

utunga / **thankful** Public

Code Issues Pull requests ZenHub Actions Projects Wi

master ...

**thankful / doc / theory.md**

 utunga Update theory.md History

1 contributor

**⚠ We found potential security vulnerabilities in your dependencies.**  
Only the owner of this repository can see this message.  
[See Dependabot alerts](#)

302 lines (174 sloc) | 34.8 KB ...

# Kinpesa - A system for building reciprocity games

## Contents:

- Introduction
- Acknowledgements
- Identity
- Nodes
- Clearing and default
- Agents
- Trust lines

Sign in now to use ZenHub

- Tokens
- Regarding decentralization
- The zero circle
- Visualization and the 'hard problems'

## Introduction

---

This is an attempt to create a framework to model (and simulate) markets with distributed information that allows for the creation of money like instruments (specifically 'credit/debt') without assuming a-priori the existence of centralized credit creating organisations (like governments or banks) or the existence of a centralized ledger (like most blockchains). That is it aims to avoid the assumption of any entities having global information.

This is also an attempt to define a general model on which various 'games' can be defined. The optimal outward form (UI and rules) of such 'games' is outside of the scope here, but the objective is that all games of interest could be defined on top of the basic structure described.

I am specifically interested in games which allow:

- the democratization of credit creation process
- for credit in the game to (nevertheless) retain value and be robust against shocks or boom/bust cycles
- for real value to be derived by participants of the game
- for the experience of interacting with money/credit to be more gentle, and enjoyable (psychologically speaking)
- for the celebration and acknowledgement of people that contribute to their community
- the support of vulnerable persons that need help

At this point, no claim can be made that games built on top of this framework can achieve these objectives, but these objectives are still noted for the record.

### Regarding the assumption of 'commodity money'

In the following, I assume 'a-priori' the existence of 'hard currency' tokens or reserves that 'hold value'. These tokens could include bitcoin, crypto tokens, pre-authorized credit card transactions, or claims on bank credit.

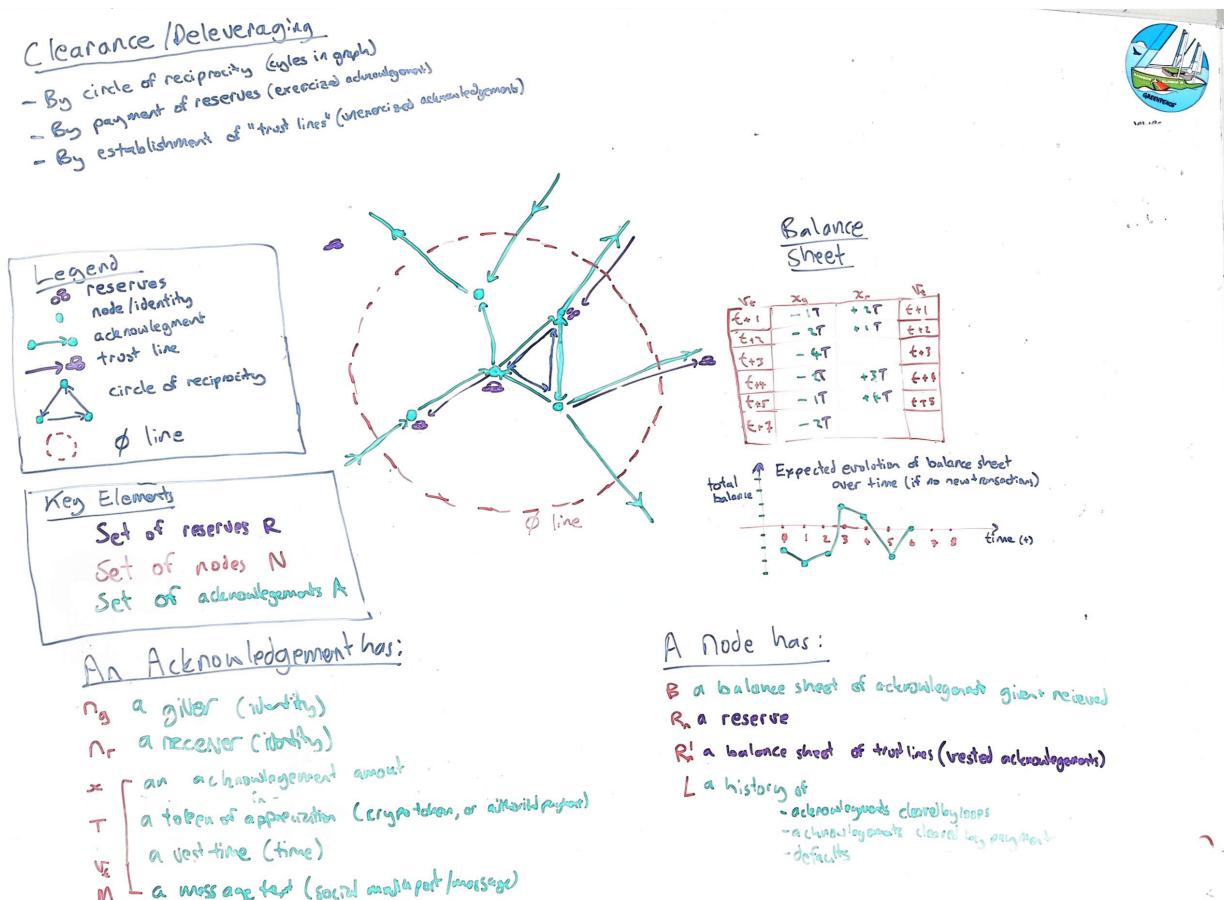
[Sign in now to use ZenHub](#)

For the discussion to proceed it is easier to assume these tokens already exist. However, once we have advanced the discussion to a certain point it will become clear that it would also be possible - in certain types of game and only if desired - to do away with such 'hard currency' tokens and still have a valid system for creating credit (i.e. entirely 'in game'). This is a bit like pulling up the ladder into an airship. At some point it is necessary to demonstrate that airship can be lighter than air - but it is easier, to explain how it works while the airship is still on the ground.

## Key elements

We imagine a directed, weighted, graph with:

- **N:** A set of nodes
- **R:** A set of reserves
- **A:** A set of acknowledgements
- **I:** A set of identities



- Acknowledgements are edges between identities
- Identities 'point to' exactly one node
- A node has one or more identities

[Sign in now to use ZenHub](#)

- Reserves are 'at' nodes
- 'Vested acknowledgements' are also referred to as 'trust lines'

## Acknowledgements

---

An "acknowledgement" is, strictly speaking, an edge between two identities. However, identities can be thought of as simply labels on nodes, so for ease of expression we may refer to acknowledgements as being between nodes in some cases.

An acknowledgement has:

- **ig**: the giver (an identity)
- **ir**: the receiver (an identity)
- **M**: a message text (text, image or attachment)
- **xT**: the acknowledgement amount (x) denominated in token T
- **vt**: a vesting time (time)

The last two properties are optional. The message text (M) can be empty if there is an amount (xT) specified. To be considered valid, an acknowledgement must also be made available via a "post" to some sort of pseudonomous public channel (such as social media).

The tweet: "Thanks @joe for the help in the garden last tuesday 2h" would be a valid acknowledgement. (Where the h refers to a fictive 'hours' token).

A bitcoin transaction can also be modeled as an acknowledgement - in this case a 'hard currency' acknowledgement with a vesting time of zero, immediately exercised and (generally speaking) an empty message.

Up until the vesting time the acknowledgement is essentially just a 'promise to pay'. In the case of 'hard currency' acknowledgements, after the vesting time the acknowledgement becomes a 'trust line' or a claim on the reserves at that node. A vested acknowledgement can be 'exercised' to transfer tokens to the receiving node or left un-exercized.

The exact meaning of 'default' is discussed in more detail below but as a first approximation if a trust line is exercised and there is not sufficient reserves the acknowledgement is considered to be in default.

[Sign in now to use ZenHub](#)

It is also possible - with agreement of all parties in the chain - to formally 'write off' or 'de-leverage' a chain of acknowledgements that form a cycle in the graph (this is called a 'circle of reciprocity' in the below).

### Sitting on top of existing distribution channels

To allow this 'game' to be distributed by posts on top of existing public/social media neither the sender nor the receiver need to have signed up with any third party service before they send or receive an acknowledgement. Furthermore the system is designed so that an acknowledgement can be made in plain text and published to normal social media channels, which can then trigger actions by trusted third parties, or identity proxies if need be.

Despite this, it is expected that most senders will have interacted with third party apps at some point. Third party apps can help guide users as to the 'rules of the game' and appropriate formatting of the message and provide many other useful services as well. However, it is critical both to the decentralization goal, and to allow virality that neither sender nor receiver strictly *need* to utilize any third parties to make acknowledgements or generally to interact with the 'games' defined in this way. The user should be able to interact entirely by signed messages on public media channels (potentially observed by trusted agents or third party apps).

It is expected that recipients of acknowledgements in particular may be first exposed to the concept aka 'introduced to the game' at the time when they receive their first public acknowledgement which therefore needs to be self evident as to its function.

[Sign in now to use ZenHub](#)

For instance, a post on social media along the lines of "Hey @Stuart thank you for helping out in the garden last weekend 2h" may (if you add an associated image and



a link for context) look like the following:

[Sign in now to use ZenHub](#)

A third party site which gathers together all the acknowledgements of this type, might look like this:

The screenshot shows a user interface for a social network or time bank platform. At the top left is a profile picture of 'Sarah Lorens'. The top right features a search bar and several icons, one of which has a red notification dot. Below the header, there's a sidebar with options like 'thank / wish / create' and a camera icon. A message box contains a thank you note to 'John' about a garden. The main content area displays a timeline of posts:

- Max:** Hey @Sue, Thanks for the help with the #sheep! (with a photo of sheep).
- Oh I Wish... (Jack):** Can anybody come next Tuesday for a working bee in my garden please? (with a photo of a garden).
- Fion:** Wow @Jo! Thanks to you the house-track is ready for adventures! #Builder hero!!! (with a photo of a wooden cabin).
- Alex:** (partially visible, showing a photo of hands working on a project).

On the right side, there are sections for 'COMMUNITIES' (showing 'KAPITI TIME BANK' with a logo) and 'LOCAL SKILLS MAP' (a map of an area with various location markers).

Third party apps or agents may choose to monitor messages delivered via public channels in this way and if necessary use their proxied identities to add meta-data or extra information and publish that elsewhere - as discussed below.

## Identity

An 'identity' is any form of meta data that establishes the *signing authority* of the **sender** (a post on a public timeline, a public/private key pair) or something which clearly *identifies* the intended **recipient** (for instance referring to the intended recipient by using their @handle in a message).

### Claiming nodes

A node is considered to exist as soon as an acknowledgement pointing to it is created (and observed). A node may, in fact never make any signed statements using the identity pointing to it. Such 'unclaimed' nodes can have only one identity.

[Sign in now to use ZenHub](#)

A recipient node created in this way can be 'claimed' at any time by someone with signing authority over the identity in question.

## Merging identities

Once a node has at least one signed identity that node can be merged with another creating a single node with multiple identities. They do this by publishing relevant signed public messages under both accounts indicating that both identities acknowledge the other. An identity may also repudiate another identity (causing a node split).

One reason that the difference between nodes and identities is important only because acknowledgements are from and to identities but reserves and reputations are calculated against nodes.

## Identity proxy

Acknowledgements can also be sent via private message, or through a third party as a proxy. In this case some form of (potentially anonymized) public record must then be made available. For instance, a message to a private facebook chat may be monitored by a bot invited into that chat who could then repost a (possibly pseudo/anonymized) representation of that acknowledgement with a signature proof indicating that the message was from the original sender.

Identity proxies could also potentially add entire contracts by reading the message text and creating smart contracts on the users behalf, configured by the message text to represent the acknowledgement, but using private keys known to the proxy, and then posting on their behalf to (for instance) the ethereum block chain.

## Nodes

---

A node has:

- **i**: 1 .. n identities
- **B**: a balance sheet of acknowledgements received and given
- **R**: a local reserve of hard currency tokens
- **R'**: a balance sheet of trust lines (vested acknowledgements) creating a 'claimable reserve'
- **L**: a public history of active and cleared acknowledgements trust lines and any previous defaults etc.

[Sign in now to use ZenHub](#)

While a node can be pseudonymous it is assumed (at this point) that the balance sheets (B) and history (L) are publically available, by appropriate querying of the public record. Acknowledgements or trust lines can also be observed (by querying the appropriate block chain) and seen to verifiably have cryptographically enforced access to reserves or tokens on the public blockchain/s. This combination of information, combined with other information also available about public identities allows for reputation and (essentially) credit scores to be built up for nodes by other nodes (or by agents acting on behalf of those other nodes) querying the public record.

## Clearing and default

---

There are two ways of 'clearing' an acknowledgment. Either (A) a circle of reciprocity can be suggested and then agreed to by all parties in that chain or (B) the acknowledgement can be cleared by vesting of the claim on the hard currency token (creating a trust line).

### Clearance by vesting (or payment)

If an acknowledgement has a 'hard currency' token and a vesting date the sender is basically making a 'promise to pay' the amount of token specified at the vesting date or to find a circle of reciprocity before that date. If not they are considered to be in default.

An acknowledgement is considered to be 'algorithmically enforced' (aka 'backed') and 'in the money' if the receiver can verify that a contract exists which will guarantee payment of tokens from the nodes addressable reserves if any such reserves exist at time of vesting ( $v_t$ ). That is that the only condition under which payment would not be possible is if the claimable reserves at the node ( $R'$ ) are below the acknowledgement amount ( $x$ ) at that time. Vesting times can, of course, be very short for immediate payment.

If an acknowledgement is measured in 'hard currency' but not algorithmically enforced, or if it is not 'in the money', the node is still making a 'promise to pay' but there is less certainty that the node will be able to support the hard currency amount by the time of that vesting date and other nodes may act accordingly.

In the case of 'hard currency' promises third party app can verify or 'validate' that a contract is 'in the money' and may choose to stamp the message as being 'validated' in such a case.

[Sign in now to use ZenHub](#)

At the time of vesting the corresponding contract can be exercised so that the token is transferred into the direct control of the receiving node (expanding their direct reserves  $R$ ). Alternatively, the acknowledgement can be left un-exercised, creating a trust line (and expanding their indirectly claimable reserves  $R'$ ). Clearance by vesting is only possible with algorithmically enforced acknowledgements.

## Circles of reciprocity

Any type of acknowledgement can be cleared by the 'circles of reciprocity' method.

Specifically, if a loop of active acknowledgements is found, having roughly equal value, all of the relevant nodes in that loop can be offered the chance to 'write off' acknowledgements they have received in return for having acknowledgements given written off.

That is, if all parties agree the acknowledgements are 'cleared'. Any algorithmically enforced contracts will no longer have a potential claim on reserves once all parties in the chain have signed (this was built into the acknowledgement contract when it was first established of course). Even if not 'algorithmically enforced' the multi-party agreement to 'write off' all related acknowledgements (or partially write them off) is noted in the public ledger by appropriate multi-party signed messages in some observable public ledger.

It is expected that the mechanics of this would often be handled by an automated agent or app acting on behalf of the node, though this is not a requirement.

If an acknowledgement has a 'fictive' token (not backed by hard currency) and a vesting date, the sender is basically making a 'promise to resolve' by establishing a circle of reciprocity prior to the vesting date, or be considered in default.

Clearance by 'circle of reciprocity' can also happen with hard currency acknowledgements, however, whether backed algorithmically or not (or including a mix of the two).

In the case of clearance in this way the acknowledgement amounts ( $xT$ ) are used *only* as a 'measure of value' and not as a transfer of value. They help provide a guide as to what 'circles of reciprocity' are likely to be accepted by the parties involved. In this case the token acts, in the same way that 'roman coins' were used as a standard measure of value on debt contracts in medieval Europe long after roman coins themselves had fully passed out of circulation.

[Sign in now to use ZenHub](#)

An example of a 'fictive' tokens would be "hours". In this case the promise to pay is an offer from the giver to do x hours of work. Another forms of fictive token would be 'a bottle of red wine', 'bags of lemons'. A range of different purely arbitrary measures of value could be used (or invented) by people as methods of measurement. They would retain meaning and 'value' only in as much as they were cleared through reciprocity loops otherwise you risk the danger of issuance ultimately devaluing such fictive tokens. Such fictive tokens would have no 'global' value only a relative local value amongst nodes who can agree on loops.

Finally, it is not even necessary for there to be any acknowledgement amount (x) or token (T) at all. By examining the text of the messages alone, using NLP techniques or by hand, it would be possible to find loops which nodes (or agents acting on behalf of nodes) may consider likely to be acceptable for closure. For instance a 'saturday sports ride sharing' app might consider all messages acknowledging 'a ride' to be basically equivalent in value and act accordingly.

At this point the airship has fully detached from the building. Nevertheless the full range of acknowledgement types, and in particular the ability to have acknowledgements which act as 'hard currency' payments when dealing with strangers and simultaneously act as purely fictive loops of locally agreed value is perhaps the most compelling option to consider.

It is also possible that the acknowledgements may have no tokens on their public face but hard currency backed acknowledgement amounts selectively revealed to agents looking to find loops or only in the case of default. (This would be done for psychological reasons, as sometimes people prefer to be discreet about such things)

## Liquidity

With circles of reciprocity the idea is that nodes (or agents) would cross off what they consider to be 'roughly equal' acknowledgements.

[Sign in now to use ZenHub](#)

Nodes would have visibility on the entire chain of acknowledgements and nodes involved in such cases (admittedly with a variable degree of anonymity). In this way they could choose to veto the write off of chains incorporating that incorporate uses of their 'money' that don't align with their values. For instance, they may have their agent boycott any acknowledgement chains that pass through arms companies, or perhaps they may choose to allow or not allow anonymous parties to be involved. Or they may choose not to allow liquidity through nodes which they consider too highly leveraged. This would decrease liquidity for parties they do not support (and increase liquidity to those they do) even if they are distantly connected in economic terms.

To increase liquidity - especially in the case of 'unbacked' acknowledgements - it is not regarded as necessary that the acknowledgement amounts ( $xT$ ) on a circle of reciprocity be of exactly the same value, or of the same type (or indeed that there are any kind of token measure at all). There is liquidity in the fact that people generally believe in 'swings and roundabouts' - up to a point.

Alternatively, one of the advantages of 'hard currency' contracts, algorithmically enforced, is that unequal amounts can be partially cleared - up to the minimum flow across the loop. However this may potentially leave 'dust' transactions and create a psychological burden to people keeping track of their balance sheet with many partially cleared acknowledgements.

## Agents

---

There is quite a lot of psychological burden to reviewing and accepting 'circles of reciprocity'. For this reason it is expected that the usual way to do such clearances would be via an 'agent' or 'app' that can make judgements on behalf of the user. However the user would have the option to configure their app, review specific clearance requests or even switch to a different agent.

As a way to fund such agents or apps, they could attempt to find loops where all parties are happy to accept the trades on both sides, but there is overall a downward trend, thus allowing the app to insert itself into the chain and collect the difference. (Essentially this is a form of arbitrage, or alternatively the user of the app may feel that it is appropriate to pay for the app in this way.) The app acts a little like a proof of stake miner, essentially, charging (and sharing) transaction fees with other relevant agents every time they close a circle of reciprocity.

[Sign in now to use ZenHub](#)

To encourage reciprocity loops and to make things fun prizes could be awarded for 'longest chain', 'most geographically distributed', or 'largest value' or 'largest number' of loops closed in this way. Perhaps a leaderboard could be created.

As noted below, clearing acknowledgements by circles of reciprocity is rationally advantageous to nodes (and to be preferred over the alternative of default or clearance by payment). For this reason apps would play a role in notifying users of any opportunities to do this (see below section on notification and reputation). In fact this may be the primary function of apps - notification of opportunities to help or sell to others and posting of requests for goods or assistance. See below for more about this.

While there may be a shared substrate of messages, apps may also choose to focus on specific domains - such as (just as an example) the domain of parents acting as ride share taxis for kids getting to saturday sports. In this example the app may also keep track of things like which kids are in which teams, where the teams are playing and where the kids live. However, importantly, the app can sit alongside or on top of other existing transport mechanisms (such as texts and facebook messenger or whatever). The app is aiming not just to keep track of which kids need to go where, but also to attempt to close as many circles of reciprocity as possible. In this way a family who has given a lot of rids to other kids will be first on the list when they need to ask for a ride. (This is, of course, a trivial example but it demonstrates the basic idea.)

## Trust lines

---

Vested acknowledgements don't have to be exercised right away. This only applies in the case of 'backed' acknowledgements, but in such case an acknowledgement creates a 'trust line' as soon as it vests - essentially giving the recipient node a claim on the reserves of the giving node at any time after that time.

It may appear that it would make sense to immediately exercise any vested acknowledgements. In the case of nodes which you are unlikely to interact with in future - ie strangers - this is unarguably the rational choice. Furthermore, when used as a form of 'payment' to strangers outside the close support network - for instance if used to purchase something at a shop - this would be the expected behavior.

### Transitivity of trust lines

[Sign in now to use ZenHub](#)

When payment is made *outside* of the immediate trust network the expectation is that this would draw down reserves within the reachable trust graph ( $R'$ ) for that node. Trustlines should be transitive in the sense that the draw down can be made from a range of reserves within the wider reachable reserves ( $R'$ ) without necessarily exhausting the local reserves ( $R$ ) of the giving node first. (The exact details of the underlying smart contracts would need to reflect this transitivity property.)

## Rationality of trust line networks

In the case of others in your local 'trust' network (neighbors, friends or family) it could be a strictly rational choice to keep an unexercized trustline open. Doing this creates an incentive on the part of the giving node to ensure that the recipient (you) is not going to *need* to exercise that trust line in future. Leaving a trustline open is a little like a form of collective insurance or collective responsibility against unexpected events. This is similar to the way that within the 'gift economies' of the Tiv people in Africa (as described in [Debt: The first 500 years](#)) the expectation over the long term, is that neighbors will approximately reciprocate but at the same time the conscious choice is made never to quite pay back a gift exactly, thus symbolizing the ongoing existence of the relationship. Rationally speaking, however, a node needs to weigh this dynamic against the 'tragedy of the commons' dynamic where it makes rational sense to grab your share of the reserves from another node before another party does the same. Excessive amounts of open trust lines also creates the danger of a 'run on the banks' dynamic. Although in a more diffuse form than with more centralized banks this still exists on such a network, more so the extent that overlapping trustlines create one giant pool of reserves. Rational actors may therefore choose to exercise their trust lines simply so as to reduce leverage in their local network.

## Support for vulnerable people

However, even putting aside such (game theoretical) 'rational' choices one of the goals for this framework (stated above) was to allow games built on top of this framework to *allow* for the support of vulnerable people (who may or may not be in the kin group of the giver).

[Sign in now to use ZenHub](#)

One way this could be achieved is simply to extend a trust line directly to another node (or person) without that person actually doing anything that requires an acknowledgement first. In this way when the receiving node (the vulnerable person) themselves chooses to pay with an acknowledgement it potentially draws down from the reachable reserves of the original giving node. However, as with any such payment, provided that there is a reciprocity loop before the vesting date - for instance one involving other vulnerable persons - the reserves need not be drawn down at all. If a number of people extend a trust line to such persons then their reachable reserves are increased.

Ideally, games could be built whereby young people, old people, and generally people unable to maintain reciprocity in their own right should still be able to be supported and invited to pay back what they can without the dignity robbing feeling of having to borrow money or be poor. It depends on the design of the game at a UI level but ideally by somewhat blurring the line between 'exact matching' of values and trust lines etc, the interaction with 'money' and credit need not feel so cold, harsh and rigid and yet - critically - still retain the value of that money that makes it able to actually support people.

In the current design trust lines are irrevocable by design. The ability to revoke a trust line and how that might work is something that requires further thought if needed.

## Notifications and incentives

---

### Preference to de-leverage

Rationally, it is in the interest of nodes to preferentially de-leverage their acknowledgements balance sheet (B) ahead of paying down acknowledgements from their reserves (R).

This is because - assuming rational risk analysis by other nodes - reserves act as 'high powered money' (in the economics sense) and have a multiplier effect on the ability to expand their acknowledgements balance sheet.

As a result nodes are incentivized to attempt to 'close the loop' and reduce both the leverage of their overall balance sheet (B) and to try and keep the total balance of their balance sheet positive.

They can do this in two ways:

- They can let nodes in their near local network know about [Sign in now to use ZenHub](#)

each other thus potentially creating an acknowledgement loop which will ultimately link back into a 'incoming' acknowledgement already on their balance sheet.

- They can attempt to create new 'incoming' acknowledgements with nodes that will ultimately form a chain back to their outstanding 'outgoing' acknowledgements.

Essentially they are looking for opportunities to 'pay back' the benefit they have already received (and acknowledged) before it vests. In the same way, a bank wants to receive deposits before they have to pay down their liabilities from their bank reserves.

## Notifications

For this reason nodes will want to use apps or agents

- That can find and notify them, preferentially, of requests for assistance or requests to purchase goods from people who are able to help them build a chain back to the recipient/s of their existing outgoing acknowledgement/s.
- That can help them connect people in their local network to opportunities that will do the same. They might do this by 'vouching' for another party as to the ability of that party to do a service for another.

This is, essentially, a fairly well defined optimization problem for agents to solve. Where all relevant information about people - such as their skill set, their locality their friend network, their reputation etc can all be used to help make relevant recommendations (or notifications) as to who to engage with.

Specifically, to maximize the chances of a closed loop they probably want to engage preferentially with nodes that are: (a) close, in a network sense, to the nodes to which they have outstanding acknowledgement debts (b) that are likely themselves to engage with other nodes in a way likely to create further acknowledgements.

It is notable that such nodes do not necessarily need to be close in a geographic sense, but rather a network sense. Generally speaking, in a small town other people that are geographically close are also 'close' in a network sense. However this is not true in a large town. This is one reason that alternative currency / green dollar type schemes work better in small towns, where geographic boundaries are a good proxy for network proximity but hardly at all in big cities where no such correlation exists. However it is possible, by examining social network data, to measure of (real world) social network proximity.

[Sign in now to use ZenHub](#)

Another piece of data that would be used when trying to optimize such recommendations, is, of course to look at the number of successfully closed loops vs defaults in the history ( $L$ ) of another node. Essentially this creates a further reputation incentive for nodes to attempt to close their outstanding acknowledgements without default. To prevent sybil attacks it may be necessary to only allow reputation to flow only through nodes that the querying node has a direct outgoing connection to.

The exact mechanism by which requests for help are recommended or the way in which they are shared, and/or the mechanism by which nodes can 'vouch' for another are outside the scope of this document.

## Defaults and reputation

As noted above if an acknowledgement has not been cleared by its vesting date then it is considered a 'default' and removed from the list of 'active' acknowledgements.

Default need not be a major problem for the giving node in question nor is it such a binary thing. Failure to 'clear' an acknowledgement will undoubtedly decrease the reputation (in terms of willingness of other nodes to interact with that node), however it is also important for resilience of the system that acknowledgements with a very long vesting period, or that have not been cleared after a long period, may be considered basically equivalent to a 'default' for reputation purposes.

For long term resilience of the system it is important acknowledgements that cannot or will not be honoured are removed from consideration in reputation systems.

It would also be possible that acknowledgements dont have a vesting cliff per se, but a vesting schedule, In this way older acknowledgements would simply be discounted over time. Equivalently, for reputation purposes, older acknowledgements (such as those without a token or with a 'fictive' token) could simply become worth less and less over time and be less likely to cause recommendations to engage to appear in other people's notifications.

## Tokens

As already discussed an acknowledgement token can take many forms. Specifically tokens used in acknowledgements can be :

- 'Hard currency' algorithmically enforced by a contract ↗ [blockchain](#)
- 'Hard currency' but not enforced (not backed) where to [Sign in now to use ZenHub](#)

measure of value

- 'Fictive' (made up words) and not backed, but still potentially valuable in local usage
- Not present at all.

It is fully realized that it is not traditional to use the word 'hard currency' to describe crypto tokens. However in the context of this document it is hoped that the analogy of such tokens to 'hard currency' relative to bank created credit makes such terminology understandable. Crypto tokens also have the advantage of allowing algorithmic and immediate enforcement of the type of logic described above (especially around trust lines).

The exact choice of crypto currency is up to the next layer but it may be preferable to use 'stable coins' or a basket of currencies. If there were a form of traditional crypto token (traditional in the sense that value derives primarily from its restricted supply) that was at the same time truly decentralized (in the sense of no global ledger or an only 'eventually consistent' global ledger) then it would be more in keeping with this project to use that as the 'hard currency' store of value on top of which the kinpesa type network was built. Such crypto tokens may already exist. IOTA, for example looks promising but more research is needed at this point.

It is also, of course, possible to algorithmically enforce a contract based on, for example, a pre-authorized credit card or legal contracts. There is nothing strictly preventing kinpesa from being built 'on top' of such currency forms but this would introduce the fragility of having to trust a centralized third party to act as a proxy or potentially wait for humans to act on their legal obligations.

## Regarding decentralization and the competing interests of the system as a whole vs individuals

---

[Sign in now to use ZenHub](#)

As a goal, even though users can choose which 'app' they use and there is no central store of data, the intention is that the incentive structure is set up in such a way that the apps should not need to enforce 'good' or prevent 'negative' behaviors that the users of the app may wish to do. Specifically every app should act 'in the interests' of the user who installed that app, and not need to (or attempt to) enforce behavior which is good for the network per se, rather than the user. That is, there should be no reason why a user would want to install an app with 'cheat codes'. Potentially this is because other apps will prevent interaction with players or apps that attempt to subvert this paradigm by spam or other methods of attack but better than that would be to design the system so that no such 'cheat codes' exist. That, at least, is the goal.

As stated above, it was also the intention that the overall system should allow games to be built that are viral in nature but at the same time allow for the creation of tokens that are stable in value. (Sometimes these two things are in conflict).

## Zero circle/s

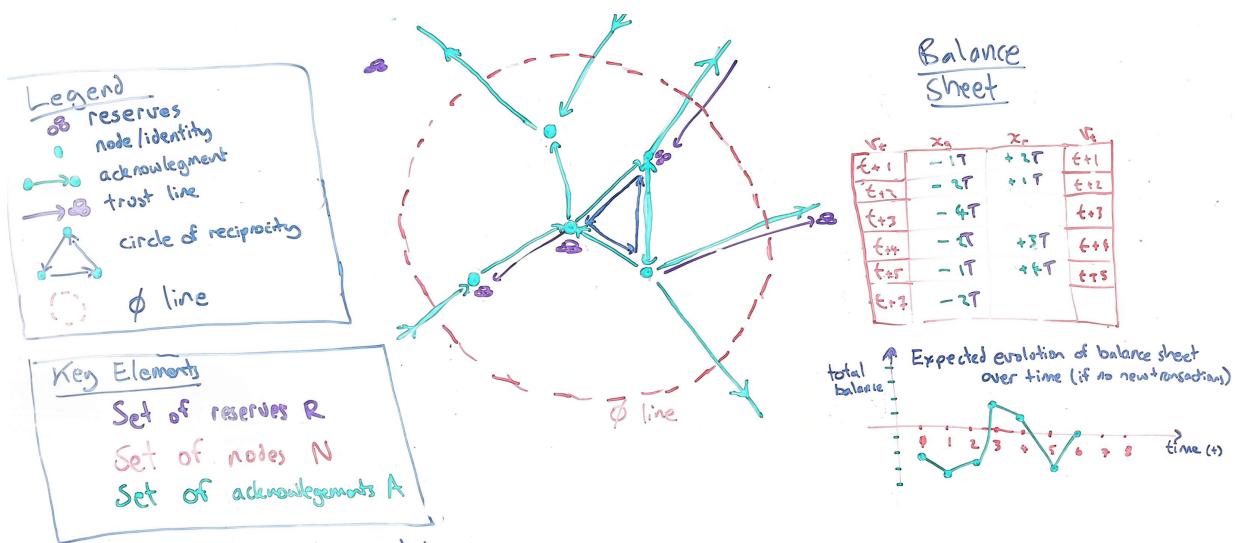
---

The claim is made that, for every node (provided all transfers of value are included in the network) there exists a network distance or rather a 'network cut' beyond which the total of acknowledgements plus reserves *transferred in* vs the total of acknowledgements plus reserves *transferred out* will not only tend towards zero but will stay near zero over time.

In the same way that gravitational acceleration is due to the total mass within a given volume and irrespective of the details of the distribution of mass within that volume the claim is made that there exists a certain boundary beyond which the details of the internal network are no longer relevant.

The reasoning behind this is that at a certain distance nodes will not consider other nodes to be in their local kinship network so that it makes sense for acknowledgements, or the overall balance sheet in hard currency terms, to zero out (or tend to zero out) across the boundary. If this is demonstrably true for all nodes it opens up opportunities for further network analysis.

[Sign in now to use ZenHub](#)



## Visualization and the 'hard problems'

As stated at the beginning the goal of this document is not to define an actual game or solution per se. Rather it is to create a framework within such games can be thought about. Even once such a game is defined there will still be 'hard problems' to resolve. For example one such hard problem would be the question of what is 'fair' vs 'selfish' in terms of behavior of nodes where such behaviors are allowed within the rules of the game but perhaps not felt to be in the spirit of that game.

In such cases the suggestion is made that we should be inspired by the creators of reddit (and twitter) who, in both cases, allowed their users to guide and/or solve hard problems, provide simple solutions by voting with their feet or by developing new 'manual' solutions and then build solutions around those behaviors.

In this domain the, two main tools by which users would be able to perform this function would be

- by adjusting the behaviors of their agents (particularly, when to accept a 'circle of reciprocity but also in other ways).
- by being able to look at visualizations of the state of the overall network, perhaps in somewhat anonymized form, so they can see what types of behavior are being developed.

For this reason, visualizations and user engagement are important parts of any game level solution - as always.

[Sign in now to use ZenHub](#)