Dash Evolution

DashPay and Social Wallet

Overview

Rev 1
Evan Duffield - evan@dash.org

NOTE: All Dash Evolution documentation, designs and source are in the research & development phase and subject to change. To access all materials, follow the development, or contribute, visit the Dash Evolution Github (https://github.com/evan82/dash/)

Contents

1 Introduction
- 1.1 Dash Evolution Design Goals

2 DashPay
- 2.1 Barriers to crypto-eCommerce
- 2.2 The first decentralized payment gateway
- 2.3 Web Integration
- 2.4 Software Development Kits (SDKs)
- 2.5 Purchase Process
- 2.6 Decentralized Gateway Merchant/Customer Benefits

3 Social Wallet Design
- 3.1 Intuitive user experience
- 3.2 Design Rationale
- 3.3 Name First Design and wallet compatibility
- 3.4 Name reservation signup steps
- 3.5 Common Experience / Syncing
- 3.6 Fee Structure
- 3.7 RPC API Safety and Masternode Quorums
- 3.8 Transaction Confirmation via DashDrive COR

4 Social Wallet Process
- 4.1 Login and Registration
- 4.2 Create New Account / Login
- 4.3 Friends List - Liking (People and Companies)
  - Friend List - Searching / Inviting / Adding
  - Wallet migration, reputations and search
- 4.4 Pay to username / email
- 4.5 Fiat Conversion Services
  - Decentralized Fiat Conversion Benefits

5 Conclusion

1 Introduction

The core mission of Dash is to build the future of money. We seek to create a currency that is public when it wants to be, private when it wants to be, perfectly fungible, instant to transact, self-governed, and financed/developed in a decentralized and predictable manner. We seek to do all these things while being user-focused from the start. We want Dash currency to be easy for anyone to use and understand.

1.1 Dash Evolution Design Goals

Here we will describe the vision and goals of Dash in the best and most thorough way possible. We seek to create money that is social, easy to use and easy to get money into and out of. Our goal is to create social wallets that are easy to get started with, easy to use, and and help users to better manage and spend their funds.

Dash seeks to build a cryptocurrency that resolves all of the issues plaguing cryptocurrencies today while providing an experience similar to a service like PayPal. The issues we seek to resolve are: ease of use, privacy and fungibility, speed, a robust and scalable network infrastructure, implementation of a decentralized API, decentralized financing, and decentralized governance.

All this requires Dash to be accessible on an incredibly fast network utilizing today's best technology. Masternodes running on high-end servers will provide the robust infrastructure necessary to implement the open source technology that forms the core of the Dash Network. InstantX, DashDrive, and Masternode quorums will all be used to create a lightning-fast, high-capacity network capable of competing with financial clearing houses such as VISA, MasterCard, and American Express.

In this document we outline a complete plan for bringing a product to market that can be used as efficiently as a centralized service like PayPal, but is decentralized, inexpensive to use, and extremely robust.

2 DashPay

2.1 Barriers to crypto-eCommerce

One of the main areas where we think cryptocurrencies like Dash need to improve usability is when used for buying goods and services on the world wide web, or retail eCommerce, a market with total transactions estimated to reach $2 trillion in 2015[1].

Today there are many cheap and easy to implement centralized payment services available for eCommerce solutions, such as PayPal, Google Wallet and Visa/Mastercard.

The problem with implementing cryptocurrencies for eCommerce is that to stay trustless, merchants and customers need to host their own infrastructure (full-node), otherwise an intermediary is required such as a centralized payment service like a web-wallet or SPV server, meaning the user does not retain control of their funds.

Hosting your own fullnode and (for merchants) maintaining it as part of your website infrastructure creates additional time and cost, isn't friendly to use or implement, and as most users instead opt to use centralized payment gateways to integrate crypto eCommerce to their websites and apps, the main benefit of decentralized currencies as a way to conduct commerce between users who retain full control of their money without needing trust of a 3rd party is lost.

2.2 The first decentralized payment gateway

What we want is to remove these barriers whilst keeping the process trustless and decentralized, by creating a web-enabled cryptocurrency that can be integrated seamlessly into existing eCommerce markets without any use of centralized services. This enables us to provide a more competitive payments option that is easy to use, cheaper than centralized systems and above all, trustless and decentralized, helping to drive mainstream cryptocurrency adoption.

To support this we are introducing the concept of a decentralized wallet and payment gateway protocol called DashPay that gives web merchants and their customers trustless access to Dash's payment functions without needing to run their own fullnode or resort to a centralized 3rd party payment or SPV service.

[1] "Democratization of Ecommerce Report - SlideShare."
<http://www.slideshare.net/bigcommerce/democratization-of-ecommerce-report>

2.3 Web Integration

DashPay uses common transport protocols such as HTTP and RCP meaning it's easy to integrate into existing eCommerce solutions in websites and apps at no extra cost to merchants

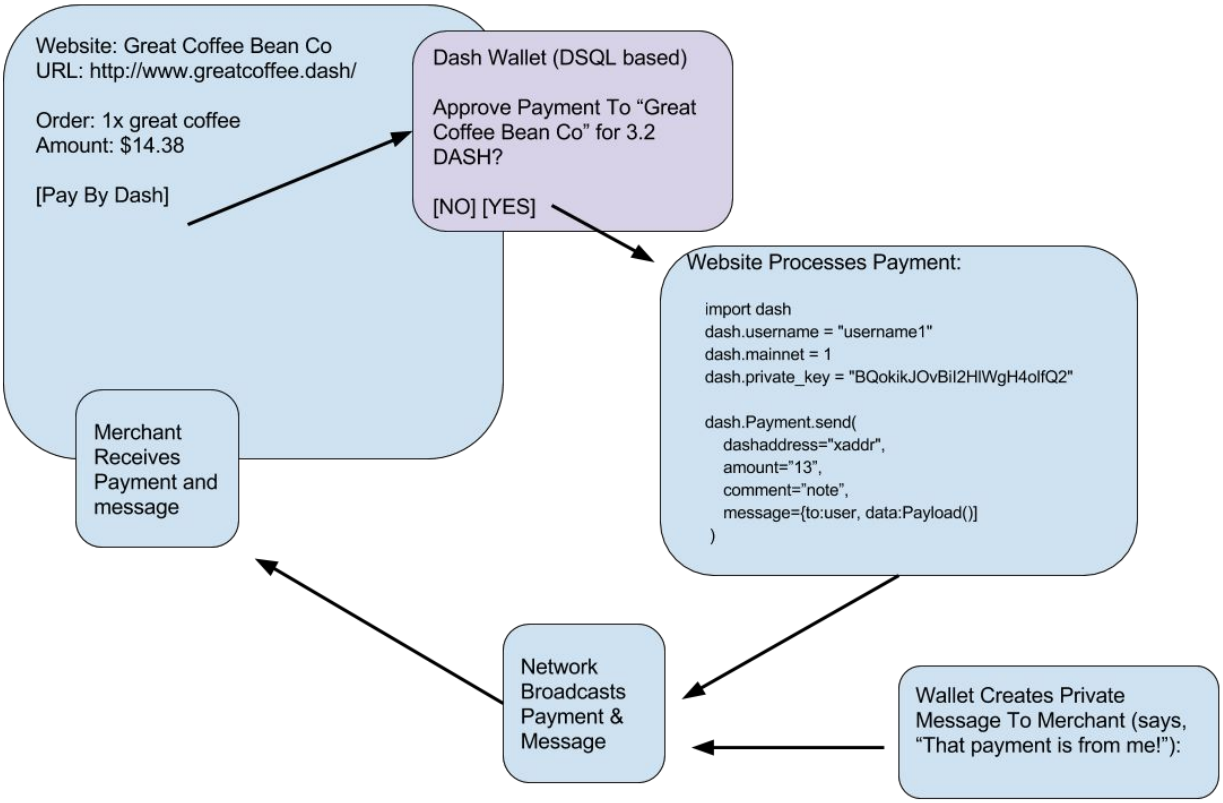
Users can buy merchandise directly from websites without the need for centralized payment services and merchants can implement a trustless eCommerce solution into their website without needing to use 3rd party payment processor or host their own infrastructure.

The DashPay protocol enables direct access to the Dash P2P Network via a decentralized API (DAPI) where requests are distributed across thousands of Masternodes operated by Dash users meaning end-users don't need to trust any 3rd party node or single point of failure to make trustless payments natively on the web.

2.4 Software Development Kits (SDKs)

DashPay will be provided with a suite of open source SDK tools that will enable developers to quickly integrate Dash Pay into websites using Javascript and into apps on all major platforms such as OSX, Windows, Linux, iOS and Android using languages and design patterns familiar to eCommerce developers.

2.5 Purchase Process



Users will be able to buy merchandise directly from websites without the need for any centralized payment services. This will allow the network to process it's own purchases, using only our hardware. This process can be subsidised by the network itself, providing a nearly free-to-use payment processing system with decentralized fiat conversation.

2.6 Decentralized Gateway Merchant/Customer Benefits

- Much lower cost than PayPal, Visa/Mastercard, GoogleWallet and no hidden fees
- Quick-integration, familiar process to centralized payment processor integration
- Robust access with JSON based communication and per-platform SDK
- B2C (merchant) focused
- High-availability decentralized infrastructure (3,300+ backend servers)
- All functions accessible through single API layer
- Advantage to merchants to retain full control of funds and receive deposits instantly
- Advantage to customers of paying direct using an intermediary

3 Social Wallet Design

Dash Evolution is designed to be user-focused and usage centers around the concept of a profile. Profiles can be public or private and users can have more than one profile. If users choose to, they can store their public information on the network so that friends and family can find them and form payment groups.

Once friends have connected, all interactions between them are completely private, including the sending of messages or money. Anonymous transactions can also be conducted with a pseudonymous ID that has no connection to the user's public profile.

We seek to create something unique and useful: a social-based cryptocurrency that is as easy to use and intuitive as cash. Imagine inviting your friends, tipping celebrities, or sending micropayments directly from a website or app without ever having to use a complicated, unintuitive payment address again. All of this becomes possible by fully leveraging the Masternode network.

Users will find the experience useful because they can form social connections with friends and contacts they can do finance with. Users can grow their network and invite more people to join with the incentive of being instantly connected and able to use funds together.

3.1 Intuitive user experience

We want to make a wallet that is easy and intuitive to use for mainstream users, where there are no complex addresses, no sync time or unintuitive interfaces, or anything else unfamiliar to your average person.

This is possible with Dash Evolution, which will support an extremely intuitive and simple design while keeping the features we love like instant transactions and privacy. This will be achieved through the world’s first Distributed Application Programming Interface (DAPI). When you sign up for an account, you will simply create a profile with username and email address. You can create the account as “private”, which means it’s not searchable on the network, or you can create an account that is “public” and can be found by your friends.

After you create an account, you can search for your friends and add them, or send them an invitation to join. If your friend adds you back, your accounts will automatically exchange five new addresses which will be used for future payments. This process will all happen behind-the-scenes without presenting confusing messages and options to the users. There will no more need for complex cryptographic addresses; users will simply transact business by using the real name of their friends. We’re calling this concept “name-first” design.

All wallets will use deterministic seeds that store your keys on the device (not on the network). Other information such as your profile (name, email, etc.) will be stored on the network and will be publicly viewable (unless you enable privacy mode). Your transaction history with your friends and your private messages will also be stored on the network in a decentralized way in an encrypted file that only you can access.

Our goal is to make cryptographic currency easy enough that anybody can use it; you no longer have to be a computer engineer use the currency. We need to redesign cryptographic currency from the ground up to give it a very accessible high level design that is consistent across the ecosystem. The key to mass adoption is giving people something they can use and understand without any instruction. We can do this by removing all of the underlying complexity and present a well designed product. This includes safe, secure retrieval of information which cannot be lost as long as the user remembers their profile name and passphrase. There will no longer be any need for the storage or backing up of wallet.dat files.

This design will offer even greater anonymity than before, because you will only access your wallet through the Masternode network. When you want to do anything on the network, you will access the network through the decentralized API. Each request will go to one of over 3,300 Masternodes, so your network footprint will be spread out..

For even more security and privacy, a user could access the network through proxies. All requests will be done through a restful API or through websockets, which are very easy to protect by using proxies.

Your device will store information on the network in two forms: public and private. Your public data will be accessible to everyone and will allow friends to find you so you can send each other money. The second form is private encrypted data, which will include your friends list, transaction history, private messages, and the key cache for sending friends money.

Each piece of data will be stored on 3-10 Masternodes in our decentralized storage network. This will allow us nearly limitless expansion ability and blockchain growth. We will also support an “archive mode” which will require setting a few individual servers with terabytes of storage. This will act as a backup in case multiple Masternodes go offline and lose a piece of data. These can be financially supported by our decentralized budget system [1].

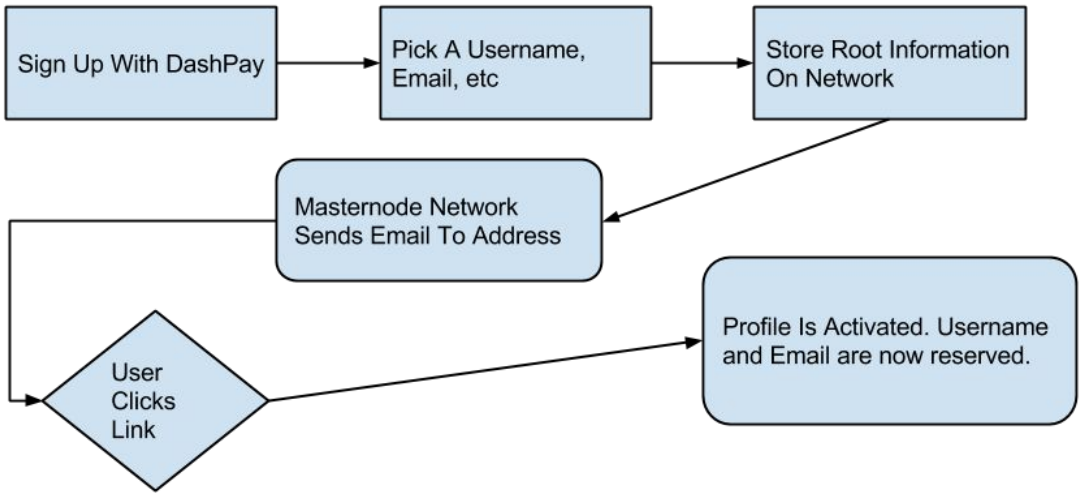
With this system your money is not stored on the network, it’s in your seed phrase, so only you have access to it. Also, no matter how or where you log on to the network, you can query the network for the information that is stored such as your profile information (name, email, etc) and private encrypted information (messages, friends list, friends key cache, transaction history). You will immediately sync with the network and have exactly the same setup no matter what device you use to log in. .

Anonymity in such a system is also much higher than running a node on the network. The only interaction you have with the system will be to store your profile data (which can be an alias and in privacy mode) and private encrypted data, which is done by communicating with a Masternode API endpoint. You will connect to 1 of over 3300 Masternodes for each query, creating small bits of information on each Masternode, but your complete information will not be identifiable by anyone on the network. For extra security, proxies will be made available. These proxies could run as extra services on some/all of the Masternodes.

3.2 Design Rationale

- Nearly all complexity is hidden. Users are never shown transaction hashes (they can see them if they wish), confirmations, or other cryptographic information.
- The network is split into two components: the front-end and the back-end. These are connected between a decentralized API layer.
- The backend can be updated independently of the front-end.
- API endpoints will be consistent and backwards compatible.
- Sending money to your friends can be done with high anonymity.
- Sending money without a friendship link will require retrieving an address from the network which was never encrypted. The public name query form of sending money is less private but is still reasonably anonymous due to the sharding system which it’s stored on.
- Sending money to websites via the SDK will also be highly anonymous.
- Friends have a trusted relationship on the network and can pass information back and forth between each other privately using encryption.
- All parts of the network can be accessed with a simple API that is completely decentralized and trustless.

3.3 Name First Design and wallet compatibility



The most promising feature of a common API for all of the wallets in the ecosystem is the fact that we can achieve a reliable and consistent user experience for all software used on the network. Users will be able to move from one wallet to another without losing any transaction history, friend lists, or other information they have acquired.

Meanwhile, software developers will find it much easier to build applications on top of the Dash API, due to the use of familiar eCommerce and social network integration patterns used and quick-start SDKs provided for all major platforms.

3.4 Name reservation signup steps

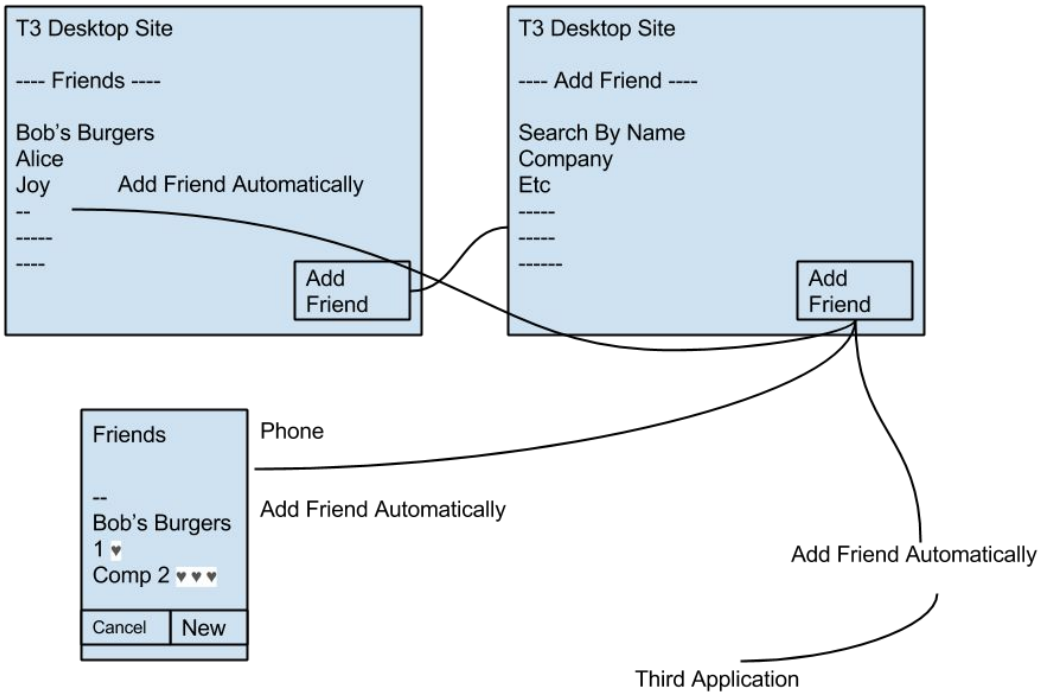
When a user decides to join the Dash Network, we want to make things as familiar as possible while providing a consistent user experience to everyone using the network. This is a hard problem for decentralized systems, such as cryptocurrencies, to solve. However, we have a solution to provide a superior experience from start to finish.

1. Name reservation - The first interaction with the network a user has is picking out a username and email. Both of these items must be unique and are permanently reserved for future use on the network.
2. After picking a username and email, a Masternode quorum will be chosen and an email will be templated and sent out via an API. Masternode operators can choose to do one of two things: they can ask for access to our email service or they will be required to get a domain and setup an email service to send from.

3. When the user receives the email, it will task them with entering a 6 digit number into the system and the same Masternode will approve the code and finalize reservation of the username and password.
4. In the case where the user was sent money via our “send money to invite user” feature, it will be credited to the account at this point.

By allowing reservation of usernames and email addresses, we can ensure when you search for the user, imposters will not show up.

3.5 Common Experience / Syncing



All clients on the network must use existing network primitives to access the network. These objects are saved on the network and must be displayed on all T3 device implementations, therefore users will enjoy a common experience for all software our ecosystem.

3.6 Fee Structure

The Dash Network uses a trust model by running an internal pagerank like algorithm on users’ average ratings. By creating an accurate trust-based rating system on the network, we can compute trust per user and allow them each a specific amount of network computational time.

All commands on the network use up this time and the quorums you activate during a command will update your user profile. If a user uses up the amount of free time on the network for a given day, month or year, they can purchase more at any time.

The Masternode network by quorum majority functions will configure these numbers until more than 90% of users are able to fully utilize the network for free, with , fewer than 10% of the userbase paying for some extra processing.

3.7 DAPI Safety and Masternode Quorums

Some of the best experiences on the internet are served via Web 2.0 technology, which has an active internet connection with a trusted third party. There are many popular services that function in this way, such as Facebook or PayPal. These technologies have provided huge improvements to our experience of using the internet, but they fail to apply to cryptographic currency due to the need for a centralized third party.

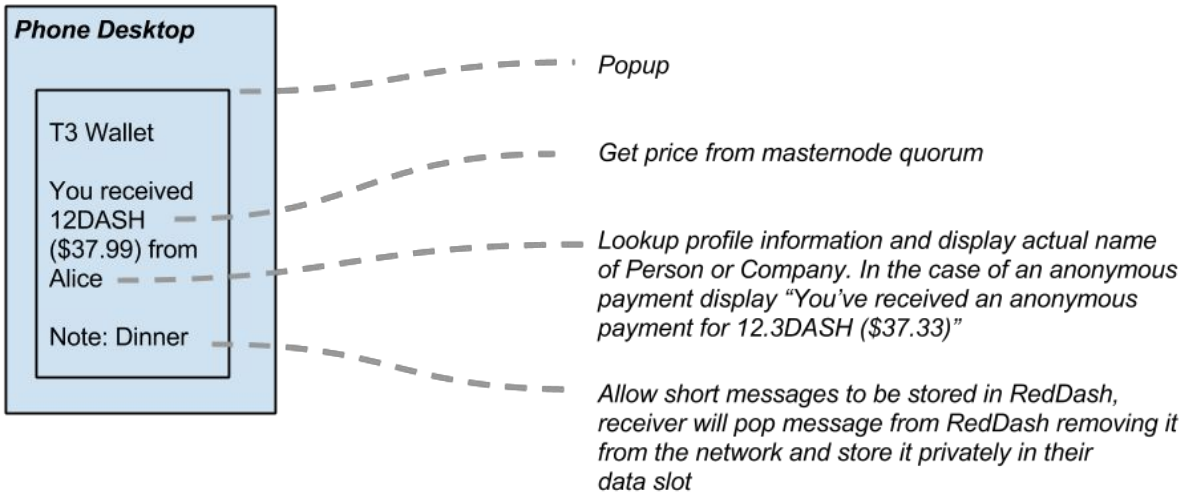
With cryptocurrencies like Bitcoin and the current version of Dash, all operations on the network must be approved by all nodes on the network. It would be impossible to create a distributed currency in this way and still scale to the level of industry leaders such as Visa. If every node has to process every transaction in parallel, centralized services will always have the advantage of requiring only a fraction of the computing power and energy that a decentralized service would require. The key to unlocking the potential of cryptographic currency is to use decentralized cryptographic currency networks in the same way we use centralized services. By using Masternode quorums, we can have hundreds of mini-networks processing transactions simultaneously yet securely. If all/most Masternodes come up with the same result, the network can consider the matter resolved and trust the outcome.

When accessing a service for money, the underlying technology must be very secure. To achieve a high level of security we rely on Masternode quorums. A user will simply execute a command to a specific Masternode then that Masternode will answer the request themselves and query each Masternode in the quorum for an approval signature. Each quorum signature signs a hash of the data being sent back to the user. This allows the user to check each signature themselves to know if there was any tampering.

Because of this, the RPC API is as safe as running a full node because the nodes can’t lie to your client and your client can trust all information it received. If it finds an issue with some of the data it has received, it can simply request it again from another Masternode, which will then repeat the process and give the same answer.

3.8 Transaction Confirmation via DashDrive COR

To improve user experience we need to be able to provide instantaneous delivery of funds in a non-reversible / non-complicated way. To achieve this we use something called “Commit or Rollback” (COR) operations on DashDrive to lock funds from an input to one or more outputs This way if a lock succeeds it’s impossible to double spend on the network. This feature allows us to vastly simplify our wallet GUI and remove confirmation information except in the cases where instant transactions fail due to attacks or network issues.



No confirmation is needed later on, therefore the GUI can simply show “You received money from Alice.” When you receive a payment, the address will be looked up in our global database and alias information will be found and displayed.

InstantX seeks to provide merchants with 99.9999% coverage from double-spending attacks and reduce the complexity of interacting with cryptographic currency.

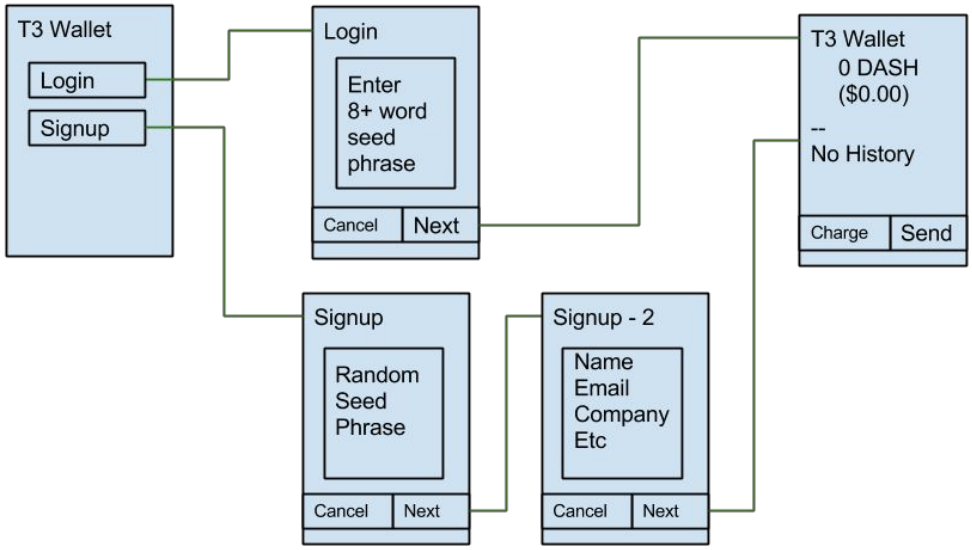
For more information on security of InstantX and Masternode quorum technology, see our whitepaper at: http://www.dashpay.io/whitepaper.pdf

4 Social Wallet Process

4.1 Login and Registration

The main design goal of the wallet is to make a user experience closer to what is found in traditional finance, something like your banking app. On startup, you will be able to login with your passphrase or you will be tasked with registering. Registration will occur and be stored in a key/value store on the 3rd tier of the network. This will allow any Masternode to access information quickly on other parts of the local network.

4.2 Create New Account / Login



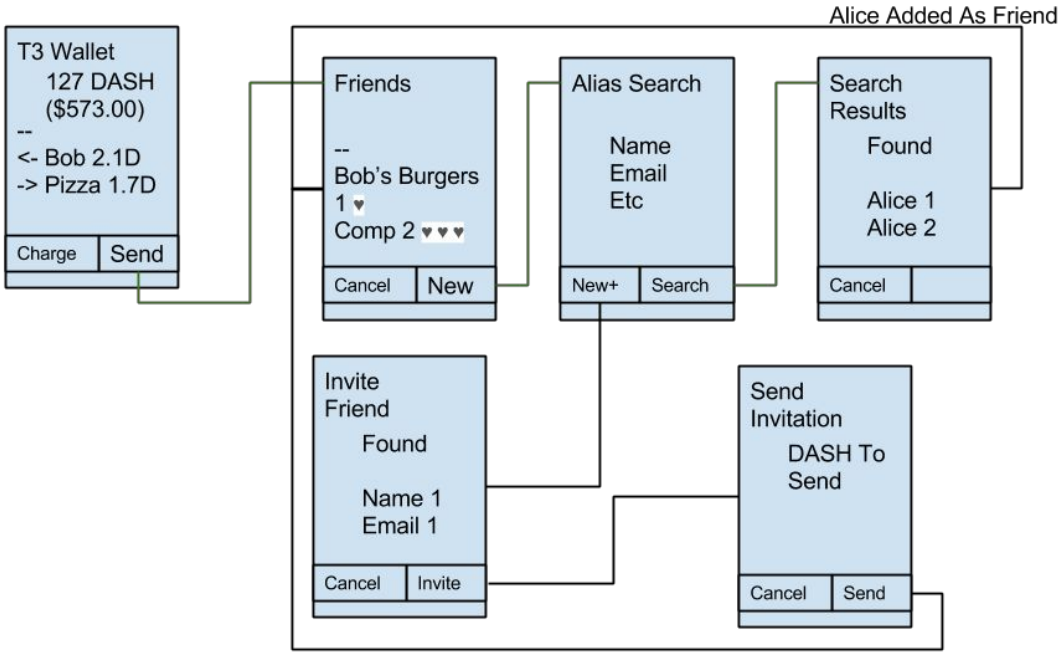
When logging into the network, you will use your full seed phrase, which will then generate the keys that will be queried on the network. Information about your friends and transaction history will be stored encrypted with redundancy on the network and not able to be accessed by others besides yourself.

Profile information for your account is not independently verified so you can use an alias to remain anonymous if you wish. However, if you would like people to be able to find your payment addresses and other profile information you will need to provide public information which will then be searchable by another end user.

4.3 Friends List - Liking (People and Companies)

After login, a user will have access to their friend list. Friends are stored on the network and encrypted using your password for the wallet. You access your friends list and can click their name to pay them. You can ask the API to get five new addresses from your friends for future use, and each time you pay them the wallet will automatically pay the next address in the list. This provides added anonymity, since these addresses are only known by the two of you.

Friend List - Searching / Inviting / Adding



Wallet migration, reputations and search

All T3 compatible wallets will use the friends list to store known addresses of people you commonly pay. As a result moving from one T3 compatible wallet to another is very simple and doesn't require manually moving any data. You can also have multiple wallets on different devices and they will automatically sync from the updated network information .

Any user can rate any other user on the network. This ratings are just a set of stars between one and five. This is a non-private rating system and will show up when looking at company/friend data.

Search data can then be sorted by the average ratings of your friends. By sorting the data by what your friends like, we will improve the search quality greatly while also making the search feature easier to use.

Companies will develop a reputation in the system quickly as our users try their service and rate them.. The best services will rise to the top and receive the most business.

4.4 Pay to username / email

If your payee doesn't exist in the system, you can pay to their email. This will send them an email through the network. By following the instructions in this email, they can sign up through the dash.org website and join the network to claim the money you sent them.
For more information about the privacy on the system, see the transaction privacy section of the DashDrive paper.

All outgoing transactions will use InstantX and all incoming transactions will be instantly confirmed as well. This wallet will be confirmation-less; the word "confirmation" will not even appear in the wallet unless there is an issue. Such problems should be extremely rare.

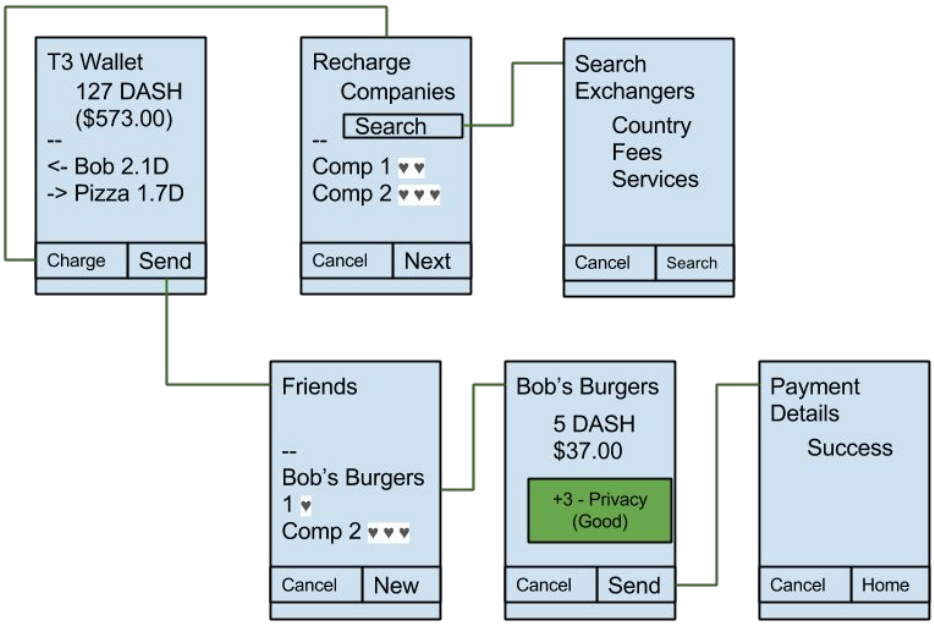
After confirming a friendship, the first task of the wallet is to request five payment addresses from the other friend and store them in your wallet private data online. When you receive money on any of these one-time use addresses, your client will know exactly who paid you and show their name in your payment history.

4.5 Fiat Conversion Services

Users can then use a local fiat converter to get money into the system. Each person will be able to use different local servers that all charge different fees and compete for business on the network. Fiat converters may use services like bank wire, bank transfer, PayPal, or credit cards and will be accessible in every country in the world.

The whole wallet will be confirmation-less, so instant delivery of funds will be a very normal thing to see when interacting with the wallet. Also, instead of "Paid to Xaddr23" your wallet will read "Paid to Alice Baker" due to our name-first aliasing system.

This will be the world's first truly accessible wallet, with zero centralization.



When using T3, all contacts appear as names and never addresses. You can search the public record for friends and add them here. When two people are friends with each other, they can send messages back and forth using the key/value system. All communication will be encrypted with the reader's pubkey.

Privacy is provided through exchange of private keys over encrypted channels, then storage on the blockchain as denominated, merged transactions. This is part of the process of sending a transaction on the Dash Network by default.

Decentralized Fiat Conversion Benefits

- Cheaper cost / lower fees
- 100% decentralized, used from within wallet
- Providers can build reputation and grow services of value
- p2p & p2b & b2p & b2b model supported
- Accessible through simple decentralized API layer

5 Conclusion

Dash addresses many problems inherent in first generation light client implementations which often lack security or scalability or both. Current solutions include Simple Payment Verification, which uses a centralized infrastructure design to support end users, or other centralized services that provide walled gardens such as Coinbase.com or Circle.com.

We propose a decentralized second tier topology of the network, which allows users to access the network in a completely secure way directly from the web or via RCP, while only hitting a portion of servers on the network.

When second tier topology is calculated ahead of time, users are connected to small secure groups and we can implement a decentralized API that is similar to that of a centralized service but also scales effectively.

Users then access the network through light, low-resource implementations built on various technologies that are familiar to the web developer community. These designs also utilize an array of simple primitives such as users, groups, and accounts. By altering these objects, users interact with the network with ease.