African Blockchain for Agriculture (AB4A)

White Paper

We believe that the most effective way to bring about sustained growth and development to the continent is through a vibrant agricultural sector imbued with the necessary technology and liquidity that facilitates production of value and effective delivery of that value to markets

October 4 2017 (Ver 2.0)

Abstract

Agrikore represents a comprehensive approach to enhancing agricultural production, produce delivery, and access to financial services to rural and agricultural communities, focusing on the African continent, but potentially extending beyond to rural and farm communities world-wide. The Agrikore ecosystem is already established with millions of registered customers and farmers, mobile banking applications in the Tingg application, and a network of service agents across several African countries. This whitepaper describes how blockchain technology utilizing a transparent public ledger, and connected system of smart contracts would form the heart of the system and help facilitate transactions in ways previously unattainable with any other technological methods, with enhanced trust between all actors.

Agriculture is the staple occupation in most regions of Africa and there are lots of land resources that is dedicated to agricultural activities. Despite this, on an efficiency basis, the output is middling and Africa continues to import food and solicit food aid to the tune of billions of dollars annually. Significant challenges are in the area of agricultural production and obtaining the necessary resources in season, as well as storage and produce delivery challenges. Cellulant, developers of Agrikore, have been engaged in providing solutions to some of these challenges for some years – helping governments of African countries in disbursing Agricultural subsidies, providing mobile payment services to farmers, developing and operating produce delivery framework for large commodity buyers. This led to the identification of a connected framework that can result in significant enhancement to food production and produce delivery in the region.

The Agrikore blockchain is built based on technology derived from the Ethereum network due to its inherent smart contract capabilities. Significant modifications were added to this framework to make it suited to the problems being addressed in the Agrikore ecosystem. Firstly, all issued assets are backed by an actual reserve in a currency trusted by the community. The blockchain network was also modified so that block times was cut down to 10 seconds allowing in person, in-store transactions. The time-consuming proof-of-work algorithm prevalent in early blockchains was replaced by one relying on trusted processing nodes with significant stake or deposit in the reserves. A completely new way of handling savings within blockchain networks is also introduced, as well as an automated and fair monetary control procedure to reduce the inherent volatility of blockchain networks that are based on fixed supply paradigm. Finally, the extensive network of smart contracts for providing liquidity and facilitating produce delivery was built inherently into the system. The current white paper discusses these developments and presents a roadmap for the growth of the ecosystem. The hope and expectation is that this project will result in significant growth of this large region of the world thus increasing its GDP and contribution to the overall world economy, and its own self-sufficiency.

- Ken Alabi (Ph.D) & Bolaji Akinboro (CEO Cellulant)

Table of Contents

1.0 Introduction

Agrikore is a blockchain wallet that digitizes & organizes the way participants with Agriculture interacts with each other for the purposes of agricultural commerce (service provisioning), payments, and financial services. Conceptually, what Uber is to transport is what Agrikore is to Agriculture. It also potentially does what ebay does (merchants aggregation), PayPal (seamless payments) & Alibaba (market place) for agriculture in a very simple and elegant way.

Agrikore is developed & operated by Cellulant, Africas # 1 Fintech company, as published in the FINTECH Africa payments & transfers category, 2016. Conceived in 2012 by Dr. Akinwunmi Adesina , President African Development Bank. Designed, developed & implemented as an eWallet solution to deliver subsidies (coupons) & organize input supply value chains. It has evolved to become an end-end blockchain enabled agricultural commerce solution that enables governments, donors, banks and large fastmoving consumer goods companies to deliver subsidies, loans, inputs and offtake produce from millions on farmers on the African continent.

Agrikore is the only large scale blockchain or smart contracting solution that is enabling the hidden $330Billion agricultural sector, according to a World Bank 2103 report [1], that will grow to $1Trillion over the next decade. Over the years servicing these communities, Cellulant has been able to identify some of the challenges related to agricultural produce delivery, storage, agricultural production financing, resource distribution, as well as the access to financial and banking services within these communities. These challenges are:

i. Liquidity & allocation of capital: - Despite the huge size of Africa's agriculture sector, the capital & commerce market for agriculture is illiquid and there is no clear mechanism for efficient allocation of capital. Those who have it can't get it to those who need it at scale and fast. This problem of capital allocation is the major hidden reason across the continent why there hasn't been sustained growth. This blockchain will be Africa's instrument for capital allocation in an efficient manner; the way that liquidity circulates amongst everyone. It is how we get liquidity to the smallholder farmer, the intervening actors, and capture the upside from the commercial activities within agriculture.

ii. Asymmetry of information: Africa and Africans are unable to trade with themselves because information related to demand and supply are not available in-country and cross-country. Millions of smallholder farmers cultivate, and information required to convert their produce to wealth is not freely available to all. Across Africa, there are mechanisms for storage – formal & informal, transport & markets that are working but because they are not integrated together find it difficult to exploit the potential. This blockchain will be Africa's ledger for Agricultural produce. Contained within this ledger is information related to primary produce; storage sites where farmers take their produce; buyers and their offer price; and transporters who move the produce. It is how we connect the last mile and first mile together; essentially it is how we organize agriculture.

iii. Democratizing & digitizing access to markets, finance & payments: How do you as a private citizen, government, donor, bank etc., participate in agriculture in a very simple way without being the person that has to go to the farm physically and being subject to complex, complicated contract instruments? The blockchain will serve as agricultural community's unit of account and digital asset for agriculture. It is how records are kept within the economic system that is understood by everyone so that everyone can trade with each other.

Some of the challenges have been addressed in process and technological ways. For instance, Cellulant developed a highly functional mobile banking application in form of the Tingg mobile App that provides banking and payment services via mobile devices. In addition, the organization also deployed thousands of physical banking kiosks in communities where the nearest bank branches are over twenty kilometers away. However, some of the challenges sill remain, and are best addressed by a well thought out and implemented blockchain system, due to the unique features that blockchain networks provide. The process and the technology behind this effort is described in detail in this report.

The Agrikore blockchain is a peer to peer permissioned network built on similar technology as the Ethereum network. The Agrikore blockchain was further enhanced with features to address the specific use cases in the communities of interest. Specifically, the blockchain does away with mining rewards, implements a low volatility monetary system, issues only assets that are backed by true reserves, and consists of smart contracts already tailored to the various agricultural contracts that form the basis of a vibrant agricultural community.

1.1 Role of the Blockchain Technology in the Agrikore System

Note that Agrikore is not solely a blockchain solution. It is not a system of mining coins that is based on pseudo-economic activities. Agrikore is an over-arching system for trading , payments & commerce that is anchored in the real world. It ensures that commerce, banking & payments all work together in on seamless agricultural ecosystem.

Figure 1 illustrates the entire system. The core currency of the system is named the MULA. The MULA is designed to be perfectly liquid and can be exchanged into local currency on demand via the Tingg app or mobile banking kiosks. The way Agrikore brings the various markets together for farmer in the ecosystem is depicted in the figure below.

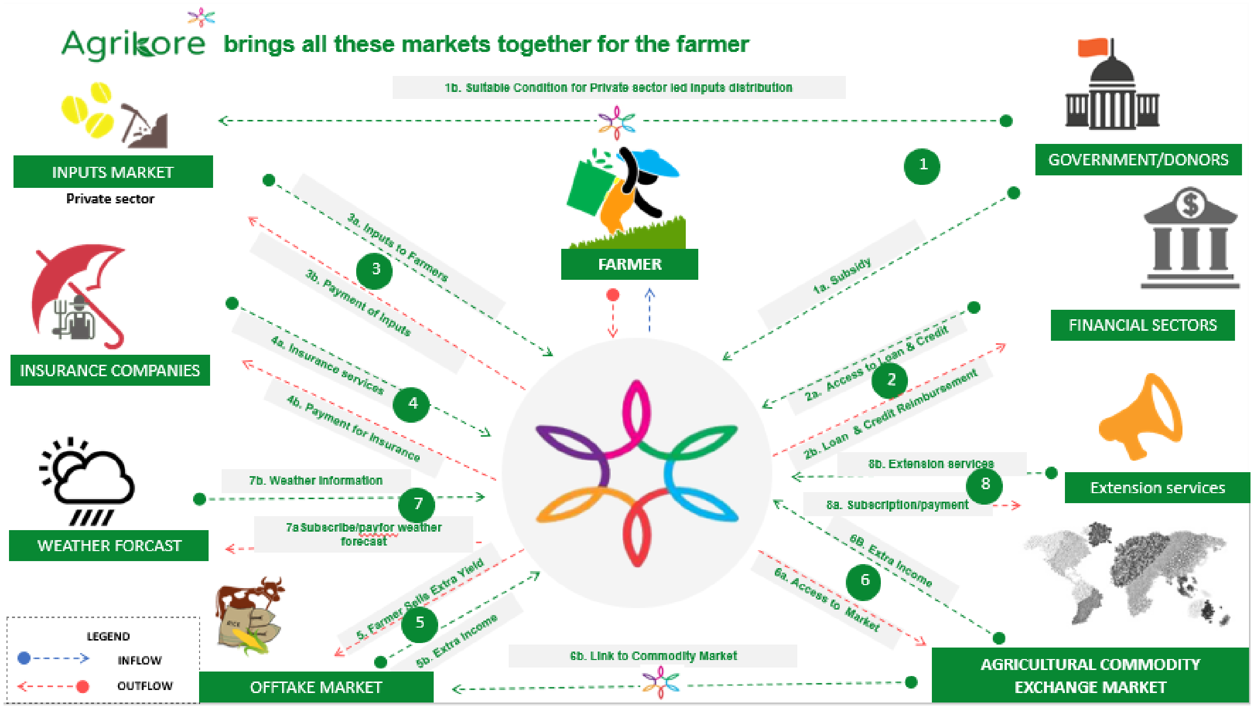


Figure 1. **The Agrikore ecosystem: Key stakeholders in the agriculture value chain and they interact**

1.2 Description of the Roles in the Agriculture Commerce Ecosystem

To ensure clarity of the various roles, and terminologies that will appear in the rest of this report, the following simple schematic illustrates the path of services, payments & finance within agriculture:

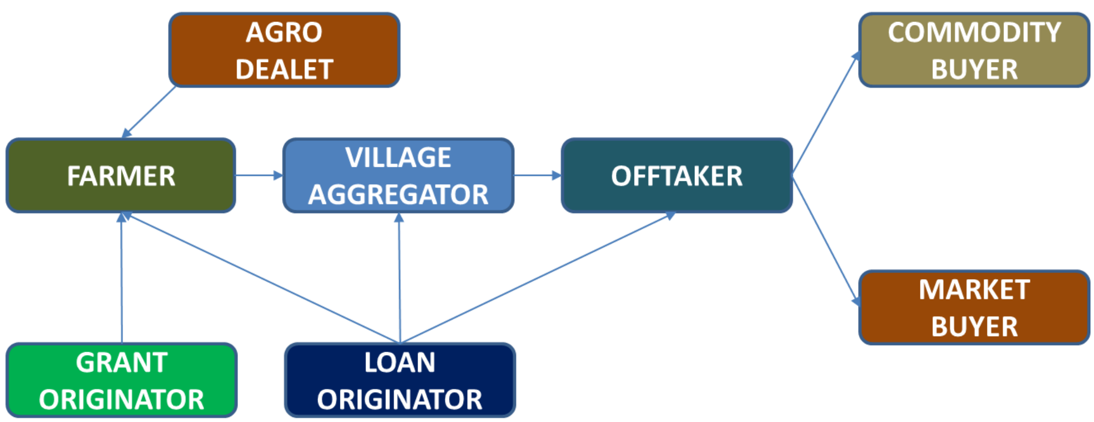


Figure 2. Players in the Agricultural Production and Produce Deliver Ecosystem

The farmer is the producer of agricultural produce in the ecosystem.

The village aggregator is a member of the community who is active and has physical presence in the farming communities, is also typically a farmer, and is able to collect farm produce in bulk for delivery to an offtaker. The village aggregator typically has access to liquidity more than the average farmer.

The offtaker is a wholesale buyer of agricultural produce.

The commodity buyer typically represents a large produce factory or processor.

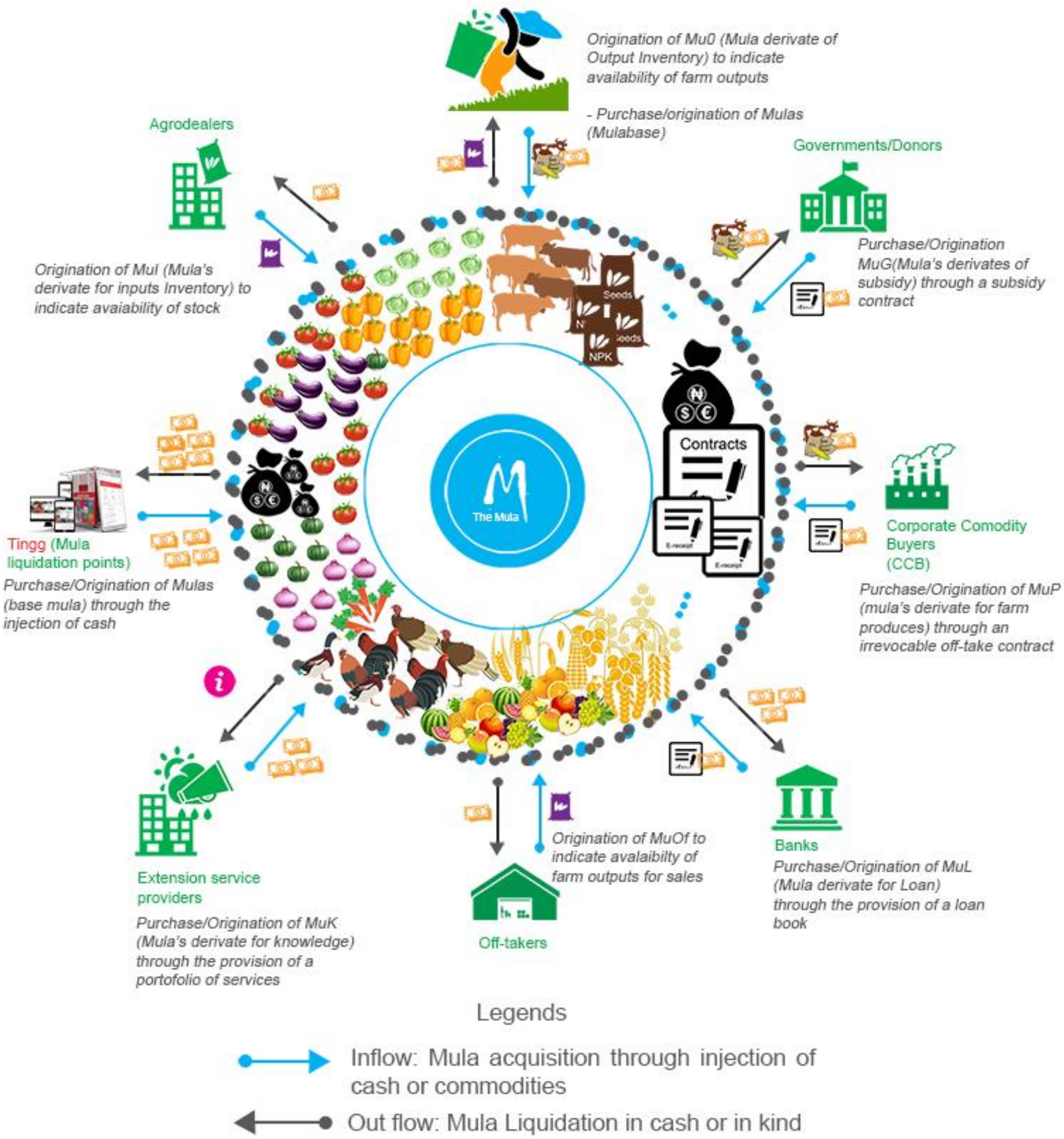


Figure 3: The Mula didgital asset & smart contract within the agrikore ecosystem

The market buyer is a wholesale purchaser of agricultural produce that gets directly distributed to the market.

The agro-dealer is a seller or lessor of agricultural resources such as seeds, fertilizers, feeds, and equipment such as tractors.

The loan originator is an individual or organization/bank that provides loans to farmers or other roles in the agricultural ecosystem.

The grant originator is usually a government body or non-governmental organization that provides subsidies, grants, price support, or developmental aid to sustain or incentivize agricultural production or produce delivery to the market.

1.3 What AgriKore does?

We utilize the Agrikore blockchain system to organize the sector for wealth creation for entrepreneurs and farmers, the same way Uber organized transport. We expect to change the lives of Millions at the same time.   The Agrikore system is the next evolution of the eWallet system that came out of Nigeria.

The injection of liquidity by corporate commodity buyers to trigger the ecosystem ensures that the liquidity that multiplies itself is available within the ecosystem. No such technology today exists on the surface of the earth and it is another example of cyber-physical (economic activities that are facilitated by technology that are anchored in the real world) services. This framework will unleash economic activities like never before in Africa as the mechanism for the allocation & exchange of resources on a digital platform with the Agric ecosystem is now in place.

The size of this opportunity is huge. There is $330B (2013) that will grow to $1 trillion by 2030 of economic opportunity [1] within Africa's agriculture ecosystem because the sector is not organized to capture that opportunity. Agrikore enables a significant part of this ecosystem to be captured. It brings the technology & organization to capture this opportunity with a payments and service ecosystem.

This paper outlines how this blockchain solution that has been implemented by Messers Cellulant delivers the promise of this economic opportunity & at the same time lifts millions of Africans out of poverty.

1.4 Issues addressed by the blockchain solution for Agriculture

There are four main area addressed by the current effort, which are found to be the most challenging for these communities:

i. Access to Financial Services (Payments/Banking)

Despite the huge size of Africa's Agriculture sector. The capital market for agriculture is illiquid and there is no clear mechanism for efficient allocation & distribution of capital. Those who have it can't get it to those who need it at scale and fast. This problem of capital allocation and distribution is the major hidden reason across the continent why there hasn't been sustained growth. This blockchain will be Africa's instrument for capital distribution and allocation in an efficient way. It is how we get liquidity to the smallholder farmer, the intervening actors, and capture the upside from the commercial activities within agriculture. The presence of financial service actors in the network who inject liquidity to the participants, and the existence of retail point of presence (Tingg points) allow both liquidity and distribution to take place.

ii. Asymmetry of information:

Africa and Africans is/are unable to trade with itself/themselves because information related to demand and supply is not available in-country and cross-country. Millions of smallholder farmers cultivate, and information required to convert their produce to wealth is not freely available to all thereby putting them at the mercy of middle men who serve as market intermediators. This blockchain will be Africa's ledger for agricultural produce. Contained within this ledger is information related to primary produce storage sites where farmers take their produce; buyers & their offer price; and transporters who move the produce. It is how we connect the last mile and first mile together; essentially; it is how we organize agriculture.

iii. Democratizing & Digitizing access to markets, payments :

Market places in Africa are fundamentally physical. No construct exist today on a large scale that democratizes access to markets the way Alibaba & Amazaon have done. This is what Agrikore seeks to achieve. It enables private citizens, governments, donors, banks etc., to participate in agricultural enterprise in a very simple way without being the person that has to go to the farm physically, or be subject to complex , complicated contract instruments. The blockchain will serve as Africans unit of accounting and digital asset for agriculture. It is how records are kept within the economic system that is understood by everyone so that everyone can trade with one other.

1.5 How the Challenges Are Being Addressed

There are some aspects of the challenges described above being imperfectly addressed by the communities themselves in some ways, some of which are listed below:

1. Farmers in some of the communities have organized themselves in ways where they are represented by a large farmer usually with greater liquidity, termed the Village Aggregator. For instance, utilizing that liquidity leverage, the village aggregator is able to purchase farm produce from the other farmers and organize to deliver them in larger

bulk to an offtaker, and earning a margin off the difference between the produce price at the offtaker compared to the farm.

2. The offtaker sometimes approaches the banks and guarantees loans on behalf of the farmers. The offtaker then monitors the farmer through the growing season, through their own staff and agents, and takes delivery of the produce at season's end, obtaining loan repayment off the top.

The Agrikore system has also put in place solutions to the above challenges in the manner described below:

1. Development of a comprehensive process, technological, and organizational structure to identify and sign up farmers so that farmers can be addressed and targeted for subsidies and grants. In these regards, Cellulant is one of the biggest organizers in farm communities and have been able to register over 17 million farmers in about five different countries. Based on this capability, Cellulant has been contracted by several governments in the region to assist in targeted subsidy delivery.

2. Development of the Tingg mobile banking application, that provides payment and banking services.

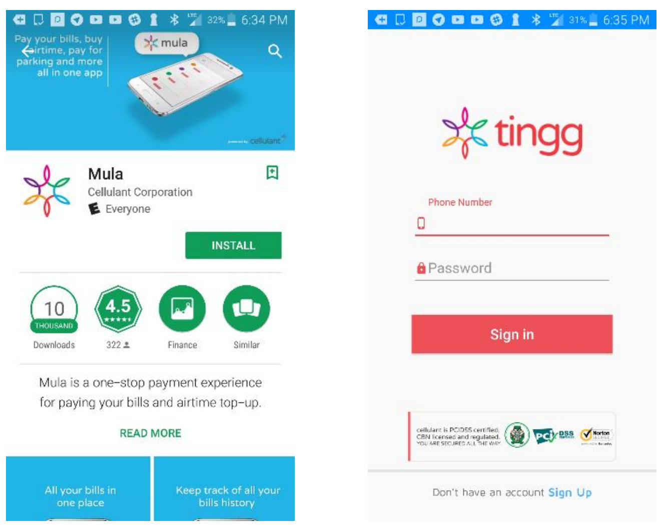


Figure 4. The Tingg Mobile Payment App

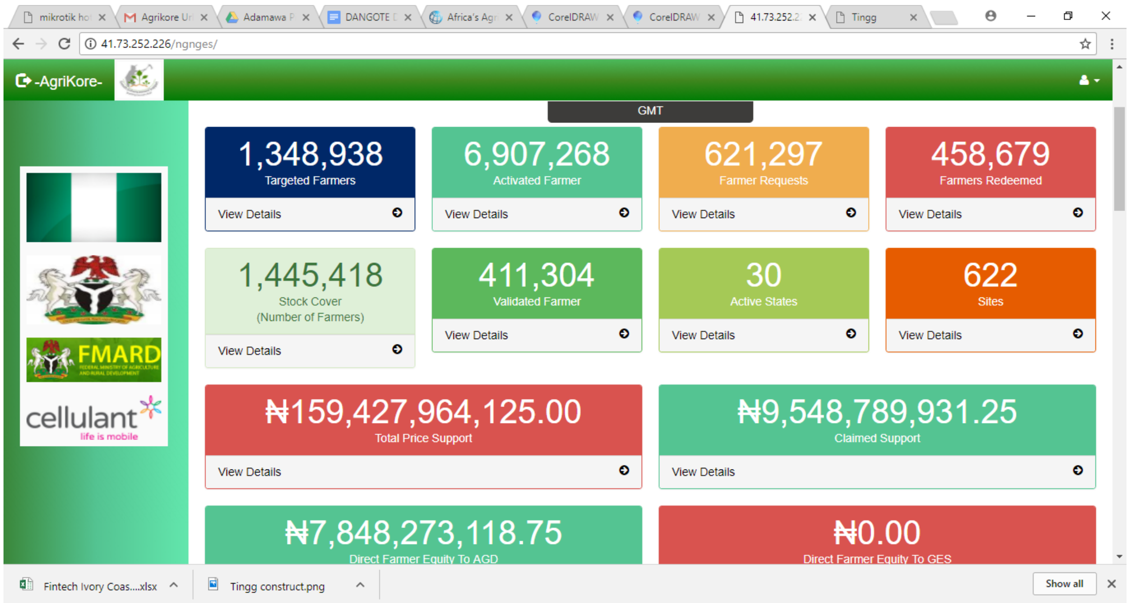
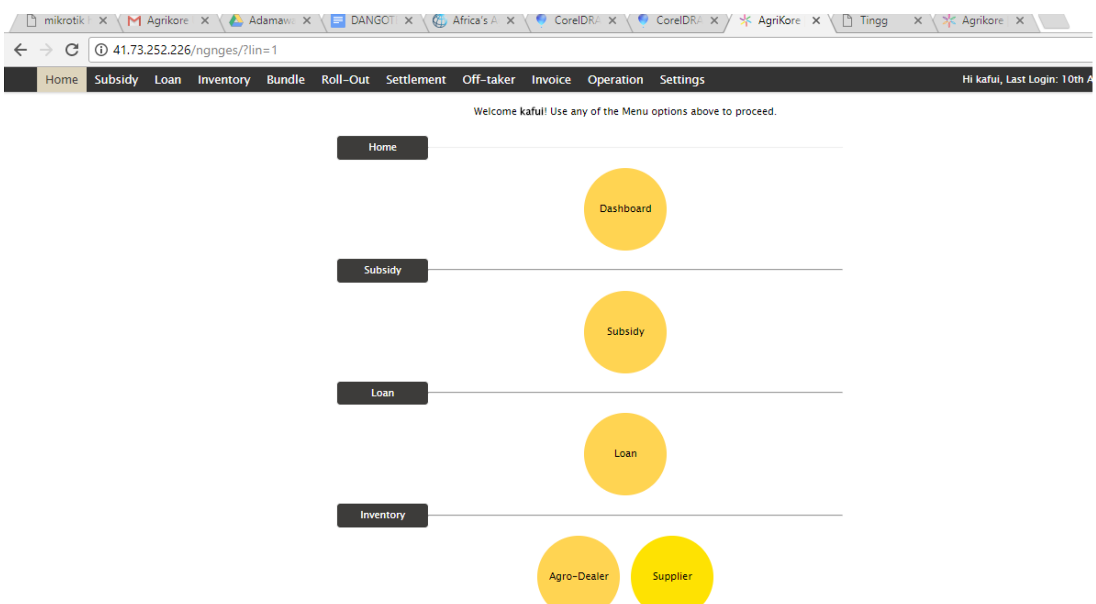


Figure 5. The Agrikore service delivery dashboard



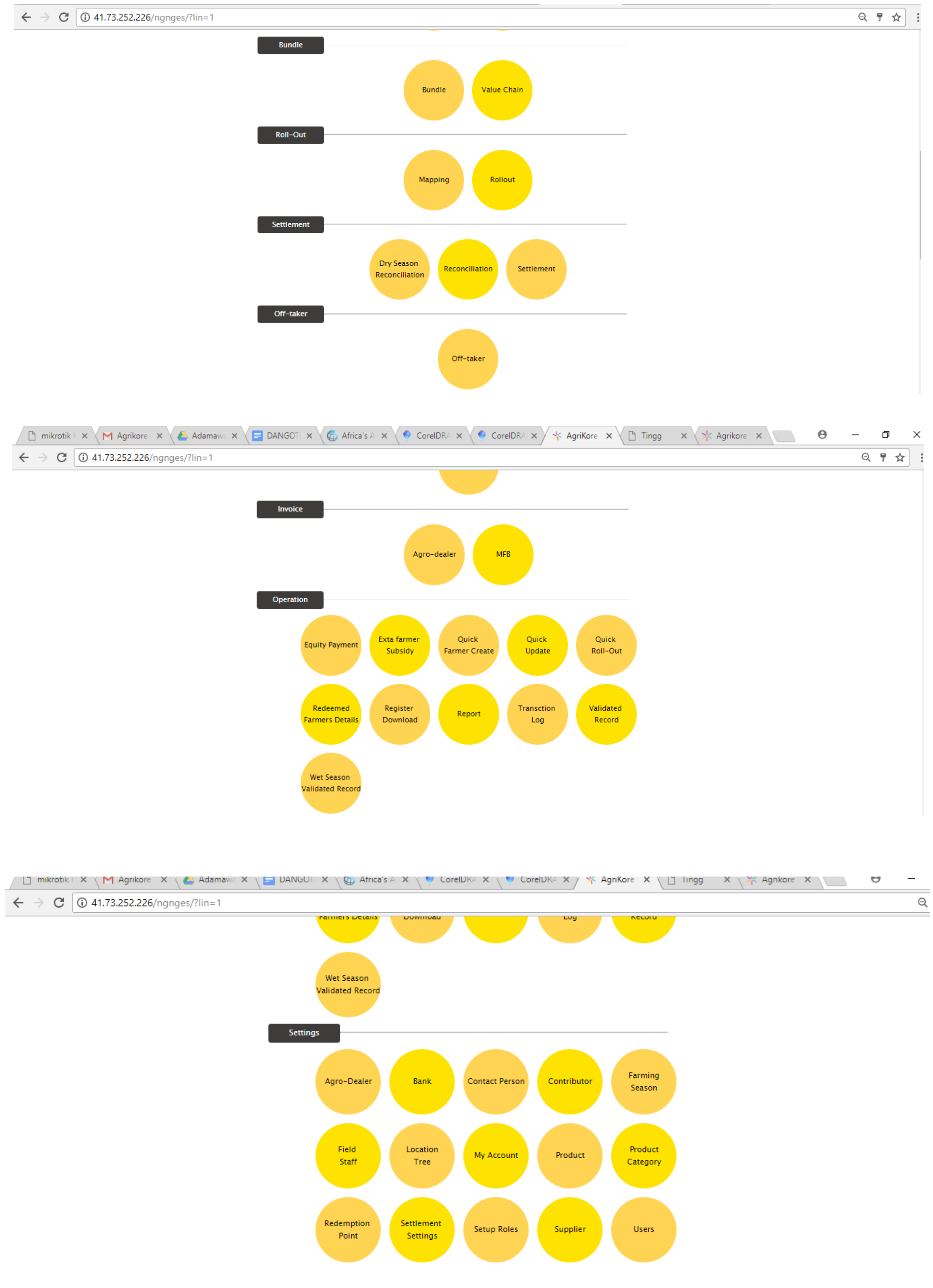


Figure 6. The Agrikore platorm core modules

3. Installation and deployment of mobile banking kiosks in rural communities where the nearest bank branches are over twenty kilometers away.

Figure 7. One of the several thousand mobile kiosks recently installed in agricultural communities

4. The Agrikore blockchain is the last piece of the comprehensive solution to enhance agricultural production and promote rapid development in these communities.

2.0 The Agrikore Ecosystem

As described previously, the Agrikore ecosystem represents a comprehensive system and solution for providing operational and financial solutions for enhancing agricultural production and produce delivery. The Agrikore blockchain occupies a critical role in the entire Agrikore system. This role is described in this section, particularly in the context of its relationship to the entire system, the functions it performs, and how the blockchain technology best performs those functions.

2.1 Role of the Agrikore Blockchain within the Overall Agrikore Ecosystem

The active roles introduced in Section 1 resolves into three types. Consumers consisting of farmers, village aggregators, offtakers, commodity buyers, members of the community using the system daily for its mobile banking and financial services, and finally government and non-governmental organizations. These users interact with the system through the Tingg mobile application or web client, or through Cellulant staff or agents in installed mobile kiosks.

The other two types of users include loan originators from banks and financial institutions, or merchant and stores. For these types of users, Cellulant provides integration and payment API and solutions for users to connect their system to the Agrikore financial and payment channel.

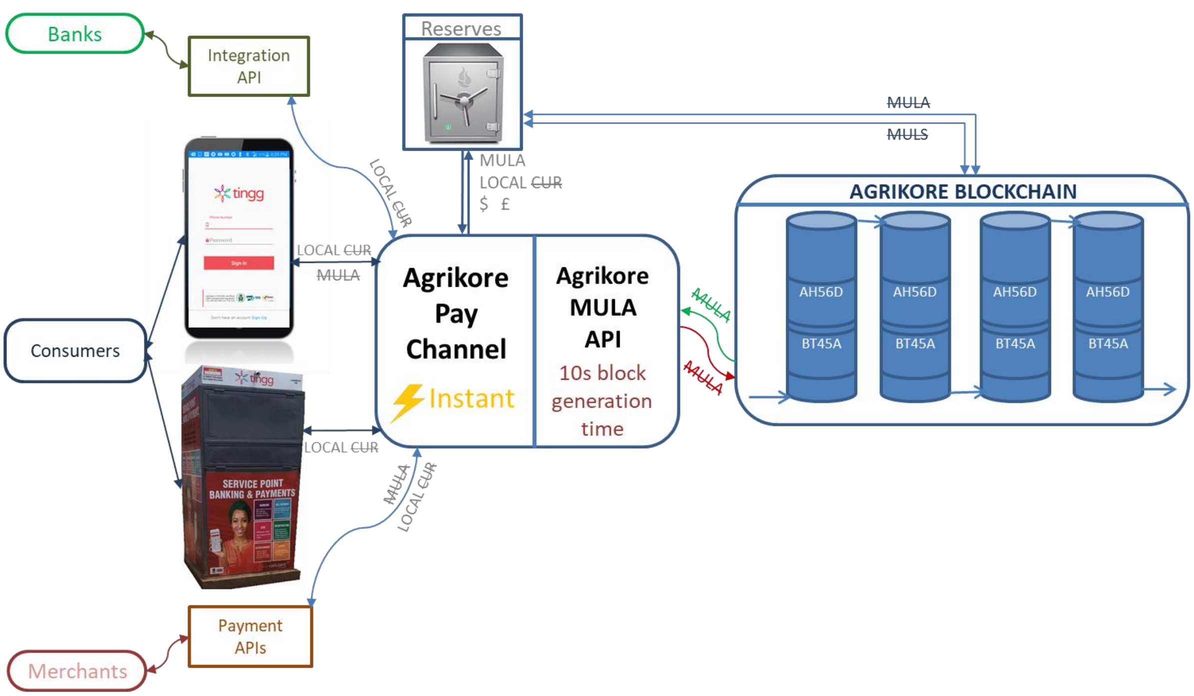


Figure 8. The Agrikore ecosystem and the flow of local currency and MULA blockchain currencies within the system

In addition, there are less active participants in the ecosystem that simply purchase the MULA due to inherent production within its ecosystem, and its non-inflationary constitution. These participants mostly hold the MULA so as to enjoy the benefits of saving MULAs in the ecosystem, and details of savings vehicles or contracts in MULAs is also described later in this paper.

For the active roles in the ecosystem, the Agrikore blockchain is designed to support the transactions that enable the entire eco-system. This is due to merits that a blockchain has that is particularly suited to some of the challenges highlighted in the prior section. These merits or properties will be clearly delineated in this section. In addition, in the development of the Agrikore blockchain, the Agrikore project team embarked on a thorough analysis of prior blockchains in crafting a unique solution suited for the agricultural ecosystem. Significant modifications to the way in which earlier blockchains were implemented were necessary and are also highlighted in this section.

2.2 The Blockchain Technology

A blockchain is a public ledger that connects blocks of transactions together in a chain. Starting from a genesis block, additional blocks are added each referencing or chained to the block before it. The blocks are chained so that transactions become immutable because any change would need to go backward to prior blocks to effect that change, which is virtually computationally impossible. This is because each block has to be signed by a time consuming and difficult computational process that ensures that redoing blocks would take an inordinate amount of time and new blocks would have been written that would render the rewritten blocks invalid.

Since the blockchain or public ledger is typically distributed among processing nodes and wallets with each maintaining its own copy, there needs to be a way to ensure that all blocks maintain and represent the same information. This process is known as consensus. Only one copy of the ledger is accepted by all nodes at any point. And any new block written is chained to that valid copy of the blockchain. The process of computationally signing a new block ensures that only one of the processing nodes is allowed to write a block successfully. The two main popular types of algorithms that is used to enforce this condition: are proof of work and proof of stake algorithms.

<u>Proof of Work (PoW)</u>

The proof of work algorithm is a computationally time-consuming exercise solving a cryptographical puzzle, for instance, to find a nonce or bit such that the hash of that nonce together with some data from the transactions has a specific form. One such form is to find some fixed number of zeros in front. Proof of work algorithm cause thousands of nodes to perform electricity consuming calculations daily in order to create blocks [1,2]. They also throttle how frequently blocks can be created and limit the transaction capacity of the networks.

<u>Proof of Stake (PoS)</u>

For proof of stake algorithms, the exact node that would process the next block is chosen by a pseudo-random algorithm. There are several different ways of ordering this selection process but the most prominent one is weighted based on the stake, or amount of the blockchain asset held by the processing node. PoS algorithms are more energy efficient and faster than PoW algorithms.

There are four main types of blockchain implementations:

1. **Bitcoin-Type Blockchains**
   Several blockchain implementations are essentially derivatives of the first popular blockchain – the Bitcoin network [4]. Bitcoin blocks consists of blocks chained together historically as described above, and in addition all transactions are related to prior ones by creating new transactions only from existing ones.

2. **Ethereum-Type Blockchains**
   Ethereum technology is also similarly built on blocks [5]. However, unlike bitcoins where transactions are chained to prior transactions and balances are inferred by checking the balance of unspent transactions, Ethereum blocks maintains balances of accounts similar to bank ledgers, and transactions are raised off those balances. A more important difference is the introduction of smart contracts or programmable assets into Ethereum. This makes it possible to assign assets including currencies created on the blockchain with extremely smart programs and instructions. The smart contract engine includes all the instructions of a full programming environment including conditional logic, loops, directive statements, and mathematical commands.

3. **Networks Based on Obfuscating Transactions**
   The above two technologies are referred to as pseudo-anonymous networks. All transactions are transparent and can be reviewed by the public. However, names are not attached to account information. Some other types of networks were designed to create complete anonymity including for transactions. Examples of these types of blockchains include the Dash network [6], Monero, and ZCash.

4. **Directed Acrylic Graph Chains**
   This type of network deviates slightly from the original Blockchain technology. In fact, transactions are no longer grouped in blocks that are chained together in time but connected directly in a directed acrylic graph (DAG)[7]. This allows some interesting properties including faster and cheaper transactions. Subsequent transactions in a DAG tree reinforce prior ones in a weighting system. However, on scrutinizing the mathematics of current DAG implementations, the probability of asset compromise appears higher than in older, more tested blockchain technologies such as Bitcoins, which has actually not seen a successful on-chain attack in several years.

The Agrikore blockchain technology is based on the Ethereum technology, with significant modifications as described in Section 3. Ethereum technology was selected as the basis for Agrikore due to its smart contract feature, which is required for implementing the comprehensive interconnected trustless interactions that are key to the ecosystem. The rationale for the changes and additions made to the protocol and code is discussed in the nest two subsections.

2.2 Some Shortcomings of Prior Blockchain Systems Addressed in the Agrikore Implementation

The following features of the blockchain makes it particularly suitable for implementing solutions to the Agrikore ecosystem:

1. **Creation of value seemingly out of nothing**
   Earlier blockchains have created digital currencies and allowed them to rise in value based on demand and the scarcity of their currencies created by their low production rates. For an existing ecosystem such as the Agrikore system, we believe this approach will not suffice. Value created in this manner may prove fragile for an existing agricultural community and we decided not to build a system on that same foundation.

   Historically, new currencies have grounded their value based on the accepted value of other proven prior means of exchange in the same economy. The dollar during its creation was initially pegged to gold, whose value was trusted and established at the time. Similarly, every asset or digital currency issued on the MULA network would be backed by a reserve of equivalent value. This process is discussed in greater details in the third section of this paper. (Please also see the Paper titled: "Digital Currencies Need a Reserve System" [8] for more discussion of this feature.)

2. **Lack of Monetary Control**
   The method of supply of most blockchain systems was by rewarding processing nodes with new minted currencies [9,10]. Adopting that method of supply on the Agrikore blockchain is problematic for two reasons:

   a. Such supply distribution is inequitable in the Agrikore ecosystem since the drivers and creators of new value are not mainly transaction processors but producers of agricultural commodities.
   b. A fixed supply mechanism results in no monetary control, resulting in currency value fluctuations that are very unpredictable. Such fluctuations will not augur well for adoption. Farmers and members of the Agrikore ecosystem are unlikely to be comfortable with seeing the value of their produce fluctuate by more than 30% in a day, as sometimes occurs for current blockchain implementations.
   c. The supply mechanism of many blockchain implementations is extremely deflationary. This result in assets that are not attractive for use in commerce, and is better suited as a short-term investment vehicle. Most farmers would not want to spend currency from their wallet today only to rue that spending as the asset's purchasing power doubles by the next week.

   The Agrikore blockchain addresses these supply shortcomings by implementing an automated feedback supply mechanism based on proportional integral derivative (PID) controller to compute its supply rate. New mint is introduced via injection of new value into the reserve in the way of agricultural loans, grants or subsidy sale of savings instruments, or an irrevocable order of produce backed by the reserves. The details and equations for this supply-side technique is presented in section 3. (*Please also see the Paper titled: "Monetary Policies and Control of Digital Currencies"* [11] for more discussion of this feature.)

3. **Slow and Wasteful Expenditure of Energy to Process Blocks**
   Earlier blockchains that are based on POW algorithm such as bitcoin expend significant time and electricity in processing blocks. We did not believe this would be sustainable for the Agrikore community. As a result, Agrikore does not utilize POW but instead implements a form of proof of stake, that includes only trusted administrative nodes which have injected liquidity into the reserve (stake). An administrative node can only create blocks with total transaction values, over some time period, that is less than their stake in the reserve. This mitigates the possibility of a node deliberately processing a bad transaction such as in a double spend attack, as their stake would be used to cover the transaction. As a result, the block production rate, for the Agrikore blockchain is set to 10 seconds, as depicted in Figure 2.

2.3 Blockchain Features that Enable and Enhance the Agrikore Ecosystem

The following features of the blockchain makes it particularly suitable for implementing solutions to the Agrikore ecosystem:

1. Blockchain implemented smart contracts allows the creation of various financial-agricultural instruments that can be programmed to inject liquidity into agricultural produce delivery and production financing in specific ways, than would be possible if actual cash were instead injected. This is because the smart assets could be programmed with conditions that ensures that they can not be easily diverted, can have time variable behavior, and conversion conditions in a manner completely impossible to accomplish with cash.

2. Blockchains by nature have transparency properties such that transactions can be verified on the blocks. Combined with the properties of immutability in a manner that does not exist with prior financial systems including in modern banking, blockchains are especially suited to processes that are more corruption prone, such as produce price support and subsidy distribution. (*Please refer to the Paper titled: "Corruption Meet Your Cryptonite – the Blockchain"* [12] for more discussion of these features.)

3. Transactions on the blockchain are potentially cheaper than traditional banking systems. They are also more fraud resistant due to their chained and immutability properties.

3.0 The Agrikore Blockchain

The technical details of the Agrikore blockchain implementation is discussed in this section. As mentioned in 2.2, the core technology on which the Agrikore blockchain was developed upon is that of the Ethereum blockchain. Modifications and additions to the network were needed to implement the changes described in 2.3 and 2.4. These are described in this section.

3.1 The Agrikore Blockchain Formulation

The Agrikore blockchain is a ledger consisting of accounts and balances. Let the set of accounts at any time be represented by $\overline{\sigma}$, where a single account σ_i has information on the balance of the account, V, as well as its cryptographic information, $P_{k,p}$, the key/value pair that needs to be presented for the system to permit any transaction on the account, and an array of transaction hashes with look up information implicitly linking them to prior blocks containing those transactions:

$$\sigma_i = \sigma(V, P_{k,p}, T[],...). \tag{1}$$

The state of the ledger is transitioned in blocks, B, with each subsequent block representing the current state of all accounts, together with the series of transactions, T, that modifies the state of the accounts from the prior block to the current one. There is only one valid block at any time, and the most recent block represents the state of all accounts in the ledger. Starting from a genesis block consisting of no account, B_0, the entire blockchain continues to evolve in time with the creation of accounts, $\Delta\overline{\sigma}$, and transactions that modify the state of those accounts:

$$B_{l+1} = \Psi(B_l, \Delta\overline{\sigma}, \overline{T}_{l+1}). \tag{2}$$

In the above equation, Ψ is the state transformation function for the entire blockchain, $\Delta\overline{\sigma}$ are the new accounts created since the last block B_l, and added to the blockchain, to create the new set of accounts in block B_{l+1}, where

$$\Delta\overline{\sigma} = \overline{\sigma}_{l+1} \cap \overline{\sigma}, \overline{\sigma} \subseteq \overline{\sigma}_{l+1}, \tag{3}$$

and noting that accounts once created are never removed from the blockchain. $\overline{T}_{l+1}$ are the set of transactions applied to the accounts in block state, B_l, and all new accounts added to the blockchain, $\overline{\sigma} + \Delta\overline{\sigma}$,to bring them to block state B_{l+1}.

$$\overline{T}_{l+1} = [T_1, T_2, ...T_{N_{l+1}}], \tag{4}$$

are the set of N_{l+1} transactions that were found in the transaction stack since block l, and included in creating block l +1. Each transaction T_j are of two types – those that are of arithmetic accounting nature that move balances between accounts, and those that are execution of a computer code representing a smart contract. For an accounting type, the transaction consists of a set of accounts, σ_m, and change to their balances ΔV_m applied to them in that transaction:

$$T([\sigma_m],[\Delta V_m]) \rightarrow (\sigma_m)_{l+1} = (\sigma_m)_l + \Delta V_m, \tag{5}$$

$$\sum \Delta V_m = 0. \tag{6}$$

In the simplest representation, a transaction between two accounts, where account *A* pays account *B*, 10 MULAs, with a fee of 0.01 MULA to the transaction processor *C*; with the payer being responsible for the transaction fees; the transaction vector would have the form:

$$T_i([\sigma_A, \sigma_B, \sigma_C],[-10.01,10,0.01]). \tag{7}$$

For any type of transaction, the transaction data must also include the right key/value pair to unlock any account, in order for any change to the state of the account to be permitted.

3.1.1 Block Chaining and Block Creation Time

Each new block contains a cryptographic hash of the prior block such that any change to the previous block invalidates the block built upon it, in that the cryptographic hash would become altered. This provides the immutability property of blockchains. To make it virtually impossible for a node to go backwards and rewrite a block, and then subsequent blocks, to catch up with and supplant a current legitimate block. Nakamoto in his seminal bitcoin paper [4] introduced a time-consuming computation that also needed to be completed in the process of hashing a block. The practice of introducing unrelated computational work into the block creation process is termed proof-of-work, POW. However, this process is considered problematic for the Agrikore blockchain for the following reasons:

- It results in long block creation time, essentially making the time it takes to confirm a transaction out of the range that allows blockchain transactions to be usable for in person commerce in a store. For instance, bitcoin blocks take about 10 minutes to write. Needless to say, there are not too many in the ecosystem that will find it feasible waiting at the store cashier's for ten minutes for their transaction to be confirmed.
- POW leads to scaling issues in time. The number of transactions that can be handled per second for bitcoins for instance is about 2.
- The POW process leads to significant energy consumption by the computers. For instance, the electricity consumption of both the Bitcoin and Ethereum networks have been noted to combined by more than several small country's total electricity consumption. The consumption process also leads to significant cooling requirement, which has also resulted in several mining operations locating to cold climates.

We do not believe this approach is sustainable nor efficient.

The Agrikore process is a modification of proof of stake. Transaction processing nodes are all trusted and approved nodes with significant deposits or contribution into the reserves. Nodes take turns in a round-robin manner in generating the next block, while all nodes validate each block written. Nodes may only create a block whose total transaction value is less than the stake of the miner's account in the reserves:

$$\sum_T \sum \left| \Delta V_m \right| < \sigma . V_{MINER} . \tag{8}$$

Besides the fact that nodes are all vetted, this ensures that nodes will not create rogue transactions since they will simply cover any such transactions with their own stake and possibly also lose processing status. As a result of these changes, the difficulty level required to create a block is set really low for the Agrikore network, and clocks are created every 10 seconds. **Thus it takes between 10 and 20 seconds to confirm a transaction on the Agrikore blockchain.**

3.1.2 No Minting Rewards to Miners

Another key deviation in the current definition compared with other blockchains, including the Ethereum blockchain is that the writing of a block does not result in minting of new value, or reward to the transaction processor, also known as the miner. So how are transactions paid for? Every transaction carries a transaction fee. The method of rewarding processing nodes with new mint, we feel is economically distortive. It might initially create an illusion of cheaper transaction costs, but that cost is simply taken from the system, and other users of the currency, in providing the rewards for growth in the system to transaction processors when actually the activity being performed is a service rather than a creative process.

3.1.3 Creative and Transformative Transactions

For simplicity, we identify two types of transactions in a theoretically isolated economic sub-domain. Those which transfer value among members of the ecosystem and those that result in completely new value that hitherto never existed in the system. In the Agrikore eco-system the transactions that are creative and results in new mint include the following:

1. Deposit of fiat currency into the reserve. This results in the creation of an equivalent amount of MULA at the current exchange rate.
2. Deposit of fiat currency into the reserve to cover a guaranteed order or to use to fund agricultural purchase orders.
3. Deposit of fiat currency into the reserve to cover an agricultural loan issued within the system.
4. Deposit of fiat currency into the reserve to cover an agricultural grant issued to cover agricultural subsidies or provide price support to farmers in the ecosystem.
5. Deposit of fiat currency or other cryptocurrency such as Ethereum that automatically results in minting of new MULAs or a special savings contract MULA deposited into the saver's account.
6. Rewards allocated to saved MULA contracts.

The first four value creation processes are described in detail in the next section. The process of releasing savings version of MULAs directly in exchange for fiat currency is described in Section 3.2.

3.2 Agrikore Supply and Monetary Policy

As introduced in Section 2.2, the process for minting new assets utilizes on monetary control procedure. The control procedure is presented in Figure 6. The procedure consists of supply function *S*(*t*) computed from a feedback control loop implementing PID components [13], and consisting of intrinsic net currency supply sources $K_s(t)$ and net currency supply reduction $K_b(t)$.

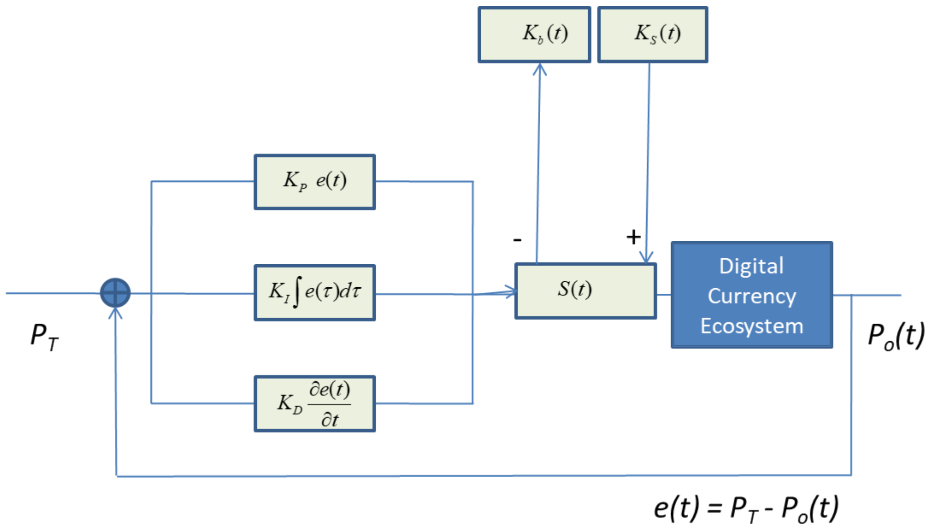


Figure 9. Feedback Loop used to Compute Release Rate for Monetary Supply Control

The procedure requires setting a target circulating supply rate, P_T. This rate is set by determining a target level of appreciation for the MULA currency exchange rate against the dollar. The target is set at 10%, but is allowed a tolerance of ±5%.

The actual circulating MULA $P_0(t)$ is computed on a 12-hourly basis and the shortfall *e(t)* is used to compute a new value of MULA *S(t)* to inject into the ecosystem. *S(t)* incorporates in its release net sources from loans, grants, and guaranteed orders that create new MULAs by injecting liquidity, as well as net removal of circulating MULAs by procedures such as saving of MULAs which locks up the MULAs from circulation into a contract saving MULA asset described in section 3. The overall algorithm for implementing this procedure in presented below.

Algorithm I. Implementation of the Feedback Loop Supply Control

1. Linearly increment the target circulating MULA, P_T
2. Compute the actual circulating MULA, $P_0(t)$, as a moving summation of all transaction values over the prior *M* hours
3. Save prior value of as $e_o(t)$.
4. Compute the current value of *e(t)*
5. Compute the proportional additional supply $K_p e(t)$
6. Compute the backward differential additional supply $K_D\left(e(t) - e_o(t)\right)$
7. Compute the integral additional supply using the prior 12*L* hours $K_I \sum_{12L} e(t)$

8. Compute the target supply $S(t)$ over the next 12 hours.

The values for K_P, K_D, K_I, M, and L will be calibrated to ensure the intended controlled currency stability is obtained once the network has a reasonable amount of traffic. Note that the monetary control procedure here remains a computed deterministic and trustless process compared to activist or arbitrarily determined, opaque, or activist control usually exerted by central authority of fiat currency authorities and regimes.

3.3 Savings Contract in the Agrikore Ecosystem

A monetary ecosystem is incomplete without a savings strategy within the system. Savers provide liquidity within the system for enterprising entities; in this case mostly famers; to produce. The system would need to provide some incentive for savings. This does require some balance as well. A system where savings is too incentivized typically results in one where many hold and do not spend and the system does not grow.

Several blockchains have introduced the concept of savings by allowing account owners to stake (lock) balances in their wallet for some length of time in exchange for some of the new mining rewards. An example is the Dash network [5]. And some blockchains are essentially completely savings mechanisms because they are so deflationary that their value keeps rising and significant enterprise or spending as a ratio of the circulation remains low; for example bitcoins.

In Agrikore, we introduce a completely new way of accommodating those who wish to save their assets while incentivizing them to do so. The savings is discounted off the reserve such that Cellulant, the administrative node has room off the reserve to make agricultural loans or microloans to farmers, that results in growth to the reserve on repayment with interest.

To accomplish this, we designed a contract asset, MULS, which has the same conversion rate and value as MULAs. However, bearers of this asset receive back a proportion of the overall growth of the reserve. To convert a MULA to MULS, a user simply needs to send their MULAs to the reserve and an equivalent amount of MULS is deposited into their account.

To end the savings, the user simply returns the MULS to the reserve and obtain the equivalent MULA back at the reserve rate. The reserve rate has a slight margin to dis-incentivize early withdrawal (similar to a withdrawal penalty on a fixed rate savings deposit.) **The conversion rate of MULA to MULS is set at 1.0 while the reserve rate is set at 0.9.** The longer an account holder holds their savings asset MULS and receive gains from the reserve on their savings, the larger their overall gain potentially. With this formulation there is no need to create CD terms like 3 months, 6 months, 12 months, etc. with different interest rates. These are continuously obtained intrinsically within the formulation.

3.3.1 Rewards Applied to the Savings Contract

The portion of the reserve growth that is distributed to all saved accounts is initially set at 30%. Again, with this formulation, the system settles into a natural equilibrium since if too many are saving compared to participating in commerce, the rewards get split by so many that it becomes less attractive and results in more of the savings returning into circulation.

A question that might arise is why is such a high percentage of gains in the reserve allocated to savings accounts? Firstly, savings in the ecosystem provides a stable base that helps reduce fluctuation in value and enhances the network. Secondly, we would love to enhance the network effect [13] of the currency by ensuring it attracts participation from outside the ecosystem, and the availability of savings mechanisms contributes to that. Thirdly, savings actually provides resources that the reserves can loan back to producers within the ecosystem. **When compared to savings instruments in fiat currency; some of which have negative interest rates when the currency's inflation rate is considered; the expectation is that the MULA and MULS will be found extremely attractive.**

3.3.2 Bounds on the Savings Contract

There are a few bounds placed on the savings assets. Firstly, the maximum annual rate of rewards that can be allocated to a savings asset or reserved amount is upper bounded at 20% per annum. Secondly, rewards begin to get allocated to saved assets after 7 days to prevent volatile hops into savings and back when windfall into the reserves are anticipated. Section 5 contains numerical simulations that demonstrate the application of these principles. The savings contract algorithm is presented below, and is executed once daily at 7am UTC.

Suppose the value of the reserve on a day, *l*, can be computed as the value of all currency in the reserve, represented in Figure 5, less the value of all circulating MULA and derived MULA assets, represented by MUL*. This is known as the **reserve margin**, and computed as:

$$\Gamma_l = \sum_{reserve} V_{All\ Currencies} - \sum_{circulating} V_{MULA} - \sum_{circulating} V_{MUL^*} \tag{9}$$

Then the increase in the reserve the next day, *l* +1 will be:

$$\Delta\Gamma_{l+1} = \Gamma_{l+1} - \Gamma_l \tag{10}$$

Algorithm 2. Implementation of the Savings Contract Rewards

1. Compute the addition to the reserves $\Delta\Gamma_{l+1}$
2. Compute the portion of the increase that gets allocated to the savings contract (30%)
3. Compute the value of all MULS in savings contracts older than 7 days, $\sum V_{MULS}$
4. The unit savings rewards will be $R_w = 0.3\Delta\Gamma_{l+1} / \sum V_{MULS}$

For each Wallet Containing Saved MULS assets:

5. If the MULS in the wallet have been saved for greater than 7 days:
    a. Compute the reward for the contract as $R_w V_{MULS}$
    b. Compute the daily maximum reward for the contract as $0.2V_{MULS}/365$
    c. Mint and allocate actual MULAs of the lower of (a) or (b) to the account containing the MULS.

3.3.2 Transparency of the Reserve

The balances of all currencies in the reserve will be published on the Agrikore public blockchain website. The values in the reserve will be maintained in reputable banking institution so that it would be verifiable that indeed the issued currencies are backed by the reserves that are intended to cover their liquidity. It should be noted, that it is implied, and expected that the Administrative nodes could withdraw to cover operational costs of running the network from the reserves up to the reserve margin. All such withdrawals will also be on the blockchain and can be viewed publicly.

3.4 Initial Currency Exchange (ICE)

A similar contract to that in 3.3 may also be utilized on the Ethereum network to offer an initial currency exchange, allowing members of the public to purchase, hold, and trade, a coin with symbol MUL on the Ethereum exchange. The purchase rate for MUL will be set the same as for MULAs converted into ETH. The reserve rate for MUL will be set at 0.8 the value of MULA as converted into dollars. Some of the value in the margin between the reserve and purchase rate of the released currency will be used to develop and operate the MULA network.

This is similar to an initial coin offering ICO, except that buyers of MUL are guaranteed the value of their currency can not fall below 80% of what they exchanged it for. Initially, the ICE target will be $80million worth of Ethereum. This will raise $16million in the reserve that can be used to develop the MULA network. Future gains in the MULA reserves will be moved into the Ethereum network to reward holders of MULs for participating in the ICE. The amount of all reserve gains allocated to saved MUL will be 10%. In addition, the exchange rate for MUL will increase with the value of MULA since MULA will be tied in value to MUL. However, in observing how crypto-currencies get bidded up on exchanges, it would not be surprising if the growth in value of MUL on the exchanges begins to outstrip that of e reserve rate, and consequently the MULA.

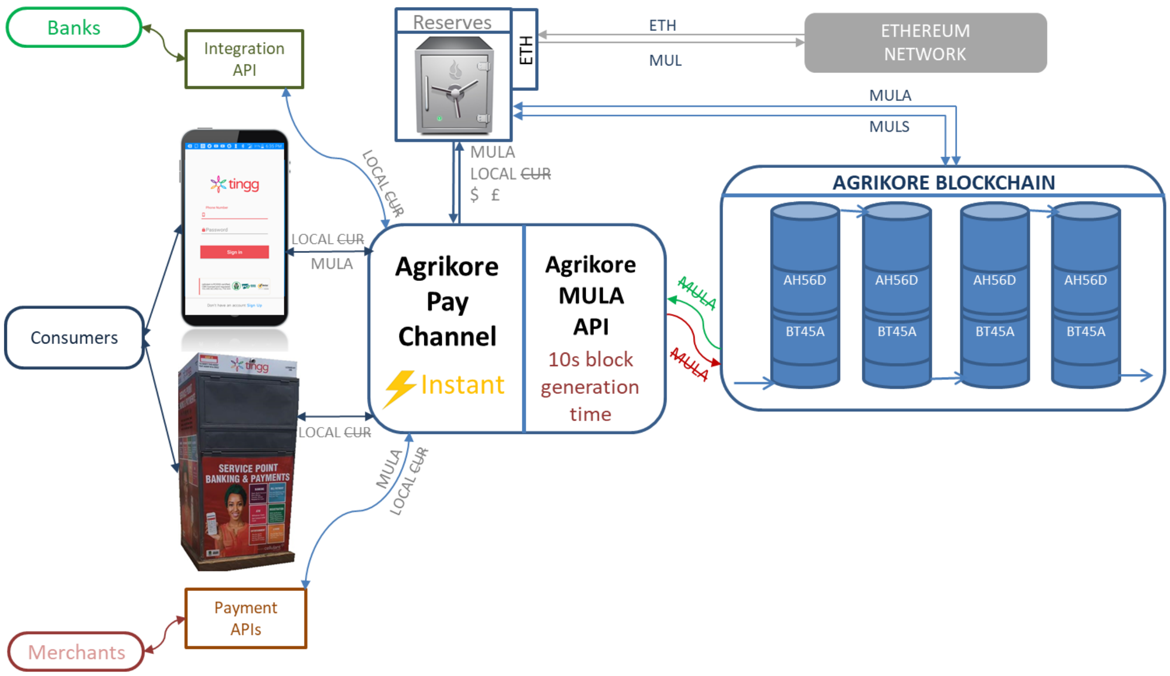


Figure 10. The interface between the MULA and Ethereum network in conducting a funding ICE

The network together with interaction with the Ethereum network in operating the ICE is shown in Figure 7. The total reserved on the Ethereum network becomes a part of overall MULA reserves, and the issued MUL just like MULS and other derived MULAs are issued assets of the ecosystem.

3.4.1 A Trustless ICE Implementation

Most financial interactions including even asset deposits requires trust in some authority. The aim behind the current ICE implementation is to encode the trust requirement or rules and terms in the contract thus removing the need to trust any third-party entity in the core manner the currency exchange works. The Ethereum solidity contract is written in a trustless manner with the following features:

1. The MUL are automatically issued to an address when ETH is sent to the contract at the published market conversion rate of ETH to MUL.
2. ETH is automatically returned to the same address when MUL are sent to the contract, at the published reserve rate.
3. The reserve rate can never be set lower than its current value at any time. This is codified in the contract.
4. Withdrawals from the reserve is hard coded such that it can never be drawn below the value of issued MUL at the reserve rate.

Note that there are two ways rewards could accrue to participants in the ICE – increases to the reserve rate which increases the value of all issued MUL, and direct issuance of ETH from the reserve to accounts holding MULs. For Agrikore MULA implementation the additions to the reserve will be typically accompanies by matching increase to the reserve rate thus increasing the value of all issued MUL. The Solidity code for MULS is presented in the Appendix.

The only aspect that is not automated or trust free in the current implementation is the transfer of 10% of the reserve gains from the ecosystem to the MUL reserve. With the establishment of atomic swaps even this aspect could be automated in future. However, every holder of MUL would be able to view and verify gains in reserves within the MULA ecosystem itself and able to confirm the expected allocations to MUL are completed. In addition, long term with all the modifications made to the MULA network to make it an improvement over prior blockchain networks, holders of MUL may well choose to transfer their MUL asset to MULS on the main MULA network, as the latter begins to grow significantly.

3.5 Allocation of Increases to the Reserve

As previously discussed in Sections 3.3 and 3.4, 30% increases to the reserve would be used to incentivize savings on the network, 10% to any initial coins purchased at the commencement of the system. The remaining will be available to administrative nodes for operational, service cost, other costs, and a portion will be used to slowly enhance the value of the currency via increases to the reserve rate.

4.0 The Smart Contracts Implemented in the Agrikore Blockchain

This section contains details of the Agriculture produce delivery contracts as implemented. There are three main objects that encompass the activities and trust contracts in the Agrikore agricultural produce purchasing system. This includes the produce or commodities, guaranteed orders, produce delivery receipts. In addition, there are seven major roles that participate in these activities. The roles are described in below:

4.1 Agrikore Produce Delivery Blockchain Roles

Commodity Buyer

Description

A commodity buyer is able to issue a guaranteed order for a produce. The order can only be placed on the blockchain by an administrative node (Cellulant).

Properties

Property Name	Variable Type	Description
Name	String	Name of the commodity, such as Corn, Rice, or Wheat
Location	Location Object	Location of commodity buyer
Score	Number	Performance score of this commodity buyer
Address	Blockchain Address	Buyer's blockchain addresses
GuaranteedOrders[]	GuaranteedOrders	Array of guaranteed orders issued by this buyer.

Methods

(1) Add Commodity Buyer
This adds a new commodity buyer to the blockchain.

Roles: Cellulant Admin

(2) Update Commodity Buyer
Updates the commodity buyer information

Roles: Cellulant Admin

Offtaker

Description

An offtaker is an intermediary buyer that is able to issue good receipts to farmers and take possession of produce. They receive payment for delivery of the produce to the commodity buyer or to the market. An Offtaker can only be set up on the blockchain by an administrative node (Cellulant).

Properties

Property Name	Variable Type	Description
Name	String	Name of the commodity, such as Corn, Rice, or Wheat
Location	Location Object	Location of offtaker
Score	Number	Performance score of this commodity buyer
ReceiptLimit	MULA Currency	Limit in value of all open produce receipt the offtaker can issue
ReceiptTotal	MULA Currency	Value of all open produce receipt issued by the offtaker
Address	Blockchain Address	Offtaker's blockchain addresses
GuaranteedOrders[]	GuaranteedOrders	Array of guaranteed orders issued by this buyer
ProduceReceipts[]	ProduceReceipts	Array of produce receipts issued by this buyer.

Methods

(3) Add Offtaker
This adds a new Offtaker to the blockchain.

Roles: Cellulant Admin

(4) Update Offtaker Information
Updates the Offtaker's information.

Roles: Cellulant Admin

Produce Certifier

Description

The produce certifier is an account on the blockchain designated as able to certify the grade of delivered produce.

Properties

Property Name	Variable Type	Description
Name	String	Name of the certifier
Score	Number	Performance of the produce certifier
Address	Blockchain Address	Certifier's blockchain addresses

Commodity	**Commodity Object**	Commodity the certifier is qualified to appraise.
Location	Location Object	Location of produce certifier

Methods

(1) Add Produce Certifier
Add the privilege to certifier a certain commodity to a blockchain account.

Roles: Cellulant Admin

(2) Remove Produce Certifier
Remove the privilege to certifier a certain commodity to a blockchain account.

Roles: Cellulant Admin

Farmer/VillageAggregator

Description

The farmer produces and delivers produce to an offtaker, market buyer, or commodity buyer and receives a produce receipt. The farmer is able to cash the produce receipt at any time based on the perfect liquidity the Agrikore blockchain guarantees for farm produce. The value of the receipt is discounted the earlier the farmer cashes the receipt along the supply chain.

Properties

Property Name	Variable Type	Description
Name	String	Name of the certifier
Score	Number	Performance of the farmer in the ecosystem.
Address	Blockchain Address	Farmer's blockchain addresses.
Loans	Loan Object	Loans taken by the farmer
LoanBalance	MULA Currency	Balance of all outstanding loans
Location	Location Object	Location of farmer

Methods

(1) Add Produce Certifier
Add the privilege to certifier a certain commodity to a blockchain account.

Roles: Cellulant Admin

(2) Remove Produce Certifier
Remove the privilege to certifier a certain commodity to a blockchain account.

Roles: Cellulant Admin

There are two additional roles which will be introduced here, but feature in the Agricultural Financing section of this document. They are Agro-Dealer and Loan Originator roles.

Agro-Dealer

Description

An agro-dealer is a supplier of farm input. This includes equipment such as tractors, tillers, irrigation equipment, or agricultural supplies such as seeds and fertilizers. Technically, any merchant on the blockchain can receive payments in MULA, the blockchain currencies. No special role or privilege is needed to participate in commerce in the ecosystem. The need for this role comes about due to special contract assets that are described in Section 4, that can only be processed by agro-dealers.

Properties

Property Name	Variable Type	Description
Name	String	Name of the certifier
Score	Number	Performance of the produce certifier
Address	Blockchain Address	Agro-Dealer's blockchain addresses.
Location	Location Object	Location of agro-dealer

Methods

(1) Add Agro-Dealer
Add the privilege to certifier a certain commodity to a blockchain account.

Roles: Cellulant Admin

(2) Remove Agro-Dealer
Remove the privilege to certifier a certain commodity to a blockchain account.

Roles: Cellulant Admin

Loan Originator

Description

A loan originator set up within the system is able to issue certain types of Agricultural loans that the blockchain system of contracts helps to manage. These loans are covered in Section 4.

Properties

Property Name	Variable Type	Description
Name	String	Name of the loan originator
Address	Blockchain Address	Loan originator's blockchain addresses.
Location	Location Object	Location of loan originator

Methods

(1) Add Loan Originator
Add the privilege to issue or have loans issued on the blockchain.

Roles: Cellulant Admin

(2) Remove Loan Originator
Remove the privilege to issue or have issued loans to the blockchain.

Roles: Cellulant Admin

Grant Originator

Description

A grant originator set up within the system is able to issue certain types of Agricultural grants or subsidies that the blockchain system of contracts helps to manage. These loans are covered in Section 4.

Properties

Property Name	Variable Type	Description
Name	String	Name of the grant originator
Address	Blockchain Address	Grant originator's blockchain addresses.
Location	Location Object	Location of Grant originator

Methods

(3) Add Grant Originator
Add the privilege to issue or have grants or subsidies issued on the blockchain.

Roles: Cellulant Admin

(4) Remove Grant Originator
Remove the privilege to issue or have issued grants or subsidies to the blockchain.

Roles: Cellulant Admin

The roles have properties that include blockchain addresses as well as location. Blockchain addresses are intrinsic properties of the blockchain. The two other objects including the produce or commodity and location descriptor are presented below.

Produce/Commodities Object

Description

Agricultural produce and commodities are at the heart of the Agrikore agricultural produce purchasing ecosystem. The ecosystem is made up of farmers who grow and sell the products, village aggregators and off-takers who initially buy the products, and corporate buyers or market buyers that are the final destination of the produce.

Commodities are basically defined by name, units, and price; such that it is possible to have two different pricing for several measurement units of the same commodity. For instance, it is possible to define a price for rice per bag and per tonne on the blockchain. A fourth identifier is location such that it is possible to define a wholesale price for rice at one location that is different from another.

Properties

Property Name	Variable Type	Description
Name	String	Name of the commodity, such as Corn, Rice, or Wheat
Units	String	Units of the commodity, such as bags, gallons or liters
MarketPrice	MULA currency	Current market price of the commodity
WholesalePrice	MULA currecncy	Wholesale price of the commodity
DiscountedPrice	MULA currency	Cash Price on delivery to the warehouse
Term	MULA currency	Number of days before the commodity price matures to the *WholesalePrice*
Location	**Location Object**	*Location at which this price holds*

Methods

(5) Add Commodity
This adds a new commodity to the blockchain.

Roles: Cellulant Admin

(6) Update Commodity Price
Updates the commodity price

Roles: Cellulant Admin

Location Object

Description

Add a location object. Location indexes are attached to Commodity price object definition so that the right price is automatically attached to deliveries. Later, geolocation capabilities would be added to the blockchain so that the location is automatically set rather than being selected in the application when setting up the Guaranteed Order.

Properties

Property Name	Variable Type	Description
LocationName	String	Location Name
Coordinates	String	Geolocation coordinates
Country	String	Country
State	String	State or region
Locality	String	Local government area or County

Methods

(1) Add Location
Add a new location.

Roles: Cellulant Admin

4.2 Produce Delivery Pathways

The system is developed for the following agricultural produce pathways:

(1) **Guaranteed Order**

In this path way a large commodity buyer negotiates an order for produce at a given price with the Administrative node (Cellulant). The commodity buyer places a deposit in the reserve, and is then able to receive produce and issue guaranteed produce receipt for the produce. The Agrikore reserve provides perfect liquidity for the receipts and provides producers guarantee of cash for their produce at any time.

Depending on their score, the commodity buyer may e allowed credit or financing through the system whereby they can issue receipts for produce in excess of their deposits. The financing is covered by the reserve or participating banks who make interest on the line of credit. The commodity buyer could also be limited to no line of credit by the blockchain requiring additional deposits into reserve as their deliveries approach their total reserve deposits. In addition, the amount of liquidity financed for the commodity buyer could also be further guaranteed by an insurance bond.

The path from the farm to the commodity buyer is depicted below:

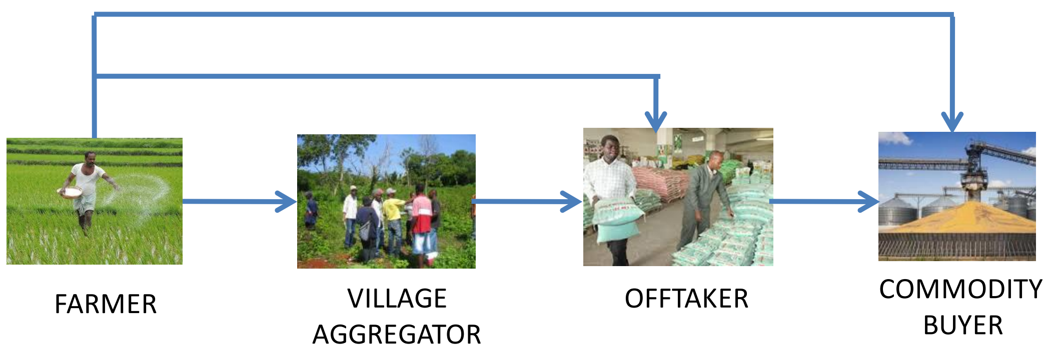


Figure 1. From Farm to Factory

The spread and incentive in the system for each of the participating roles is illustrated in the table below.

Role	Income/Incentive	Comments
Farmer	• Commodity.DiscountPrice if produce is delivered to VillageAggregator or Offtaker for cash • Commodity.WholesalePrice at any time the produce is sold • GuaranteedOrder.GuaranteedPrice if produce is delivered to Commodity Buyer	The discount cash applies for cash at delivery since farmer is being financed by the system while the produce is not yet delivered to the Commodity Buyer.

Village Aggregator	• Difference between Commodity.WholesalePrice and Commodity.DiscountPrice	Village Aggregator pays farmer in cash to collect produce. Adds value in terms of transporting aggregated produce. Finances farmer until the produce is delivered to the offtaker and then the Commodity Buyer. Village Aggregator can also cash out earlier in the cycle for less than the full spread
Offtaker	• Difference between Guaranteed Order price and Commodity.WholesalePrice assuming all receipts are not cashed until term. Spread increases the earlier holder of the receipt cashes it.	Offtaker finances producer/farmer until produce is delivered to the Commodity Buyer. Adds value in terms of storing and transporting produce.
Commodity Buyer	• Able to master their supply chain by having guaranteed product and guaranteed price.	The Agrikore system provides perfect liquidity for the system allowing suppliers to deliver produce in a trustless manner with the knowledge that payment at agreed prices are guaranteed.

- *Note that Cash in terms of local currency or MULA is used synonymously in the above table*
- *Note also that in this new high liquidity environment, there is no application-level difference between a farmer and a village aggregator. A farmer cashing out his produce at Wholesale price could also have purchased (and aggregated) that produce from another farmer paying them the Discount price. (Thus the Village Aggregator function becomes an Operational distinction that exists outside of the blockchain system.)*

(2) **Offtaker Direct Purchase**

In this path way, the farmer or village aggregator sells or stores the produce to the offtaker. This path allows the farmer to take advantage of market conditions and sells when prices are high and hole when they are lower.

In the case of a sale, the offtaker then sells the produce to a market buyer, or even a guaranteed order if one is exists at the time of going to market. In the case of storage, the offtaker charges the farmer storage for the warehousing or storing the produce. The storage fees may accrue since the offtaker has custody of the produce as collateral, and may be settled from the proceeds of the sale of the produce.

The path from the farm to the large market buyer is depicted below:

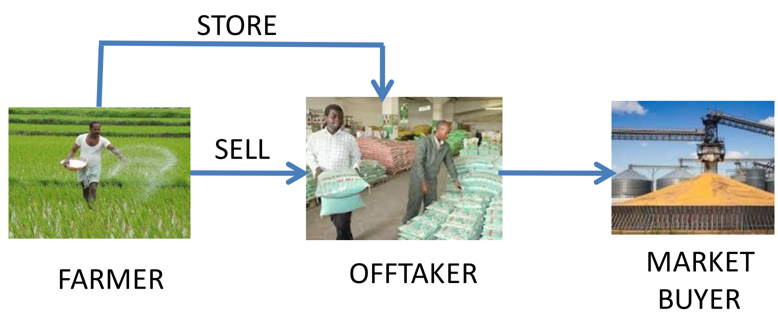


Figure 2. From Farm to Market

The role of the blockchain in this path is to provide liquidity to the system such that the farmer can cash out their produce receipt at any time. Secondly, the blockchain receives commodity price feeds such that the farmer may benefit from commodity value appreciation.

Role	Income/Incentive	Comments
Farmer	• Prevailing Commodity.DiscountPrice if produce is delivered to VillageAggregator or Offtaker for cash • Commodity.WholesalePrice variable with prevailing commodity price at any time the produce is sold	The discount cash applies for cash at delivery since farmer is being financed by the system while the produce is not yet delivered to the Commodity Buyer. Farmer can more easily participate in the gains in commodity value and utilize market trends to their benefit.
Village Aggregator	• Difference between prevailing Commodity.WholesalePrice and the strike (at time farmer was paid cash) Commodity.DiscountPrice	Village Aggregator pays farmer in cash to collect produce. Adds value in terms of transporting aggregated produce. Finances farmer until the produce is delivered to the offtaker and then the Commodity Buyer. Village Aggregator can also cash out earlier in the cycle for less than the full spread.
Offtaker	• Difference between Commodity.MarketPrice and prevailing Commodity.WholesalePrice assuming all receipts are not cashed until term. Spread increases the earlier holder of the receipt cashes it. • Storage fees in the case where the produce is stored beyond a	Offtaker finances producer/farmer until produce is delivered to the Commodity Buyer. Adds value in terms of storing and transporting produce.

Market Buyer	• Able to view history and dates of all produce purchased and received.	The Agrikore system enables an unprecedented level of transparency for produce recorded on the blockchain. Like rings on a tree, once a produce is recorded in the blockchain at any step from warehouse all the way to the market the record becomes immutable as well as available for public consumption..

- *Note that Cash in terms of local currency or MULA is used synonymously in the above table*
- *There is no blockchain level distinction between the farmer and aggregator role. At the blockchain level, the aggregator is simply a farmer with enough liquidity to buy off the produce of other farmers and sell them at the next level with the spread from financing the activity. Any farmer on the blockchain has the access technically to perform that role. Subsequently, there is no separate village aggregator role on the blockchain.*

4.3 Produce Delivery Contracts

Guaranteed Order Object (MULAB)

Description

A guaranteed order is typically placed by a commodities buyer. Once placed, the order drives significant agricultural and economic activity in the ecosystem to fulfil it, proportional to the size of the order.

A guaranteed order can only be placed with an Administrative node of the blockchain. The guaranteed order is a smart contract that allows several participants in the ecosystem guarantees to take certain actions. To ensure the system performs as intended, the aim of the Agrikore system is to make this contract as Ricardian as possible. To that end, there are three ideal scenario for the order:

1. The commodity buyer places an equivalent amount to cover the order into the blockchain reserve. That amount then covers all deliveries made to fulfill the order. (The reserve system will be discussed separately and is a core feature of the Agrikore blockchain system.)
2. The commodity buyer places a deposit to cover a portion of the order in the system, and continues to top up or make deposits as the order is being fulfilled and the order receipts cashed off the reserves.
3. The commodity buyer places a deposit to cover a portion of the order, and also provides an insurance bond to the Administrative node in the event that they do not cover the rest of the order once it is delivered. This third scenario results in interest payment by the commodity buyer for the portion of produce delivery that represents credit to the buyer from the system.

Once a guaranteed order has been set up in the system, the bearer is able to issue receipts for agricultural produce deliveries for the type of produce covered by the order. Recipients of the receipt will be farmers and village aggregators. Recipients may cash off the value of the receipts from the blockchain at any time.

A core value of the blockchain is the liquidity it guarantees to agricultural producers to provide them value for their goods at any time following a set of guaranteed parameters in the ecosystem.

Properties

Property Name	Variable Type	Description
Beneficiary	Blockchain Address	Commodity buyer's account
DateCreated	Date and Time	Guaranteed order date
DateExpiring	Date and Time	Expiry date of order
GuaranteedPrice	MULA Currency	Guaranteed order price
GuaranteedQuantity	Number	Guaranteed order quantity
TotalValue	MULA Currency	Value of order
ValueDeposited	MULA Currency	Value deposited to cover order
ValueBonded	MULA Currency	Value of any bond covering order
AllowedCreditPercent	Percentage	Percentage by which total receipt can exceed deposit
ProduceReceiptCommission	Percentage	Commission on produce receipts redeemed off Reserves

InterestRate	Percentage	**Daily**Interest rate applied on the difference between receipts and deposit
ValueOfIssuedReceipts	MULA Currency	Total value of all issued receipts
TotalRedeemedValue	MULA Currency	Total value of the receipts cashed off the reserves
Type	**Commodity object**	Commodity type covered by this order
Status	String	Issued, Suspended, Revoked, Expired
ComputedInterest	MULA Currency	Total interest computed
PaidInterest	MULA Currency	Total interest paid
ComputedCommission	MULA Currency	Total commission computed
PaidCommission	MULA Currency	Total commission paid
ListofReceipts	List	An array of all receipts issued

Methods

(1) Add Guaranteed Oder

This adds a new guaranteed order to the blockchain. This method also adds the new guaranteed order to the Commodity Buyer's list of guaranteed orders.

Roles: Cellulant Admin

(2) Add Offtaker

Add an offtaker as able to offtake produce based on this guaranteed order, from among the offtaker objects

Roles: Cellulant Admin

(3) Suspend Guaranteed Order

Suspend a guaranteed order. A suspended order will be unable to issue new receipts to farmers and aggregators for produce on the blockchain.

A guaranteed order can also be automatically fall into suspension if the value of receipt issued over the amount deposited exceeds its AllowedCreditPercent value.

Roles: Cellulant Admin

(4) UnSuspend a Guaranteed Order

Reverse the suspension of a guaranteed order. Once reversed the beneficiary can again resume issuing receipts that are guaranteed on the blockchain.

Roles: Cellulant Admin

(5) Revoke Guaranteed Order

Revoke a guaranteed order. A revoked order will be unable to issue new receipts to farmers and aggregators for produce on the blockchain. A revoked order can not be revoked. A new guaranteed order would need to be set up if the beneficiary must regain the privilege of issuing guaranteed receipts off the blockchain.

Roles: Cellulant Admin

(6) Compute Interest

Computes the interest on a guaranteed order.

$$C_r = V - D$$

$$Interest = C_r \times \mathrm{r}$$

Increment *ComputedInterest*property with *Interest* computed

where

V = *TotalRedeemedValue*

D = *ValueDeposited*

r =*InterestRate*

Interest is computed on any guaranteed order in Issued, Suspended, or Expired status.

Roles: System. The system automatically completes this at 7am GMT each day.

(7) Pay Interest

Pay interest on a guaranteed order to the reserves.

Increment *PaidInterest*property with *Interest* computed

Roles: Beneficiary (Commodity Buyer, Offtaker)

(8) Take Interest

Take interest payment on a guaranteed order from a beneficiary's MULA account and increment the interest payment credited to the guaranteed order contract.

Increment *PaidInterest*property with *Interest* computed

Roles: Cellulant Admin

(9) Expire Guaranteed Order

Expires a guaranteed order in an orderly fashion when its expiry date is reached. All issued receipts are coverted to cash (MULA currency) by redeeming their cash value from the reserve. This is an automated process where the blockchain checks the expiry date of any guaranteed order and applies this function when on the expiry date of an order.

If (DateExpiring<= Today)

For every receipt in ListofReceipts[]

Perform RedeemProduceReceipt (see section on Produce Receipt)

Note that ComputeInterest continues to be called on an expired order for any difference in TotalRedeemed and TotalDeposited as before.

Roles: System. The system automatically completes this at 7am GMT each day.

(10)Make Reserve Deposit

Deposit funds into the reserves to cover potential receipts

Increment ValueDeposited property with *Interest* computed

This has the effect of reducing potential interests on issued receipts. However, already computed interests in *ComputedInterest* remains the same and needs to be paid using the [Pay Interest] function.

Roles: Beneficiary (Commodity Buyer, Offtaker)

Produce Receipt Object (MULAC)

Description

Any user (beneficiary) that has been set up with a guaranteed order object is able to issue produce receipt to a bearer of agricultural produce of the type specified in the guaranteed order.

The bearer is able to trust in such receipts because it gets recorded on the blockchain and reflects in their mobile device as credit. Bearers are made to understand that a produce receipt legitimately issued on the blockchain and reflecting in their Tingg App is as good as cash and can be redeemed off the Agrikore blockchain reserve.

Properties

Property Name	Variable Type	Description
SourceGuaranteedOrder	**Guaranteed Order**	Commodity buyer's account
IssuedBy	Blockchain address	Issuer's Bearer's blockchain address
IssuedTo	Blockchain address	Bearer's blockchain address
CurrentHolder	Blockchain address	Current bearer at any time (receipts are transferable)
PendingHolder	Blockchain address	Bearer to transfer to during a transfer process

DateIssued	Date and Time	Percentage by which total receipt can exceed deposit
Commodity	**Commodity Object**	**Agricultural produce object**
CommodityUnit	Units	Total value of all issued receipts
Quantity	Number	Total value of the receipts cashed off the reserves
ValueAtIssue	MULA currency	Value at issue (Commodity.DiscountPrice)
StrikePrice	MULA currency	Wholesale price on delivery (Commodity.WholesalePrice)
ValueNow	MULA currency	Computed value now
Type	String	Currently not used. This will in future allow Produce receipts to be indexed to commodity prices
Status	String	Initiated, Issued, Redeemed, Pending, Expired
Certifier	Blockchain address	Certifier blockchain address
ProduceReceived	Blockchain address	Receiver blockchain address
Grade	String	Produce grade (**Discussion**)

Methods

(1) Add Produce Receipt

Add a new produce receipt to the blockchain. Include the commodity from the selected commodity objects set up in the blockchain.

Set *ValueAtIssue*= *Commodity*.DiscountedPrice
Set *Status*= "Initiated"
Check SourceGuaranteedOrder.TotalValue will not be exceeded (Path 1)
Check *SourceGuaranteedOrder* receipts in credit will not exceed *AllowedCreditPercent* (Path 1)
Increment the Offtaker's list of Produce Receipts
Increment the Offtaker's value of ProduceReceipts with *ValueAtIssue*

A new produce receipt initially has a status of "Initiated" and has no value yet on the blockchain. Initiated receipts that are not yet certified do not yet show up in the bearer's Ting application until it is certified (so bearers know they have no receipt yet until it is certified.)

Roles: Commodity Buyer/Offtaker

(2) Certify Produce Receipt

A produce certifier accredited within the system by an Administrative node inspects produce and certifies it on the blockchain.

Change *Status* to "Issued"
Increment *SourceGuaranteedOrder*'s*ValueOfIssuedReceipts*
Add to*ListofReceipts*for*SourceGuaranteedOrder*

Roles: Produce Certifier

(3) Deliver Produce

On delivery of the produce, the commodity buyer inspects and accepts delivery. Offtaker or delivery agent must receive indication on the blockchain that buyer has accepted delivery. Then the blockchain system automatically cashes out the produce receipt at Commodity.WholesalePrice, and the Offtaker or current owner of the produce receives the difference between the Commodity.WholeSalePrice and GuaranteedOrder price.

Change *Status* to "Redeemed"
Credit Offtake GuaranteedPrice - Commodity.WholeSalePrice
Perform the Redeem Produce Receipt activity (Method 7)

Roles: Commodity Buyer

(4) Recompute Produce Receipt Value

For guaranteed orders computes current value of the produce receipt.

$$V = Vo + (\text{Today} - \mathit{DateIssued}) \times (P - V_o)/T_m$$

where

V = *ValueNow*

Vo = *ValueAtIssue*

P =*Commodity.WholeSalePrice*

T_m = *Commodity.Term*

For Path 1 Produce Receipt, P = strike price.

For Path 2 Produce Receipt, P = prevailing or current price based on commodity market prices.

Roles: System. The system automatically computes this at 7am GMT each day.

(5) Initiate Produce Receipt Transfer

Initiate the transfer of a produce receipt to a different bearer. For instance, a farmer bearing a produce receipt can initiate its transfer to an Agent. The value in MULA paid by the new bearer is the ValueNow. The upside to its recipient is that the value of the receipt continues to increase until it reaches its term value. The value to the initial bearer is that they get cash immediately.

PendingHolder= New Bearer Address

Roles: Bearer of Produce Receipt, which is any blockchain account.

(6) Accept Produce Receipt Transfer

Accept ownership of a Produce Receipt transferred into account. Only the account indicated in *PendingHolder* a can accept it.

CurrentHolder = *New Bearer Address*

Roles: Produce Receipt *PendingHolder*.

(7) Admin Transfer Produce Receipt

Transfer a Produce Receipt from one account to another. This is intended to be an administrative function in case of loss of access to account or such situation.

CurrentHolder = *New Bearer Address*

Roles: Cellulant Admin.

(8) Redeem Produce Receipt from Reserve

Redeem a produce receipt from reserve. The bearer of the receipt can cash it off the reserve at any time thus providing perfect liquidity for agricultural production activities. This is completed at the current value of the receipt.

Credit*CurrentHolder* with *ValueNow*
Increment*SourceGuaranteedOrder*'sTotalRedeemedValue with*ValueNow*
Increment*SourceGuaranteedOrder*'s ComputedCommission
with*ValueNow**ProduceReceiptCommssion*

Roles: Bearer of Produce Receipt, which is any blockchain account.

Incentives and Vibrancy of the Agrikore Reserve

The *ProduceReceiptCommission* is the variable that ensures that the Agricultural Produce delivery process enabled by the Agrikore blockchain results in an increase to the Reserves, and hence the ecosystem. Every ProduceReceipt that is cashed off the reserve results in a debit to the Commodity Buyer for the convenience and financing inherently provided to them in using the system. Note that the *ProduceReceiptCommission* is a property of the guaranteed order so is negotiated situationally with each guaranteed order depending on several external factors that may additionally be a part of the service, such as logistics, warehousing, transportation etc.

The *InterestRate* variable also negotiated with each guaranteed order is the second way the Reserve increases from the benefits provided by the system in enhancing produce delivery. Where the Commodity Buyer negotiates and allows the Reserve to directly finance produce delivery, the interest is automatically computed and accumulated in the ComputedInterest variable of the asset.

4.4 Agricultural Produce Storage Contracts

Storage is a key linchpin in the agricultural production and delivery process. As detailed in Section 1 and 2, the issues addressed by Agrikore ensure that firstly information on storage facilities for different types of produce is available to farmers, secondly that contract instruments are available within their financial service infrastructure to ensure trust and certainty is established between farmers and storage provider. This allows this storage provider networks to flourish and develop and build more facilities, and reduces the risks in financing storage infrastructure for loan providers. The storage contract allows provisions for auto-payment of fees and facility rental, ties the produce to the contract thus securitizing it, and allows connectivity of all involved actors with their other activities and reputation within the system.

Storage Contract (MULT)

Description

The storage contract represents an agreement between a farmer and an owner of a storage facility. Any role is able to negotiate a storage contract with other roles. The key aspect of the storage contract is that it is backed by a commodity. And proceeds from selling the commodity is used to pay for any amount owning on the storage.

Properties

Property Name	Variable Type	Description
IssuedBy	Blockchain address	Issuer's Bearer's blockchain address
IssuedTo	Blockchain address	Bearer's blockchain address
PendingHolder	Blockchain address	Bearer to transfer to during a transfer process
DateIssued	Date and Time	Percentage by which total receipt can exceed deposit
Commodity	**Commodity Object**	**Agricultural produce object**
CommodityUnit	Units	Total value of all issued receipts
Quantity	Number	Total value of the receipts cashed off the reserves
ValueAtIssue	MULA currency	Value at issue (Commodity.DiscountPrice)
ValueNow	MULA currency	Computed value now
DailyRate	MULA currency	Daily storage rate
AutoPayments	Yes/No	Automatically withdraw payments from beneficiary
StorageFeesAccrued	MULA currency	Storage fees accrued
StorageFeesPaid	MULA currency	Storage fees paid
AccruedPenalty	MULA Currency	Total accrued penalty
PaidPenalty	MULACurrency	Total penalty paid
LatePaymentPenalty	MULA Currency	Late payment penalty
Status	String	Active, Outstanding, Defaulted, Expired
Certifier	Blockchain address	Certifier blockchain address
Grade	String	Produce grade (**Discussion**)

(1) Initiate Storage Contract

Create a new storage contract. Indicate the owner of the produce (IssuedTo)

Roles: Any Role (Owner of the storage facility.)

(2) Accept Storage Contract

Accept an Initiated storage contract. Only the IssuedTo account may accept an initiated contract.

Roles: Any Role (Owner of the produce.)

(2) Recompute Storage Produce Value

For storage contract, compute the current value of produce in the storage, based on the Commodity's current wholesale price.

Roles: System. The system automatically computes this at 7am GMT each day.

(3) Initiate Storage Contract Transfer

Initiate the transfer of a storage contract to a different bearer.

PendingHolder= New Bearer Address

Roles: Bearer of the Storage Contract, which is any blockchain account.

(4) Accept Storage Contract Transfer

Accept ownership of a Storage Contracttransferred into account. Only the account indicated in *PendingHolder* a can accept it.

CurrentHolder = New Bearer Address

Roles: Storage Contract *PendingHolder*.

(5) Admin Transfer Storage Contract

Transfer a Storage Contract from one account to another. This is intended to be an administrative function in case of loss of access to account or such situation, **or where the owner has defaulted**.

CurrentHolder = New Bearer Address

Roles: Cellulant Admin.

(3) AutoPay Storage Fees

Pay storage fees on a contract set as AutoPayment by automatically withdrawing the funds from the beneficiary's account to the storage IssuedBy account.

Roles: System. The system automatically completes this check sweep for all loans at 7am GMT each day.

(4) Make Storage Fees Payment
Pay storage fees on a storage contract. This function as will all user facing functions will be available to storage contract bearers via the Tingg App.

Add payment penalty if payment is late

Increment *AccruedPenalty* property with *LatePaymentPenalty* computed

Roles: Beneficiary (Commodity Buyer, Offtaker, Farmer)

(5) Take Interest
Take interest payment on a loan from a beneficiary's MULA account and increment the interest payment and principal payment credited to the loan contract.

Add payment penalty if payment is late

Increment *AccruedPenalty* property with *LatePaymentPenalty* computed

Roles: Cellulant Admin

(6) Make Penalty Payment
Similar to make payment except that the amount increments *PaidPenalty*

Roles: Bearer of the Storage Contract, which is any blockchain account.

(7) Take Penalty Payment
Similar to make payment except that the amount increments *PaidPenalty*

Roles: Cellulant Admin.

4.5 Agricultural Production Financing

The prior sections already introduced significant amount of financing and liquidity into the system. For instance, produce deliveries are guaranteed and financed through the reserve such that farmers are able to deliver produce in a trustless manner. In addition, commodity buyers are able to make arrangement to finance their guaranteed order through the reserve system. These are intrinsic financing of agricultural production within the system enabled by the Agrikore blockchain. Note that some of the intrinsic liquidity provided through the reserve system could be provided by banks, members of the public that chose to purchase MULAs and hold the currency as savings (ICO), subsidies and agricultural grants from governments and non-governmental organizations, as well as direct financing by the administrative nodes (Cellulant).

In addition to the intrinsic financing of produce deliveries, the Agrikore blockchain also makes provision for direct financing of agricultural production through participating banks and financial institutions. The following types of agricultural loans are contracts directly supported within the system and devised to be as trustless as possible:

- Loans to Village Aggregators to collect farm produce and pay farmers the discount cash value of the produce. From the previous section that the Aggregator simply has the liquidity to buy off farm produce from other famers. Such liquidity could come in the form of a loan from the Offtaker or a participating financial institution.

- Loans to Offtaker on behalf of farmers. Offtaker uses loans to pay agro-dealer to supply farmer with production inputs, and monitors farmer through the production season. Offtaker is able to reclaim loan on harvest after the farmer delivers and sells the produce.

- Loans to farmers specifically for agricultural production.

The Agrikore blockchain built-in smart contracts are able to best implement and enforce terms of these loans in ways cash loans never could previously. This is accomplished as follows:

Agricultural productions loans are disbursed as a special currency or asset type that can only be spent in designated blockchain addresses under certain conditions. For instance, loans for fertilizer or tractor and farm equipment rentals would be created on the blockchain such that only designated agro-dealers that supply those equipment or sell those products can redeem them. This Ricardian arrangement where the code also enforces the rules potentially prevents tempting abuses possible with cash, such as a farmer that proceeds to spend agro-loans and child's wedding leaving the system without produce and in collection position.

The loan object properties and functions are described below.

Agricultural Loan Object (MULL)

Description

An agricultural loan on the blockchain has special features in that their disbursement creates a smart type of MULA that can only be monetized at specific vendors and time.

Property Name	Variable Type	Description
Loan Originator	Loan Originator	Loan originator
Beneficiary	Blockchain Role	Commodity buyer, Offtaker, Farmer/Village Aggregator role
BeneficiaryType	List	Commodity buyer, Offtaker, Farmer/Village Aggregator
DateCreated	Date and Time	Guaranteed order date
Loan Type	List	Loan type
Term	Number	Loan term in months
Moratorium	Number	Time in months before payments begin to accrue
Principal	MULA Currency	Value of order
InterestRate	Percentage	Interest rate
PrincipalPayments	MULA Currency	Monthly principal repayment amount
InterestPayments	MULA Currency	Monthly interest payments
AutoPayments	Yes/No	Automatically withdraw payments from beneficiary
DayToPay	Integer	Day to pay (1-28)
Payments	Transaction IDs	Array list of payments
LastPaymentDate	Date and Time	Last payment date
Status	String	Pending, Issued, Matured, Default, Revoked
AccumulatedInterest	MULA Currency	Total interest computed
PaidInterest	MULA Currency	Total interest paid
PaidPrincipal	MULA Currency	Total principal paid
AccruedPenalty	MULA Currency	Total accrued penalty
PaidPenalty	MULACurrency	Total penalty paid
LatePaymentPenalty	MULA Currency	Late payment penalty
DaysBeforeDefault	Number	Number of days before loan is considered in default

The disbursement of the loan results in the creation of a special type of currency or MULM that is very similar to the blockchain's MULA but can only be converted to regular MULA or cash by an agro-dealer. Loans are not transferrable. However, the disbursed assets has all the features of cash in that it can be split or transferred between farmers and agro-dealers. However, it can only be converted to the regular currency of the blockchain (or effectively cash) by an agro-dealer.

Methods

(8) Add Agricultural Loan

This adds a new agricultural loan (MULL) to the blockchain. This method also disburses the loan into the loan beneficiary's account in the form of the conditional currency MULM.

The loan issue process goes through three steps:

1) Loan originator deposits cash (MULA) into the reserve to cover the loan *Principal*.
2) Pending. The loan is created on the blockchain on behalf of the loan originator

3) Issued. The loan is accepted on the blockchain by the beneficiary. On acceptance, the loan is disbursed (by crediting the *Principal* in MULM) into the beneficiary's address or wallet.

The amount of MULM disbursed is foundationally equal to the MULA placed in reserve. To incentivize agricultural loan activity, Agrikore variable **AgriculturalLoanRate** set slightly below 1, could be applied to the amount the Loan originator deposits in reserve in MULA to disburse the MULM. On the other hand, to grow the reserves **AgriculturalLoanRate** could be set slightly above 1. That is:

MULA Reserved = **AgriculturalLoanRate** x Principal

Roles: Cellulant Admin

(9) Compute Interest

On the loan's DayToPay, add the InterestPayments to the AccumulatedInterest on the loan .

Increment *AccumulatedInterest* property with *InterestPayments* computed

Interest is computed on any guaranteed order in Issued status.

Roles: System. The system automatically completes this at 7am GMT each day.

(10)AutoPay Interest

Pay interest on a loan set as AutoPayment by automatically withdrawing the funds from the beneficiary's account to the loan originator's account on the loan's DayToPay.

Increment *PaidInterest* property with *InterestPayments* computed
Increment *PaidPrincipal* property with *PrincipalPayments* computed
Debit Borrower Reserve's *AutoInterestCommission* (see end of section)

Roles: System. The system automatically completes this check sweep for all loans at 7am GMT each day.

(11)Make Interest Payment

Pay interest on a loan. This function as will all user facing functions will be available to loan beneficiaries via the Tingg App.

Increment *PaidInterest* property with *InterestPayments* computed
Increment *PaidPrincipal* property with *PrincipalPayments* computed
Debit Borrower Reserve's *InterestCommission* (see end of section)

Add payment penalty if payment is late

Increment *AccruedPenalty* property with *LatePaymentPenalty* computed

Roles: Beneficiary (Commodity Buyer, Offtaker, Farmer)

(12)Take Interest

Take interest payment on a loan from a beneficiary's MULA account and increment the interest payment and principal payment credited to the loan contract.

Increment *PaidInterest* property with *InterestPayments* computed
Increment *PaidPrincipal* property with *PrincipalPayments* computed
Debit Borrower Reserve's *InterestCommission* (see end of section)

Add payment penalty if payment is late

Increment *AccruedPenalty* property with *LatePaymentPenalty* computed

Roles: Cellulant Admin

(13)Make Penalty Payment

Similar to make payment except that the amount increments *PaidPenalty*

Roles: Bearer of the Storage Contract, which is any blockchain account.

(14)Take Penalty Payment

Similar to make payment except that the amount increments *PaidPenalty*

Roles: Cellulant Admin.

(15)Check Loan Status

Check all active loans on the blockchain.

a. Flag in *Default* any loan where *LastPaymentDate* exceeds *DaysBeforeDefault*
b. Flag as *Issued* any loan in default where *LastPaymentDate* is less than *DaysBeforeDefault*
c. Flag as *Mature* any loan where *Term* > *Now*.

Roles: System. The system automatically completes this check sweep for all loans at 7am GMT each day.

(16)Convert to Blockchain Cash

Convert the agricultural cash (MULM) to regular (MULA) cash on the blockchain. The conversion between MULM and MULA is at ***AgriculturalActivityRate***, which is foundationally one to one. The smart contract limiting such conversion to an Agro-Dealer simply ensures that it is spent on agricultural activity.

Conversions performed by an Agro-Dealer can be audited since all activities are visible on the blockchain. Obviously, Agro-Dealer that abuse their status (potentially washing agricultural cash into real cash without actual purchase of agricultural produce or service) risk losing that privilege or status on the blockchain.

Roles: Agro-Dealer

(17)Revoke Loan

Revoke an agricultural loan. All disbursed loan assets, MULM, in the beneficiary's account are immediately claimed and returned to Reserves. The assets are then converted into the blockchain currency and credited back to the loan originator.

Roles: Cellulant Admin

Properties of Disbursed Loan Currency Assets

Disbursed agricultural loans (MULM) are deposited in the wallet of the beneficiary and are programmed with all the same properties of cash as programmed for the blockchain's currency (MULA) but with the following restrictions and properties:

(1) They can only be transferred (used in payment) to an blockchain address owned by an agricultural role on the blockchain including an Agro-Dealer, Farmer, or Offtaker.
(2) They can be exchanged for the blockchain's currency (MULA) by an Agro-dealer.
(3) They are fungible and there is no distinction between 1 MULM received from one agricultural loan to another. They only need to be eventually spent on agricultural activity through an Agro-Dealer.
(4) Once disbursed, just like cash they can never expire even after the loan mature and is fully repaid. However, the limitations on their property remains in place until they are exchanged for cash by an Agro-Dealer.

Incentives and Vibrancy of the Agrikore Reserve

Rapid growth and sustainability of the Agrikore blockchain system will depend on agro-dealers participating, and even preferring transacting in the currency. The same is true of loan originators (financial institution). As a result, the following potential incentives are built into the above transactions:

1. The conversion rate at which a loan originator injects new cash into the system could initially be incentivized at a small premium.

   This is applied by using a loan origination purchase rate controlled by the internal blockchain property termed ***AgriculturalLoanRate*** in this section. Consequently, if that property is set at 0.99, say, then a loan originator injects a loan on $1000 into the system and disburses that amount to farmers in the equivalent MULM, they would need to inject an $990 worth of base MULAs into the reserve to cover the loans.

On the other hand if is set slightly greater than 1, the Loan Originator pays a small premium into the reserve to issue the loan in the blockchain system.

Note that, on repayment, the loan originator receives back the principal ($1,000) plus interest. That remains the incentive for any loan profiting business. Therefore, ***AgriculturalLoanRate*** property when set as an incentive, would only serve as an advertising purpose or talking point for the system than the main incentive for loan originators. The incentive is financed by the reserve so will need to be properly subjected to staging and scenario analysis prior to go live.

2. The rate at which Agro-Dealers convert what is essentially agro cash (MULMs) to regular cash (MULAs) could also enjoy a small premium. The global property that controls this is ***AgriculturalActivityRate.*** Thus, if that variable were set at 1.05, then an agro-dealer sells $100 of fertilizers to a farmer and receive $100 in MULMs, converting it to MULAs would result in an agro-dealer receiving $105 in MULAs, thus creating a slight preference for transactions in Agrikore than in the old cash system.

   Just as for the loan incentive, this incentive could be introductory as well to incentivize agro-dealers to set up their system to transact on the blockchain. And just as the loan incentive, it is also financed from the blockchain reserve and should be appropriately subjected to game-theory to properly compute what level it should be set at for maximum effectiveness.

3. The third global property ensures that the reserve continues to grow with economic activity. The variables are the ***AutoInterestCommission*** and ***InterestCommission.*** These are applied to every interest paid within the eco-system on agricultural loans. The properties is applied as a percentage value of the amount paid. (The existence of two variables allows the system to additionally incentivize borrowers that use auto interest repayment loans, which result in lower loan administration costs and rates.)

   The offset for participants of the ecosystem that choose to take loans within the system compared to traditional bank loans outside the system is that the smart contract enforcement inherent within the Agrikore blockchain system potentially reduce risks and loan administration costs, and will eventually reduce cost of the loans to loan originators. The reduce costs can then be passed on to borrowers at lower rates. This should result in an equivalent loan in the Agrikore blockchain becoming cheaper compared to a similar loan taken outside of the system.

   The ideal scenario for these set of incentives would be to set the above variables such that:

   Where ***AgriculturalLoanRate*** < 1:

Sum of all intake into reserves from (***Auto***)***InterestCommission***	>	Sum of all incentives to agro-dealers from ***AgriculturalActivityRate***	+	Sum of all incentives to loan originators

from
AgriculturalLoanRate

Where ***AgriculturalLoanRate*** > 1:

Sum of all intake into reserves from (***Auto***)***InterestCommission***	+	Sum of all incentives to loan originators from ***AgriculturalLoanRate***	>	Sum of all incentives to agro-dealers from ***AgriculturalActivityRate***

Ultimately, what will drive the growth of the Agrikore blockchain system are two factors:

- Reducing cost of loans through lower default rates, lower administration costs, increased transparency, and lower payment processing costs. This is the power that the blockchain and the more Ricardian nature of the formulation disclosed here brings to bear. If the cost of loans issued on the blockchain is lower, then lower rates can be passed on to consumers which potentially increases customers (demand) and attracts loan originators (supply.) Ultimately, it is expected that some of the savings in lower cost loans will be retained by the reserve through a ***AgriculturalLoanRate*** setting that as greater than 1.
- Convenience and ease of use is also important. If blockchain loan costs are potentially lower, it must first attract borrowers for loan originators to realize this benefit. This then results in growth of the system. The Tingg application and the Agrikore network of mobile banks that allows customers that can not easily access banking facilities participate in the financial ecosystem addresses the issue of convenience and access that potentially attracts participants into the system. Considering that the Agrikore system already has over 17 million already registered farmers provides the path to converting customers to take advantage of the agricultural financing options in the blockchain.

4. 5 Developmental and Agricultural Grants and Subsidies

Cellulant has been a leader in assisting governments in Africa disburse agricultural subsidies to farmers in rural areas for years. In the process, Cellulant has developed a physical, operational, and technological process for reaching farmers, organizing the communities, registering the farmers, and disbursing the subsidies in an effective and transparent way.

However, the process can be improved and more cost effectively administered using a blockchain infrastructure; and this aspect of Cellulant's services was part of the impetus that commenced the planning for the blockchain enabled system. The Agrikore blockchain system enhances and complements the existing process in the following ways:

1. Disbursing targeted agricultural subsidy such as fertilizers as cash grant leaves room for it to be diverted; at every stage from the top down.
2. Cellulant developed a method of disbursing the subsidy as coupons so that it can only be cashed by agro-dealers in exchange for fertilizers. Coupons are obviously superior to cash for this specific purpose. However, coupons are not easily divisible in the manner that cash is. A blockchain currency that is programmable such that it has properties of cash but able to be programmed to be conditionally cashed in specific accounts delivers a superior solution.
3. Programmed grant assets can in addition now have properties of assets in balance sheet terms in a manner that a basic coupon system does not.
4. Accounting for coupons that are printed and disbursed is not as exact as an asset on a blockchain where the amount of subsidies cashed is instantaneous and more easily monitored and administered.
5. The blockchain also allows a level of transparency not easily available from other system. Information on the blockchain can be accessed from anywhere by both the public and the sponsors of the grant in a way that is not currently possible.
6. Information on the blockchain, like rings on a tree is virtually immutable. This provides the trust to sponsors that access the information on their grant and subsidy campaigns that is not possible with third-party system that can be modified easily by its custodian.
7. Administration of grants via the blockchain is very cost effective compared to any other known method currently. A key metric for all charities and donation system is usually the cost of administration. And for many, even some of the more notable and popular charities, this number is appalling. In general, disbursement of rural aid and grants via the blockchain brings the costs down dramatically. **Cellulant is able to put the Agrikore blockchain available to not only governments, but also non-governmental organization in disbursing grants in a transparent, effective, and accountable way.**

The resulting formulation developed for agricultural grants was devised to be generally applicable for not just fertilizers, but any subsidy, discount, grant, or even charitable donation that is designed to be targeted and utilized for specific developmental or agricultural purposes.

Agricultural Grant Object (MULG)

Description

An agricultural subsidy or grant on the blockchain has special features in that their disbursement creates a smart type of MULDs and MULFs that can only be monetized at specific vendors and time. The grant itself is created on the blockchain after negotiating and injecting its equivalent value in the reserve.

Property Name	Variable Type	Description
Grant Originator	Grant Originator	Grant originator
ReservedValue	MULA Currency	Total value of MULA paid into reserve to cover the grant
DisbursementAmount	MULA Currency	Total value of the MULD, MULF, or MULA released
CashedAmount	MULA Currency	Total value of MULD or MULF cashed by recipients
GrantLimitAmount	MULA Currency	Limit of disbursed MULF or MULD that can be spent/utilized during this grant
DisbursementAddress	Blockchain Address	Blockchain address or wallet
BeneficiaryType	List of Roles	Farmer
DateCreated	Date and Time	Guaranteed order date
ExpiryDate	Date and Time	Date the grant disbursement ends, if not used
Grant Type	List	Agricultural Subsidy (MULD), Fertilizers only (MULF)
Status	String	Active, Completed, Expired, Suspended, Stopped

There are four types of agricultural subsidies currently programmed on the blockchain:

- General agricultural subsidy (MULD)
  This can be used for any agricultural equipment or input.
- Fertilizer subsidy (MULF)
  This can only be used for purchasing fertilizers
- Cash subsidy (actual MULA)
  Some grant distributors may decide to directly disburse cash as their grant. Note that the transparency properties of the blockchain still allows a superior accountability for the disbursement compared to cash. Cash subsidies can be used for price support subsidies
- Produce receipt backed cash subsidy (MULA)
  This subsidy is implemented such that it can only be disbursed to an account that holds a produce receipt (MULC). This can be used for price support subsidies just as direct cash subsidy but provide an even more accountable control for a price support subsidy programme.

The first two assets can only be converted to the regular currency of the blockchain, MULA (effectively cash,) by an agro-dealer.

Methods

(1) Add Agricultural Grant
This adds a new agricultural grant (MULG) to the blockchain. This method also disburses the new

grant asset, MULD or MULF, into the disbursement address from which it will be distributed into qualified recipient's addresses.

The grant issue process goes through three steps:

1) Grant originator deposits cash (MULA) into the reserve, *ReservedValue*, to cover the grant *GrantLimitAmount*.

2) The grant is created and in "*Active*" status. DisbursementAmount worth of MULD or MULF in MULAs is placed in the *DisbursementAddress* (wallet or account.) Note that could be several times larger than *ReservedValue* or *GrantLimitAmount* since not all recipient of the grant cash would end up using it before they expire. It is the job of the blockchain system to actually limit the usage to *GrantLimitAmount*. For certain grant types, the *DisbursementAmount* could be set the same as *GrantLimitAmount*. This is a set up decision by the Administrative node based on their knowledge of the grant type.

3) The grant manager accepts applications from *BeneficiaryType* for the grant and selects recipients and issues MULF or MULD into their account.
(Note: the *DisbursementAddress* may be an account belonging to the Grant originator if they are self administering their grant, or to an address created by the Administrative node if the grant is managed by the Agrikore node administrator as a service.)

The value of MULD or MULF disbursed is foundationally equal to the MULA placed in reserve. To incentivize agricultural loan activity, Agrikore variable **AgriculturalGrantRate** set slightly below 1, could be applied to the amount the Grant originator deposits in reserve in MULA to disburse the MULD/MULF. On the other hand, to cover the cost of administering the grant or grow the reserves **AgriculturalGrantRate** could be set slightly above 1. That is:

ReservedValue = **AgriculturalGrantRate** x *GrantLimitAmount*

Note that a separate service fee could be negotiated outside of the blockchain smart contracts to the administrative node for administering the grant, and separately paid into the administrator's account for the service.

Roles: Cellulant Admin

(2) Check Grant Status

Check all active grants on the blockchain.

a. Flag as "*Expired*" any grant where *ExpireDate* > *Now*.
b. If grant has expired, Close Grant Assets (next method below.)

Roles: System. The system automatically completes this check sweep for all grants at 7am GMT each day.

(3) Close Grant Assets

Mark all MULD or MULF associated with the grant as expired, and can no longer be converted to cash.

Roles: System, Cellulant Admin, Grant Originator.

(4) Convert to Blockchain Cash

Convert the agricultural cash (MULD) to regular (MULF) to cash on the blockchain. A grant blockchain cash is automatically converted to cash on acceptance by an Agro-Dealer (since grant cash can expire.) The conversion between MULD/MULF and MULA is at ***AgriculturalActivityRate***, which is foundationally one to one. The smart contract limiting such conversion to an Agro-Dealer simply ensures that it is spent on agricultural activity, and performs the following actions:

If Grant *CashedAmount* + (MULD/MULF) > *GrantLimitAmount*, Set Status = "*Completed*" and Close Grant Assets, or else

Increment Grant (MULD) *CashedAmount* by MULD/MULF value

Cashing of agricultural assets performed by an Agro-Dealer can be audited since all activities are visible on the blockchain. Obviously, Agro-Dealers that abuse their status (potentially washing agricultural cash into real cash without actual purchase of agricultural produce or service) risk losing that privilege or status on the blockchain.

(5) Transfer Grant

Transfer a grant from a beneficiary's account to another. This action is reserved for an administrator and is expected to be invoked in very rare cases; such as when recipient has lost access to their account. *This function is an example of an activity that is very difficult or nearly impossible to pull off consistently with coupons*.

Roles: Cellulant Admin

(6) Suspend Grant

Suspend a grant. A suspended grant will be unable to disburse assets currently in the *DisbursementAddress*. Already disbursed MULDs and MULFs in every wallet will also be indicated as suspended and can no longer be spent with an agro-dealer.

Roles: Cellulant Admin, Grant Originator

(7) UnSuspend a Grant

Reverse the suspension of a grant. Once reversed the grant and all its disbursement becomes active again on the blockchain.

Roles: Cellulant Admin

(8) Stop Grant
Discontinue a grant. A stopped grant can no longer disburse its cash to recipients. All disbursed assets not yet cashed can no longer be used. Stopping a grant is not reversible.

Roles: Cellulant Admin

Properties of Disbursed Grant Assets (MULF and MULDs)

Disbursed agricultural grants have the same properties as the base MULA (cash) except with the following additional properties and limitations:

(1) They carry identifying information on the source grant (MULG). This allows them to be accounted for, stopped, or expired when the grant ends.
(2) They can only be accepted and exchanged for the blockchain's currency (MULA) by an Agro-dealer.
(3) They are fungible and there is no distinction between 1 MULF received from one agricultural grant to another, in terms of its value and use. They only need to be eventually spent on agricultural activity through an Agro-Dealer. However, because they can expire, MULF/MULDs are not as fungible as MULAs or MULMs.

Incentives and Vibrancy of the Agrikore Reserve

The transparency and the lower cost of administering their grants, charitable distributions, or subsidies is the main incentive for using the system to a grant originator. In addition, access to existing target members of the communities such as farmers is also a very strong incentive. Subsequently, since the system has farmers registered in the system, and if the number of registered farmers continue to grow, organizations that wish to disburse development grants to farmers will find the system attractive. The incentive to farmers, on the other end, is the access to the grants and subsidies made available by grant originators through the system.

The ensure growth of the reserve, the ***AgriculturalGrantRate*** is expected to eventually be set to ensure that the reserve returns a positive margin from the net *ReservedAmount* less the *GrantLimitAmount*. In addition, the cashing procedure also pre-computes the addition of a grant amount to the *GrantLimitAmount* before cashing the asset (see Convert to Blockchain Cash method) to ensure the reserve always remains positive with respect to any grant campaign.

In other words, for a vibrant reserve, the following equation must be assured for all grant campaigns

Net ***ReservedAmount*** - ***GrantLimitAmount***	>	Sum of all incentives to agro-dealers from ***AgriculturalActivityRate***

4.6 Multi-Role Formulation

The Agrikore blockchain role formulation is a mult-role setup. This means, for instance, that an individual could be set up (in Tingg or web client) with an address, and then that address be used as the primary address while creating a farmer role for the individual. (Notice that each role's constitution includes a blockchain address in Section 4.1.) If the same individual chooses to issue loans to a fellow farmer, a loan originator role can be cumulatively added to their profile, using their blockchain address, and contracts created on the blockchain network that ensures the loan agreements in the manner described in Section 4.4.

This multi-role formulation would allow an Offtaker to additionally be set up with loan originator roles when they establish loan contracts with farmers within the ecosystem, as in the Outgrower programme.

4.7 Summary of Contract Assets

The current section of this paper has laid out a set of pair of smart contracts devised to create a nearly perfectly liquid agricultural eco-system, with nearly perfect information flow to all participants within the system.

The introduced asset pairs in the system include:

- Guaranteed and Market Orders (MULBs) balanced by Purchase Receipts (MULCs)
- Agricultural Loans (MULLs) balanced by disbursed smart Agric Cash (MULMs)
- Agricultural Grants (MULGs) balanced by disbursed smart Agric Cash (MULDs/MULFs)

5.0 Numerical Simulations and Projections

The following examples of use cases and simulations are designed to demonstrate in numerical ways the effect of the key interactions in the ecosystem.

5.1 Numerical Simulation of Key Economic Activities in the EcoSystem

Initial Injection of Liquidity by Administrative Node (Cellulant)

Purchase Rate: 10 (MULA exchange rate to $)
Reserve Rate: 11.11 (MULA exchange rate to $ /90%)

Activity	Circulating	Reserve
1. Cellulant injects $1m into the reserve.	+10,000,000MU	$1,000,000
Total needed in reserve at end of transactions, in $	$900,000	
Total in reserve at end of sample transactions, in $		$1,000,000

Simulations of Purchase of MULAs by Members of the Ecosystem

Transaction Fees: 0.5%

Activity	Circulating	Reserve
2. Several farmers, merchants, and members of the ecosystem Purchase a total of $1,000,000 worth of MULAs from the MULAs reserve balance	+10,000,000MU	$1,000,000
3. 8,000,000 worth of MULAs exchange hands over the course of the month. Transaction Fees go into the Reserves	-40,000 MUs	+40,000 MUs
Total needed in reserve at end of transactions, in $	$8,964,900	
Total in reserve at end of sample transactions, in $		$1,004,000

(Note that MULAs could be sold to members of the public by Cellulant or via new mint in exchange for fiat deposited in reserves. In the above, it is assumed that the MULA is sold from Cellulant's own reserve in exchange for the liquidity injected. The reserve margin becomes now available to Cellulant compared to when it was simply provided by the Administrative node at the start.)

Simulations of Guaranteed Order and Produce Delivery in the Ecosystem

Transaction Fees: 0.5%
Wholesale Price of Grain: $900
Guaranteed Order Price $1,000
Discounted (Early Cash) Price: $800
ProduceReceiptCommission: 1%
InterestRate on Financed Purchase: 0.5%/day
AllowedCreditPercentage: 20%

Activity	Circulating	Reserve
Considering the following starting initial state	10,000,000MU	$1,000,000
1. Commodity buyer places a $30,000,000 guaranteed order. Deposits $5,000,000 in the reserve to start, for covering Produce Receipt		$6,000,000 60,000,000MU minted for purchase receipt
2. Farmer A delivers $10,000 worth of grains to Offtaker. Receives Produce Receipt MULC worth 100,000 MU		
3. Offtaker delivers grains to Commodity buyer. System credits Offtaker difference between wholesale price and guaranteed order price	+10,000MU	-10,000MU
4. Farmer redeems MULC. Receives 90,000 MU. Produce Receipt Commission of 1,000MU accrues to the order	+90,000MU	-90,000MU
5. Farmer B delivers $1,000 worth of grain to Offtaker. Receives Produce Receipt MULC worth 10,000 MU		
6. Farmer B cashes out early for 8500 MU	+8,500MU	-8,500MU
7. Offtaker delivers grains to Commodity buyer. System credits Offtaker difference between wholesale price and guaranteed order price	+1,000MU	-1,000MU
8. Offtaker redeems the MULC for the remaining wholesale price. Makes the spread from Reserves for the early farmer cash out. Produce Receipt Commission of 100MU accrues to the order	+500MU	-500MU
9. Guaranteed order now being financed up to $2,000. Continues for 10 days. $100 interest accrues to the order		
. . .	. . .	. . .
The total $30,000,000 order is filled. Total amount financed in product delivery was $1,000,000 for 10 days. Total accrued in Interest at 0.5% per day is $50,000 Total accrued Produce Receipt Commission at 1% is $300,000	Net +60,000,000MU	Net -60,000,000MU +$50,000 +$300,000
Total needed in reserve at end of transactions, in $	$7,000,000	
Total in reserve at end of sample transactions, in $		$7,350,000

(Note that exchange of MULAs in other commercial activity generated in fulfilling this order will result in additional transaction fees that will accrue to the Reserves. Ditto for the additional circulating MULA that will generate transaction fees when eventually spent within the system.)

Simulations of Agricultural Loan Issued in the Ecosystem

Transaction Fees: 0.5%
AgriculturalLoanRate: 1.01
InterestCommission/AutoInterestCommission: 20% of interest paid
AgriculturalActivityRate (to Agro-Dealers): 0.1%

Activity	Circulating	Reserve
Considering the following starting initial state	10,000,000MU	$1,000,000
1. Bank issues $20,000,000 worth of Agricultural and Developmental loans within the system at 10% p.a for 6 months. Loan is disbursed as MULMs to loan recipient farmers within the system. Reserve deposit is at AgriculturalLoanRate	+200,000,000MULM	+$20,200,000 200,000,000MU minted to cover the MULMs
2. Farmer spends some MULMs to purchase agricultural equipment, say tractor, for $1,000. Agro-Dealer cashes out MULMs for 10,000MUs. Agro-Dealer receives the AgriculturalActivityRate incentive on cashing out MULMs	-10,000MULM +10,000MU +10MU	 -10,000MU -10MU
3. Farmer pays monthly interest to bank. 1,666,667 MU is paid from farmer to the bank's MULA account. Interest Commission to the Reserve at 20% is 3,333,333 MU		+3,333,333 MU
. . .	. . .	. . .
The total $30,000,000 in loans is spent on agricultural activities. Total InterestCommission paid to the reserve is $400,000 Total incentive to Agro-Dealers at 0.1% AgriculturalActivityRate is $20,000 Transaction fees accrued to the reserves from $20m in spending at 0.5% is $100,000	Net +200,000,000MU -4,000,000 +200,000MU -1,000,000MU	Net -200,000,000MU +4,000,000 -200,000MU +1,000,000MU
Total needed in reserve at end of transactions, in $	$20,520,000	
Total in reserve at end of sample transactions, in $		$21,680,000

(Note that agricultural produce resulting from the production created from this loan will result in additional commercial activity, and/or produce receipt that will accrue even more to the Reserves. Ditto for the additional circulating MULA that will generate transaction fees when eventually spent within the system.)

Simulations of Agricultural Grant Issued in the Ecosystem

Transaction Fees: 0.5%
AgriculturalGrantRate: 1.01
AgriculturalActivityRate (to Agro-Dealers): 0.1%

Activity	Circulating	Reserve
Considering the following starting initial state	10,000,000MU	$1,000,000
1. Grant originator or governmental organization issues $30,000,000 worth of grants into the system for fertilizer subsidies. The deposit into the reserve is completed at the AgriculturalGrantRate. The grant is disbursed as MULFs to farmers who qualify within the system.	+300,000,000MULF	+$30,300,000 300,000,000MU minted to cover the MULMs
2. Farmer spends some MULFs to purchase fertilizers using MULFs worth $1,000. Agro-Dealer cashes out MULFs for 10,000MUs. Agro-Dealer receives the AgriculturalActivityRate incentive on cashing out MULFs	-10,000MULM +10,000MU +10MU	 -10,000MU -10MU
. . .	. . .	. . .
The total $30,000,000 in loans is spent on agricultural activities. Total incentive to Agro-Dealers at 0.1% AgriculturalActivityRate is $30,000 Transaction fees accrued to the reserves from $30m in spending at 0.5% is $150,000	Net +300,000,000MU +300,000MU -1,500,000MU	Net -300,000,000MU -300,000MU +1,500,000MU
Total needed in reserve at end of transactions, in $	$30,880,000	
Total in reserve at end of sample transactions, in $		$31,420,000

(Note that agricultural produce resulting from the production created from this subsidy will result in additional commercial activity, and/or produce receipt that will accrue even more to the Reserves. Ditto for the additional circulating MULA that will generate transaction fees when eventually spent within the system.)

Simulations for Savings of MULA as MULS

Purchase Rate: 1.0 (MULA exchange rate to MULS)
Reserve Rate: 0.9 (MULS exchange rate to MULA)

Activity	Circulating	Reserved MULS Value	Reserve
Considering the following starting initial state	100,000MU		$10,000
Savings Initiated			
1. Individual A sends 10,000 MULA to contract to save	-10,000MU	10,000 MULS $900	$10,000
2. Individual B Sends 30,000 MULA to contract to save	-30,000MU	30,000 MULS $2,700	$10,000
Total needed in reserve at end of transactions, in $	60,000MU ($6,000)		
Total in reserve at end of sample transactions, in $		$3,600	$10,000
Net gain in reserve margin	$400		
Savings Rewards			
Reserve Gains Resulting in Savings Rewards R_w = 0.01 Applied to Savings Assets After *x* Days	+ 100MU to A +300MU to B		-400MUL
Savings Withdrawal			
Individual A cashes out some of their savings by sending the 1,000 MULS back to the contract	+900MU	-1000MULS	
Total needed in reserve at end of transactions, in $	61,300MU ($6,130)		
Total in reserve at end of sample transactions, in $		$3,510	$9,960
Net gain in reserve margin	$320		

Note that in the absence of any gains in the reserves no savings rewards are paid out to savings accounts, essentially guaranteeing the reserve from being over-run by the issuance of savings assets.

Initial Currency Exchange (ICE) Activity Simulations

Purchase Rate: 3000 (MULA exchange rate to ETH)
Reserve Rate: 3750 (MULA exchange rate to ETH /80%)

Activity	Circulating	Reserve
3. Individual A sends 100 ETH to contract	+300,000MUL	100 ETH
4. Individual B Sends 400 ETH to contract	+1,500,000MUL	500 ETH
5. Individual B Sends 50 ETH to contract	+1,650,000MUL	550 ETH
Total needed in reserve at end of transactions, in ETH	440 ETH ($132,000)	
Total in reserve at end of sample transactions, in ETH		550 ETH ($165,000)

5.2 Projections

This section includes projections over the first three years of fully operating the Agrikore blockchain system. Note that Cellulant already engages in these activities, absent the blockchain. Consequently, Cellulant operators have a grasp of the demand and potential size of each of the above activities, and these are presented below. (Note that the numerical simulations presented above are not projections, but simply game the effect of the various activities on the network.

Please redo. This is a placeholder using limited info I have on some initiatives like Dangote & Letshego

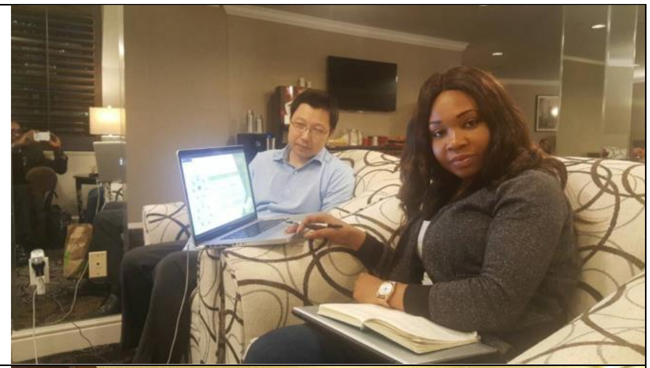
Production Activity	2018	2019	2020	2021	2022
Guaranteed Orders	$70,000,000	120,000,000	200,000,000	$300,000,000	$400,000,000
Accrued to Reserves	$3,500,000	$6,500,000	$10,000,000	$15,000,000	$20,000,000
Agricultural Loans	$100,000,000	$200,000,000	$280,000,000	$350,000,000	$400,000,000
Accrued to Reserves	$4,500,000	$9,000,000	$13,000,000	$17,000,000	$20,000,000
Agricultural Grants	$80,000,000	$100,000,000	$120,000,000	$140,000,000	$150,000,000
Accrued to Reserves	$1,600,000	$2,000,000	$2,400,000	$2,800,000	$3,000,000
Accrual to Reserves on Other Commercial and Agricultural Activities	$4,000,000	$10,000,000	$18,000,000	$29,000,000	$38,000,000
Total Reserve Increase	**$13,600,000**	**$27,500,000**	**$43,400,000**	**$63,800,000**	**$91,000,000**

Cellulant already has a Guaranteed order negotiated to start off this process with one of the largest rice packaging companies in the region. By the second year, on rolling this out across the continent, we expect the size of the ecosystem to grow significantly. Similarly, for loan origination, several banks have already expressed interest in participating in the ecosystem. For some of the banks that have approached Cellulant, the ecosystem presents a very low-cost way of expanding their customer base and acquiring new customers rapidly; customers that will continue to generate transactions within the bank's system longer term. Subsequently, we expect significant participation from all groups within the ecosystem, and enhancement in production in terms of agriculture, and finance in the communities served.

6.0 The Agrikore Team

To get to this point, Cellulant commissioned a project team consisting of internal technology experts and external advisors and consultants to implement the Agrikore blockchain. Initial meeting of team members for the planning phase was completed in London in April 2017. Programming and development commenced soon after with the teams working in coordination and remotely. The targeted timeline for deployment of the solution is included below.

Project Team meeting at the Montcalm in London, April 2017



Project Team meeting at the Radisson in New York, September 2017

6.1 Agrikore Deployment Roadmap

Agrikore Roadmap				
September 2017. The MULA blockchain network is opened for simulations.	**October 2017.** Initial MULA Exchange for ETH tokens is opened to the public.	**November 2017.** MULA is integration into the Tingg App is completed.	**December 2017.** MULA is launched and rolled out to current Tingg users and Cellulant customers.	**January 2018.** MULA held on ETH network are transitioned to the MULA network.

6.2Team Members

Bios of team members

Bios of a Notable Advisors

Technology and Blockchain Advisor, Dr. Alabi

Computing and Technology Advisor, Dr. Li

6.3 About Cellulant

Description of Cellulant including details such as goals, infrastructure and technology as it pertains to the current initiative, organizational structure and background.

7.0 References

1. The World Bank, March 4 2013, "Africa's Agriculture and Agribusiness Markets Set to Top US$ One Trillion in 2030", http://www.worldbank.org/en/news/feature/2013/03/04/africa-agribusiness-report. Accessed October 6, 2017.

2. Bitcoin Energy Consumption Index. Accessed May 20 2017. http://digiconomist.net/bitcoin-energy-consumption

3. William Suberg, Jun 30 2017, "Ethereum Mining Needs More Energy Than Cyprus, Cambodia, Brunei" Cointelegraph, https://cointelegraph.com/news/ethereum-mining-needs-more-energy-than-cyprus-cambodia-brunei. Accessed September 12 2017.

4. Nakamoto, Satoshi (24 May 2009). "Bitcoin: A Peer-to-Peer Electronic Cash System" (PDF). http://bitcoin.org/bitcoin.pdf, Accessed 20 May 2017.

5. Vitalik Buterin, Sep 1 2014, "Ethereum Whitepaper: A Next-Generation Smart Contract and Decentralized Application Platform", https://github.com/ethereum/wiki/wiki/White-Paper, Accessed 20 May 2017.

6. Evan Dufffy & Daniel Diaz, June 14 2016," Dash: A Privacy-Centric Crypto-Currency", Dash WhitePaper, https://dashpay.atlassian.net/wiki/display/DOC/Whitepaper.

7. Serguei Popov, "IOTA WhitePaper: The Tangle," https://iota.org/IOTA_Whitepaper.pdf, Accessed September 20 2017.

8. Ken Alabi, July 2017, "Digital blockchain networks appear to be following Metcalfe's Law", Electronic Commerce Research and Applications, Volume 24, July–August 2017, Pages 23-29 https://doi.org/10.1016/j.elerap.2017.06.003

9. "Bitcoin Supply Strategy", Bitcoin Wiki. https://en.bitcoin.it/wiki/Controlled_supply. Accessed July 12 2017.

10. Vitalik Buterin, Jun 27 2017. Tweet on ETH supply, Internet tweet. https://twitter.com/VitalikButerin/status/879858608091144193

11. Ken Alabi, May 25 2017, "Why Cryptocurrencies are not yet Making a Big Impact in Payment Processing", Published on Medium, https://medium.com/@alabi.ken/why-cryptocurrencies-are-not-yet-making-a-big-impact-in-payment-processing-3ea1f71d2dee

12. Ken Alabi, Jun 16, 2017, "A Macro-Mathematical Model for the Observed Value of Digital BlockChain Networks", Published on Medium. https://medium.com/@alabi.ken/a-macro-mathematical-model-for-the-observed-value-of-digital-blockchain-networks-23cc8e0dc7ea

13. The Economist, May 15 2014, “Money from Nothing”, https://www.economist.com/news/finance-and-economics/21599053-chronic-deflation-may-keep-bitcoin-displacing-its-fiat-rivals-money

14. Michael Del Castillo, “United Nations sends Aid to 10,000 Refugees that utilized the Ethereum Blockchain.” Internet article. http://www.coindesk.com/united-nations-sends-aid-to-10000-syrian-refugees-using-ethereum-blockchain/, Accessed July 12 2017.

15. Vitalik Buterin, Jun 13 2017. “Vitalik Lauds UN Aid to Refugees making use of the Ethereum Blockchain”, Internet tweet. https://twitter.com/VitalikButerin/status/874603372141318145

16. Li, Y. and Ang, K.H. and Chong, G.C.Y. (2006) Patents, software and hardware for PID control: an overview and analysis of the current art. IEEE Control Systems Magazine 26(1):pp. 42-54.

17. Cooper, Douglas. "PI Control of the Heat Exchanger". Practical Process Control by Control Guru. http://controlguru.com/pi-control-of-the-heat-exchanger/

18. "Gold Standard Act of 1900". The Statutes at large of the United States of America. Washington: Government Printing Office. http://www2.econ.iastate.edu/classes/econ355/choi/1900mar14.html . pp. 45–50. Accessed July 16 2017.

19. “The Gold Standard Act [March 14, 1900” http://www.historycentral.com/documents/GoldStandard.html. Accessed July 16 2017

20. Board of Governors of the Federal Reserve System (U.S.), 1935- and Federal Reserve Board, 1914-1935.,June 1989,"The International Gold Standard and U.S. Monetary Policy from World War I to the New Deal” Federal Reserve Bulletin (June 1989). https://fraser.stlouisfed.org/scribd/?item_id=20805&filepath=/files/docs/publications/FRB/1980s/frb_061989.pdf. Accessed on July 16, 2017.

21. Alan Kaplan, Nov 9 1987, “Full Faith and Credit of U.S. Government Behind the FDIC Deposit Insurance Fund” https://www.fdic.gov/regulations/laws/rules/4000-2660.html. Accessed July 16 2017.

22. The Bancor Protocol, 2017, https://www.bancor.network/. Accessed July 16 2017.

23. Emin Gun Sirer & Phil Daian, June 19 2017, “Bancor is Flawed”, http://hackingdistributed.com/2017/06/19/bancor-is-flawed/. Accessed July 16 2017.

24. The Kimberley Process, 2000, https://www.kimberleyprocess.com/en/about, Retrieved 2017-09-13.

25. Pamela Amber, Forbes Magazine, Sep 10 2017, “Diamonds Are The Latest Industry to Benefit from Blockchain Technology”, Retrieved 2017-09-13.