# Atto¢ash whitepaper v0.6

*Edward Keyes <ed@attocash.com>*

Last modified: 2023-07-17

**An internet-scale layer-2 micropayment system for Web2 and Web3**

## Introduction

As physicist Richard Feynman said about nanotech, “There’s plenty of room at the bottom.” Blockchain technology has created distributed financial applications of many sorts, but the internet still runs on advertisements. One reason for this is because there is not an available micropayment system suitable for replacing ad-supported content at scale. Paying a penny to read a news article or to post a tweet is not practical because the minimum effective credit-card charge is around $1.00. Transaction fees for popular cryptocurrencies like Bitcoin and Ethereum are even higher, and payments may take minutes to finalize.

Instead we are designing Atto¢ash as a cryptocurrency system which is specifically optimized for the micropayment scenario. It has two key advantages:

- **Digital cash**: Instead of needing to interact directly with the blockchain for each transaction, the user’s browser extension can generate a small, anonymous payment ticket which can be embedded right into an HTTP request to a website. The site can validate and redeem it immediately while the reply page is being generated, or if more convenient can just store it offline to redeem in bulk later.

- **Internet scale**: The system is architected to be able to handle a very large transaction volume, suitable for competing with ad-supported infrastructure even for sites like Google and Facebook. In addition to such Web2 uses, it can transition easily into the Web3 world, where the mechanisms for supporting Metaverse creators and service providers are still being established.

However, the tradeoff we are explicitly making to achieve this requires abandoning some of the guarantees of immutability, reliability, and auditability provided by most blockchains. Instead, we tolerate a small level of inconsistency and even fraud, under the assumption that most transactions will be tiny and not worth the effort to rigorously police. This is similar to how internet advertising already works: click fraud and ad blockers are ubiquitous, but the system still provides enough benefit to both users and providers to be worth using.

The Atto¢ash system is purposefully simple compared to many DeFi projects. There’s no volatile currencies, no staking rewards, no tokens to HODL, no leveraged flash loans. There’s just the high-velocity movement of money with minimal overhead, to promote efficient commerce on the internet for everyone to benefit from.

## Scale

What does it mean to operate at "internet scale"? As one example, Google currently handles about 100,000 search queries per second, each of which has advertisements associated with it. Beyond its own pages, its AdSense network also distributes ads to provide revenue for many other websites across the internet. In comparison, classic blockchain networks like Bitcoin and Ethereum operate at only around 10 transactions per second. Next-generation chains such as Avalanche and Solana are designed for more like 1000, which is also comparable to traditional worldwide payment-processing systems such as Visa.

As a strawman design point, Atto¢ash is instead targeting 1,000,000 transactions/sec, or around 100B/day. Assuming each payment is $0.01, this might correspond to a scenario of 1B internet users spending $1.00/day across 100 transactions, most of which would be browser-automated. This totals up to $1B/day of payments, or $400B/year, about the same as Google and Facebook's revenues combined. Assuming a reasonable 1% payment-processing fee, the Atto¢ash system would itself generate $4B/year, which should be more than enough to support the necessary compute infrastructure.

## Protocol

The system is designed as a layer-2 augmentation to a next-generation blockchain. (We are currently targeting Avalanche, but this is not essential.) Users would start with a normal on-chain transaction, using stablecoins or credit cards to purchase, say, $10 worth of Atto¢ash credits into a local wallet. Their browser extension would gradually deplete these over the next 1000 transactions until it needs to be refilled again. On the other side of the transaction, sites that receive the payment tickets would validate and redeem them using the Atto¢ash server API, and they would periodically "cash out" those credits back to the layer-1 blockchain as stablecoins or into the traditional financial system. Since the blockchain is only involved at the aggregated entry and exit points, the load on it is kept manageable.

How does this work in more detail?

1. Atto¢ash users initially establish a ledger **account**, with a public key and a chosen amount of starting currency. This is a regular blockchain transaction, so the account would be funded with stablecoins obtained through any existing financial on-ramp, or directly with traditional payment methods.

   Unlike regular blockchain accounts, these funds are spend-once only. Rather than a permanent account with a fungible balance that goes up and down, this is more like traveler's checks, as the funds are tagged and single-use, though the account can be augmented with additional funds later if desired.

2. To make a payment, the user's wallet creates a **ticket**. Instead of containing a simple amount of currency, the ticket applies to a specific **span** of funds, imagining them as if laid out on a number-line. For example, if an account is funded with $1.00, then one ticket could spend $0.02 with the span of 0.00 to 0.02, and the next ticket could spend $0.05 with the span of 0.02 to 0.07 in that dollar. Once a span is used, those specific funds are considered spent and are unavailable for future transactions. Note that small fractions of a cent are supported in spans, as of course atto- is the metric prefix for $10^{-18}$.

   The ticket does not need to include a recipient's specific account, as it is redeemable by whoever possesses it. However, sites would typically request that the ticket include a simple identifier such as the website domain name or a random nonce or timestamp to demonstrate that the user generated the ticket on demand, to reduce fraud.

   The ticket is signed with the spending account's private key, known only to the user and kept in their wallet. Note that ticket creation can happen completely offline, as it does not touch the blockchain or even the layer-2 servers.

3. The recipient of the ticket can hold it offline if they wish, but common practice would be to immediately provide it to the Atto¢ash system for redemption. That server checks the ticket signature against the public key of the account-holder, and checks the additional layer-2 information for whether that particular span has already been spent or not. If it passes, the funds are added to the recipient's account provided with the redemption request. This addition extends the available spendable span in the account: if the account was originally funded with $1.00 and it receives a ticket for $0.05, then now it can write its own tickets in the span from 1.00 to 1.05.

4. Periodically, the recipient can choose to transfer portions of their spendable funds back into the layer-1 blockchain, typically converting it to stablecoins or going through other financial off-ramps. A specific span or spans must be selected so those funds can be marked as unspendable going forward, though often recipients would opt to just cash out all of their available funds for simplicity.

The core mechanism here is the use of spans to track ticket spending, which is a little unusual. As an analogy, imagine that your bank gave you a checkbook with a separate check for every penny in your account. You could tear them off one by one and give however many of them you want to businesses like cash, who could redeem them at the bank. And when you run out of checks, everyone knows you have also run out of money.

In addition to a wallet and browser extension, there can be some lower-overhead ways of using the system. For example, a user might be able to create a ticket directly with a credit-card or PayPal transaction, and either send it to someone else or use the Atto¢ash website to split it into smaller denominations. The tickets themselves can be provided as URLs in text or QR codes, so that the recipient can simply follow the link to redeem them immediately via PayPal or a VISA gift card, for instance. This would only be suitable for larger transactions, but it would be useful to let people participate in the ecosystem without commitment at first.

## Security

There are several different categories of security concerns that arise with a digital-cash system.

**Double spend**

This is the most important vulnerability, since data can be infinitely duplicated by its very nature, so any cash-like system needs to worry about someone simply copying a digital dollar and spending it again and again. In Atto¢ash the use of spans to track spending is the key feature to prevent this, as each iota of the funds is tagged uniquely. Any duplication can be detected immediately, instead of having to wait until the user exhausts their whole account balance and the system can start to recognize that they have written $100 worth of outstanding tickets from only a $10 starting amount.

The Atto¢ash servers are looking specifically for such cases of double spending. If the system receives two tickets with different recipient identifiers but covering the same span, both correctly signed by the user, then this is clear evidence of fraud. As a punishment, the user will be assessed a penalty fee from any remaining funds in their account, which is also an incentive for the server to take the trouble to check for this possibility.

**Double redemption**

On the other hand, the system generally does not flag cases of double redemption as evidence of fraud, since it's an easy mistake to make for a site to accidentally submit the same ticket twice, if the first request seemed to fail. Instead, it's an advantage to treat such operations as idempotent: if the ticket is still valid, the system can redeem it, and if it is not, it is just ignored as already redeemed.

Note that the user can also retain a copy of the data, so it is possible for them to issue a ticket to a website, and then rush to self-redeem the same ticket before the site does, retaining the funds. However, this is not really a fruitful procedure, as the website will typically detect the ticket as invalid within milliseconds and deny whatever was being purchased, the same as if the user had supplied garbage data to start with. It could succeed if sites only process tickets in bulk at a later time, though in such a case, supplying garbage data would also pass, so this is not really recommended except when the user is otherwise trusted or the chance of fraud is minimal, such as with a voluntary tip jar. This loophole does also allow the user to reclaim funds which were accidentally sent to a website that didn't support the protocol, for example.

**Eavesdropping**

A ticket can be redeemed by anyone who possesses a copy of it, so it is important that transactions between users and websites be encrypted with standards like TLS/SSL. Otherwise someone could theoretically sniff a network's WiFi traffic and grab any tickets passing by to front-run the redemption process into their own account.

Tickets are a relatively small amount of data, and have a well-defined text format, so it is possible to send them by virtually any mechanism: emails, texts, chats, QR codes, NFC, etc. However, some care will need to be taken against eavesdroppers when doing this, similar to issues encountered in sending paper cash by postal mail, so these use cases are best handled in a trusted environment. In a high-trust situation, tickets could even be created with an empty recipient, like making a check out to “cash”, and circulated repeatedly before being redeemed.

**Anonymity**

Atto¢ash accounts are considered semi-anonymous. Their creation is registered on the public blockchain, so the transaction history of the funds up to that point can be tracked, and likewise the cash-out processes are likely going to be restricted by Know Your Customer requirements. If more anonymity is needed between those steps, tickets could be recycled into a new account.

The actual tickets are encrypted with an Atto¢ash public key, so to the recipient they are opaque data blobs, fully anonymous just like cash. In processing redeemed tickets, the system sees the user’s account and could in theory compile data like a list of paid websites they frequent, but the detailed layer-2 transactions are kept only temporarily instead of in permanent blockchain records, and having an explicit data-destruction policy is advantageous to avoid subpoenas.

**Insiders**

There are also cases of internal fraud to consider, such as from a malevolent Atto¢ash server. We have mostly been considering this system as needing to favor high-speed efficiency over trustless operation and therefore being run as a commercial service instead of a fully open network. However, consensus mechanisms could be implemented for that too, such as having multiple nodes process the same tickets in parallel and compare their results, with any discrepancies punished by penalty fees against the erroneous node.

**Hackers**

The security advantages of the Atto¢ash account should also be noted. It is not necessary to grant a site any sort of wallet permissions to send it a payment, and the only funds at risk of a mis-click are the small amounts already set aside for micropayments, rather than a user’s main blockchain account balances, their NFTs, etc.

In addition to a minimal server API, the attack surface area of the blockchain infrastructure used is also relatively small, with just a few smart contracts for interacting with the layer-2 system. Compared to most complicated DeFi projects, there is little possibility for subtle vulnerabilities like flash-loan or oracle-manipulation attacks.

**Denial of service**

Since the system is designed for distributed processing and scaled to handle large amounts of traffic, there aren’t a lot of opportunities for denial-of-service attacks. However, attacks directed against a specific account may warrant some consideration, as typically that traffic would be routed to a subset of the servers to take advantage of caching and consistency.

# Architecture

The scale of data processing in an internet-wide system like this is not to be underestimated. Recall that Atto¢ash is targeting around 100B transactions/day, so if it had to retain precise blockchain records of every payment indefinitely, this would rapidly climb into petabytes and be difficult for normal distributed nodes to manage.

Instead, the layer-2 system only needs to retain a fraction of the data to successfully operate. For each account, we just need to store a compressed representation of the spans which are still available out of its spendable funds. Under most circumstances, spans will be spent in order, so all of the tickets can be efficiently collapsed into a single span plus perhaps a few as-yet-unredeemed holes. This is how Atto¢ash is able to operate with orders of magnitude more transaction volume than even highly-engineered modern blockchains.

In this minimal implementation, the details about the individual transactions and their recipients are discarded as soon as each ticket has been processed. However, this impacts the system's ability to detect double-spending. Any double-spent span would still be rejected, but without the prior ticket's recipient and signature to compare against, fraud cannot be proven with certainty, as it could be an innocent double-redemption instead.

To solve this problem, we can enlist some economic incentives. As fraud becomes more prevalent, it becomes a more attractive tradeoff for the nodes to decide to retain some individual ticket data for a while in order to attempt to catch a double-spend and claim the penalty fee as a reward. Heuristics about which accounts are higher-risk may enter into this, such as a gap in the spans indicating that a ticket may still be pending offline, but in the end the system should come to equilibrium at whatever level of policing is "worth it".

We envision the Atto¢ash servers as forming a peer-to-peer network which distributes the layer-2 data via consistent hashing of the account address, so that a small group of redundant nodes will be responsible for each one. In this way, any ticket which references that sending account can be routed directly to the group for processing, and the results can be forwarded to another group responsible for the receiving account. Redemptions are tracked as idempotent operations, so they can be consistently applied across a distributed system even when multiple nodes process the same ticket in parallel.

The layer-2 servers must also interact with the layer-1 blockchain, to support account funding, public key management, and balance transfers back out. However, they don't require any special blockchain privileges or features, as all of those operations can be handled with normal tokens and standard smart contracts. This means that Atto¢ash is compatible with a wide variety of potential blockchains, or possibly multiple chains simultaneously to make transfers in and out of the system as flexible as possible.

## Use cases

Although we have mentioned the canonical use case of a website charging users a penny per page, there are a number of variations worth mentioning:

- **Replacing ads**: Many sites and creators are currently forced into running ads on their platforms because there is no feasible way to directly charge users for the content without driving them away. As Atto¢ash becomes ubiquitous and supported, this is something that could change the character of the internet.

- **Metaverse**: Similar issues will certainly apply in Web3 experiences, where there are open issues for how to fairly compensate content creators and service providers, especially when the experiences need to be much more immersive and high-quality, where ads may easily become too disruptive.

- **Subscriptions**: Currently many subscription services tend to be in the $5-20/month range, and users must carefully choose which ones they want to commit to. It is a classic economic principle that lowering prices increases demand, so if it were practical for a site to offer a $0.25/month subscription, it is likely that many new users would sign up, and many services might start offering such terms who currently cannot.

- **Tip jars**: Voluntary contributions by fans of content creators are already a phenomenon, and just like subscriptions, offering a practical option for small, inconsequential payments might exponentially increase participation in such activities, and even a penny per “like” could add up quickly. This would also apply to value-added contributions where the user gets some recognition in return.

- **User earning**: We have mostly been talking about a flow of funds from users to websites, but there is no reason why this could not be reversed, as every Atto¢ash account can also receive funds. Micropayments would be great for incentivizing users to take small actions, like filling out a survey or resharing a social-media post.

- **Peer-to-peer**: Payment systems like Venmo have opened up all sorts of possibilities for transactions between friends and between strangers on a 1:1 basis, and Atto¢ash can support the same sort of cash-free niche. There’s nothing restricting transactions to penny sizes, so you could easily pay your share of dinner with a phone-generated ticket.

- **Internet of Things**: Similar to the Helium network, remote devices could pay for their data usage by attaching locally-generated payment tickets when connecting to a wireless gateway. On the flip side, remote sensors could be incentivized to provide environmental data by paying them for each reading.

- **Spam prevention**: Email inboxes and social-media forums might be improved if senders could attach a small payment to messages in order to bypass spam filters or boost their post’s visibility. Genuine spammers wouldn’t be able to afford even fractions of a cent for their massive campaigns, but legitimate users and businesses might like the tradeoff.

There could even be a custom for the recipient to refund the money if they agree it's a message they actually wanted, or the sender can simply self-redeem the ticket some time later if it doesn't get processed.

## Business

There are several possibilities for how Atto¢ash's business model can work, since there are multiple revenue sources available:

- **Processing fees**: The most obvious approach is to charge a small percentage of the funds passing through the system, similar to how credit cards impose about a 3% overhead plus a flat fee on every transaction. Above we assumed a 1% percentage with no flat fee, since the system is optimized for small transactions. It would likely be a psychological advantage if such fees were invisible to the user, with their wallet showing 100% of the supplied funds, so the fee can be applied at redemption instead.

- **Deposit interest**: Users supply funds to the system as stablecoins or traditional currency, and the system ends up holding these until recipients finally cash out their balances. Possession of these deposits can be a revenue source through the simple mechanism of earning interest on zero-risk investments like money-market funds.

- **Exchange fees**: Users and websites, particularly in different countries, may wish to use different currencies to fund accounts and withdraw from them, including potentially volatile cryptocoins. The system could provide this sort of currency-exchange service for an extra fee.

- **Bridging fees**: Similarly, it will be an advantage for the system to have on- and off-ramps connected to as many blockchains and financial outlets as possible, so some extras fees could come from its use as a cross-chain bridging service.

- **Credit advances**: In the initial deployment, we want to operate as a strictly pay-first system, but as it becomes more mature, it would be a natural move to transition into offering credit to users, charging them interest on the outstanding balance.

- **Penalty fees**: This is likely a small revenue source, but any detected instances of fraud will result in a penalty being assessed against the user's account. If nothing else, this can be used to self-support the extra fraud-detecting infrastructure in the system.

As mentioned above, Atto¢ash is initially being conceived as a commercial service, in order to provide low-latency and high-reliability performance across the whole internet. However, if this is not feasible, it can also be deployed as an open-architecture distributed network, with nodes run by individual operators who share in the revenue generated by the system, in proportion to the traffic they are handling.

## FAQ

**Does this exist? Where can I download it?**
At present this is mostly a proposal, intended to gather feedback and attract interested collaborators. Since the initial publication of the whitepaper, we have implemented a proof-of-concept web service on the Lamina1 testnet, so if you're interested in trying that out, you can find it via their Discord server.

**Why is it called Atto¢ash?**
The standard metric unit prefixes go: milli, micro, nano, pico, femto, **atto**. So the name is appropriate for a digital-cash platform optimized for extremely small transactions. It's not just micropayments or nanopayments, but attopayments! And of course the cents mark is just a cute bonus instead of using a boring mid-caps AttoCash.

**Can I use Dogecoin? How about triple-wrapped Bithereum?**
Technically, yes, the protocol could work with any currency. However in the interest of reducing complexity for users, the primary cases will likely be with "real" money and an emphasis on efficient commerce, like current ad and subscription systems use.

**Have you heard about the similar project / new blockchain X?**
Probably not! The DeFi world is vast, so if there's something out there we should know about, please point us to it.

**What's the connection to Lamina1?**
Lamina1 is an Open Metaverse project based on Avalanche, who have talked about the need for large-scale payment infrastructure, so there's some potential synergy. But the only actual connection is that Ed sometimes hangs out on their Discord server.

**Who is this Ed guy anyway?**
The whitepaper author is an ex-Google software engineer who believes that money gains value from its velocity, not its scarcity. There's no such thing as a free lunch, but I'll trade you my potato chips for your apple, and we both win.

**How can I help?**
That depends, are you a VC? Right now, we're mainly looking for feedback… Is this a good idea? Is this feasible, both technically and economically? Are there vulnerabilities or use cases that we have missed?

## Contact

Please reach out to us at: *feedback@attocash.com*