ChainIDE: A Cloud-based Integrated Development Environment for Cross-blockchain Smart Contracts

Han Qiu*, Xiao Wu† Shuyi Zhang†, Victor C.M. Leung‡, and Wei Cai§[1]
*Telecom Paris, Paris, France
†White Matrix Inc., Nanjing, China
‡Shenzhen University, Shenzhen, China
§The Chinese University of Hong Kong, Shenzhen, China
han.qiu@telecom-paris.fr, {ling, tim}@matrixdapp.com, vleung@ieee.org, caiwei@cuhk.edu.cn

***Abstract*—Blockchain has become novel solutions for many traditional issues in computer science and finance. Recently, with the release of Libra blockchain from Facebook, the decentralized finance concepts have another huge development. However, there are already many different blockchain systems existing now and the development of blockchain systems have become more and more complicated since each kind of blockchain development environment will consume time to be built. To make the programming on different blockchain systems more easily, we propose a novel cloud-based solution, namely ChainIDE, for the development of blockchain-based smart contracts on multiple kinds of blockchain systems. With chainIDE, cross-chain developing of smart contracts on different blockchain systems can be easily done without any time consumed by building the environment. Based on the operation statistics in this paper, we served more than 310,000 compilations in the past 30 days which makes us the most popular cloud-based cross-chain development platform in the world.**

***Index Terms*—Blockchain, Smart Contract, Cross-chain Platform, Libra, Cloud IDE**

I. Introduction

Since the first blockchain system, Bitcoin [1], was proposed nearly one decade ago, a huge number of novel properties and concepts were proposed such as decentralization that re-addressed some of the traditional problems such as accessibility and trustworthiness. The decentralized topology could ensure that no single entity controls the network; open access, which allows anybody with an internet connection to participate; and security through cryptography, which protects the integrity.

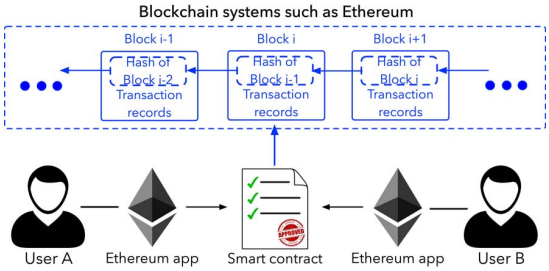

Fig. 1. A smart contract can be easily realized based on blockchain systems such as Ethereum.

Nowadays, many decentralized applications and systems such as cryptocurrencies [2], decentralized games [3], and smart contracts [4] are implemented based on the new properties and concepts brought by blockchain. For instance, a blockchain-based smart contract is a cryptographic box which stores information, processes inputs, writes outputs and is only accessible to the outside if certain predefined conditions are met. For instance, as shown in Fig. 1, the smart contracts can be realized by inserting code comprising a Turing complete set of operations that will be executed by the Ethereum [5] network once a smart contract is called. Such a computation results in several outcomes: alteration in the state of the smart contract, returning a result and transferring monetary value.

There are many blockchain systems today but the complexity of developing on blockchain systems is also rapidly increasing. For instance, before programming on the Libra blockchain [6], the setting up process is slow as users must first setup and build the environment which will consume more than one hour and Windows system is not supported. For the other blockchain systems, the downloading and building processes are all different from each other which will make the developing of cross-chain very costly.

In this paper, our contribution is to develop a cloud-based solution for developing on multiple blockchain systems by avoiding downloading, setting up, and building process. The current version of our system, named chainIDE, supports direct cloud-based coding on six popular blockchain systems including Ethereum [5], Libra [6], Cocos [7], IOST [8], Nervos [9], and Ultrain [10]. We propose a cloud-based solution for the programming of smart contracts for the needs of the cross-chain development. With a brief example, we prove that our solution can significantly reduce the cost of building the development environments for smart contracts which also makes us one of the most frequently used cross-chain development platform in the world.

We present the system architecture of chainIDE in Section 2, evaluation on saved time in Section 3, operation statistic analysis in Section 4, and conclusion in Section 5.

II. System Architecture

According to our knowledge, our platform is the first in the world that can support online cross-chain programming with a cloud-based approach to realize. In this section, the architecture of our platform will be illustrated. A brief demo of how the code is also presented in this section.

[1]Wei Cai is the corresponding author.

2330-2186/19/$31.00 ©2019 IEEE
DOI 10.1109/CloudCom.2019.00055

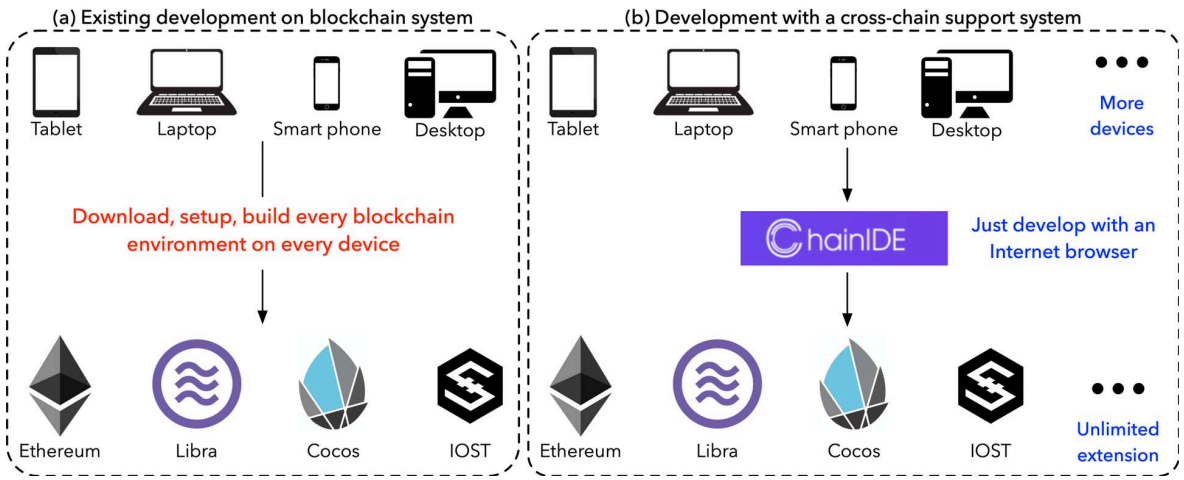

Fig. 2. The improvement of chainIDE: (a) current development on blockchain system requires complicated install, setup, and configure process on every blockchain system on every device; (b) chainIDE can support unlimited extension of blockchain system but users just need an Internet browser to develop.

An overview of the system architecture is shown in Fig. 2. The main function we realized now is to provide a cloud-based developing platform for the developing of several famous blockchain systems including Ethereum, Libra, Cocos, IOST, Nervos, and Ultrain. ChainIDE is built as a cloud-based solution that provides the development environment as a web-based interface for the blockchain developers, which in turn interacts with the blockchain. Based on the user's viewpoint, ChainIDE provides a graphical user interface with also the tutorial and help documents for the blockchain developers which can directly interact with the blockchain systems. Other than the uniformed graphical interface and pre-installed compilation environment for many famous blockchain systems, we also deployed the cache mechanism for the developers such that the uncompiled code will be saved in case the Internet browser was closed. Also, Cloudflare is deployed to accelerate the compiling process for special network environments. Thus, as shown in Fig. 2, with such cloud-based approach, users can program on any digital devices without any preparation of the develop environments as long as they have the Internet connection and the Internet browser that supports Java Script.

Our platform also serves the most quantity of the compilations of the new released Libra blockchain system. Along with the release of the Libra, a brand new programming language dedicated to the Libra blockchain system called 'Move' is also released by Facebook [11]. According to [11], 'Move' is the smart contract platform language mainly designed to (1) issue cryptocurrencies, tokens, and digital assets; (2) handling blockchain-based transactions; (3) managing validators.

On our platform, Move language is well adopted and supported. As shown in TABLE I, there are more than 120k compilations on Libra through our platform in the past 30 days. We also give a brief example of the Move programming language compiled on Libra blockchain as shown in Fig. 3. The small program is used to check the target address balance. In line 4 of Fig. 3, the variable `record\_balance` is set as u64 type. Then, the `LibraAccount` is used to get the balance of the target address `targetAddr`. In line 6, the error code `18` will be returned once the `assert` operation fails with the condition of `record_balance > 1`.

```
1  import 0x0.LibraAccount;
2
3  main(targetAddr: address) {
4      let record_balance: u64;
5      record_balance = LibraAccount.balance(copy(targetAddr));
6      assert(copy(record_balance) > 1, 18);
7      return;
8  }
```

Fig. 3. A demo code with Move language of Libra [6] on chainIDE.

III. Evaluations

As pointed in Section I, there are many practical obstacles for slowing down the developing on blockchain systems since the downloading and the configuration will cost time. In this section, we measure the time cost by the necessary steps before one user could start developing with the examples on the Libra blockchain. Firstly, as shown in Fig. 4, there are mainly two necessary steps for a user before he could start the programming on Libra blockchain. Same with any other blockchain systems, there are instructions from Libra that can help users easily perform the necessary steps to download, setup and build the development environment for Libra.

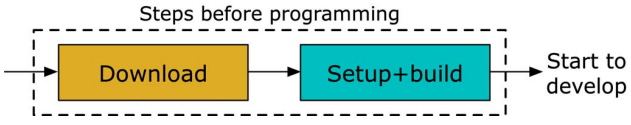

Fig. 4. Two steps before programming on blockchain.

Practically speaking, these two steps will still consume time and a brief comparison is given in Fig. 5. Although the download time is based on the network condition, the users in China will have a special network environment which makes the download time extremely long (around one hour) compared with the users in USA or France. We also give the time consumed by the setup and build steps on three different machines including AWS ec-2 medium [12], AWS

ec-2 small, and a Mac Pro PC as shown in Fig. 5 (b). The fastest machine in our test will consume 45 minutes to setup and build the environment for only Libra blockchain. However, with using the chainIDE with an Internet browser, there is no preparation before programming on Libra blockchain and the time consumed by making the developing environment will be zero minutes. Moreover, considering some use cases that the users need to develop on multiple blockchain systems or to develop on different devices, using cloud-based chainIDE will help to save more time consumed by setting up and building.

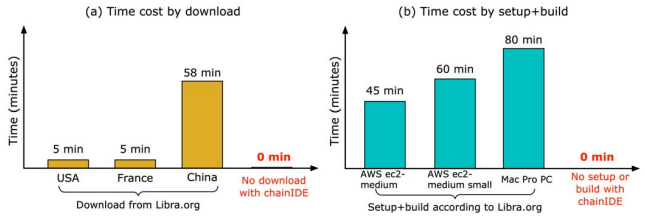

Fig. 5. Time consumed for the necessary steps before users can start programming on blockchain.

IV. Operation Statistics

Currently, there are a huge number of developers on our platforms from academics and industrial. In the past 30 days, there are totally **310,212 smart contracts compiled** on the chainIDE. Based on the statistical results, currently, we are the most popular developing IDE for blockchain systems such as Libra and IOST. The category distribution of the 310k compiled smart contracts is given in Fig. 6. Libra is the most popular and mostly used blockchain systems for deploying smart contracts. Particularly, there are **120,362 smart contracts compiled on Libra blockchain** in the past 30 days through the chainIDE.

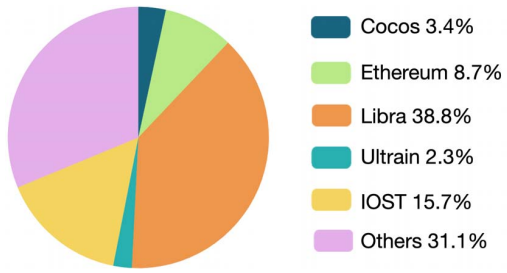

Fig. 6. The distribution of the compilations through chainIDE on different blockchain systems.

We also list the category distribution of smart contracts compiled on the chainIDE in TABLE I. According to our statistical analysis, there are more than 282k compilations (more than 91%) are based on the tutorial codes of smart contracts which means there are many beginners to program on smart contracts. Then, for the other categories, the most popular smart contracts are used for the Decentralized games (Dapp games). There are more than 18.9k compilations (more than 6%) for Dapp games which means the Dapp games are very popular today. This is due to the decentralized topology of blockchain system that could help to avoid the gaming rules manipulated by the big game companies and the cheating [13]. For the other categories, the Decentralized Finance (De.Fi) [14] is also a popular kind of smart contracts for financial use and the is seen as one of the next generation financial services. This is also the most popular smart contract on Libra blockchain for the online decentralized digital banking.

TABLE I
Libra compiling distributions of smart contracts.

Category	Smart Contract	Tutorial	Dapp Games	Others
Quantity	310,212	282,293	18,924	8,995
Ratio	100%	91.0%	6.1%	2.9%

V. Future Work and Conclusion

Blockchain systems have become a powerful tool to change the existing solutions in computer science and financial domain. However, how to efficiently develop on blockchain became an urgent issue since there are too many blockchain systems in the world now and the developing environment is becoming more and more complicated to build. Also, there are various kinds of programming languages for blockchain systems that are not friendly for beginners. Thus, we presented chainIDE that can make the developing on blockchain systems easily done with ultra-efficiency.

We believe with the future deployment, there will be more blockchain systems supported by chainIDE and the developing on chainIDE will be more and more friendly for everyone. Meanwhile, we can also collect the tutorial documents and code demos to further help the blockchain developers.

References

[1] S. Nakamoto *et al.*, "Bitcoin: A peer-to-peer electronic cash system," 2008.
[2] A. M. Antonopoulos, *Mastering Bitcoin: unlocking digital cryptocurrencies*. " O'Reilly Media, Inc.", 2014.
[3] W. Cai, Z. Wang, J. B. Ernst, Z. Hong, C. Feng, and V. C. Leung, "Decentralized applications: The blockchain-empowered software system," *IEEE Access*, vol. 6, pp. 53 019–53 033, 2018.
[4] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in *2016 IEEE symposium on security and privacy*. IEEE, 2016, pp. 839–858.
[5] G. Wood *et al.*, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum project yellow paper*, vol. 151, no. 2014, pp. 1–32, 2014.
[6] A. Z. et al., "The libra blockchain," 2019. [Online]. Available: https://libra.org
[7] T. Min, H. Wang, Y. Guo, and W. Cai, "Blockchain games: A survey," *arXiv preprint arXiv:1906.05558*, 2019.
[8] I. O. T. Group, "Iost," 2018. [Online]. Available: https://iost.io/#community
[9] D. L. Terry Tai and K. Wang, "The Nervos network," 2018. [Online]. Available: www.nervos.org
[10] E. L. Ray Guo and W. Li, "Ultrain," 2018. [Online]. Available: https://ultrain.io/
[11] S. B. et al., "Move: A language with programmable resources," 2019. [Online]. Available: https://developers.libra.org/docs/move-paper
[12] Amazon, "Amazon ec2 instance types," 2019. [Online]. Available: https://aws.amazon.com/ec2/instance-types/
[13] H. Y. Yuen, F. Wu, W. Cai, H. C. Chan, Q. Yan, and V. Leung, "Proof-of-play: A novel consensus model for blockchain-based peer-to-peer gaming system," in *Proceedings of the 2019 ACM International Symposium on Blockchain and Secure Critical Infrastructure*. ACM, 2019, pp. 19–28.
[14] P. Treleaven, R. G. Brown, and D. Yang, "Blockchain technology in finance," *Computer*, vol. 50, no. 9, pp. 14–17, 2017.