P2P bitcoins trading system

Gzmask Late
Gzmask.github.com

November 15, 2011

Abstract

This is a paper of a P2P system that let users trade digital works online. There are some basic peer-review processes that guarantee the works are paid fairly.

1 Introduction

Online trading using credit cards or paypal-like system is popular. A famous exampe is Ebay.com. All these systems are centralize, thus the trades are slow and controlled by some private parties in some degree. As bitcoins emerged, pure P2P system is getting recognized. Here I proposed a very basic and simple design for a P2P trading system.

2 Important constants

N: Total user amount (updates weekly)
K: Total voter amount (updates three days)
O: offer of the posted Need
P: process fee of the dispute.

$$P = smaller(\frac{O}{10}, \frac{O}{K})$$

R: reward for each voter.

$$R = \frac{P}{(K \times \frac{correct\_votes}{total\_votes})}$$

3 Models

3.1 Node

This model stores in every node
Properties

- IP address
- is_voter

3.2 Need

This model stores in every node
Properties

- time_available
- offer
- src_node_address

3.3 Work

This model stores in posted node
Properties

- proposal
- time_available
- price
- src_node_address

- des_node_address
- solution
- accepted

3.4 Dispute

This model stores in posted node
Properties

- needer_ip_address
- worker_ip_address
- vote
- process_fee
- next_voter_address

4 Actions

4.1 Post a Need

1. a node U_1 boardcasts$O(N)$ a need N_1 in the network
2. all other nodes U_x receive the boardcast of N_1 and store it locally
3. after the time_available expired, each node removes N_1 permanently.
4. when U_1 logs off, it will not longer be able to accept proposed works. U_1 is supposed to be online until a solution is accepted.

4.2 Propose a work

1. any node can propose a work for a need, in this example, N_1.

2. let U_2 be a node proposing a work W_1. U_2 send a message to U_1, telling U_1 that W_1 is proposed at U_2.
3. U_1 gets notified that W_1 is proposed. There can be multiple works that are proposed by other nodes.
4. U_1 can accept one of the proposed work, or wait. If N_1 expires, all proposed works expire at the same time and the case is over.

4.3 Accept a work solution

1. let U_1 accepts W_1 from U_2. Then U_1 sends a message to U_2, W_1 is accepted and U_2 can give a solution to W_1.
2. After U_2 submits a solution for W_1, U_1 receives a message notification that he can review the solution from U_2. Now U_1 can either chose to accept or reject the solution.
3. If the solution is accepted, the deposit bitcoins in N_1 will be transfered to U_2.
4. If the solution is rejected, then U_1 submits a dispute D_1, which is boardcasted to the network. And starts the peer-review voting process.

References