Разработка прототипа блокчейн-системы с динамически задаваемыми ограничениями на смарт-контракты

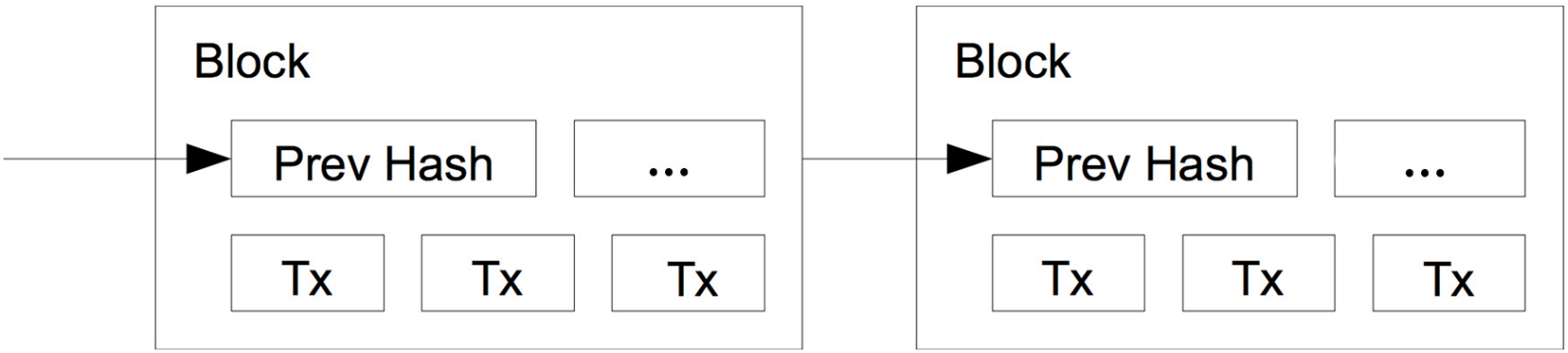
Алексей Андреевич Фефелов
Научный руководитель: ст. преп. Я. А. Кириленко
Консультант: к. ф.-м. наук Д. А. Березун

Кафедра системного программирования СПбГУ

22 мая 2019 г.

Блокчейн

- Одноранговая (p2p) сеть
- Транзакции объединяются в блоки
- Блоки объединяются в цепь



- Смарт-контракт — программируемый объект, работающий поверх блокчейна

Основные преимущества блокчейна

- Неизменяемость
- Прозрачность
- Распределенность
- Отсутствие центрального органа управления

Основные недостатки блокчейна

- Высокие финансовые и вычислительные издержки на поддержку сети
- Наличие известных уязвимостей
- Низкая скорость обработки транзакций
- Юридическая неопределенность

Постановка задачи

Цель работы — разработка прототипа блокчейн-системы, ограничения на выполнение смарт-контрактов в которой могут динамически задаваться третьими лицами

- Сделать обзор существующих блокчейн-систем
- Разработать архитектуру системы
- Реализовать систему
- Провести тестирование и апробацию

Блокчейн-системы

public

- Bitcoin
- Ethreum
- ...

private

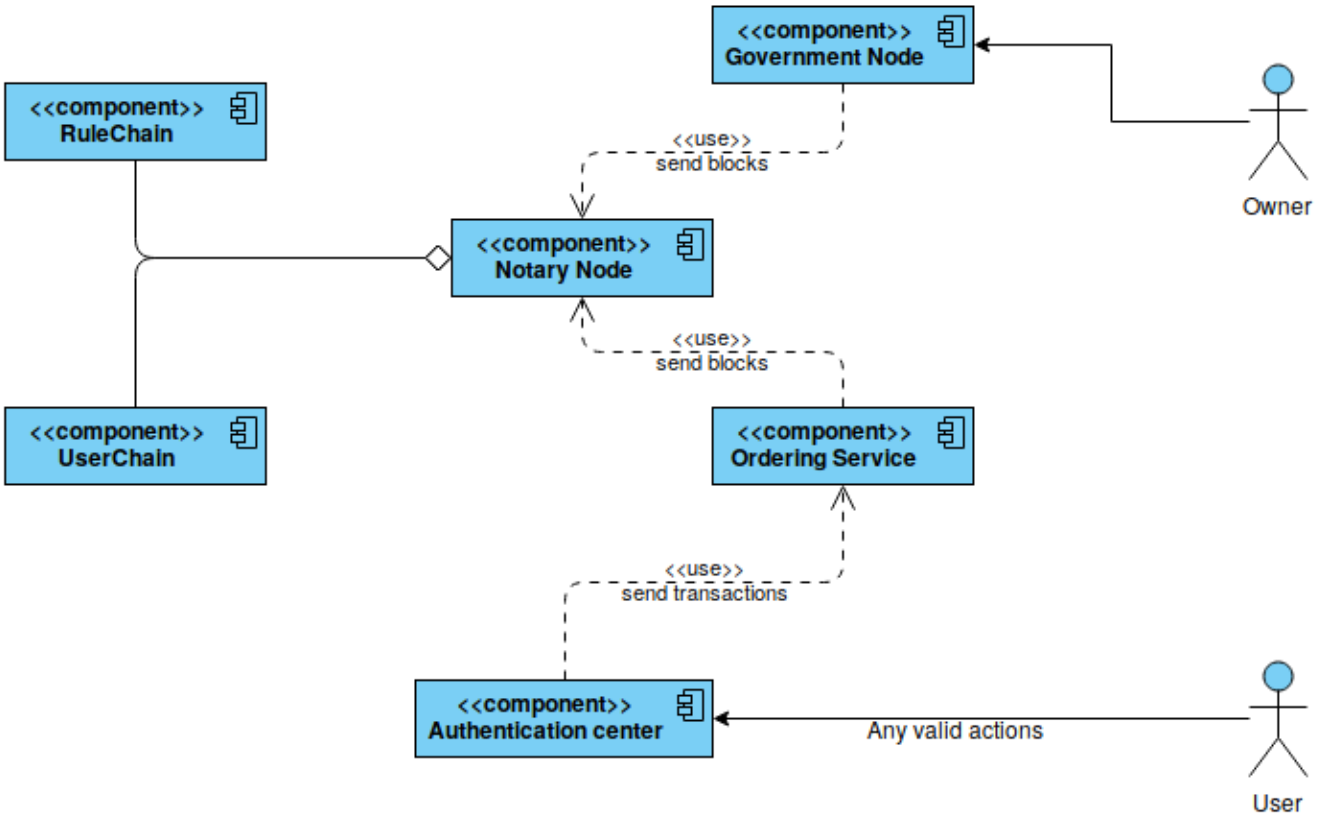
- Hyperledger-Fabric
- Cosmos
- ...

Статья: Survey on blockchain technology, consensus algorithms, and alternative distributed technologies

Описание системы

- Система состоит из двух блокчейнов, взаимодействующих друг с другом
- В первом хранится набор допустимых действий и ограничений (требований)
- Можно заключать только те типы сделок, для которых заданы требования
- Код: github.com/fefaleksey/FinChain

Диаграмма компонент



Пример контракта

```
public class Contract
{
    public bool IsActive { get; private set; } = true;
    public AccountAddress Owner { get; }

    public Contract()
    {
        var bytes = new byte[16] {1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1};
        var guid = new Guid(bytes);
        Owner = new AccountAddress(guid);
    }

    public void Execute(IActionExecutor executor, AccountAddress sender, params object[] paramsList)
    {
        if (!sender.Equals(Owner))
        {
            return;
        }
        var bytes2 = new byte[16] {2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2};
        var guid2 = new Guid(bytes2);
        var receiver = new AccountAddress(guid2);
        int amount = 100;
        object[] @params = {receiver, amount};
        executor.Execute(ActionType.TransferMoney, sender, @params);
        IsActive = false;
    }
}
```

Результаты

- Сделан и опубликован обзор существующих блокчейн-систем
- Разработана архитектура системы
- Реализован прототип системы
- Проведена апробация