ASTRA: A Decentralized Financial Services Network

A hybrid architecture for a robust, transparent, and user-centered financial services ecosystem on the Ethereum blockchain

Astra

https://astra.finance

September 12, 2017 // Draft V0.3

ABSTRACT

We describe a scalable network architecture that empowers a robust, decentralized ecosystem of peer-to-peer financial services. Transactions within the system are powered by smart contracts in coordination with decentralized application (dApp) clients for both end users and service providers. Client data discovery is facilitated by off-chain oracles and contract offer discovery is managed with minimal latency through off-chain financial services “matching” and data “discovery” channels. The ASTRA network runs on the Ethereum blockchain with access to and usage of the network facilitated by a ERC20 standard compliant “AST” protocol token. It is our strong opinion that the successful launch of a new financial network that is valuable for consumers must coincide with user-friendly dApps at the application layer that are ready for use at inception. In creating this network, we strive to: initiate a transparent, programmatic, and data-driven, decentralized marketplace for financial services; empower end users with intelligent applications to optimize their cash flow and debt through the diversity and liquidity of the available services on the network; and align the incentives of end users and service providers towards an improved and healthier overall system for personal finance.

1. INTRODUCTION

Our financial health in the United States is fraught with problems and inefficiencies at both a systemic and an individual level. It is a latent and unspoken truth while the scale of the challenges we have both individually and as agents within a financial services marketplace is breathtaking. Student loan debt in the US has surpassed $1.3 trillion[1], savings rates for Americans under 35 is negative 2%[2], and 46% of Americans cannot afford an unanticipated expense of $400 without using credit[3]. In seeking products that can help us to reduce our debt burden and increase our financial safety net, we are left with few choices that are either costly or illiquid, that are systematically opaque, or that do not outpace the rate of inflation in return[4]. We are in desperate need of smart technological solutions to bridge this gap and reset our collective course towards financial stability. We founded Astra Inc. to take on this problem with a mission to make your finances more tangible, empowering, and personal.

While the modern internet has enabled new offerings by financial technology startups, banking by and large still runs on a centralized "old rails" system dominated by relatively few big banks[5]. Since the financial crisis, the macroeconomic indicators mentioned above have only continued to worsen while the number of operating banks has decreased through failures, consolidations, and a dearth of newly chartered institutions. From a technical perspective, the majority of recent innovative solutions for financial services reside within an improved user experience (Simple[6]) or smart application layer (Stripe[7]) of core banking technology, while continuing the use of underlying legacy protocols like the Automated Clearing House[8] (ACH). While this is frequently referred to as the "unbundling of banks,"[9] the new diversity of offerings remains at the top layer of the technology stack and assumes the inefficiencies and illiquidity of a consolidated banking industry utilizing centralized technology. In 2016 alone, banking in the United States generated $553 billion in revenue[10] including $15 billion in overdraft and bounced check fees[11]. When operating within today's banking ecosystem, users are left to expect misaligned incentives (why am I hit with compounding overdraft fees?), lack of transparency (why specifically wasn't I approved for a loan?), and a lagging user experience (why isn't my banking experience as intelligent as when I use Google?).

---

[1] https://www.forbes.com/sites/zackfriedman/2017/02/21/student-loan-debt-statistics-2017/#2f4c26bf5dab
[2] https://www.creditdonkey.com/average-american-savings-statistics.html
[3] https://www.washingtonpost.com/news/wonk/wp/2016/05/25/the-shocking-number-of-americans-who-cant-cover-a-400-expense/?utm_term=.bef73149eeeb
[4] Excluding gambling or risky options such as the stock market
[5] Chase, Bank of America, Wells Fargo, US Bank
[6] https://www.simple.com/
[7] https://stripe.com/
[8] The Automated Clearing House system is 40 years old and is managed by the NACHA, composed of the same centralized institutions that offer banking services
[9] https://www.cbinsights.com/research/disrupting-banking-fintech-startups-2016/?utm_source=CB+Insights+Newsletter&utm_campaign=fc293aa2e6-Top_Research_Briefs_5_28_2016&utm_medium=email&utm_term=0_9dc0513989-fc293aa2e6-87493377
[10] IBIS Reports 2016 https://www.ibisworld.com/industry-trends/market-research-reports/finance-insurance/credit-intermediation-related-activities/commercial-banking.html
[11] http://fortune.com/2017/08/05/overdraft-fees-2016/

While the last 8 years have been marked by inertia and increased centralization in the banking industry, in parallel, we have seen the founding of new decentralized economic systems, based on the technological innovations of blockchains (Bitcoin) and smart contracts (Ethereum). These networks have been initialized, their associated cryptocurrencies (bitcoin and ether) have seen dramatic appreciation (as well as a lot of volatility), and their development platforms are flourishing as seen through the many launches of native cryptocurrencies and Ethereum-based network tokens just this year[12]. The trustlessness inherent in the economics of the decentralized systems has demonstrated valuable in applications of cross-border remittances, store of value, and low-volume or infrequent payment processing; however, many challenging questions remain regarding securities regulation, network scalability, and broader user adoption. Fundamentally though, a decentralized network architecture with programmable transactions has the potential to shift the landscape of banking and redefine the relationship between end users and the providers of their financial services such that end users own their data, gain access to a more diverse and liquid market that they themselves create, and, most importantly, are empowered to improve their financial health.

While we are certain that the foundational technologies of blockchains and smart contracts can be game changers for financial services, it is important to note that we advocate for a pragmatic, balanced approach to defining which elements of a decentralized architecture will offer value to the end user. In terms of value creation, much has been made about the dichotomy between the older internet paradigm "skinny" protocols where value is created most at the application layer and the newer crypto-enabled "fat" protocols[13] where value is created most at the protocol layer. The argument for this shift in approach is strong, yet taken at face value it ignores two key points. First, creating a "fat" protocol assumes that there will be many development teams that build on your network and that you have incentivized all parties effectively. This "build it and they will come" perspective must achieve a goldilocks scenario for users, developers, investors, organizations, and miners at launch. Second, weighting the importance of the decentralized protocol over the apps that can be built on top of it given the current limitations of on-chain data storage, blocktime, and computation cost, reduces the opportunity for leveraging other advanced contemporary technologies. This is especially true when solutions to larger problems, such as personal finance, require such technologies to succeed; and as "artificial intelligence is eating software"[14] in the public domain, users are becoming more accustom to smarts embedded in their user experience. To be effective and create value for users, AI requires large, centralized databases that would currently be too expensive to maintain and/or too slow to run on-chain. Therefore, we propose the ASTRA Network as a hybrid "balanced" protocol (at least at launch)

---

[12] For better or worse, the volume of and funds raised by Initial Coin Offerings (ICOs) is enough to define the token market as lava hot. World-wide some 120 businesses have raised about $1.5 billion through coin offerings this year, up from about $256 million by 43 companies last year, according to CoinDesk's ICO Tracker. https://www.wsj.com/articles/coin-mania-forces-vcs-to-sidelines-on-cryptocurrency-1505388633

[13] http://www.usv.com/blog/fat-protocols

[14] https://www.technologyreview.com/s/607831/nvidia-ceo-software-is-eating-the-world-but-ai-is-going-to-eat-software/

where centralized services and a decentralized network run in concert and which creates a flywheel of user adoption and development energy to scale.

2. EXISTING WORK

The recent increase in momentum for blockchains and smart contracts to enable an internet of money indicates the potential for cryptocurrencies and protocol tokens to offer value to end users and developers alike. In existing networks and protocols under development, we point to key achievements in token liquidity, network latency, external data connections, and user-centered incentives. Each example below demonstrates productive advancements in the respective focus areas which tend towards exchange dynamics, but none discussed nor at large provide a generalizable architecture for diverse services for personal finance as we propose.

As the broader ecosystem of protocol tokens expands, a primary challenge is exchange market making, particularly when trading volume is low for a given token thereby making it illiquid. To solve this scenario, Bancor[15] has defined a "smart token" so that, when attached to another reserve token and a programmed constant reserve ratio, the change in token value can be discovered before the buy or sell order is executed.[16] The Bancor approach is to make *external* tokens from their network more liquid and therefore more readily exchanged; however, we apply this transparent price discovery mechanism to a token internal to a network with USD and ETH in balanced reserve, effectively functioning to both stabilize the token's value as a treasury reserve would and offer clear outcomes attached to the minting or liquidation of the protocol token.

The increased usage of blockchains such as Ethereum, directly or by way of ERC20[17] tokens, also presents a challenge to latency within the network. While Ethereum as the "world computer" creates value by trading speed for decentralization and immutability[18], the existence of projects such as Raiden[19], Plasma[20], and 0x[21] demonstrate the demand to increase the volume for payments and the execution of smart contracts. Whether through state networks, nested child blockchains, or order relay respectively, these projects seek to move the volume transactions off-chain to increase performance and subsequently settle the results on-chain. Of specific value for a hybrid on-chain, off-chain model is 0x's "broadcast orders," which provide insight as to how any decentralized network might aggregate any type of on-chain data to external end points within a centralized service. Although their ambition is to facilitate signaling of orders and diversify

---

[15] https://www.bancor.network/
[16] https://www.bancor.network/static/bancor_protocol_whitepaper_en.pdf
[17] https://theethereum.wiki/w/index.php/ERC20_Token_Standard
[18] A valuable tradeoff! Note that improvements to blocktime and blocksize on core blockchains for Bitcoin and Ethereum to increase throughput are also in the works; however, such updates still trail volume for centralized options such as the Visa payment network. https://www.coindesk.com/information/will-ethereum-scale/
[19] https://raiden.network/
[20] http://plasma.io/
[21] https://0xproject.com/

the creation of exchange order books, we adopt this messaging mechanism for on-chain contracts to communicate to a central pool of offers by both end users and providers so that they may choose pairing within a "matching channel" for a two-sided financial services marketplace.

Within more complex[22] financial products such as lending, the traditional formal application process typically involves assessment of creditworthiness by way of one's history with an institution and/or cross-institution credit reports by third parties such as Experian[23] or Equifax[24]. A new financial services network that utilizes the benefits of a decentralized network must also be able to facilitate data discovery by providers and end users alike.[25] Augur[26] and Oraclize[27] demonstrate off-chain means of betting market settlement and data carrier connections respectively to dApps through oracles. More widely used cloud data stores can then be leveraged to provide the requested data back to the programmatic features of smart contracts within the network[28]. Furthermore, as we propose with matching channels such a data connection can be opened by logic on the blockchain as a "discovery channel" in a centralized, searchable database, with the possibility for the user to specify a range of cryptographically signed permission levels for that user's data at either a global level or granularly per contract.

Permissioned access to user data can benefit the process of requesting a loan within the network as in the example above, but we further generalize this feature such that data discovery can also be the central focus of the contract. In the case of detailed consumer data search or broader market research, end users receive the network token per query of their data thereby incentivizing participation and discovery on both sides of a two-sided marketplace. A parallel example of such a structure is 21.co[29] where users are rewarded predefined amounts of bitcoin for various tasks as programmatically requested by developers. 21.co has also just announced a token to manage these rewards[30].

Lastly, prior to proposing the Astra Network, Astra Inc. has spent the past year developing a beta web application for personal finance. Our Early Adopter Program allowed us to prove our balance forecasting technology that is powered by deep learning, and define a schema for and compile an initial collection of financial data, and gather key feedback from end users. The architecture of this app followed a conventional centralized architecture for purposes of efficient and speedy development as well as to effectively structure our datasets for consumption by our AI models. From the onset, we envisioned it as a means of developing the first version of a user-focused component for future inclusion within a broader system as well as a container within which we could develop decentralized modules including a wallet for AST tokens.

---

[22] Beyond basic one-time transactions, such as recurring and/or variable payments
[23] http://www.experian.com/
[24] That comes with the security risks of a centralized identity and credit service, including a massive hack in September 2017 https://www.equifax.com/
[25] In as much as a lender should be able to assess one's financial health, so too should a lendee be able to assess the history of the organization or peer offering the service.
[26] https://augur.net/
[27] http://www.oraclize.it/
[28] https://blog.ethereum.org/2014/07/22/ethereum-and-oracles/
[29] https://21.co/
[30] https://21.co/token/

3. SPECIFICATIONS

The Astra Network defines a generalized, scalable architecture for financial services. This architecture, and its component modular smart contracts deployed on the Ethereum blockchain, is initialized as a hybrid system where the security, robustness, and performance benefits of on-chain versus off-chain services are coordinated for maximum value add to the end user and maximum utility across the network. The AST "usage token" that powers the network provides access to engage the services available to both end users and financial service providers[31] and is governed by a transparent, explicit formula to establish clear mechanics for token value across time.

The design of the Astra Network is guided by the following principles:

- Enable transparent, smart, data-driven financial services
- Increase the diversity of financial services available to end users
- Increase liquidity for the variety of financial services offered by providers
- Empower users and service providers with information
- Establish user ownership over financial data with outlets to reinforce services requested and further monetize that data
- Create a generalized financial services network with balanced incentives throughout the resulting ecosystem

3.1. SYSTEM ARCHITECTURE

The high-level system architecture is composed of modular elements that establish a multi-sided marketplace for financial services, as depicted in the diagram below. Each element has an assigned colloquial name for ease of reference with technical definition in subsequent sections. At the center of the technology stack are a series of linked smart contracts (magenta) deployed to the Ethereum blockchain that programmatically define interaction between agents in the network and the management of the AST token that powers the network. Oracles (green) are hosted as off-chain channels required for communication and data exchange between parties related through the smart contracts. Distributed applications (blue) reside on either side of the financial services marketplace and serve as interaction points for end users and marketplace participants.

[31] https://thecontrol.co/on-token-value-e61b10b6175e

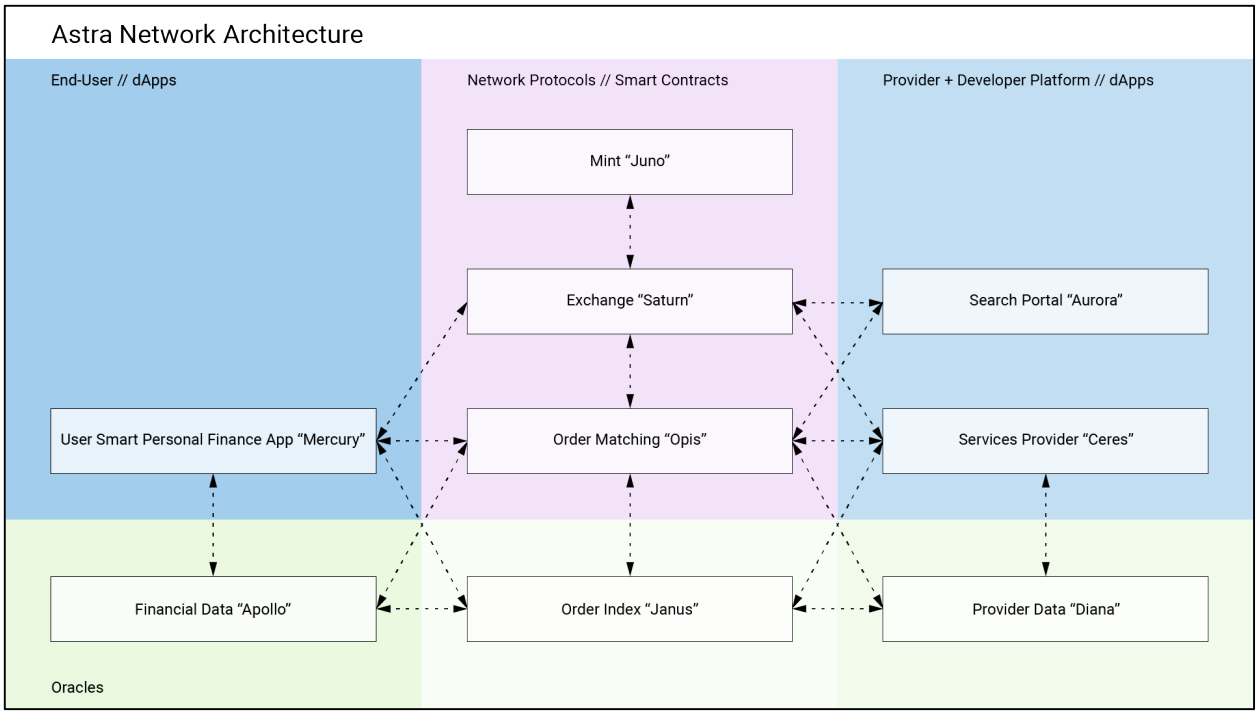

Figure 1: Astra Network Architecture

3.2. SYSTEM MODULES

3.2.1. SMART CONTRACTS

All smart contracts live on the Ethereum blockchain with messages received by each including a version number so that contracts may be updated.

3.2.1.1. MINT (“JUNO”)

The Mint contract serves to facilitate the exchange of currency types into and out of the network as well as to manage the network’s reserve currency held in USD and ETH. Transactions received by the Mint contract address define the minting of newly issued tokens or liquidating of tokens by removal from the supply, which can be done in any combination of reserve currency. The contract will then relay the corresponding change to the two reserve addresses based on the pre-defined relative constant reserve ratio. One reserve address will store USD and the second will store ETH with the Mint contract periodically rebalancing the amounts of currency in each reserve address to accommodate changes in the respective value of the currencies.

Extending the price discovery equation by Bancor, we further define the price of the AST token as follows, where the value can be calculated for both current and post transaction execution[32].

[32] See Appendix A.1.4.

$$P = \sum_i \left(\frac{R_i * C_i}{S * \frac{F_i}{G_i}} \right) = \sum_i \left(\frac{V_i}{S * \frac{F_i}{G_i}} \right)$$

Additionally, the Mint reserve achieves more stable pricing by minimizing currency fluctuation with dynamic rebalancing[33].

$$O_i = \Delta V * G_i - \Delta V_i \quad where \quad O = \sum_i O_i = 0.0$$

Further details and additional formulas guiding the value of the AST token and Mint contract functions can be found in the Appendix A.1.

As seen in the below spreadsheet, various scenarios varying the flow of reserve currency into and out of the network results in updated token price and rebalancing of the respective reserve currencies. Additional comparison of balanced and unbalanced reserves show the stability in AST pricing when volatility exists in the exchange rate of reserve currencies relative to USD value. More detail can be found in the published version of the spreadsheet[34].

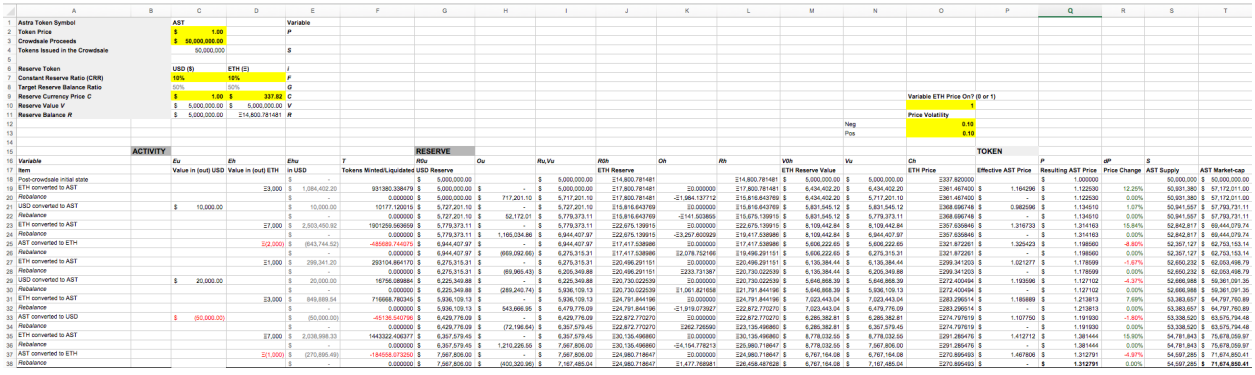

Figure 2: Mint Reserve and Token Pricing

Interaction with the Mint contract through the Astra Network is achieved through the following parameters:

JUNO CONTRACT PARAMETERS		
Parameter	Data Type	Description
version	string	Contract version number

[33] Appendix A.1.7.
[34] https://drive.google.com/file/d/0BxUbdpphIu8sMVpkNng2eFNFN0U/view?usp=sharing

reserves	address[]	Addresses for reserves held in USD, ETH
ratio	uint[]	Ratio values (1-100) per reserve address (default 10,10)
rebalanceFrequency	uint	Frequency reserves rebalance to ratios
rebalanceThreshold	uint	Ratio difference threshold forcing off cycle rebalance (default 5)
requester	address	Address of contract call

3.2.1.2. EXCHANGE (“SATURN”)

The Exchange Contract manages actions between agents in the network, service contracts they deploy, and the AST and reserve currencies. Any calls to the Mint contract flow through the Exchange to accommodate validation by all parties to a financial service or transaction being executed. While it may seem as though multiple signatures could be managed in the Mint contract itself, running such transactions through the Exchange allows the Mint rebalancing functionality to be isolated and ensures the flow of executed transactions to be verified and generalized for modularity. Simple peer-to-peer exchange of AST tokens need not call the Exchange contract as in vanilla Ethereum; however, any financial service offered in the network marketplace or any transaction utilizing oracles or accessing user data must call the Exchange contract and supply the requisite user or contract signatures.

SATURN CONTRACT PARAMETERS		
Parameter	Data Type	Description
version	string	Contract version number
order	address	Address of executed contract
funder	address[]	Address of funding contract/user
receiver	address[]	Address of recipient contract/user
signatures	address[]	Verified signature
exchange	unit[]	Value of currency to exchange [AST, USD, ETH]

3.2.1.3. ORDER (“OPIS”)

All peer-to-peer financial services flow through the Order contract as smart contracts themselves. The Order contract facilitates the interaction between agents in the network by receiving requests or offers from dApps on each side of the marketplace, relaying the requested and offered contracts to the correct Oracle, and sending confirmed orders to the Exchange contract. The Order contract's architectural model is a hub with spokes connecting the agents and contracts in the network allowing agent generated contracts and parameters to be shared, discussed, and eventually executed. It acts as a meta-contract where requested contract flow exists on-chain so that order contracts, terms, and addresses may captured securely and added to the immutable log in the blockchain while the intermediate steps of refining terms and matching requesters and takers of the orders are facilitated off-chain through Oracles without the cost, delay, or inefficiency of posting all order updates on-chain. This enables the contracts that pass through the Order contract to be readily searchable and malleable during negotiation between agents as well as to be linked to or driven by the rich data stored within the Oracles. Lastly, once terms have been finalized, the Order contract sends the network fee and the updated version of the agreed upon contract to the Exchange for signature verification and execution of all programmed token or reserve currency transactions.

OPIS CONTRACT PARAMETERS		
Parameter	Data Type	Description
version	string	Contract version number
order	address	Address of executed contract
funder	address[]	Address of funding contract/user
receiver	address[]	Address of recipient contract/user
signatures	address[]	Verified signature
exchange	uint[]	Value of currency to exchange [AST, USD, ETH]
fee	uint	Value of network fee [AST]

3.2.2. ORACLES

Oracles are off-chain data sources established for data monetization or contract enrichment. At initialization, the Astra Network includes three oracles – two for data discovery and a third for order matching. Agents in the network may define the level of access to oracles storing their data but are incentivized through the earning of tokens to grant access to that data on a per contract or per detail level.

In general, all oracles act essentially as query-able API endpoints accessed and authorized through the Astra token.

3.2.2.1. END USER DATA DISCOVERY CHANNEL (“APOLLO”)

When consumers sign up for an Astra account that grants them access to the End User dApp, they are prompted to securely connect their various accounts at financial institutions. This connection between the institution includes account balances and transaction history and is maintained as a live connection over time. This data is essential for the intelligence built into the dApp and is further utilized through the End User Data Discovery Channel. Agents in the Astra Network may request user data through this oracle in order to conduct market research, verify financial details relevant to a contract executed on the network, or to assess credit worthiness for a financial service. This data may be queried by the public key of a given user or balance or transaction details, with search results hashed by default. The querying agent may then request further detail from the user through off-chain communication with access granted and logged on-chain through the Order contract. Similarly, when requesting a financial service, an end user may pre-define the access settings for agents who may accept the offer. All queries are also tied to a fee in AST split between the oracle provider (Astra Inc) and all users whose data was found in the search.

APOLLO ORACLE ENDPOINTS (Example list, not exhaustive)	
URL	Returns
/users/<search>?<search>=<terms>/	List of hashed user public keys by search criteria
/users/credit?num_cards='>2'/	List of hashed user public keys by search criteria
/users/<user_token>/accounts/	List of hashed account ids
/users/<user_token>/accounts/<account_id>/	Account details for given user token's hashed account id, including current balance
/users/<user_token>/accounts/<account_id>/historical/	Historical balances for given account
/users/<user_token>/accounts/<account_id>/forecast/	Forecasted balances for given account
/users/<user_token>/accounts/<account_id>/txs/	List of hashed transactions for given account
/users/<user_token>/txs/	List of hashed transactions for given user token
/txs/<search>?<search>=<terms>/	List of hashed transactions by search criteria
/txs/cat?cat='shops'/	List of hashed transactions by search criteria

3.2.2.2. PROVIDER DATA DISCOVERY CHANNEL (“DIANA”)

Service providers must also sign up for an Astra account that grants them access to the network and the Provider dApp. In the same manner that end users’ data is discoverable through the Apollo oracle, so too is the provider user and their in network activity discoverable. End users who are considering accepting an offer for a service may request data about their provider. The returned data may help a consumer decide which provider to go with for the service and may aid in a strong relationship between the agents engaged in an on-chain contract. This data may be queried by the public key of a given provider or service contract address, with search results hashed by default. The querying user may then request further detail from the user through off-chain communication with access granted and logged on-chain through the Order contract. Just as in the Apollo oracle, a provider may pre-define the access settings for users who may accept offers. All queries are also tied to a fee in AST split between the oracle provider (Astra Inc) and all providers whose data was found in the search.

DIANA ORACLE ENDPOINTS (Example list, not exhaustive)	
URL	Returns
/providers/<search>?<search>=<terms>/	List of hashed provider public keys by search criteria
/providers/contracts?num_contracts=’>200’/	List of hashed provider public keys by search criteria
/providers/<user_token>/contracts/	List of hashed contract addresses
/providers/<user_token>/contracts/<contract_token>	Contract details for given provider hashed contract addresses
/contracts/<contract_token>	Contract details for given hashed contract address
/contracts/<search>?<search>=<terms>/	List of hashed contract addresses by search criteria

3.2.2.3. ORDER MATCHING CHANNEL (“JANUS”)

Whereas the primary function of the Apollo and Diana oracles is data discovery, the Order Matching Channel facilitates communication surrounding and updating of financial service orders. The dApps for end users and providers alike may search for orders and then engage the agents requesting or offering those orders. Messaging occurs in the respective dApp and is logged with both the user/provider public key and

the address of the contract under discussion. In addition to GET and POST endpoint actions, the Order Matching Channel also includes PUT actions to update the terms of an order's contract.

JANUS ORACLE ENDPOINTS (Example list, not exhaustive)	
URL	Returns
/orders/<search>?<search>=<terms>/	List of hashed order addresses by search criteria
/orders/type?type='loan'/	List of hashed order addresses by search criteria
/orders/<contract_token>	Contract details for given hashed contract address
/orders/<contract_token>/terms/	Contract terms for given hashed contract address
/orders/<user_token>/	List of open orders by user/provider hashed public key

3.2.3. DISTRIBUTED APPLICATIONS

Distributed applications for each agent type within the Astra Network are essential to facilitate the on-chain actions and off-chain channels. Each dApp must include a wallet for purchasing, spending, receiving, and holding AST tokens, communicate with the network with the user's associated public key, and access the appropriate oracles, while offering valuable usability to the respective user type.

3.2.3.1. END USER DAPP ("MERCURY")

The End User dApp is a personal finance management app with an associated wallet to hold Astra tokens. A beta version of this app (without the token wallet) has already been developed and released through an Early Adopter program. As in the beta currently, the dApp allows a user to sign up for an Astra account (and thereby establish a public key for communication to the network and private key for contract authorization), link their financial accounts to the app[35], and manage their finances through interactive charts, insights, and deep learning powered balance forecasting. The cloud servers that maintain the connection to the user's financial institutions communicate with the dApp to provide relevant data to the app for the user's consumption and also serve any queried data to the Apollo oracle with the hash specifications defined by the user in the dApp.

MERCURY FEATURES

[35] No credentials are stored in the Astra system after the accounts are linked.

Parameter	Description
user_id	User log in id/email address
private_key	Private key for contract execution
public_key	Public key for engaging contracts on network

3.2.3.2. PROVIDER DAPP (“CERES”)

The Provider dApp is a financial service smart contract creation tool with an associated wallet to hold Astra tokens. It allows an individual or organization to sign up for an Astra account (and thereby establish a public key for communication to the network and private key for contract authorization), link their financial accounts to the app to fund their offerings, and create, curate, and manage their financial services on the network. Within the app, a provider can specify terms in conventional web/mobile forms and fields to create a financial service (such as a loan) and submit that service to the network in a user-friendly way. The service with its associated contract address can be tracked in the app, queried through the network, and serve as the subject matter for communication with agents interested in accepting the offer across the Janus matching channel.

CERES FEATURES	
Parameter	Description
user_id	User log in id/email address
private_key	Private key for contract execution
public_key	Public key for engaging contracts on network

3.2.3.3. SEARCH PORTAL DAPP (“AURORA”)

The Search Portal dApp is a market research tool with an associated wallet to hold Astra tokens. It allows an individual or organization to sign up for an Astra account (and thereby establish a public key for communication to the network and private key for contract authorization), link their financial accounts to the app to fund their queries, and submit searches to the Apollo oracle to gather anonymized information about consumer behavior. Before executing a search, the dApp user must have funds held in AST and then contribute AST to the Order contract when the search is executed. The equivalent of user reach and

budgeting as seen in Facebook promotions[36] may be gathered before execution. The gross fee for the query after execution is then broken into micropayments to all users whose data was returned in the search.

AURORA FEATURES	
Parameter	Description
user_id	User log in id/email address
private_key	Private key for contract execution
public_key	Public key for engaging contracts on network

4. USE CASES AND ILLUSTRATIONS

To illustrate the interaction between Contracts, Oracles, and dApps, the following use cases for the Astra Network are described with data flow diagrams: Astra end user premium subscription, market research, and loan execution both with and without user data discovery as a part of the process.

4.1. USER SUBSCRIPTION

Astra Mercury dApp users are offered a premium subscription option that unlocks additional features in the dApp and automates a monthly contribution held in AST. This pseudo-savings program is designed such that with positive behavior across agents in the network the value of the funds held should appreciate. The monthly contribution is sent in USD through a simple smart contract through the Exchange contract to the Mint Contract and back to the user's wallet. The scenario sequence in the network and appreciation calculation are as follows:

[36] https://www.facebook.com/business/a/boost-a-post

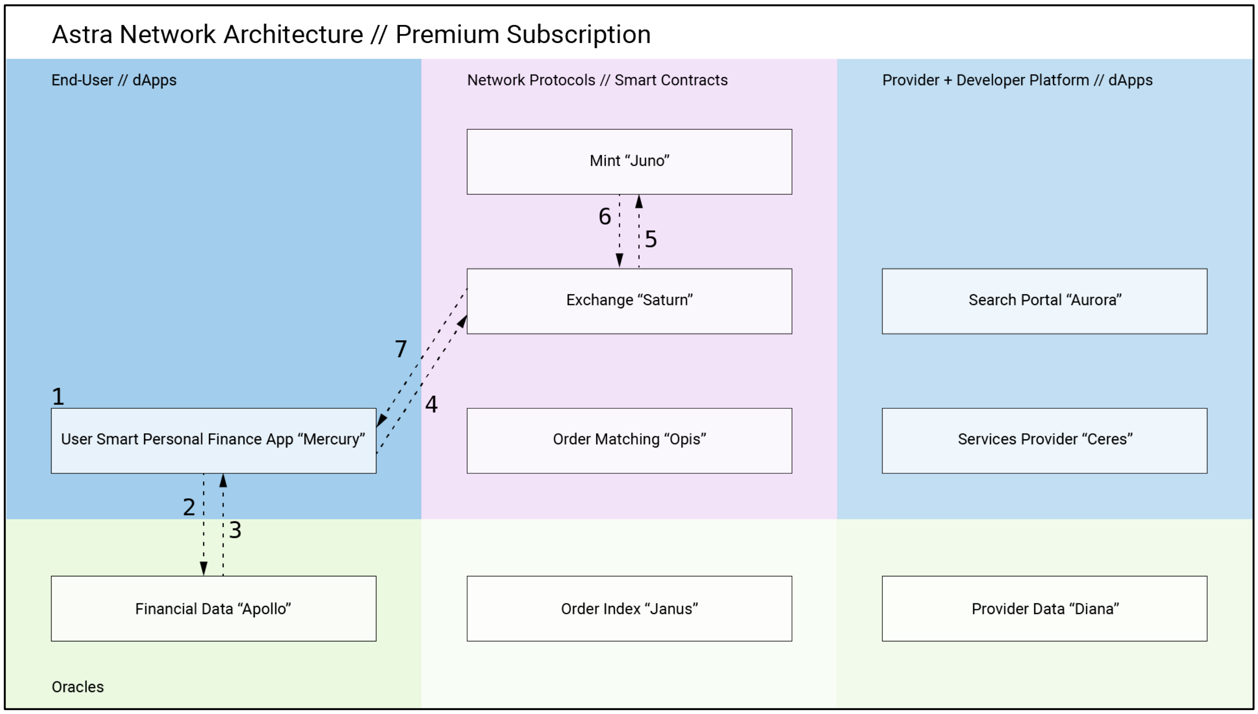

Figure 3: Premium Subscription Sequence

1. User signs up for the Astra Mercury dApp and securely connects financial accounts (checking, savings, credit) earning tokens
2. Live connection to financial accounts established on the app back end and corresponding balance and transaction data is stored on centralized database enabling the Apollo data Oracle for the network and the creation of AI models for Mercury dApp client
3. Formatted financial data (including transactions, balances, forecasted balances, categorizations, etc) is sent back to the Mercury dApp client
4. The Mercury dApp sends the monthly subscription fee ($12/month) from one of the user's connected bank accounts in fiat currency to the Exchange contract on-chain
5. The Exchange contract splits the fee into two portions sending the saved amount ($10) to the Mint contract and the remainder ($2) to Astra Inc's wallet
6. The Mint exchanges fiat currency returning the equivalent minted value in SOL tokens and rebalances the 20% constant reserve for the system (10% in USD and 10% in ETH, rebalanced at X min increments to stabilize value and encourage appreciation)
7. The Exchange contract sends the SOL tokens back to the user's wallet in the Mercury dApp

Init Subscribers	10000
Growth Rate	35%
Monthly Contributions	$ 10.00

	ACTIVITY					RESERVE				
Variable		*Eu*	*Eh*	*Ehu*	*T*	*R0u*	*Ou*	*Ru,Vu*	*R0h*	*Oh*
Item	*Subscribers*	**Value in (out) USD**	**Value in (out) ETH**	**in USD**	**Tokens Minted/Liquidated**	**USD Reserve**			**ETH Reserve**	
Post-crowdsale initial state				$ -		$ 5,000,000.00		$ 5,000,000.00	Ξ14,800.781481	
Monthly Subscribers	10000	$ 100,000.00		$ 100,000.00	99111.237237	$ 5,100,000.00	$ -	$ 5,100,000.00	Ξ14,800.781481	Ξ0.000000
Rebalance				$ -	0.000000	$ 5,100,000.00	$ 100,000.00	$ 5,200,000.00	Ξ14,800.781481	-Ξ279.260028
Monthly Subscribers	13500	$ 135,000.00		$ 135,000.00	131061.597208	$ 5,335,000.00	$ -	$ 5,335,000.00	Ξ14,521.521453	Ξ0.000000
Rebalance				$ -	0.000000	$ 5,335,000.00	$ (41,500.00)	$ 5,293,500.00	Ξ14,521.521453	Ξ114.745457
Monthly Subscribers	18225	$ 182,250.00		$ 182,250.00	169010.180057	$ 5,475,750.00	$ -	$ 5,475,750.00	Ξ14,636.266910	Ξ0.000000
Rebalance				$ -	0.000000	$ 5,475,750.00	$ 147,082.50	$ 5,622,832.50	Ξ14,636.266910	-Ξ373.097130
Monthly Subscribers	24603	$ 246,030.00		$ 246,030.00	221995.388086	$ 5,868,862.50	$ -	$ 5,868,862.50	Ξ14,263.169780	Ξ0.000000
Rebalance				$ -	0.000000	$ 5,868,862.50	$ (207,357.49)	$ 5,661,505.01	Ξ14,263.169780	Ξ542.261652
Monthly Subscribers	33214	$ 332,140.00		$ 332,140.00	279438.744197	$ 5,993,645.01	$ -	$ 5,993,645.01	Ξ14,805.431433	Ξ0.000000
Rebalance				$ -	0.000000	$ 5,993,645.01	$ (364,222.68)	$ 5,629,422.34	Ξ14,805.431433	Ξ1,024.172718
Monthly Subscribers	44838	$ 448,380.00		$ 448,380.00	368541.078898	$ 6,077,802.34	$ -	$ 6,077,802.34	Ξ15,829.604150	Ξ0.000000
Rebalance				$ -	0.000000	$ 6,077,802.34	$ (252,337.11)	$ 5,825,465.23	Ξ15,829.604150	Ξ716.724344
Monthly Subscribers	60531	$ 605,310.00		$ 605,310.00	489070.551306	$ 6,430,775.23	$ -	$ 6,430,775.23	Ξ16,546.328494	Ξ0.000000
Rebalance				$ -	0.000000	$ 6,430,775.23	$ (331,782.33)	$ 6,098,992.90	Ξ16,546.328494	Ξ951.895078
Monthly Subscribers	81716	$ 817,160.00		$ 817,160.00	622857.821161	$ 6,916,152.90	$ -	$ 6,916,152.90	Ξ17,498.223572	Ξ0.000000
Rebalance				$ -	0.000000	$ 6,916,152.90	$ (408,580.00)	$ 6,507,572.90	Ξ17,498.223572	Ξ1,172.230285
Monthly Subscribers	110316	$ 1,103,160.00		$ 1,103,160.00	780968.419010	$ 7,610,732.90	$ -	$ 7,610,732.90	Ξ18,670.453857	Ξ0.000000
Rebalance				$ -	0.000000	$ 7,610,732.90	$ (226,201.36)	$ 7,384,531.54	Ξ18,670.453857	Ξ589.981441
Monthly Subscribers	148926	$ 1,489,260.00		$ 1,489,260.00	958621.641972	$ 8,873,791.54	$ -	$ 8,873,791.54	Ξ19,260.435299	Ξ0.000000
Rebalance				$ -	0.000000	$ 8,873,791.54	$ (929,243.29)	$ 7,944,548.26	Ξ19,260.435299	Ξ2,551.226258
Monthly Subscribers	201050	$ 2,010,500.00		$ 2,010,500.00	1116606.821774	$ 9,955,048.26	$ -	$ 9,955,048.26	Ξ21,811.661556	Ξ0.000000
Rebalance				$ -	0.000000	$ 9,955,048.26	$ (1,323,031.93)	$ 8,632,016.33	Ξ21,811.661556	Ξ3,948.226338
Monthly Subscribers	271417	$ 2,714,170.00		$ 2,714,170.00	1347929.786062	$ 11,346,186.33	$ -	$ 11,346,186.33	Ξ25,759.887894	Ξ0.000000
Rebalance				$ -	0.000000	$ 11,346,186.33	$ (1,529,725.33)	$ 9,816,461.00	Ξ25,759.887894	Ξ4,755.256409
		$ 10,183,360.00	0	$ 10,183,360.00	**6,585,213**					

					TOKEN				
Variable	*Rh*	*V0h*	*Vu*	*Ch*		*P*	*dP*	*S*	
Item		**ETH Reserve Value**		**ETH Price**	**Effective AST Price**	**Resulting AST Price**	**Price Change**	**AST Supply**	**AST Market-cap**
Post-crowdsale initial state	Ξ14,800.781481	$ 5,000,000.00	$ 5,000,000.00	Ξ337.820000		$ 1.00[illegible]		50,000,000	$ 50,000,000.00
Monthly Subscribers	Ξ14,800.781481	$ 5,300,000.00	$ 5,300,000.00	Ξ358.089200	$ 1.008967	$ 1.037943	3.79%	50,099,111	$ 52,000,000.00
Rebalance	Ξ14,521.521453	$ 5,300,000.00	$ 5,200,000.00	Ξ358.089200	$ -	$ 1.037943	0.00%	50,099,111	$ 52,000,000.00
Monthly Subscribers	Ξ14,521.521453	$ 5,252,000.00	$ 5,252,000.00	Ξ361.670092	$ 1.030050	$ 1.053849	1.53%	50,230,173	$ 52,935,000.00
Rebalance	Ξ14,636.266910	$ 5,252,000.00	$ 5,293,500.00	Ξ361.670092	$ -	$ 1.053849	0.00%	50,230,173	$ 52,935,000.00
Monthly Subscribers	Ξ14,636.266910	$ 5,769,915.00	$ 5,769,915.00	Ξ394.220400	$ 1.078337	$ 1.115659	5.87%	50,399,183	$ 56,228,325.00
Rebalance	Ξ14,263.169780	$ 5,769,915.00	$ 5,622,832.50	Ξ394.220400	$ -	$ 1.115659	0.00%	50,399,183	$ 56,228,325.00
Monthly Subscribers	Ξ14,263.169780	$ 5,454,147.53	$ 5,454,147.53	Ξ382.393788	$ 1.108266	$ 1.118406	0.25%	50,621,178	$ 56,615,050.13
Rebalance	Ξ14,805.431433	$ 5,454,147.53	$ 5,661,505.01	Ξ382.393788	$ -	$ 1.118406	0.00%	50,621,178	$ 56,615,050.13
Monthly Subscribers	Ξ14,805.431433	$ 5,265,199.66	$ 5,265,199.66	Ξ355.626223	$ 1.188597	$ 1.105963	-1.11%	50,900,617	$ 56,294,223.37
Rebalance	Ξ15,829.604150	$ 5,265,199.66	$ 5,629,422.34	Ξ355.626223	$ -	$ 1.105963	0.00%	50,900,617	$ 56,294,223.37
Monthly Subscribers	Ξ15,829.604150	$ 5,573,128.11	$ 5,573,128.11	Ξ352.069961	$ 1.216635	$ 1.136251	2.74%	51,269,158	$ 58,254,652.25
Rebalance	Ξ16,546.328494	$ 5,573,128.11	$ 5,825,465.23	Ξ352.069961	$ -	$ 1.136251	0.00%	51,269,158	$ 58,254,652.25
Monthly Subscribers	Ξ16,546.328494	$ 5,767,210.57	$ 5,767,210.57	Ξ348.549261	$ 1.237674	$ 1.178362	3.71%	51,758,229	$ 60,989,928.99
Rebalance	Ξ17,498.223572	$ 5,767,210.57	$ 6,098,992.90	Ξ348.549261	$ -	$ 1.178362	0.00%	51,758,229	$ 60,989,928.99
Monthly Subscribers	Ξ17,498.223572	$ 6,098,992.90	$ 6,098,992.90	Ξ348.549261	$ 1.311953	$ 1.242352	5.43%	52,381,087	$ 65,075,728.99
Rebalance	Ξ18,670.453857	$ 6,098,992.90	$ 6,507,572.90	Ξ348.549261	$ -	$ 1.242352	0.00%	52,381,087	$ 65,075,728.99
Monthly Subscribers	Ξ18,670.453857	$ 7,158,330.19	$ 7,158,330.19	Ξ383.404187	$ 1.412554	$ 1.389061	11.81%	53,162,055	$ 73,845,315.44
Rebalance	Ξ19,260.435299	$ 7,158,330.19	$ 7,384,531.54	Ξ383.404187	$ -	$ 1.389061	0.00%	53,162,055	$ 73,845,315.44
Monthly Subscribers	Ξ19,260.435299	$ 7,015,304.97	$ 7,015,304.97	Ξ364.233978	$ 1.553543	$ 1.467932	5.68%	54,120,677	$ 79,445,482.56
Rebalance	Ξ21,811.661556	$ 7,015,304.97	$ 7,944,548.26	Ξ364.233978	$ -	$ 1.467932	0.00%	54,120,677	$ 79,445,482.56
Monthly Subscribers	Ξ21,811.661556	$ 7,308,984.40	$ 7,308,984.40	Ξ335.095260	$ 1.800544	$ 1.562716	6.46%	55,237,283	$ 86,320,163.25
Rebalance	Ξ25,759.887894	$ 7,308,984.40	$ 8,632,016.33	Ξ335.095260	$ -	$ 1.562716	0.00%	55,237,283	$ 86,320,163.25
Monthly Subscribers	Ξ25,759.887894	$ 8,286,735.67	$ 8,286,735.67	Ξ321.691449	$ 2.013584	$ 1.734810	11.01%	56,585,213	$ 98,164,609.99
Rebalance	Ξ30,515.144303	$ 8,286,735.67	$ 9,816,461.00	Ξ321.691449	$ -	**$ 1.734810**	0.00%	56,585,213	**$ 98,164,609.99**

Figure 4: Premium Subscription Token Effects

4.2. CONSUMER DATA SEARCH

Consumers using the Mercury dApp to manage their finances own their data. The live connection to their various financial institutions is collected and parsed in the Apollo oracle so that it may be used as input to intelligent tools for the user's benefit. Additionally, a user may choose to varying degree how much of that data to share to service providers or researchers and marketers, on a per contract or global setting. Marketers may be interested to know across the entire user base, how certain demographics are spending money or what categories of transactions are trending in time. The Aurora dApp allows these marketers to create such queries and execute searches so that users whose data is found as results may earn micro transactions worth of AST. The scenario sequence in the network is as follows:

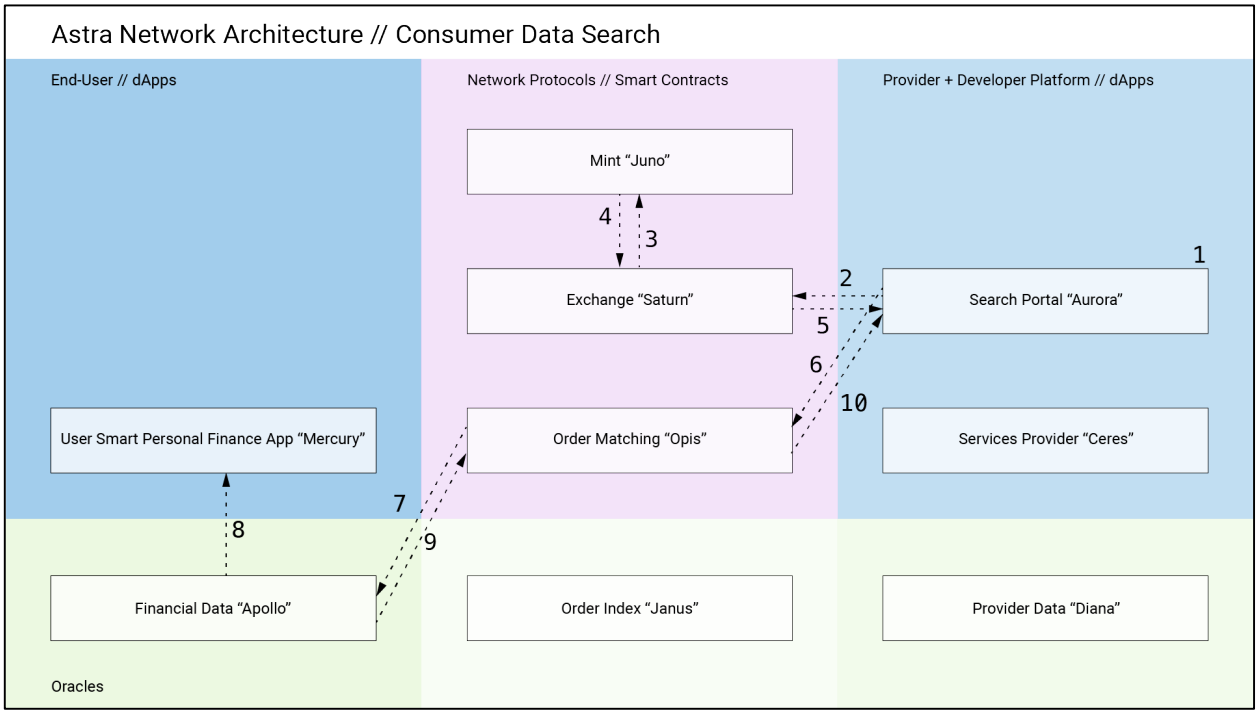

Figure 5: Consumer Data Search Sequence (Requires updating after Step 7)

1. User/enterprise creates a registered account with Astra's Search Portal aka "Aurora" dApp
2. User sends request to exchange contract with fiat currency to access search functionality by pre-purchasing SOL tokens to fill search fees
3. The Exchange contract splits the fee into two portions sending the majority amount to the Mint contract and the remainder fee to Astra Inc's wallet
4. The Mint exchanges fiat currency returning the equivalent minted value in SOL tokens and rebalances the 20% constant reserve for the system (10% in USD and 10% in ETH, rebalanced at X min increments to stabilize value and encourage appreciation)
5. The Exchange contract sends the SOL tokens back to the user's wallet in the Aurora dApp
6. The Aurora dApp sends a search request to the Order Matching contract "Opis"
7. The Opis contract queries the Apollo Oracle for results

8. Search results returned to Opis include Mercury user public key
9. Found public keys are used to distribute a predetermined portion of the search fees and sent to associated wallet address in the Mercury dApp

4.3. END USER LOAN REQUEST

Within the Mercury dApp an End User may create a request for a financial service, such as a loan, that is sent as the meta data and terms of a smart contract describing the service to the Order contract Opis. This request is then sent to the order matching channel Janus so that service providers may search for as well as offer to fill the request. Upon acceptance by both agents, the Exchange contract executes the smart contract order with existing AST tokens from the Service Provider's dApp wallet to the End User's dApp wallet. From there, the user may execute any number of actions to utilize those funds. If the service is a loan, any repayments or other transactions will be automated by the smart contract according to the agreed upon terms.

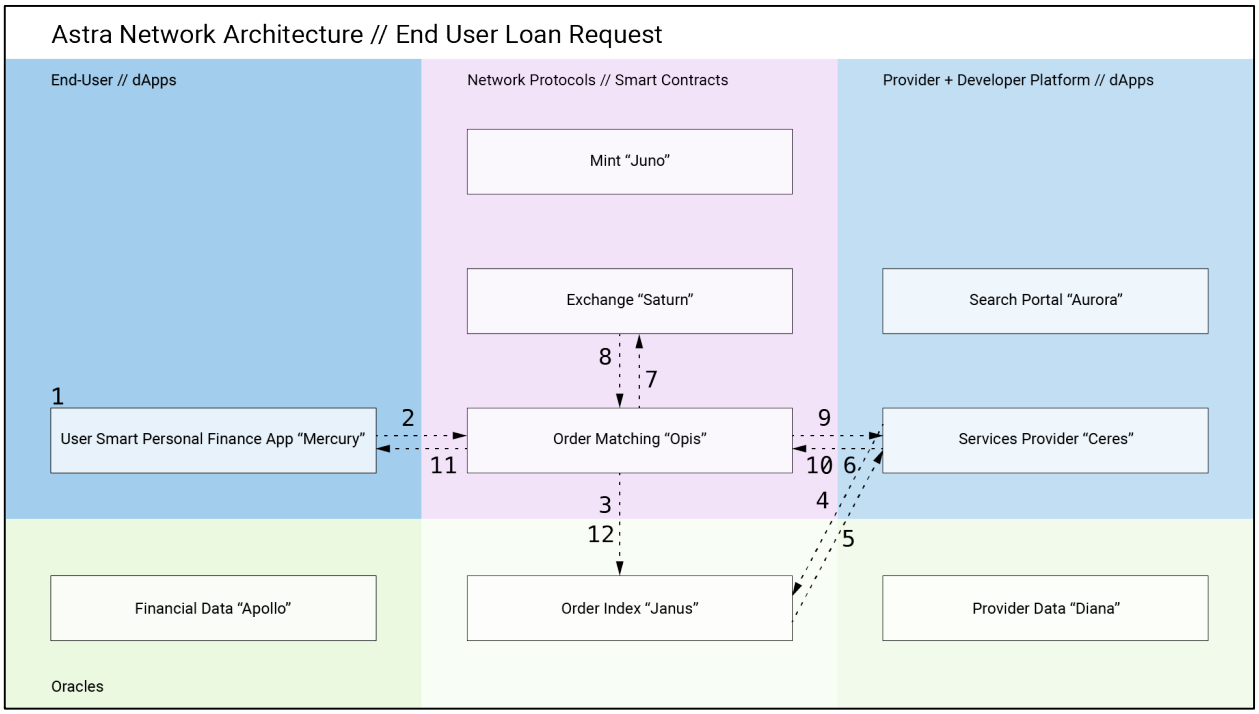

Figure 6: End User Loan Request Sequence (Needs updating)

1. Existing Astra Mercury dApp user
2. User sends a loan request to the Order Matching contract (Opis)
3. Opis broadcasts the request to the Matching Channel Order Index (Janus)
4. Registered Service Provider holding SOL tokens already uses the Ceres dApp to search through open orders
5. If provider finds an order to fill, Janus returns the public key of the initial loan request contract to Ceres
6. Ceres sends request key and required order filling signature to Opis
7. (Clarify)
8. (Clarify)

9. Opis sends order confirmation to Ceres
10. Ceres sends SOL tokens to Opis
11. Opis sends loaned funds and request confirmation details to Mercury dApp wallet (user has option to exchange SOL for fiat currency into one of their connected accounts)

4.4. END USER LOAN REQUEST WITH TWO WAY DISCOVERY

As in the prior example with a End User service request, once the order is sent to the order matching channel Janus, both user and provider parties may wish to query additional information about the other's user profile. After finding the request, the provider may pay a small fee to discover more details about the user requesting the service, which may range from use of credit cards to forecasted account balances. The End User must grant access to their data and may start a discovery process themselves for information about the provider i.e. what is their contract volume by number or value. The provider must also grant access to this information. Direct communication back and forth between the agents may continue through each agent's dApp and the matching channel. Once the terms and service request are agreed upon, the same process of smart contract creation and execution proceeds.

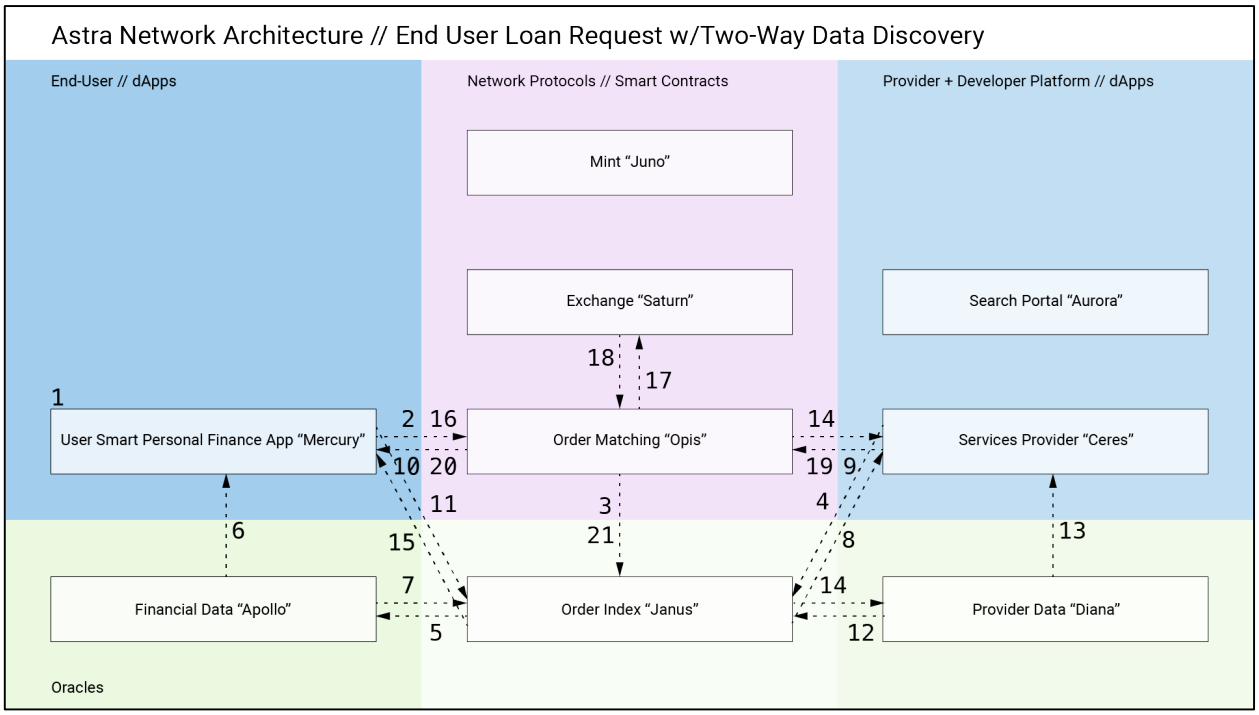

Figure 6: End User Loan Request with Two-Way Data Discovery Sequence

1. Existing Astra Mercury dApp user (with minimum value held in SOL wallet?)
2. User sends a loan request to the Order Matching contract (Opis)
3. Opis broadcasts the request to the Matching Channel Order Index (Janus)
4. Registered Service Provider holding SOL tokens already uses the Ceres dApp to search through open orders in Janus
5. If provider finds an order to fill, Janus sends search request to Apollo to review requester's financial data (search fee to Astra Inc)

6. Found public keys are used to distribute a predetermined portion of the search fees and sent to associated wallet address in the Mercury dApp
7. Requester search results returned to Janus
8. Requester results returned to Ceres for review
9. Provider sends a request fill message to Opis with provider public key
10. Opis sends fill message with provider public key to Mercury dApp
11. Mercury sends search request to Janus
12. Janus sends public key of provider to Diana
13. Provider search results returned to Janus
14. Provider search results return to Mercury
15. If user approves of provider and fill offer, Mercury sends approval to Opis
16. Opis sends the financial service contract to Saturn
17. Saturn executes contract and returns fill message
18. Opis pulls funds from Ceres
19. Opis sense funds to Mercury
20. Opis updates Janus with completed / liquidated loan request

THE ASTRA FOUNDATION

The Contracts on the Astra Network are released as open source network protocols. To manage the development of the of these smart contracts and to act as stewards to the reserve currency and protocol token, we propose the founding of a non-profit Astra Foundation. Astra Inc will contribute significant financial and engineering support to the development of the open source software and the Astra Foundation will assume policy, public communication, and research support activities. The foundation will work to benefit the broader Ethereum platform as well to support the ERC20 token ecosystem. Additionally, any funds raised beyond the token offering target will be held by the foundation and subsequently released in tranches to Astra Inc and any other contributors after successful milestones are reached.

INITIAL TOKEN OFFERING

The Astra Network will be initialized with an Initial Token Offering. This offering will only occur after all Contract, Oracles, and dApps are live and audited, with proceeds managed by the Astra Foundation. Our basic ambitions and goals are listed as follows, with more detail to be determined[37]:

- Create a user incentive pool to encourage signing up, connecting financial institutions, and referring contacts

[37] http://vitalik.ca/general/2017/06/09/sales.html

- Create a bug bounty pool for the foundation to reward the identification and resolution of system bugs
- Limit token purchases to discourage purchases from holding significant percentages of tokens
- Define token offering settings such that anyone wanting to purchase tokens may without additional cost or penalty for rushing purchase
- Do not separate individual holdings of early contributors or investors from those of Astra Inc to maintain alignment of development incentives

SUMMARY

Through the Astra Network, we seek to define “new rails” for consumer finance where the individuals own their data and are empowered to achieve a brighter financial future. To attain success, this core infrastructure for financial services must include the elements we describe herein: accessible, user-friendly applications for agents in the network; a generalized, hybrid centralized/decentralized developer platform, a marketplace for the creation and exchange of user created financial services; and a transparent and aligned incentive structure to reward positive behavior. We are passionate that a robust ecosystem built with these principles and the technical definition under development for the Astra Network can help us as individuals and as a broader economy achieve that future.

ACKNOWLEDGEMENTS

First, we would like to acknowledge the work and commitment of all of the Ethereum contributors and the Ethereum Foundation. Without the Ethereum platform and the protocols it enables, we would not be able to envision nor execute the components of the Astra Network. We also would like to express our sincere gratitude to the reviewers, advisors, and mentors for their feedback and contribution to this whitepaper.

APPENDIX

A.1. FORMULAS

A.1.1 ASSUMPTIONS

For ease of calculation of examples, we assume that the Network in initialized with a token sale of 50,000,000 AST tokens at a starting value of $1.00 that can be purchased in either USD or ETH. The Constant Reserve Ratio (CRR) for AST is 20% equally weighted between USD and ETH at 10% each. The conversion rate from ETH to USD is $337.82.

A.1.2. VARIABLES

FORMULA VARIABLES			
Variable	Description	Data Type	Value at Network Initialization
P	Price of AST Token	USD	$1.00
R_i	Reserve Balance	{USD, ETH}	{5,000,000 USD, 14,800.78 ETH}
V_i	Reserve Value	{USD, USD}	{5,000,000 USD, 5,000,000 USD}
C_i	Reserve Currency Price	{USD, USD}	{1.00, 337.82}
S	Supply of AST Token	AST	50,000,000 AST
F_i	Constant Reserve Ratio (Per Reserve Currency)	{float, float}	{.10, .10}
E_i	Value Paid into/Received from Network	{USD, ETH}	
T	Tokens Minted/Liquidated	AST	50,000,000 AST
O_i	Offset Balance (Per Reserve Currency)	{ΔUSD, ΔETH}	
G_i	Target reserve balance ratio (Per Reserve Currency)	{float, float}	{.50, .50}
X_i	Amount of Reserve Currency for Exchange during Rebalancing	{float, float}	{0.00, 0.000000}
i	Reserve Currencies	{1, 2, … m}	{USD, ETH}
*	Subscript for variables in examples	{u, h}	{USD, ETH}

A.1.3. TOKEN PURCHASE CALCULATION[38]

To calculate the tokens received in exchange for any combination of E_i reserve currencies:

$$T = S_0 * \left(\left(\prod_{i=1}^{m} \left(1 + \frac{E_i}{R_{i_0}} \right)^{F_i} \right) - 1 \right)$$

A.1.4. TOKEN PRICE CALCULATION

[38] https://drive.google.com/file/d/0B3HPNP-GDn7aRkVaV3dkVI9NS2M/view

To calculate the price of AST tokens relative to both Reserve Currencies Balances R_i at Prices C_i:

$$P = \sum_i \left(\frac{R_i * C_i}{S * \frac{F_i}{G_i}} \right) = \sum_i \left(\frac{V_i}{S * \frac{F_i}{G_i}} \right)$$

A.1.5. TOKEN SUPPLY CALCULATION[39]

Tokens can be bought and sold for any combination of reserve currencies, so in general:

$$\Delta S = \sum_i \left(\frac{\Delta R_i}{C_i} \right)$$

To calculate the token supply at any time step *t*, given that bi-directional flows of each reserve currency may occur at a prior time step:

$$S_t = S_{t-1} * \prod_{i=1}^{m} \left(\frac{R_{i_t}}{R_{i_{t-1}}} \right)^{F_i}$$

A.1.6. TARGET RESERVE BALANCE RATIOS CALCULATION

The Astra Network holds two currencies in reserve (USD and ETH), with a target reserve balance ratio *G* per currency Constant Reserve Ratio *F* calculated as:

$$G_i = \frac{F_i}{F} \quad where \quad G = \sum_i G_i = 1.0$$

A.1.7. RESERVE VALUE CALCULATION

The Astra Network holds two currencies in reserve (USD and ETH), with respective values V_i in USD (as Reserve R_i multiplied by Currency Price C_i) and total value *V* and difference of value across time ΔV:

$$V_i = R_i * C_i \quad and \quad V = \sum_i V_i$$

$$\Delta V_{i_t} = \left(V_{i_t} - V_{i_{t-1}} \right) \quad and \quad \Delta V = \sum_i \Delta V_i$$

A.1.8. OFFSET VALUE CALCULATION

The relative value per reserve currency *V* is balanced periodically by offset value O_i per currency calculated as:

$$O_i = \Delta V * G_i - \Delta V_i \quad where \quad O = \sum_i O_i = 0.0$$

[39] https://drive.google.com/file/d/0B3HPNP-GDn7aRkVaV3dkVI9NS2M/view

A.1.9. REBALANCE AMOUNT CALCULATION

The relative amount per reserve currency to exchange X_i as calculated from the offset value O_i:

$$X_i = \frac{O_i}{C_i}$$

A.2. FUTURE WORK

A.2.1. ON-GOING WORK

Currently, Astra Inc is conducting an Early Adopter program for the precursor app to the Mercury dApp. More information can be found at the below URL – the app is live and demonstrates our application of deep learning development to personal finance:

https://beta.astra.finance

Additionally, we see this whitepaper as a living document that will be hosted on Github. We will continue to add detail to this whitepaper and seek feedback.

https://github.com/gilakos/astra-network-whitepaper

A.2.2. FUTURE WORK

Beyond the use cases illustrated here, we envision the generalized Astra Network infrastructure enabling more complex financial services and products. Specifically of note are: consortium service providers (that may fund a group provider smart contract address to service more or larger contracts); derivatives market making through synthetic instruments (bundling service contracts for resale); and additional dApps and oracle services to extend the functionality and data richness of the network (financial advising offered to end users through their dApp and an underwriting oracle for additional agent assessment).

A.2.3. OPEN QUESTIONS

As a draft of a high-level network architecture, we acknowledge this whitepaper leaves open questions about certain details and second order consequences:

- While we plan to embrace standard “know your customer” and “anti-money laundering” conventions for applications and not the pseudonymous options available to decentralized systems,

current and trending governmental policy, specifically in the United States, leaves many regulatory questions and implications undefined.

- Given that our reserve currency rebalances periodically, at scale the network may be exchanging large amounts of USD/ETH in the broader market. This could lead to scenarios where external actors front run our transaction, thereby affecting the market price of either currency and negating the intention of our Mint contract. We may need to add consensus or randomness rules to this protocol to achieve the intent while maintaining transparency of the token value mechanisms.
- In the current design, the Order contract requires a small fee to store the request or offer contract on-chain and agents must grant permission to requests for their data through their respective oracles. This presents challenges for a user-friendly system in that this could potentially lead to either spamming the system if the fees are not calibrated and/or the spamming users with data requests if the volume is high or the settings obtuse to use in the dApp. Users should own their data and have control over its visibility, but that needs to be easy for the average user to engage.
- The intelligence offered in the dApps and data served through the network's oracles are enabled by more standard cloud services run in a centralized schema. This is required to facilitate features for users but presents a security risk not present in the decentralized elements of the system. While we do not store credentials for users' financial institutions and the majority of the data stored is not personally identifiable, additional measures for hashing that data or establishing secure connections to the oracle endpoints may be necessary.