
SAGA

TABLE OF CONTENTS

INTRODUCTION

Abstract

Digital decentralised tokens raise ever-growing interest in the promise they hold.
However considerable their potential, it remains to be realised in the form of true currency.

To unlock the vast monetary and social possibilities brought by blockchain technologies, a sea change is needed, a fundamental shift in our approach to the conception of money.

This challenge sits on a brutal truth: it's extremely hard to trust the new. Until digital currency is trusted it cannot be fully used for savings, trade or investment.

The Saga Foundation is a non-profit organisation that aims to create a reliable, global digital currency, by tying itself to established financial structures that already enjoy widespread trust.

Designing the Saga currency requires an interdisciplinary approach, combining best practice from financial services with the openness of blockchain. Saga is backed by a variable reserve, stored in established banks and algorithmically governed to minimise volatility. To ensure accountability, all participants undergo a formal KYC procedure.

The goal is simple: when the value of money is responsibly balanced, when governors and participants alike are held to fair account by a set of agreed, impartial guidelines, money will benefit from a new standard of trust and coherence.

Of Society & Money

Societies are an agreement formed by people sharing core values. A currency aims to provide an objective embodiment of these values, rendering them storable, exchangeable, and accounted for. As society's structures and values change, currency should keep in step. Currency enables individuals to exchange value both within and between societies.

The consolidation of national language, supported by the print revolution, led to the birth of the modern Nation-State, which remains the organising framework for governance since the industrial revolution. The resultant currency system serves us, nearly unchanged, to this day.

We have seen society change dramatically during the past few decades.
The forging of communities and identities no longer constrained by national boundaries - a natural effect of the internet and data era - demands answers to reflect these changes. The current system, should it remain stagnant, will be unable to meet the needs of increasingly global identity.

We face a challenge to reform the current system and regain the common ground.
Any new model must re-establish coherence between the power entrusted by participants and their fair representation in policy making.

The monetary contract, a fundamental pillar of society, is ripe for change.
Crafting a suitable monetary contract is a delicate task. Balance must be struck to preserve achievements, while introducing new governance methods.

Raison D'Etre

Saga aims to become a complementary global digital currency, designed to operate as a true means of exchange. Below we briefly explore these terms: 'global' in the context of globalisation; 'digital' from the perspective of governance; 'means of exchange' as an ambition for any successful currency - an important distinction from being a solely static store of hypothetical value.

Globalisation

While the nation state remains the primary organising principle of society, many aspects of identity have migrated into the global sphere. Information is now abundant and shared globally, and this powerfully affects our economic life.

When trading and consuming goods and services globally, state-governed fiscal policies often seem irrelevant, even burdensome. National currencies have not kept up with the information revolution.

Since 2009, with the invention of Bitcoin - and the blockchain as its underlying technology - several attempts to design new global currencies have raised widespread interest and have been supported by significant financial investments. The Saga currency does not aim to replace existing state-issued currencies, but rather complements them.

Digital Decency

Reducing Governance Bias

Existing fiat currencies were initially founded on sensible and coherent models; participants perceived themselves to be fairly represented.

Over time, as the environment changes, models are forced to to respond with short-term patches. These temporary fixes become harder and harder to maintain. Models become stressed and lose coherence.

Blockchain overcomes this structural challenge - to some degree.

The decentralised governance at the heart of blockchain ensures actions are 'kept honest' by participants: the design assures alignment of interests. In contrast, where governance is centralised, some participants acquire additional goals, inevitably to the detriment of other participants.

The current monetary system played a dramatic part in the progress gained during the last centuries. However, constrained by an outdated model, current state currencies no longer provide a fair distribution of risk and reward.

A less monolithic system may better serve society's needs, by expanding alternative models. Regaining participants' trust in their representation is key to achieve a stable, coherent new paradigm.

The information revolution and blockchain technology make this achievable. Saga presents an alternative solution that favours long-term sustainable growth, balancing resilience with efficiency.

Reducing Governance Cost

A currency is a contract that needs to be governed. The cost of governing a currency is represented by the cost of transacting.

To bear this cost, currencies are gathered and normalised, safely housed under a unified regime which requires specialised institutions. Inevitably, as with any normalisation, this results in a loss of value.

Transacting through state-issued currencies, spontaneous and straightforward exchange is often sacrificed in favour of an aggregated value mechanism. For example, instead of micro-transactions that remunerate a musician when fans listen to her music, various aggregators offer subscription or advertising models. Naturally, wherever there is an aggregator there are costs, borne by the public.

Technology - put to serve a social purpose - can lift barriers. Consider what printing did for the diffusion of knowledge, or what the internet achieved for the exchange of information. We believe blockchain is geared to advance the exchange of micro-transactions. The repercussions could be colossal. The transfer of governance from the monetary contract 'to the machine' allows for freer exchange of value, allowing reductions in transaction cost at several orders of magnitude.

Means of Exchange

One of the main challenges for a currency to become a means of exchange is its acceptability to both people and institutions.

While digital currencies gain ground as stores of value, their high volatility (relative to prominent state-issued currencies), prevents them from becoming a means of exchange. For this, a currency must target low volatility vis-a-vis prominent surrounding currencies. Saga therefore deploys a reserve anchoring algorithm, serving to stabilise the currency in terms of leading state-issued currencies. As Saga gains trust, its reserve ratio will decrease in favour of an independent establishment of value.

Identity is Accountability

Any social contract must be governed. The purpose of such governance is to enforce the rules that every contract participant undertakes to respect.

The decentralised traits of digital currencies encourage the governor's accountability towards its participants. However, participant anonymity prevents this same accountability towards others.

The identification of participants is key for a mutually accountable system.

Saga, like other digital currencies, offers no alternative jurisdiction, no replacement of the existing nation state. Saga aspires for all participants to be fully accountable, and therefore must respect existing institutions.

Identity must only be used to maintain the accountability of participants. For this to be the case, privacy must be paramount, and rigorously protected. The unveiling of identity must be restricted to legitimate purposes, as determined by participants' jurisdictions.

Saga implements a KYC (Know Your Customer) that deploys a multi-signature on-chain solution. Three signatures are issued: one to the participant; the second to their jurisdiction; the third remains in the hands of Saga. Two out of three signatures are required to reveal the identity of a participant.

How We Got Here

The Evolution of Money

Money, like other social constructs, constantly evolves alongside society. By adapting itself to changing environments, growing in sophistication and functionality, money becomes ever more useful, and eventually essential, for all societies.

Money & Economy

Money has three essential roles. It is:

- a medium of exchange
- a unit of account
- a store of value

Money as a means of payment makes trade possible and is key to almost all economic relationships and processes. Money widens the ability of humans to transact and trade, making economic relations convenient, easy to enact, safe, and open to evaluation. Without trade, each person (or household) could consume only what it individually produced. In such a primitive economy, transactions would be limited to barter, production would be inefficient, choices constricted by geography and happenstance.

As a unit of account and measure of value, money enables specialisation in production, which facilitates technological advancement. As a store of value, money is the essential and basic component of the financial system; it makes possible both saving and investment, key drivers for progress.

Alongside fiscal policy, monetary policy is a basic tool for policy makers to influence the economy; promoting growth, employment and general welfare.

Money & Society

Money plays an important role in any well-functioning society. It facilitates wide and extensive interaction between individuals and institutions. Money allows individuals to coalesce into larger groups, for economic or public purposes, and allows governments to act for the public good. It expands the individual's options over space and over time.

Historically, money has connected geographically distant areas and diverse cultures, facilitated interactions between different groups, and allowed not only the exchange of goods and services, but also the flow of ideas.

Initially local and private in nature, money issuance gradually acquired such an important social function that it became a monopoly of central government. It has both contributed to - and benefited from - progress over time. At all times, money's essential attribute was acceptability to its users.

While constantly developing, money's limitations and disadvantages remain.

Roles & Forms of Money

As barter expanded in extent and variety, certain frequently traded and widespread products assumed a larger, occasionally symbolic role. Convenient-to-transport, durable, easily valued and measured goods became widely used as proto-currencies.

Salt is a good example. Beginning as something with intrinsic value - metals such as silver or gold - money became increasingly more symbolic; seashells, tally sticks, and eventually coins and bills.

With no intrinsic value, modern fiat money is no different to these ancient proxies for value. Based solely on trust, the token of fiat currency is only as strong as the credibility of its issuer, usually a government or central bank. A Dollar, Euro or Dinar has no intrinsic value. Usefulness is determined by a currency's ability to pay debts or represent a credit. This simple role makes modern economies work.

Any individual can write an IOU – "I Owe You five Ducats". If a large enough number of people accept this IOU as payment, it becomes money. As the prominent economist Hyman Minsky commented, "Anyone can create money...the problem lies in getting it accepted."

Eventually, dematerialised versions of fiat currency became ubiquitous, as book-entry recordings of credits and debits eliminated the need to physically exchange coins or paper. Paper checks have been in use since at least the eighteenth century, and are widely expected to be eliminated within the next decade.

The recent growth of information technology accelerated this dematerialisation; currently the vast majority of money exists in electronic form only. In many ways the essence of money is a system of recording credits and debits which can be easily and safely transferred between owners. Money functions as such based on a broad consensus of credibility and acceptability.

Credit cards, electronic transfers, debit cards and mobile payment systems make the daily use of money both convenient and efficient. In essence, these modern tools rely on the same ancient symbolic elements. Ledger books, coins, banknotes, checks and credit cards are all simply tokens representing an underlying relationship recording stored and exchanged value.

THE MONETARY CONTRACT

Saga as Currency

Saga is a digital currency that operates on the blockchain, taking advantage of its benefits: reduced risk of governance bias; reduced governance cost; ease of transfer; ease of storage.

A Saga token (SGA) is an electronic currency unit that represents value. Saga tokens are backed by a reserve held in a regulated banking institution on behalf of Saga customers. The reserve is stored in assets against which Saga aims to stabilise; these assets are established, widely adopted currencies. The reserve comprises liquid low-risk assets in a mixture of fiat currencies, whose composition replicates the International Monetary Fund's SDR.

For a currency to function properly, it requires widespread acceptance. However this is difficult to achieve for a young, small currency. Therefore we designed Saga to develop in a healthy, measured manner that allows for sustainable growth. To achieve this goal, Saga operates a variable reserve ratio. This balances the dual aims of growth and stability, as the currency gains traction.

SGA are not pre-issued. Rather they are generated by the Saga contract, which is deployed on the Ethereum network. A participant that wishes to buy Saga tokens sends ETH to the contract. In return, the contract generates and returns an appropriate amount of SGA. The amount of SGA issued is determined by Saga's Price Discovery Model. Most ETH received by the contract is converted into fiat currencies according to the SDR makeup, while some is maintained as an ETH liquidity buffer.

Conversely, a participant selling SGA sends it back to the contract. ETH is extracted from the liquidity buffer to reimburse the seller, and the SGA is burned.

Price Discovery Model

The Saga contract acts as a market maker for SGA. The contract buys and sells SGA at different prices - the bid and ask prices respectively[1]. These are adjusted by Saga's Price Discovery Model according to the number of SGA in circulation. Prices rise and fall in tandem with the total number of tokens.

Within the range set by the contract's bid and ask prices, SGA's price is determined by market forces. However, when the price in secondary markets deviates from the Model's bid/ask range, the contract intercedes, altering the supply of SGA to restrain price volatility.

The primary aim of the Price Discovery Model is to allow Sagas' economy to grow, while ensuring acceptable price stability. The delicate balance between growth and stability is controlled by a variable varying reserve ratio.

1 Prices represented herein refer to the mid-point of the model's bid and ask prices.

Reserve Ratio

The prime feature of the Price Discovery Model is the reserve ratio, defined as the percentage of Saga's market cap backed by the reserve. The price of an SGA token derives from both its backing in the reserve and its inherent value[2]. The reserve ratio dictates the weight of these two sources of value when setting the price.

Whereas the value from reserve backing is stable, the inherent value is prone to volatility. With a low reserve ratio, the price heavily leans on SGA's inherent value, which, being more volatile, reduces price stability. On the other hand, a lower reserve ratio allows a faster increase in price. Thus, the choice of reserve ratio represents the balance between stability and growth.

A reserve ratio of 100% means full weight is given to the reserve backing. In this case, SGA price is fixed, and does not reflect changes in market trust. SGA at a reserve ratio of 100% is no more than a digital SDR currency board.

A reserve ratio of less than 100% means part of SGA price depends on its inherent value. As the economy grows and the inherent value of SGA rises, the price follows suit.

Three Phases of Growth

In an early Saga economy, SGA market cap is small and its inherent value deemed to be highly volatile. At this stage, stability should be emphasised over growth, so the model enforces a reserve ratio of 100% - a fixed price. In addition, this fixed price puts all early participants on an equal footing and prevents a 'run on the bank'.

Then, as the economy grows, the inherent value becomes less volatile, and the reserve ratio can be gradually decreased to support faster growth.

Finally, once SGA market cap reaches first-rank prominence, it will no longer make sense to benchmark SGA value against SDR. A new permanent system, to be determined as needed, will secure SGA as a standalone currency.

Key Benefits

A market-making function based on the above principles enjoys several benefits.

Primarily, by determining the reserve ratio, the Model controls the divergence of Saga's price from the value of its backing, thus curbing Saga's price volatility. This serves as a buffer against erratic and extreme volatility, which has hitherto plagued blockchain currencies.

A second benefit is asynchronous price discovery. Whoever wishes to trade SGA always enjoys the option to engage the contract directly. Minimal bid/ask spread is provided until the secondary market is well established.

Thirdly, the Price Discovery Model's inner workings are transparent and pre-coded into the contract. Governance is automated and resistant to tampering.

The transparency of Saga's Model could be a double-edged sword. Hostile parties can access open information to attack the reserve, aiming to drain its funds. To remedy this vulnerability, the Model maintains path independence, ensuring immunity from such attacks.

---

[2] *The inherent value is derived from a combination of factors: market confidence, usefulness as means of exchange, sentiment, and future prospects.*

Price Band

SGA price in secondary markets is free to fluctuate within the band formed by the Model's bid and ask prices.

Revenue from this bid/ask spread is deposited into Saga's reserve[3], elevating the SGA price. When volatility triggers the market-making mechanism, this additional revenue is funnelled into the reserve, effectively compensating long-term SGA holders for weathering periods of volatility.

In the early stages of Saga's economy, the price band is narrow: the contract acts as principal market maker, effectively determining SGA price. The width of the price band increases as Saga's economy grows, affording a greater role to the secondary market.

Interest Rate

The Saga reserve is held on behalf of SGA holders at major banks, where it accrues interest. If this is not reflected in the price, Saga's usefulness as a store of value would be compromised.

Accrued interest is deposited into the reserve, causing SGA price to increase. This acts as effective interest paid to SGA holders, a phenomenon that doesn't exist with other digital currencies.

The price increase is done continuously, in order to prevent manipulations - namely someone buying SGA just before a price increase and then immediately selling at a profit. This mechanism ensures Saga holders receive their fair value while reducing opportunities for arbitrage.

Saga Genesis

The resources to build Saga's core ecosystem are funded from proceeds raised by early supporters and investors, who support Saga prior to the launch of its economy.

They should naturally receive compensation commensurate with their risk. However, high prospects can only come with high volatility; ironically, the exact issue Saga seeks to reduce.

To reconcile this tension, early investors are recompensed not with SGA tokens, but with a different token called Saga Genesis (SGN), a voucher token convertible to SGA. The amount of SGA received per SGN token - henceforth termed the 'conversion ratio' - depends on the strength of Saga's economy: when small, the risk remains high and SGN is lower than SGA; as the economy grows, the price of SGN rises accordingly and surpasses SGA.

Saga's economic model is primed with a series of market cap milestones, known as 'Genesis Minting Points'. Whenever Saga's economy reaches a new Genesis Minting Point, SGA tokens are minted, to be allocated to SGN holders upon conversion to SGA.

- The first Genesis Minting Point occurs when Saga's economy hits a market cap of 25M SDR.
- The break-even point for early investors occurs when Saga's economy reaches a 610M SDR market cap.
- The SGN conversion ratio continues to increase until Saga's economy reaches a 1.5T SDR market cap. At this point the conversion ratio reaches its final cap: 1 SGN will be worth 15 SGA.

---

[3] after first being used to cover direct trading costs, mainly Ethereum transaction fees and bank commissions

The Mathematical Model

Technical Details: the Price Discovery Model

Start by denoting:

N - number of SGA units in circulation

P - current value of one unit of SGA

R - value of the main reserve

r - reserve ratio

C - SGN conversation ratio. The number of SGA an SGN holder will receive upon conversion.

i - total accrued interest rate

General Framework

The relation between Saga's price and reserve ratio is determined by two equations:

The first is $r(N) = \frac{R(N)}{N \cdot P(N)}$, which is simply the definition of the reserve ratio.

The second equation is: $R(N) = \int_0^N P(n)dn.$

This states that: the amount of money in the reserve, after N SGA tokens have been issued, is the aggregate of all the money deposited into it when issuing these tokens.

Combining these two equations gives:

$$P(N) = \frac{1}{N \cdot r(N)} \int_0^N P(n)dn \qquad (1)$$

We now explore two possible reserve ratio functions and their corresponding solutions for the price function, given by the above equation. Both options will be used in Saga's economy.

Option 1: Fixed Reserve Ratio[4]

Fix $r = \alpha$.

Solving equation (1) for P, gives $P = P_0 \cdot \left(\frac{N}{N_0}\right)^{\frac{1}{\alpha}-1}$

Option 2: Linearly Decreasing Reserve Ratio

Set $r = \alpha - \beta N$.

Solving equation (1) for P, gives $P = P_0 \cdot \left(\frac{N}{N_0}\right)^{\frac{1}{\alpha}-1} \left(\frac{r_0}{r}\right)^{\frac{1}{\alpha}+1}$

Here P_0, N_0 and r_0 are constants giving the price of SGA, number of SGA in circulation and reserve ratio respectively at a given point.
Note that Option 1 is a special case of Option 2, with $\beta = 0$.

---

[4] *This is the Bancor case and the corresponding solution is known as the Bancor formula. We would like to thank the Bancor team for fruitful discussions.*

Saga Genesis Conversion Mechanism

At each Genesis Minting milestone, new SGA tokens are minted and kept aside for SGN holders.

Consider the basic equation that the Saga economy must satisfy: $R(N) = r(N) \cdot N \cdot P(N)$ Minting new SGA tokens for SGN holders increases N, without any increase to R. Therefore, the product $r(N) \cdot P(N)$ must decrease.

We opt to keep $P(N)$ constant, so that SGA holders are not immediately disadvantaged by the minting of new tokens. Instead, the reserve ratio, r, is decreased: if n new tokens are minted, r decreases by a factor of $\frac{N}{N+n}$

The conversion ratio of SGN to SGA is given by:

$$C = \frac{\text{no. SGA minted for SGN holders - no. SGA already paid upon conversion}}{\text{no. SGN in circulation}}$$

Note that the total number of SGA minted for SGN holders depends only on the number of Genesis Minting Points that have so far occurred - i.e. the conversion ratio is a function of the high-water mark of the Saga economy.

This opens up a challenge: SGN holders may attempt to artificially increase Saga's high-water mark by buying and immediately selling SGA tokens against the Saga contract.

To protect against this, new SGA tokens are minted for SGN holders only once Saga's market cap has remained above the milestone for a period of seven days.

Price and Reserve Ratio Model

The Price Discovery Model splits Saga's economy into three distinct stages. At each stage a different choice of reserve ratio function is applied. From one stage to another, solutions are 'stitched' together, maintaining continuity in both the price and reserve ratio functions.

The Three Stages Are:

- **Small / Fragile Economy**
  In this stage, a fixed reserve ratio of $r = 100\,\%$ is chosen, in order to support the Saga economy when it is still weak. This means that price is kept constant, putting all early buyers of SGA on an even pegging. In addition, this avoids a 'run on the bank' scenario in a shrinking economy. This stage lasts until the market cap of Saga's economy reaches 20M SDR.

- **Growing economy**
  Here we support the growth of the Saga economy by slowly reducing the reserve ratio until it reaches $r = 10\,\%$ (at a market cap of 3T SDR). The reserve ratio reduction is a result of both linearly decreasing the reserve ratio (Option 2 above) and of minting SGA tokens for SGN holders.

- **Standalone economy**
  Once the Saga economy reaches this strong stage, there is no need to benchmark against the SDR. The reserve-ratio-based Price Discovery Model is abandoned and replaced by a new system, to be determined as needed.

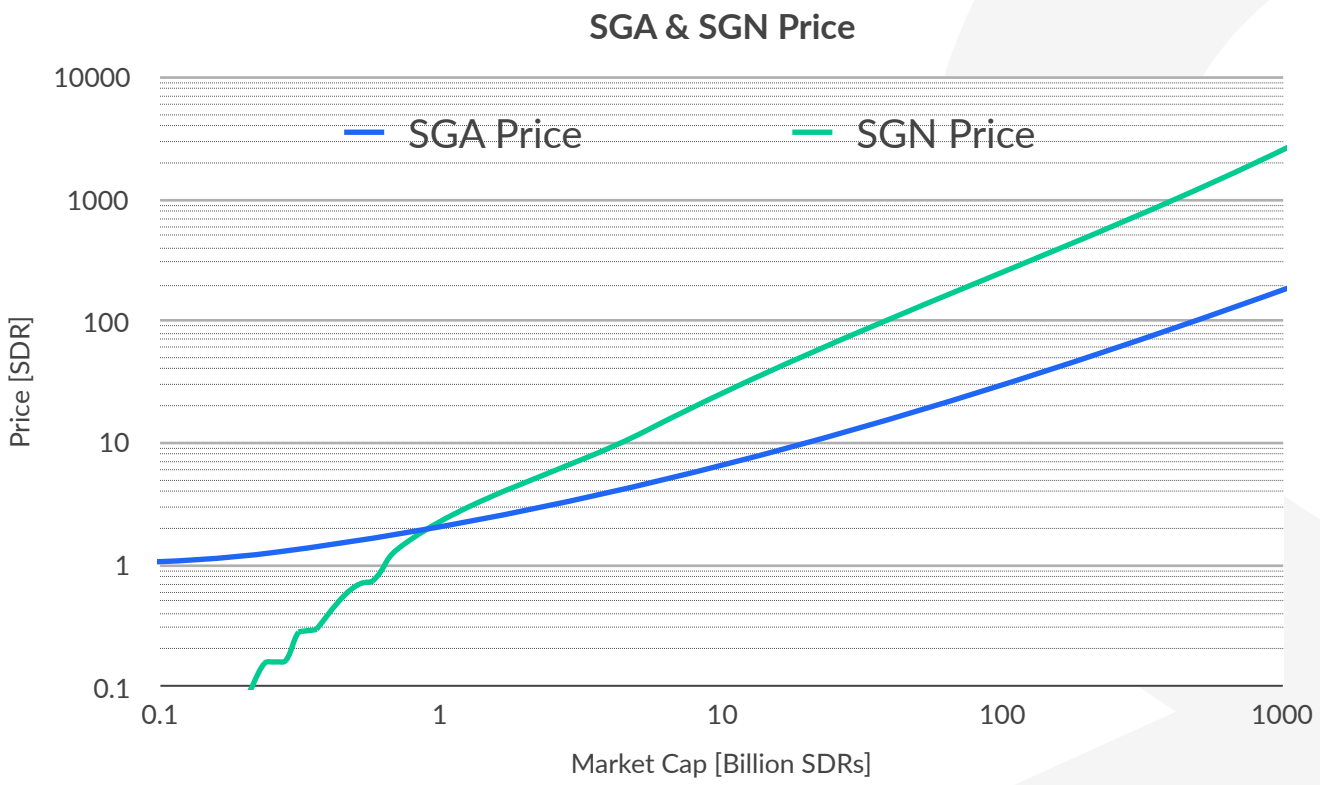
Price Band Width

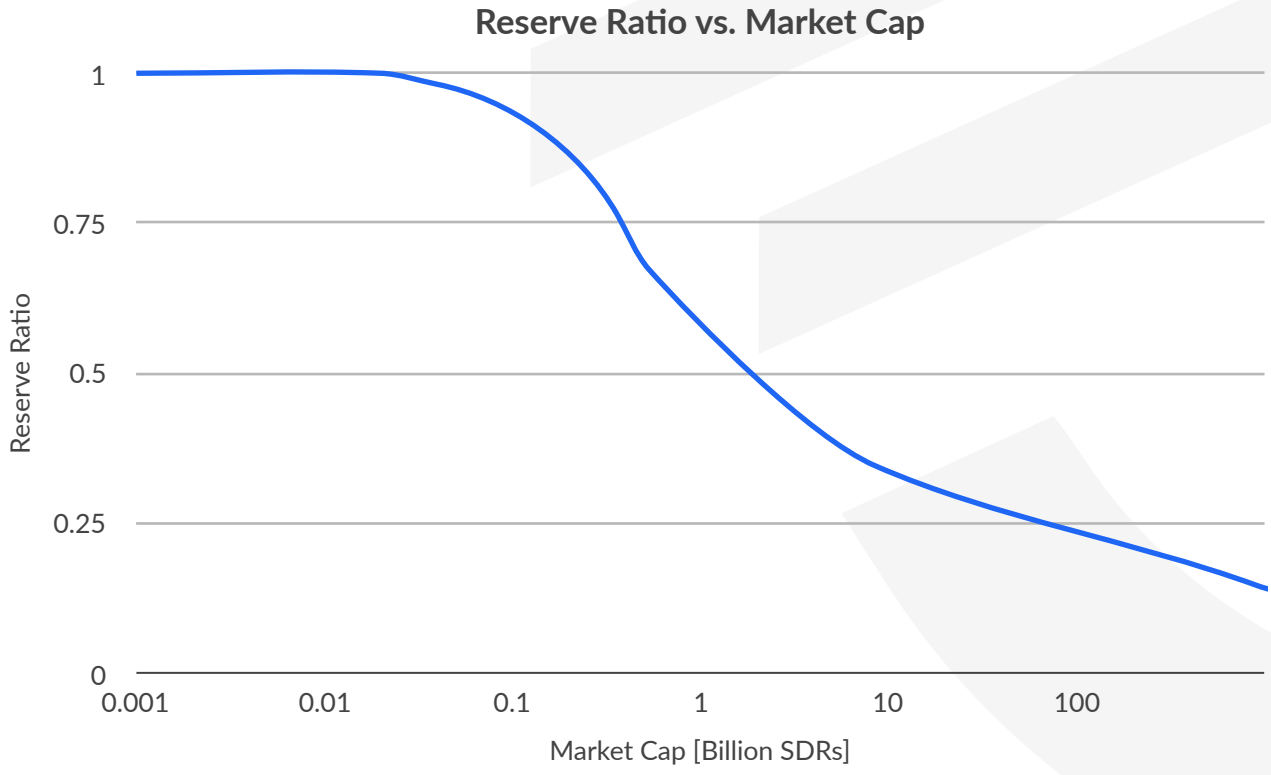
The width of the price band determines the extent to which SGA prices in the secondary market can deviate from the price set by The Model. The contract intervenes to restrain price movements whenever prices in the secondary market fall outside the limits of the band, in either direction.

The price band width increases as Saga's reserve ratio decreases; the band's maximum width of 15% is reached when the reserve ratio hits its 10% minimum.

However, until a secondary market is well established, the width of the price band remains fixed at 0.15%, covering only direct transaction costs.

Figures & Tables





SGN and SGA Prices

Market Cap	SGA Price	SGN Price	SGN to SGA Conversion Ratio	Reserve Ratio	Total Funds in Reserve
SDR	SDR	SDR	-	%	SDR
100 M	1.07	0.02	0.02	94%	94 M
200 M	1.20	0.08	0.06	87%	173 M
300 M	1.31	0.28	0.21	79%	235 M
400 M	1.44	0.46	0.32	73%	291 M
500 M	1.56	0.69	0.45	69%	342 M
600 M	1.65	0.99	0.60	66%	394 M
700 M	1.71	1.35	0.78	63%	441 M
1 B	1.95	2.37	1.2	58%	572 M
1.5 B	2.30	3.34	1.4	53%	787 M
2 B	2.68	4.39	1.6	50%	989 M
3 B	3.27	6.59	2.0	45%	1.4 B
4 B	3.80	9.20	2.4	42%	1.7 B
5 B	4.25	11.9	2.8	40%	2.0 B
10 B	6.32	25.3	4.0	34%	3.4 B
50 B	18.0	125.7	7.0	27%	13.4 B
100 B	29.3	260.2	8.9	23%	23.3 B
200 B	48.7	529.5	10.9	21%	41.7 B
500 B	99.9	1272	12.7	18%	90.2 B
1 T	180.4	2555	14.2	14%	142.5 B
2 T	334.6	5020	15.0	11%	227.4 B
3 T	487.3	7310	15.0	10%	300.0 B

USE CASES

Legend of Entities

Saga - The legal entity, a not-for-profit foundation registered in Zug, Switzerland.

SGA - Symbol for Saga's currency.

Alice - A participant in Saga's contract.

Main Contract - Smart contract that creates, burns, and transfers SGA.
Triggers two sub contracts: Reserve Manager and Liquidity Manager.

Sub-Contract: Reserve Manager - Instructs Reserve Traders.

Sub-Contract: Liquidity Manager - Decides the percentage of Saga's Reserve that is held in ETH as a liquidity buffer.

Reserve Traders - Receive arguments from the Reserve Manager. Instruct the Bank and/or Currency Exchange to perform exchange/deposit/withdrawal transactions.

Currency Exchange - External service provider offering digital currency conversions, e.g. ETH to fiat.

Saga KYC - A process used to identify a Saga user online, ensure conformity with regulatory directives, and cross-check credentials against wallet addresses.

Saga Wallet - Identified participant-held ERC 20 wallet; KYC approved and authorised to hold and/or trade SGA tokens.

Reserve Wallet - The Saga Contract's ERC 20 wallet. Receives, holds and transfers ETH for reserve anchoring.

Two Types of Transaction: Buy & Sell

BUY - A customer purchases SGA with ETH. Saga uses a currency exchange provider to convert some of the ETH to fiat, which is immediately sent to reserves held by regulated banks.

SELL - A customer sells SGA to the Main Contract, receiving ETH in return. Saga draws fiat from its reserves and uses a currency exchange to convert it to ETH.

Transaction Flow - BUY

- Description: Alice, a participant in Saga's contract, purchases SGA from the Main Contract. ETH is the base asset for the payment.

1. Alice creates a personal Saga Wallet.
   a. Alice first must undergo a KYC procedure.
   b. The KYC procedure verifies Alice's personal credentials, and then matches Alice's identity with Alice's wallet address.
   c. Alice's wallet is then authorised to hold or transfer SGA.

2. Alice deposits ETH into the Saga Wallet.

3. Alice sends a desired amount of ETH to the Main Contract.

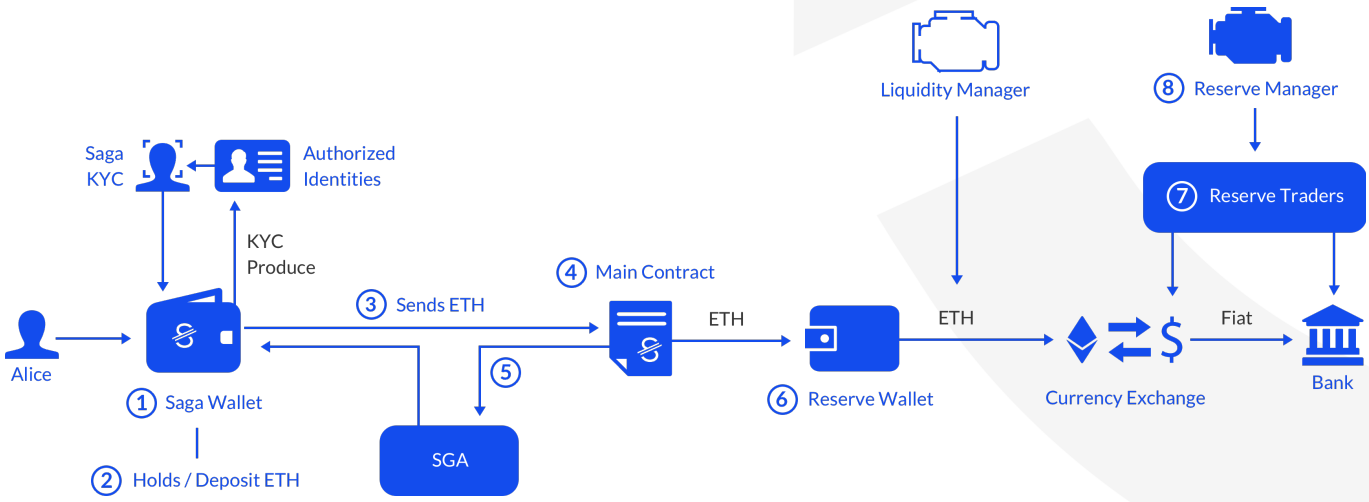
4. The Main Contract calculates and creates a determined amount of SGA.
   a. The amount of SGA created is dictated by the price of SGA/ETH, given by Saga's Price Discovery Model.
   b. The price of SGA/ETH is updated after each transaction block.

5. Main Contract performs two actions:
   a. Sends Alice the newly-created SGA.
   b. Directs ETH to the Reserve Wallet.

6. The Reserve Wallet receives ETH, then forwards the correct proportion to the Currency Exchange, for conversion into fiat.
   a. The amount sent for conversion to fiat is determined by the Liquidity Manager.

7. Reserve Traders instruct the Currency Exchange which assets to convert the ETH to, and also specify which banks to send it to.

8. All instructions from Reserve Traders are ultimately determined by the Reserve Manager.



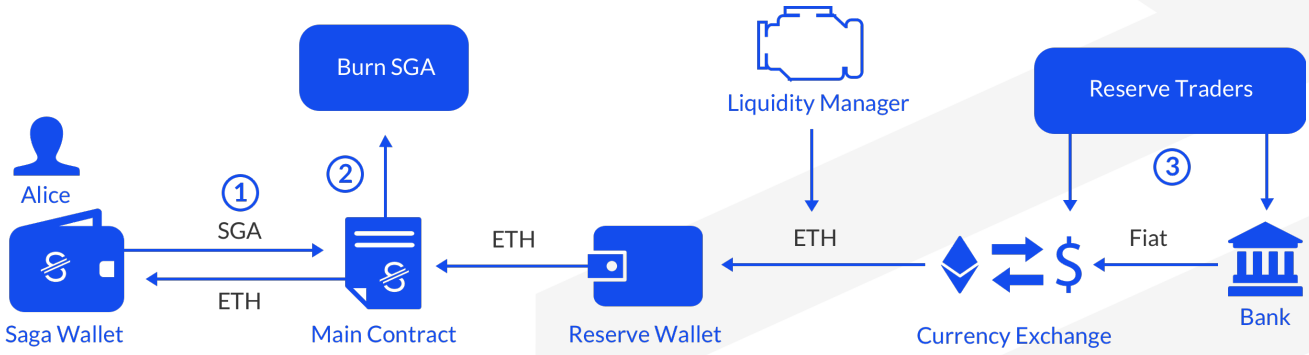
Transaction Flow - SELL

- Description: Alice, a participant in Saga's contract, sells SGA to the Main Contract and receives ETH.

1. Alice sends SGA to the Main Contract.
    a. Alice must have an authorised Saga Wallet in order to hold or send SGA.

2. Main Contract preforms three actions:
    a. Pulls ETH from the Reserve Wallet
    b. Sends the ETH to Alice. The amount of ETH sent to Alice is determined by the price of SGA/ETH at the precise moment of the transaction.
    c. Burns the SGA.

3. Reserve Traders instruct the bank to transfer a determined amount from the fiat Reserves to the Currency Exchange. Here they are converted to ETH and forwarded to the Reserve Wallet.
    a. The percentage of reserves sent is determined by the Liquidity Manager.



ARCHITECTURE

Overview

The purpose of this section is not to detail the specifics of any existing implementation. Rather the wish is to outline our general technical approach, acknowledging that there will be various challenges to the implementation of Saga's architecture.

However thorough our investigation has been, it is clear that obstacles and nascent solutions will result in alterations to what is presented below. We are happy and obliged to share changes as they develop, and will update our GitHub accordingly.

The Saga architecture is designed according to the following principles:

On Chain Implementation - Decentralised, Public and Immutable
Wherever possible, we implement our solutions on-chain. We run on Ethereum, part of the blockchain, which is transparent by design. This means governance integrity is preserved, regardless of the operator.

Centralised, Enforced Transparency:
To interface with traditional institutions, Saga requires a degree of centralised implementation. We ensure transactions are both transparent and immutable. Our discovery tools allow participants to monitor system transactions, allowing participants to assess the integrity of governance.

Contained Blind Trust
While current implementation does not require this layer, should Saga ever need to operate solutions that are prone to 'game-ability', we will obfuscate them for participants' common good. When such a solution is introduced, the amount of blind trust required will be contained by the contract.

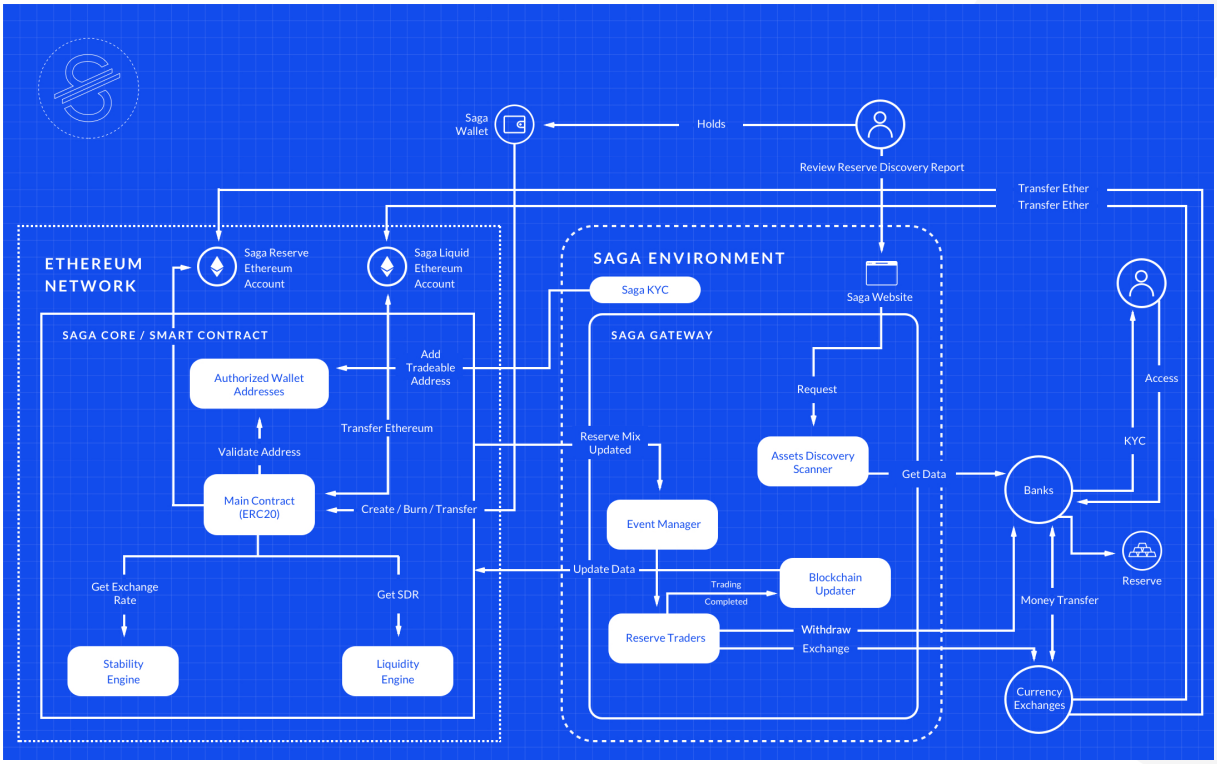
Design Considerations:

Transparency
While on-chain implementation offers out-of-the-box transparency, centralised solutions are not transparent by design. We have taken upon ourselves to explicitly implement a discovery service providing a similar level of transparency off-chain, keeping the ledger public at all times.

High Availability
Saga is a mission critical ecosystem. Accordingly, any downtime due to stress, maintenance, or updates, is to be avoided. High availability patterns are favoured to ensure SLA and information redundancy.

Scalability
As the Saga economy expects rapid growth, both in number of participants and transaction volume, the system relies upon distributed architecture and corresponding technology stacks. This avoids the formation of bottlenecks and allows the preservation of system responsiveness during spikes.

High-Level Architecture



Saga's On Chain Implementation

Saga has chosen the Ethereum network to host its smart contract, developed to manage the Saga monetary contract.

Why Ethereum?
Ethereum is the first platform to allow a Turing-complete implementation of smart contracts over the blockchain. As part of a young industry, it is clear that Ethereum faces scalability challenges. We expect it to overcome these and to emerge as a prominent ecosystem. The ERC20 token standard, which we use to facilitate the transfer of SGA, is becoming increasingly mainstream. Moreover, the relative liquidity of ETH as an underlying asset can facilitate the ease of SGA acquisition. Hopefully Ethereum will prove a robust framework for Saga's nascent currency.

Programming language over the blockchain
Saga's On Chain programming language is Solidity, the primary language on Ethereum. Solidity is a statically-typed OOP programming language.

Saga Core: On Chain Components

1. **Main Contract**
2. **Authorised Wallet Address Manager**
3. **Authorised Wallet Address Data Source**
4. **Stability Engine**
5. **Liquidity Engine**

1. Main Contract

Role
Creating / burning of SGA ERC20 tokens. Main Contract receives an asset (ETH, SGA) and calculates the amount of return assets according to SGA price.

Contract Interface
Implements ERC20. Saga's ERC20 implementation also considers KYC constraints. For example, transfer between addresses requires KYC compliance on the receiving side.
buy - accepts ETH and returns SGA, based on the price of SGA returned from the Stability Engine. To complete the operation, Main Contract calls Authorised Wallet Address to check whether the address is authorised.
sell - accepts SGA and returns ETH, based on the price of SGA returned from the Stability Engine. SGAs are burned with this action.

2. Authorised Wallet Address Manager

Role:
Determines the eligibility of a wallet address to trade SGA.

Contract Interface:
The contract implements IAuthorizedWalletAddressManager interface:
isAuthorized - Receives a wallet address as a parameter and returns true or false based on the AuthorizedWalletAddressDataSource list property.

3. Authorised Wallet Address Data Source

Role:
Contains a list of wallet addresses permitted to trade and hold SGA.
A consumer that wishes to hold SGA must undergo a KYC process. At the conclusion of the identification process, the user's wallet address is added to the list of authorised addresses.

Contract Interface:
The contract implements IAuthorizedWalletAddressDataSource interface:
add - adds a tradable wallet address.
contains - returns a boolean detailing whether an address is contained within the Authorised Wallet Addresses list.
remove - removes a wallet address from the Authorised Wallet Addresses list.

4. Stability Engine

Role:
Implements the Price Calculation Model.

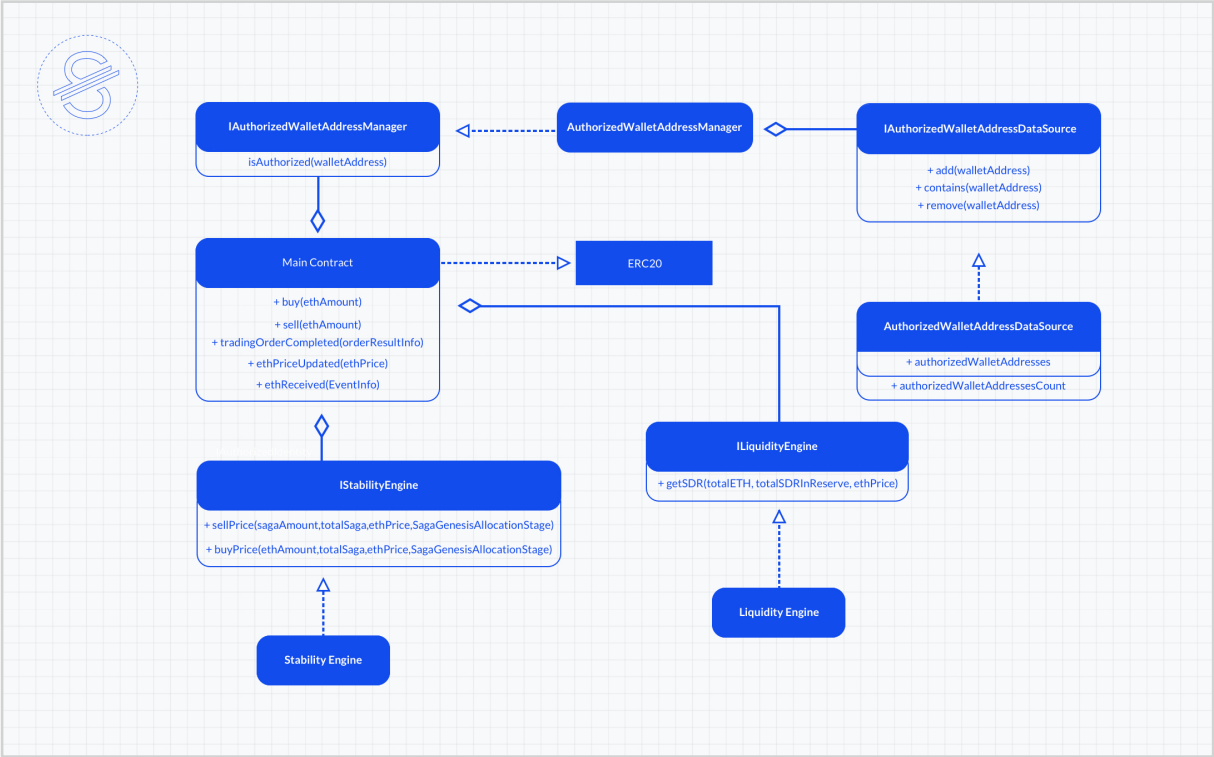
Contract Interface:
The contract implements IStabilityEngine interface:
buyPrice - how many SGA do I get for a given number of ETH?
sellPrice - how many ETH do I get for a given number of SGA?

5. Liquidity Engine

Role:
Implements the SDR calculation model.

Contract Interface:
The contract implements ILiquidityEngine interface:
getSDR - calculates the amount of ETH to transfer to/from the SDR reserve.

Initial On Chain Class Diagram



Saga Gateway: Off Chain Implementation

Main Design Patterns

Service-oriented-architecture: Non-monolithic system that enables each service to be updated with low cost of change. Permits fine-tuning for each service according to scalability and technology demands.

Event-driven-architecture: Services are decoupled and work in asynchronous mode. Services can be updated independently from each other; in case of a service failure, other services are not affected.

Programming Language

Saga's programming language for the Off Chain world is Scala, which enjoys the best of both functional programming and OOP. Scala allows us to build systems with asynchronous messaging and multithreaded development. JVM-based and compatible with Java libraries. Scala is our team's area of expertise.

Off Chain Development Stack

- Apache Storm: Real-time distribution system used to handle processing of large amounts of data. Integrates well with Apache Kafka.
- Apache Kafka Distributed Messaging System - open source distributed messaging system. Supports high throughput, includes fault tolerance for handling real-time data feeds, and offers the ability to reach horizontal scalability.
- Akka Http: REST services are built using Akka Http modules, which implement full server-client HTTP stack on top of akka-actor and akka-stream.

- Docker: Software container platform. Allows code collaboration between developers. Saga's REST and Akka-based services are dockerised and managed by container orchestration. We will perform POCs for different container orchestration systems, such as Kubernetes and Docker Swarm.

There are 7 Services in the Off Chain Solution:

1. Event Tracker
2. Blockchain Updater
3. Blockchain Event Listener
4. Trading
5. Saga Price Discovery
6. Reserve Discovery
7. Ether Pricing Updater

How does Saga connect with the blockchain?
To maintain a connection with the blockchain, aspects of our system require bespoke technology.

Saga will perform POCs to determine which technologies best suit our needs. We tend towards Ethereum because of its compatibility with Scala - a derivative of its 'pure-Java support'.

Event Tracker

Role
Some of the events passed between environments require a handshake. The event tracker is responsible for listening to off-chain events, and sending an acknowledgment to the on-chain.

Input
Events that require handshaking. Event Tracker locates them by listening to different Apache Kafka topics.

Processing
When an event occurs, Event Tracker creates a corresponding acknowledgment message to complete the handshake.

Output
Sends acknowledgment message to an Apache Kafka topic that the Blockchain Updater listens to.

Stack
Event Tracker service based on Akka stream, using Reactive Kafka (an Akka Streams connector for Apache Kafka).

Blockchain Updater

Role
Acts as a gateway between the Off Chain and On Chain environments. Listens to off chain events that require passing on to on chain contracts.

Input
Blockchain Updater. Listens to events from Apache Kafka that require updates to be made to On Chain contracts.

Processing
According to a predefined logic, **Blockchain Updater** updates the relevant On Chain contract, based on the event and its contained data.

Output
Relevant update to On Chain Contract.

Stack
Blockchain Updater service based on Akka streams, using Reactive Kafka (an Akka Streams connector for Apache Kafka).

Blockchain Event Listener Service

Role
Monitors events arriving from the blockchain; extracts particular data from the event and sends data to Trading Service.

Input
An event raised from the Saga contract on the blockchain.

Processing
The event undergoes data extraction to interpret actionable instructions.
A new event is created based on data from the received event.

Output
Sends the newly created event to Apache Kafka topic.

Stack
Blockchain Event Listener Service based on Akka streams, using Reactive Kafka (an Akka Streams connector for Apache Kafka).

Trading Service

Role
Translates the desired reserve size (given in SDR units) to actual buy / sell orders converting ETH to and from fiat currencies.

Input
Desired reserve size from the On Chain.

Processing
Sends multiple requests in parallel to each provider, asking for account balance and exchange rates.

Once information has been received from each provider, the data is processed with predefined rules and a set of actions for each provider is generated.

Each provider executes the actions and returns a response with:

- Did Operation Succeed? - True/False
- Exchange rate used to fulfil the order
- New account balance in each fiat currency.

All results from providers are aggregated into a single event that shows:

- Total reserve amount in SDR
- Order ID as originated On Chain

Output
The event is sent to the On Chain contract through the Blockchain Updater.

Provider: a financial institution which physically holds Saga's reserve and executes instructions.

We have yet to determine the financial institutions that will serve us as providers, and therefore cannot predict the process flow, requirements, APIs etc. However we do know the interface each provider would need to implement in order to engage with Saga.

Saga Price Discovery

Role
Exposes an API for simulating Saga sell and buy values.

Stack
Implemented as a REST service built by using Akka Http modules.

Reserve Discovery

Role
Serves as Saga's transparency policy, by reading the current reserve mixture directly from the Financial Provider, and making data publicly available.

Current implementation is based on the assumption that Saga will employ a small number of providers, extracting data in real time. The service uses cache mechanisms as needed.

As Saga is yet to determine its providers, the optimal method for handling information requests remains unknown. We assume read-only access to data via REST API.

Stack
Implemented as a REST service built by using Akka Http modules.

Ether Price Checker

Role
Monitors ETH price at every given moment, and alerts the On Chain.

Input
Receives a real-time stream of ETH price from the relevant Kafka topic, via a dedicated data provider.

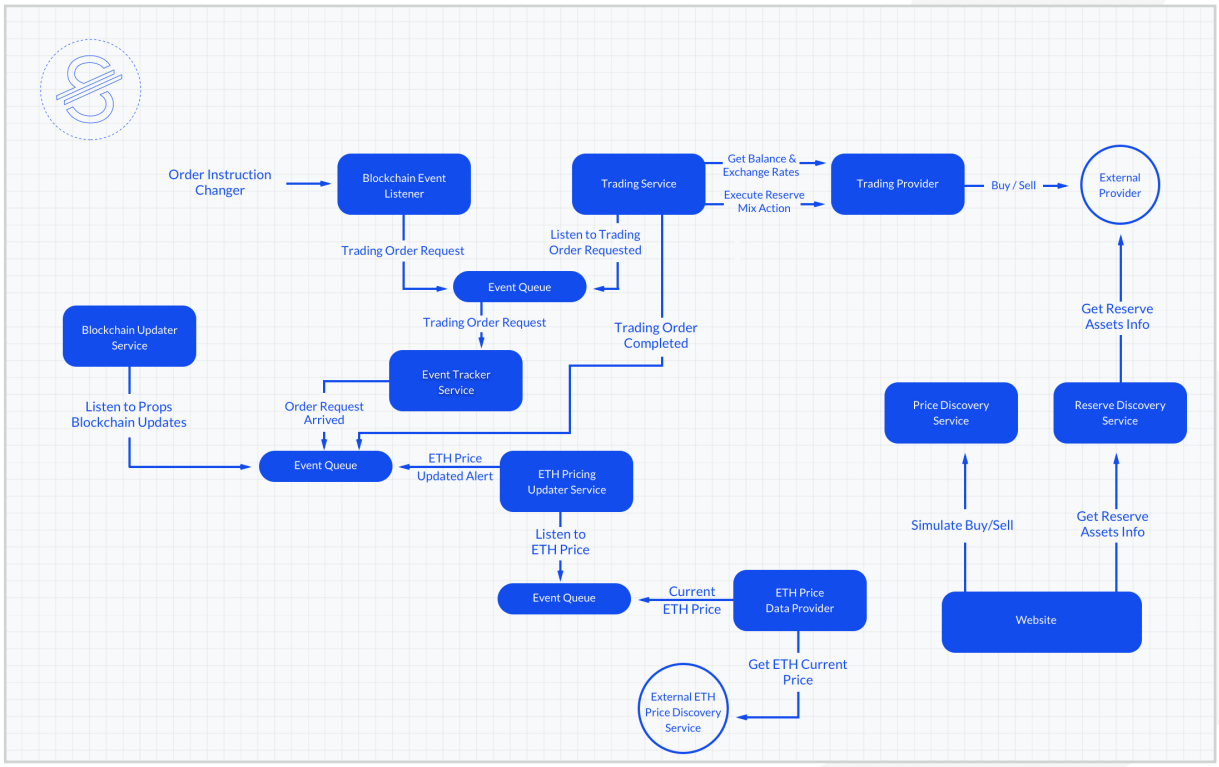
Processing
Data analysed based on a sliding window over a predefined time interval. Based on an algorithm which contains different parameters that determine if a price change is outstanding.

Output
In the event that an ETH price change requires attention, a new event 'ETH price alert' is created and sent to the relevant Apache Kafka topic. Blockchain Updater listens out for this topic.

Stack

Apache Storm topology listens constantly to Apache Kafka ETH price events and handles them as they are created.

High-Level Off Chain Architecture:



FOUNDATION

Purpose

Saga is a non-profit Foundation obliged to work under an arm's length principle: it exists merely to serves its publicly-accountable purpose.

The Foundation's purpose is to promote and develop new technologies and applications, especially in the fields of new open and decentralised software architectures. An important - but not exclusive - focus is promotion of the so-called Saga protocol and its related technologies.

- The Foundation is not for profit.
- The Foundation may be active both inside of Switzerland and abroad.

Structure

The Foundation is established under the name 'Saga Foundation', being a foundation in the sense of Article 80 et seqq.

Swiss Civil Code (the "CC") with its legal seat in Zug, Switzerland.

The Foundation chose Switzerland in order to enjoy the country's attractive regulatory, political and socio-economic environment.

The Foundation will be audited and supervised by the Swiss supervisory authority.

For day-to-day operations, The Saga Foundation engages suppliers to build and implement Saga's core.

Composite bodies of the Foundation are:

- The Foundation Council
- The Statutory Auditor (unless the Foundation has been released from its obligation to appoint a statutory auditor by the supervisory authority).

Foundation Council

The Foundation Council consists of five persons who will, in principle, work on a voluntary basis.

The Foundation Council will be composed in such a way that responsible fulfilment of the Foundation's purpose is ensured. If possible, a majority of the members of the Foundation Council will be professionally experienced, with specialised knowledge in the area of new, open and decentralised technologies and / or corporate governance.

The Foundation Council will constitute itself and adopt a special resolution concerning representation and signatory rights.

The Foundation Council will manage the business and assets of the Foundation.
This Council represents the Foundation at external forums.

The Foundation does not have a profit-making purpose and does not seek any profits.
Within its purpose, the Foundation may be active both in Switzerland and abroad.
The founders expressly reserve the right to change the Foundation's purpose in accordance with Article 86a CC.

In support of its purpose, the Foundation and the Foundation Council can:

- Support and finance appropriate research and development activities and projects
- Promote and educate the public concerning the Saga protocol
- Issue, receive, spend, and hold digital information units (no speculative trading)
- Engage various businesses, partners, banks, regulators, authorities and other third parties to the benefit of the Saga ecosystem
- Distribute received contributions and engage private companies
- Organise conferences and other events that support and promote the Saga Protocol
- Conduct and promote all business and enter into all transactions, and generally perform acts deemed necessary, appropriate, incidental to, or desirable to, assist in promoting the objects described herein

Foundation Council

Ido Sadeh Man, Founder & Foundation Council President

Mr. Ido Sadeh Man is Co-Founder & Partner of The Singulariteam Technology Group, Israel's most active VC in Israel in 2015 & 2016, with recent successful exits totalling $250M.

Mr. Sadeh Man spent the last 10 years leading product and technology organisations, including Odysii (sold to Gilbarco Veeder-Root, NYSE: FTV), and Mobli, where he transitioned from VP of R&D to COO, after successfully completing the company's Series C investment round led by Carlos Slim-owned, America Movil.

Mark Tluszcz

Mr. Tluszcz is Co-Founder and CEO of Mangrove Capital Partners and Chairman of Wix (NASDAQ: WIX). Mark's notable successes include being the first investor in Skype, which was sold in 2005 to eBay for $3 billion and Wix, currently trading on the NASDAQ for near $3.5 billion. His firm has invested in over 100 companies and manages near $1 billion across 5 funds.

Barry Topf

Mr. Topf joined Saga after a 33-year career at the Bank of Israel, where he served as one of the founding members of the Monetary Policy Committee and as Senior Advisor to the Governor, Stanley Fischer. He also held positions of Head of Market Operations, Head of the Foreign Currency Department, and Chief Investment Officer. In his capacity as an IMF Consultant, Mr. Topf has advised over 25 countries on economic policy.

Moshe Hogeg, Co-Founder

Mr. Moshe Hogeg is Co-Founder & Chairman of The Singulariteam Technology Group, and is considered to be one of the most fruitful serial-entrepreneurs in Israel. Under Mr. Hogeg's management, Singulariteam became Israel's most active VC in Israel in 2015 & 2016, with recent successful exits totalling $250M.

Today Mr. Hogeg is Israel's most active investor in blockchain ventures, with Bancor and Kik among his investments. Most recently, Mr. Hogeg co-founded and is the active Chairman of Alignment, a blockchain hub that consults, develops and funds early-stage blockchain projects and existing companies, from inception through ICO, and beyond.

Guido Schmitz-Krummacher

Mr. Schmitz-Krummacher is an executive with 25+ years experience in Board and CEO roles throughout the globe. He joined the crypto-industry two years ago and focuses on representing its emerging ecosystem. Mr. Schmitz-Krummacher is a German lawyer specialising in international and corporate law and a trained Swiss board member.

Advisory Council

Senior Advisors

Jacob A. Frenkel, PhD

Dr. Jacob A. Frenkel, is a Laureate of the Israel Prize in Economics. He is Chairman of JPMorgan Chase International and Chairman of the Board of Trustees of the Group of Thirty (G30). Previously, he served as the Governor of the Bank of Israel, as the Economic Counsellor and Director of Research of the IMF, and as the David Rockefeller Professor of International Economics at the University of Chicago.

Prof. Myron Scholes

Prof. Scholes is a Nobel Laureate in Economic Sciences for his new method of determining the value of derivatives. He is the Frank E. Buck Professor of Finance, Emeritus, at the Stanford Graduate School of Business, and co-originator of the Black-Scholes options pricing model. Scholes is currently the Chief Investment Strategist at Janus Henderson Investors.

Leo Melamed

Mr. Melamed is the founder of financial futures. He is the former Chairman of the CME, where he created and served as Chairman of the IMM. Melamed led the creation of the CME FX, interest rate, and stock index futures contracts. He also led the creation of Globex, the CME's electronic transaction system. He is the founder and permanent advisor to the National Futures Association and an advisor to the International Advisory Council of the CSRC in China.

Prof. Dan Galai

Prof. Galai is the co-developer of the Chicago Board Options Exchange's Volatility Index (VIX). He served as Consultant to the Bank of Israel as well as to various other financial institutions. He is an international expert in the area of options, financial engineering, and risk management. Prof. Galai is also the former Dean of the School of Management at Hebrew University, Jerusalem.

Advisors

Prof. Emin Gün Sirer

Prof. Sirer is a professor at Cornell University. His research interests centre around blockchains and cryptocurrencies. He is known for characterising the security of the Nakamoto Consensus protocol and for exploring the scalability limits of cryptocurrencies. He has also worked on protocol modifications to decentralise blockchains and devised functionality improvements to avoid thefts.

Prof. Raz Chen-Morris

Prof. Raz Chen-Morris is a Professor of History at the Hebrew University of Jerusalem. He holds an MA in the history of medieval and Renaissance science, and a PhD from Tel Aviv University.

Liraz Lasry, PhD

Dr. Liraz Lasry holds a PhD in Business with a specialisation in Marketing and Branding. She is an academic manager and lecturer in the MBA and executive programs at Tel-Aviv University and at New York University, focusing on Brand Management, Trends and Marketing strategy. She is also an experienced personal advisor to entrepreneurs, investors and executives.

Matan Field, PhD

Dr. Matan Field holds a PhD in Physics. A blockchain researcher and entrepreneur, he founded Backfeed to develop the protocol basis for decentralised autonomous organisations (DAO), a work in progress at the forefront of the blockchain revolution.

Eyal Hertzog

Eyal is a venture-backed technology entrepreneur with over 20 years of experience. Eyal has been working in the cryptocurrency space since 2014, at AppCoin, empowering local communities with local currencies; and now at Bancor Protocol, an on-chain, fully decentralised conversion solution.

Zeev Suraski

Zeev Suraski is one of the principal designers of the PHP programming language - which powers over 40% of the Web - and has been involved in its development and proliferation for almost 20 years. He is an emeritus member of the Apache Software Foundation, and the co-founder & CTO of Zend Technologies. Zeev holds a degree in Computer Science from the Technion, Israel Institute of Technology.

Saga Core Team

Keren Orian Nadel, Managing Director

With over 15 years' worth of experience in strategy, product, marketing, operations and P&L management, Keren has held global senior management positions in both corporates environments (Microsoft, Haaretz Media) as well as startups. Keren holds a BA in Political Science and an MA in Public Policy from Tel-Aviv University.

Barry Topf, Chief Economist

Mr. Topf joined Saga after a 33-year career at the Bank of Israel, where he served as one of the founding members of the Monetary Policy Committee and as Senior Advisor to the Governor, Stanley Fischer. He also held positions of Head of Market Operations, Head of the Foreign Currency Department, and Chief Investment Officer. In his capacity as an IMF Consultant, Mr. Topf has advised over 25 countries on economic policy.

Yaron Shalem, CFO

With 15 years' experience as a CFO of investment, industrial and high-tech companies, Yaron has been at the helm of several publicly-traded companies on both the TASE and NASDAQ. Yaron is a CPA and holds a BA in Economics & Accounting from Tel Aviv University, and an Executive MBA from Bar Ilan University.

Roy Eshkol , CTO & Blockchain Architect

With over a decade of experience in technology and infrastructure, Roy leads Saga's architecture design and development. He holds an MSc in the Management of Technology and Information Systems.

Ron Sabo, PhD, Head of Research

With a PhD in experimental condensed matter physics from Weizmann Institute of Science, Ron leads Saga's Research department. A Clore Scholar, he also lectures at the international program in Electrical & Electronics Engineering at Tel Aviv University.

Itzik Adler, Chief Product Officer

With over 10 years' experience in Product Development, Itzik's skills lie in transforming complex ideas to fully functional outputs. He holds a BSc in Computer Science from Ben-Gurion University.

Roy Epelshtein, Blockchain Engineer

A senior developer with 6+ years' experience in developing distributed, high-availability systems from scratch, Roy implements Saga's blockchain design.

Barak Manos, Blockchain Engineer

With over 15 years of experience in software engineering, Barak focuses on implementing Saga's financial model under the computational constraints of the Blockchain infrastructure. Passionate with software, and with a particular affinity for mathematical problem-solving, Barak holds a BSc in Computer Science from the University of Haifa.

Yoni Berger, Researcher

A graduate of Trinity College, Yoni holds a Master's degree in Mathematics from Cambridge University. At Saga, Yoni converts economic models to mathematical algorithms.

Vlad Bachynskyi, DevOps Engineer

With 10+ years of experience in systems engineering and networks, Vlad leads Saga's DevOps where he employs his expertise in developing and deploying IT infrastructures from scratch to production.

Katia Dubnik, Delivery Manager

Executing CI/CD processes for the past 9 years, Katia's expertise is delivering software on time, while maintaining agility and high-quality execution. At Saga she liaises across departments to ensure timely delivery and quality production.

Oleh Semenovych, Development Engineer

With over 4 years in front-end development, Oleh brings to Saga his expertise in JS and a myriad of frameworks.

Yoni Roll, Product Manager

A blockchain enthusiast and evangelist since 2012, Yoni has an in-depth understanding of the blockchain space and community.

Alex Leibo, Art Director

With over 5 years' experience as a full-stack product designer, Alex specialises in digital consumer products. His love of creating minimalist, simple and cutting-edge design has earned him both honourable awards and mentions.

Yair Hadar, Product Marketing Manager

With a rich background in online marketing, Yair has extensive experience in establishing and executing content and business development strategies for startups. Yair holds a BA in Communication and Media.

Nathan Lyons, Communications Director

A former Googler and consultant to companies such as Expedia, Nathan specialises in Systematic Inventive Thinking. A communications specialist, he holds a BA and MA from Oxford University.

Silvi Krispin, Office Manager

With 10+ years in administrative roles, Silvi heads Saga's administrative and operational infrastructure. Previously, she held a similar role at Mobli and worked in the IPO department at Bank Leumi.

RISK FACTORS

Operating across two formerly unconnected industries - the blockchain and traditional finance - Saga is likely to encounter challenges in each domain. It is reasonable to expect to meet obstacles that cannot be entirely predicted, as we strive to bridge between these two complex worlds.

Here, we briefly outline our assessment of the main directions of risk, and likely mitigations.

Adoption

Since Saga is a brand new venture, much of the initial risk lies in creating sufficient relevant interest, both amongst investors and participants.

Risk	Mitigation
Participant Adoption Although Saga has the quality to become a means of exchange, by definition - initially - the currency cannot truly be so. Means of exchange are defined by the public being willing to use the currency to exchange value. Saga therefore relies on a critical mass of adoption and usage, certainly a challenge for a nascent currency.	By implementing a mathematical and financial model that is designed to tame volatility and create a climate of sensible growth, Saga is built with the potential to provide the qualities of storing value. Hence, Saga does not depend on its value as medium of exchange in order to gain sufficient growth in the early stages. Once it has demonstrated quality as store of value, we believe Saga stands the chance of becoming a viable means of exchange.
Language and Communications - Partnerships Saga needs to create strong partnerships in the core industries that are essential to its viability; central banks, regulators, banking institutions. Introducing a new concept, combining technological tools and shifting the governance paradigm requires the formation of a glossary to enable strong agreement with these partners. The approach will be scrutinised by trained economists, lawyers and business people, who will naturally detect amateur errors.	Saga recruits leading practitioners from the worlds of finance, economics and blockchain. We hope to bridge gaps in contemporary language and form a new language that can gather acceptance among a diverse palette of partners. Saga will produce a regular blog, developing the discussion and providing explanations for readers. Initially aimed at recruiting brand-new readers, over time the blog will become a repository of knowledge about Saga. We will also feature written discussions with thought leaders, including Advisory Board Members, and potentially launch a webinar series. Saga will publish a ‘glossary of terms’ for public consumption and debate.

Risk	Mitigation
Market Sentiment: Excitement Saga's proposition is long term, stable and sensible. This is far less dynamic than the promise of many other actors in our sphere. Saga needs to make 'sensible' attractive in a hyped bull market.	Saga aims at a high volume of participants with a low risk - rather than a smaller number with a high risk, like most early-stage digital currencies. While high volatility lends itself to speculations and can account for fast-growing value in a small industry, we believe that in order to become useful and adopted by people in their daily lives, responsibility and stability are paramount. Hence we believe that volume of participation and our ability to cater to non-speculative uses will make up for a less energetic terminology.

Regulation

Currency issuance was not long ago the monopoly of states.

Digital currency, being a new phenomenon, challenges current regulatory frameworks with key issues such as Identity and Taxation.

Risk	Mitigation
Identity The approach of governments and regulators to digital currencies has been unclear to date. Their actions have mostly been limited to the enforcement of anti-money-laundering policies. Regulators have been reluctant to allow digital currencies to operate within their jurisdictions, stating both economic stability and anonymity as key threats to their enforcement abilities.	Saga provides an online KYC at international standards. This makes the currency potentially more appealing to, and accessible by, nation states and their executive bodies. Saga has engaged regulators and governors, who are working with us to harness the benefits of a digital currency while preserving the integrity of their ecosystem. As demonstrated in this paper, volatility taming is a fundamental pillar of the Saga model.
Taxation Tax policies regarding digital currencies have not been determined across jurisdictions (as yet). Issues such as the applicability of VAT and capital gain remain unclear.	Saga participates in several fora, where these issues are discussed with other interested parties across the digital economy. It is our belief that the eventual adoption of digital currencies and the manner in which they are used by the public, will define taxation policy.

Economy

Several of Saga's risks belong in the macroeconomic category.
The main issues are: a possible run-on-the-bank scenario, front-running and high-water-marking.

Risk	Mitigation
Run on the Bank In the classical definition, at a moment of panic, all participants try to withdraw their deposits during a short space of time. They fear the bank's vaults will run dry. A similar risk might be relevant to Saga, which also operates a reserve that participants cash out from.	When Saga's economy is very small (lower then 25M USD), Saga is fully backed and all participants are guaranteed an exact return of their money when they sell their SGA tokens back to the Contract. Moreover, as Saga's economy grows, the decrease in reserve ratio is gradual, reflecting the slowly-gained trust in Saga. In a shrinking economy, the reserve ratio is increased gradually, causing a corresponding gradual reduction in the value of SGA. This lowers motivation for any 'run on the bank', as all participants can be compensated. It is worth noting that the order in which people sell their SGA tokens to the Contract determines what price they receive. In any case, even the final seller will reclaim some value.
Front Running Saga participants are exposed to the risk of front running by miners and non-miners alike. Their mitigations will be different.	The risk of front running is known and widely discussed in recent literature. Part of this risk is mitigated by the model's price band. In addition, Saga is exploring several additional approaches, including: • Commit / Reveal • Price Averaging (within a block) • Gas Limits • Minimal Return
High-Water Marking The need to compensate early backers (Saga Genesis), causes Saga's pricing function to depend on the largest size the Saga economy has ever reached - the *high-water mark*. The price of SGA at a given market cap may be reduced depending on Saga's high-water mark.	In a shrinking economy, Saga's reserve ratio does not trace back the trajectory of its growth. Instead, it charts a faster increase, thereby limiting price dependence on the high-water mark. In addition, the influence of the high-water mark is proportional to the size of the shrinkage to the Saga economy. The probability of 'high influence' is therefore low: it can only happen if the Saga economy both reaches a very high market cap and then hits a major recession.

Risk	Mitigation
Pump and Dump The SGN to SGA conversion ratio depends on the high-water mark of Saga's economy. Major SGN holders may attempt to pump and dump the Saga economy, in order to exploit this mechanism and maximise their return. If this occurs, SGA price will drop as a result, due to the minting of new SGA tokens for SGN holders. It also causes unwanted volatility in SGA price.	The conversion ratio is increased through a series of small steps, limiting motivation for 'pump and dump' for a single step. Pumping through a few steps together requires a large investment; only a very limited number of wealthy players could 'game' the currency in this manner. In addition, a Genesis Minting Point is only considered achieved - and the conversion ratio increased - if Saga's economy has stayed above the Minting Point for one week. Any SGN holders that try to tamper with the size of Saga's economy will have to support the resulting higher SGA price for an entire week. This could carry huge costs to the 'pumper' if other participants believe that the elevated price does not represent the currency's true value.
Ether Price Vulnerability Saga's currency is run on the Ethereum Platform. Initially the only way to purchase Saga from the Contract will be in Ether, the currency of the Ethereum platform. This exposes early-stage Saga to changes in the Ethereum ecosystem, particularly to fluctuations in the price of Ether.	Ether is merely a liquidity buffer for Saga. This means that we use ether to reimburse sellers of Saga, so that we don't need to perform an ether/fiat conversion for every transaction. Saga plans eventually to employ banks not just as reserve holders, but as resellers of Saga tokens. Customers of these banks will be able to acquire SGA through the bank, using fiat currency.

Technological

There are two main issues confronting Saga from a technological angle: Cyber Security and Scalability. The latter can be subdivided into issues relating to Throughput, and those connected to SLA (Service Level Agreements).

Risk	Mitigation
Scalability - Throughput. Saga is dependent on the blockchain infrastructure. If there is a sudden spike in demand on the network, or any kind of impact its running, it can have a knock-on effect on Saga's functionality. **Scalability - SLA.** Ethereum is intrinsically quite a slow network. At times of high demand, the network may struggle to handle all transactions.	Saga has chosen Ethereum as the platform on which to build and operate our currency. This dependency exposes the currency to the fate of the Ethereum network; naturally we will carefully monitor Ethereum's plans and actions. In order to keep open the option of migrating to other chains, Saga will investigate and perform POCs on additional networks, as they come into existence. Saga is designed to be flexible enough to migrate to another chain.
Cyber or Security Threats A hacker may seek to exploit security breaches in Saga's systems. There could be many motivations: to steal or change Saga's data; to add backdoors to our systems; to slow down or damage the flow. We cannot know in advance the mechanisms that hackers may employ, or the vulnerabilities they may exploit. Certainly there can be many loopholes in the blockchain outside of Saga's purview.	Saga's contract is audited using third-parties, who proactively check for vulnerabilities. We only partner with libraries that enjoy proven success. Ethereum has the strongest track record of all blockchains; able to withstand assaults with good, rapid recovery times. The maturity and volume of this network mitigate against any major risk.