

A Decentralized Resource Allocation System

Abstract

Resource allocation is a commonly seen task. Typical cases are module selection, or labor division and allocation. This proposal first describe one widely used resource allocation model implemented by centralized architecture, followed by its weakness, and then propose a decentralized improvement by blockchain technology.

A resource allocation model

One process for resource allocation task is that, initially users claim their request, and then users could get the resources they needed if there are enough resource, or otherwise resources would be randomly allocated to users demanded. After that users could release the resources they don't want any more, and compete for spare resources, usually in a "First Come First Serve" order.

Often this model implemented by a client-server architecture. Inherently it has some weakness. First of all, allocation results might be manipulated by central authority. And from the technical perspective, Dos attack to the central server, or burst traffic, is always an annoying problem.

Improvement by blockchain

We could improve the model above by blockchain technology. Since blockchain is a decentralized architecture, the weakness inherited from C/S architecture would be solved naturally.

At the first stage, users claim and broadcast their requests. After deadline of the first stage, the system collect all the requests and allocate resources according to the resource quote list: if enough then allocate the resources to the user who needed, or otherwise randomly. This first stage don't adopt the "Fist Come First Serve" principle because that might result in severe competition among all users and thus network congestion. the result of the first stage become the genesis block.

At the second stage, users can release resource they don't want any more, by submitting a release transaction. Also they can submit a request transaction, in this stage "Fist Come First Serve" principle is adopted. Each time a node mines a block, it will organize the transactions (both release and request) in a sequential order, and does addition and subtraction on resources quotes. Valid transactions will be accepted into blocks, while invalid transactions will be abandoned. The Request an Release operations form the backbone of this allocation model. Furthermore, we can add more features into this system, such as limiting the number of resources a user could hold, or adding resource exchange functions.

Development Schedule

We use Ethereum as the infrastructure to develop this allocation system. Ethereum has encapsulated complicated procedures like digital signature, proof of work (stake) and mining, so that we can focus on application layer. Also, we might need a user friendly interface. Web app usually serves this purpose.

2 or 3 persons will do the blockchain development, and 2 persons do the user interface will be enough. The whole project could be completed in less than 1 month, since this is just a prototype demo.

Time	Job
3rd Feb	Submit proposal

Time	Job
9th Feb	First minimal releasable product
16th Feb	Second minimal releasable product
23th Feb	Third minimal releasable product
1st March	Final system, all features frozen

Among the schedule, the first minimal releasable product should have basic Request and Release operations and a simple UI (second stage of the model). The second minimal releasable product should tackle the initial allocation job (first stage of the model).

Team Information

Team Name: **BlockDance**

Team Members:

Stu No.	Name
Yunxiang Cai	1906262
Yi Li	1901668
Ziwei Yang	1925274
Shijia WU	1905583
Ruiqi Zhu	1904349