2i2i.io the infobox

1m1

email@1m1.io

Abstract— The *infobox* is a generalization of the *inbox*. An *outputing entity* sends *info* to an *inputting entity* with *coin_s attached, which signify social importance. The coins represent an arbitrary bundle of energy or information. The infobox is ordered according to social importance following the theory of , which incorporates* coin_s of both *objective* as well *subjective* values *fair* ly.

Index terms—Scientific writing, Typesetting, Document creation, Syntax

Lorem ipsum dolor sit amet, consectetur adipiscing.

I. Begin

1. O sends info attached with coins to I.
2. I receives the coins if it replies.
3. O can cancel anytime.

II. Definitions

III. SmartTreasury

IV. FairVesting

V. Tokenized Time

async, access to info is access to time, fair as time costs energy

VI. dApp design

VII. Tokenomics

The info box. The io highlighted. It's a generalization of an inbox. It's where info comes to you in the form of text messages. And Ordered by Importance Defined by Society. Each Info in the infobox comes attached with some currency. The currency presenting an arbitrary bundle of energy or information, in the form of any coin. the 2i2i theory gives us the ordering of this inbox. And fairly distributes time and access to information. Note that it is not simply more coins giving more access, which would be an auction market, rather the 2i2i theory formalizes a generalized market containing the auction, the fixed and the barter markets as special cases e.g..

Currently many sources of information do not give any access to the world at all. Lots of people do not publish their direct email. Because they would otherwise be inundated with messages. Instead everyone should openly allow access, which is ordered according to social importance, yet given only at individual will. The more someone needs to reach another person, the more importance in the form of coins can be attached to the info, which can be of objective xor subjective value. This is just like the google algorithm developed in the 90s. That created gmail. That sorted the inbox according to relevance to one's own chosen search term. 2i2i improves the inbox into a social importance based infobox. Any entity can send a message to any other entity. The recipient of the info receives the coins if it replies. The sender of the info can anytime. Cancel. The mess in which case the message disappears. From the inbox. Of the other. And the sender, retrieves. It's coins.

The inbox is a smart contract allowing for a fully trustless social importance driven inbox.

The recipienth has two simple parameters, which in real time can change the order of the messages. Such a change only takes effect for new messages. Old messages have their order guaranteed.

These two parameters are explained in detail in the 2i2i white paper.

Links are of the following format: https://2i2iio?a=yourid Here a can be changed to refer to different types of addresses. 2i2i will be available "omni"DLT. All major ones. The dapp, the front end, will also exist on some decentralized storage as bundle of WASM and javascript.

The smart contracts will hold literally any coin on any DLT supported. The recipient, on reply, will unlock the coins. The recipient can at any time retrieve all unlocked coins from the inbox.

Alternatively, as an improvement, the inbox could auto transfer unlocked coins, as a setting, though less efficient from a fees point-of-view.

If you turn on the auto-feature, the frontend initiates transaction but you still have to sign it. You don't want to give away power. You always have to sign into for security, but it can be auto initiated if you choose so, or just let it accumulate. Which then saves transaction costs. The only costs are transactions.

The sender should try to cover the transaction costs of the receiver to reply. The problem is that the Costs can be difficult to predict depending on the DLT. The costs on the most widely adopted DLT, the EVM, e.g. can change significantly even over short periods of time.

The sender could send the best estimate. If we make it an option, it complicates the system for the user. Better we choose some way. Since the sender has to pay for its own transaction cost anyways some gas has to be paid. The sender must have at least some. Estimate as best as possible, and just add that to the initial send transaction making the recipients life easier. Otherwise, we're basically saying to the sender, you need to have some coins to send the message, which comes with some coins. Make sense. But to say to a recipient, you need some coins in order to receive coins is more awkward. Delegate the recipient's transaction costs to the sender whenever possible.

Each DLT contains a treasury that is sent ten percent, rounded down. The treasury in return provides 2i2i coins in exchange to the inbox. The inbox paid a tax and got the 2i2i coins in exchange. Any sender can pull from there. Of course, that costs storage. And if you don't well, you're missing out but there is locked energy there. It is energy and this is the taxes' is worth.

These tokens allow for governance to update the FX smart contract. The treasury is a smart treasury, a buy-back-and-make machine.

The FX smart contract given a currency either returns "subjective" in case the currency is of subjective value xor returns a numerical value which defines the FX rate of the input currency versus a globally defined base objective currency. The smart contract goes to liquid markets to retrieve the current spot rate, as a TWAP or middle of the spread.

The community via governance can move a currency between the dichotomy of objective and subjective value currencies.

Or update liquidity requirements or links to the markets used for the data. E.g., switching to another DEX.

The end goal is complete decentralization with 2i2i coin holders deciding everything.

The frontend which lives on decentralized storage has a centralized location which links to the different versions, including the latest. This allows for users to receive seamless updates in a decentralized manner. The code being open source of course allows users to access older versions, if wanted as well.

The asynchronous nature of replying to messages, whenever and if ever one wants to, makes this a no cost, only potential profit proposition.

Everyone should create and publish it's 2i2i address and communite with people.

Finally, everyone gets access to everyone. In a fair manner.

REFERENCES