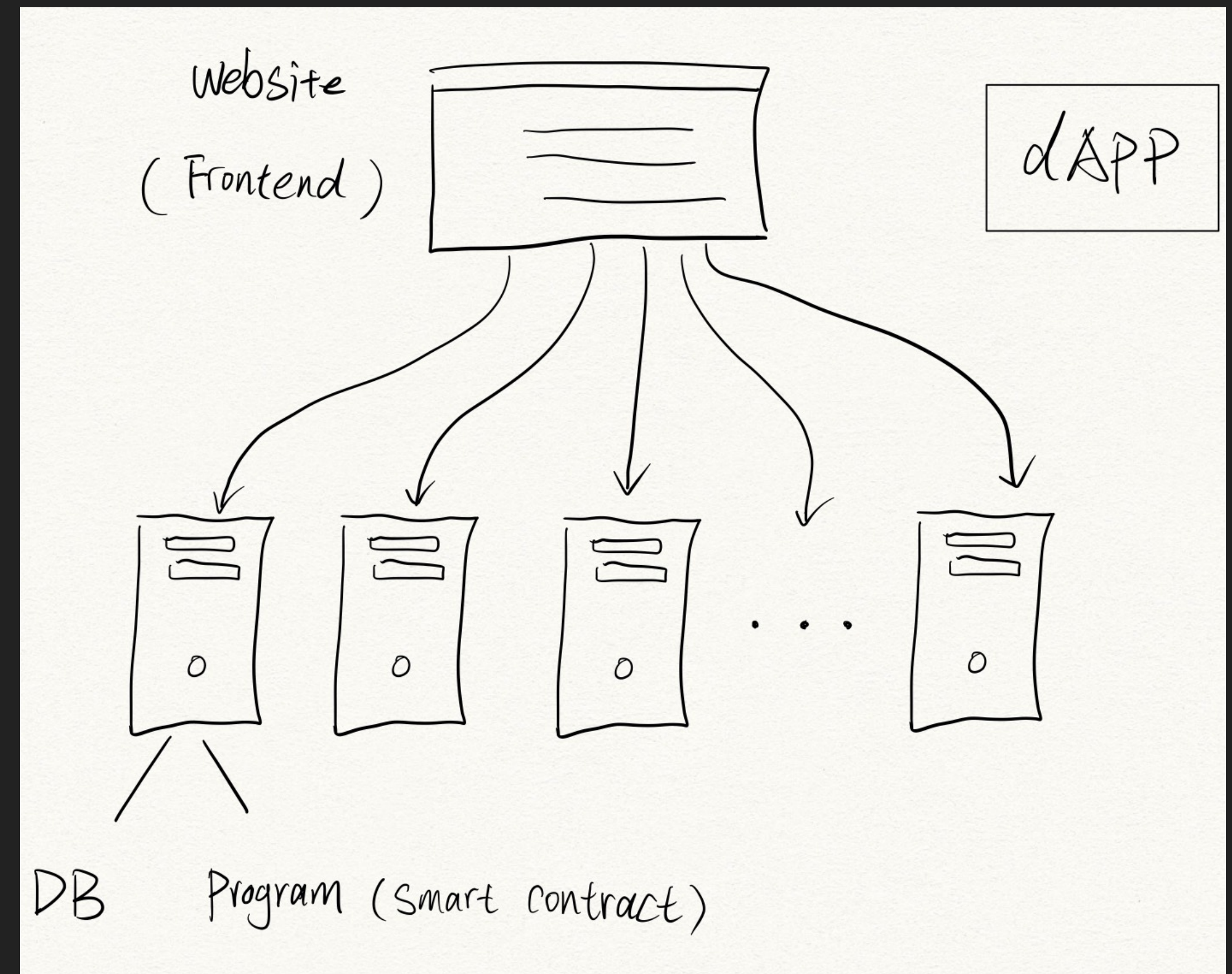
Centralized Application

- ▶ User Interface - Frontend
- ▶ Server - Backend
 - ▶ Program - Application logic
 - ▶ Database - Store data
- ▶ A centralized application is that:
 - ▶ one server receives a request and process it



Decentralized Application

- ▶ User Interface - Frontend
- ▶ Servers - Backend
 - ▶ Program - Application logic
 - ▶ Database - Store data
- ▶ The servers are running all over the world, NOT owned by one organization or company or person
- ▶ Each server has the same copy of the program and the database within the same network
- ▶ A decentralized application is that:
 - ▶ all the servers receive the same request and process it



TERMINOLOGY

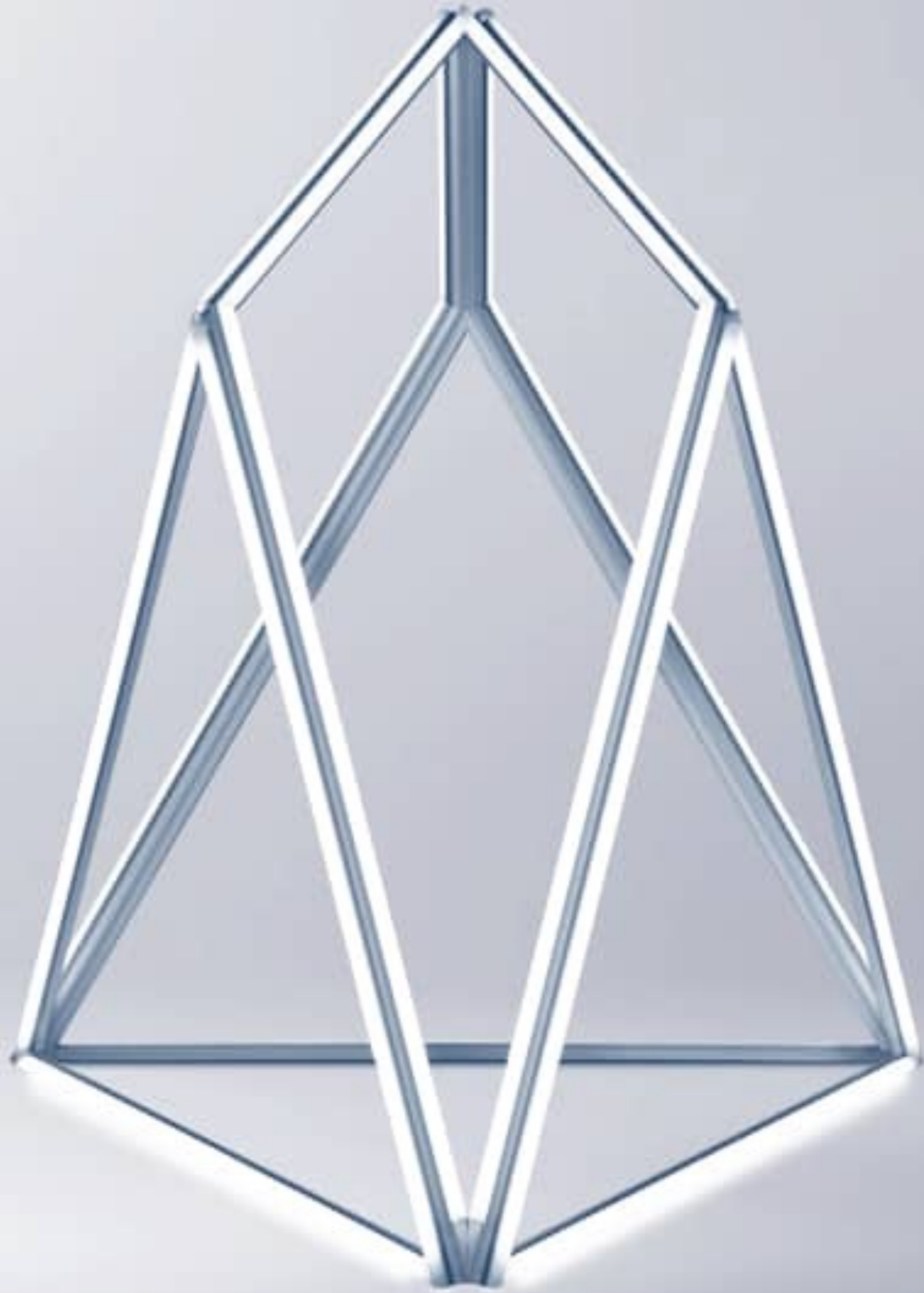
- ▶ Decentralized Application
- ▶ Server
- ▶ Program
- ▶ Request

TERMINOLOGY IN BLOCKCHAIN WAY

- ▶ dApp
- ▶ Node
- ▶ Smart Contract
- ▶ Transaction

DEMO

gd.eosstudio.io:300001



BLOCK.ONE

EOSIO



OBSIDIANS LABS

EOS STUDIO IDE

app.eosstudio.io

DATA STRUCTURE

Post Table

- ▶ title
- ▶ content
- ▶ cover image
- ▶ author
- ▶ category
- ▶ create time
- ▶ update time
- ▶ meta

Category Table

- ▶ category name
- ▶ count

Blog Information Table

- ▶ name
- ▶ description
- ▶ cover image
- ▶ version
- ▶ meta

SMART CONTRACT TERMINOLOGY

Terminology



CODING

WRAPPING UP

Terminology

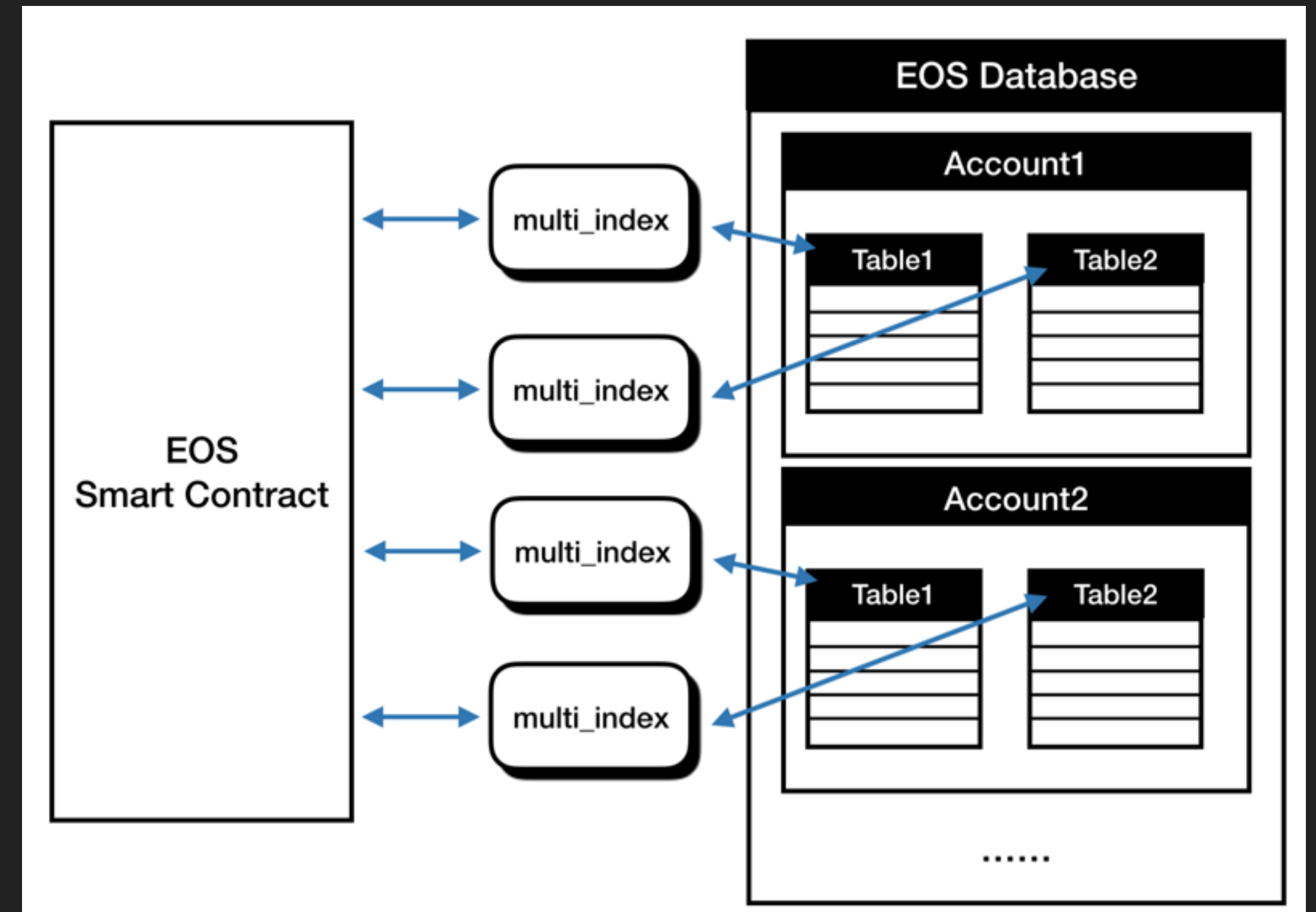


Terminology

- ▶ multi_index
- ▶ primary/secondary key
- ▶ 'name' data type
- ▶ CONTRACT / ACTION / TABLE
- ▶ emplace / modify / erase / find
- ▶ scope
- ▶ eosjs

multi_index

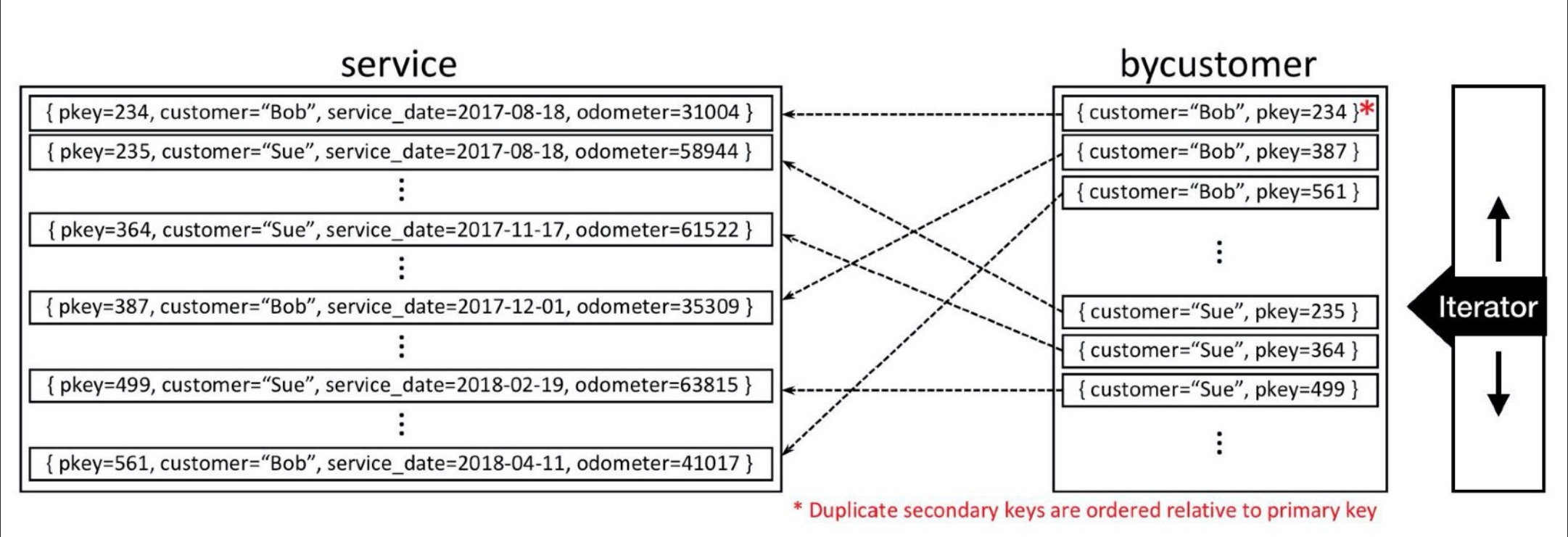
- ▶ EOSIO Multi-Index API provides a C++ interface to the EOSIO database
- ▶ EOSIO Multi-Index table requires exactly a `uint64_t` primary key



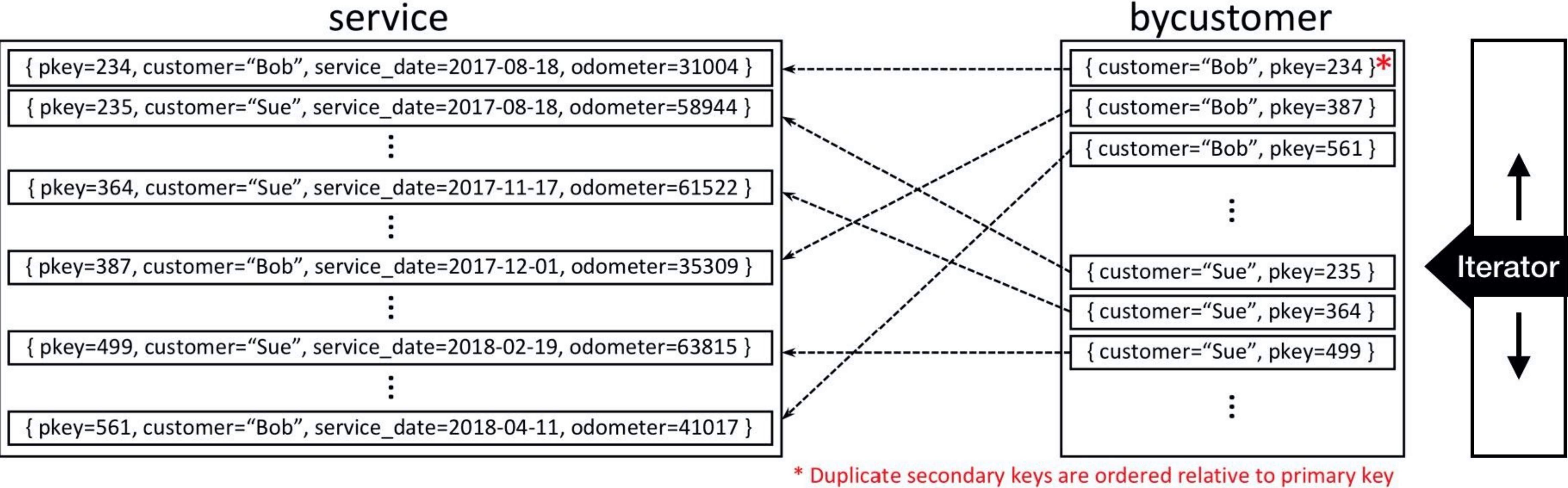
* `uint64_t` means 64 bit unsigned integer

secondary index

- ▶ Keep a table of all records with primary keys and secondary keys
- ▶ EOSIO Multi-Index table also supports up to 16 secondary indices. The type of the secondary indices could be any of:
 - ▶ uint64_t / uint128_t / uint256_t / double / long double



secondary index



'name' data type

- ▶ It can be converted to integer
- ▶ Can only contain the characters `.abcdefghijklmnopqrstuvwxyz12345`
- ▶ Must start with a letter
- ▶ Must be 12 characters
- ▶ Must not end in a dot

CONTRACT / ACTION / TABLE

- ▶ CONTRACT - smart contract class annotation
- ▶ ACTION - smart contract action(function/method) annotation
- ▶ TABLE - smart contract table annotation

emplace / modify / erase / find

- ▶ **emplace** - create a new record
- ▶ **modify** - modify an existing record
- ▶ **erase** - delete an existing record
- ▶ **find** - find the pointer (iterator) of a record

scope

- ▶ Scope is the group of table, you could separate data into different groups storing in the same table
- ▶ One record is stored under a particular scope
 - e.g. a post is stored under 'leonseosblog' account name
- ▶ Scope is using 'name' type

eosjs

- ▶ Javascript API for integration with EOSIO-based blockchains using EOSIO RPC API
- ▶ It offers a Javascript way to interact with EOSIO blockchain
 - we use '`eosjs.transact()`' to call the smart contract action;
 - and use '`eosjs.getTableRows()`' to fetch the data of a smart contract
- ▶ It translates the function call to the http request

THANK YOU