BUIP156: Andrew Stone For Developer

Raised by Andrew Stone
Submitted: 23 November 2020

I would like to continue my efforts on your behalf to push the state-of-the-art in cryptocurrency technologies, to grow the Bitcoin Cash cryptocurrency, and to build awesome software under the Bitcoin Unlimited banner.

My Philosophy:

In the last few years, much progress has been made, although much still needs to be done. As BU developer, its my job to both steer the direction of Bitcoin Unlimited's technology, and to work on my own development work. In regards to steering, I've been encouraging BU to excel on multiple fronts. I believe that BU needed and still needs to broaden our focus beyond the full node, since the most valuable thing anyone could be doing right now is working towards end-user use and adoption. But gaining adoption is hard acting solely as software buried deep in the bitcoin software "stack" -- we need to provide tools and services closer to the end user.

My Leadership:

The BU development team has continued to deliver a leading Bitcoin Cash full node, with an admirable defect count (in comparison with ABC's multiple hard-fork vulnerabilities and block production problem).

We lead the space in scalability and have delivered innovative features such as Graphene block transfer and deep unconfirmed transaction chains. We continue to explore and develop new technologies, such as miner validated tokens, new opcodes, blockchain voting technology, Bobtail block confirmation, and Storm probabilistic confirmations.

We have supported wider efforts that are aligned with our goals, such as Tom Zander's doublespend proof development and our own Bitcoin Cash Specfication effort, in conjunction with Bitcoin Verde as our content partner.

At the same time, we have dramatically broadened the scope of Bitcoin Unlimited's software offerings. Most importantly, we support and have integrated the Electrscash server into our full node. This allows full nodes to present blockchain information in a manner easily consumable by SPV (phone and light) wallets, which is arguably an important service of non-economically significant (home user) nodes. And we support and develop a blockchain explorer bch-rpc-explorer.

I feel that I have been responsive to BUIP requests coming from members. You can see the status here: http://www.bitcoinunlimited.net/buip/summary. But remember that the BUIP system is made of checks-and-balances -- a passed BUIP is not a decree that BU engineers must drop everything and implement the feature. Rather, its a pre-determination that *if such a feature gets developed, it will be merged*. A lot of people who are critical of the BU process as a democracy ("tyranny of the masses", "benevolent dictators work best for software projects" type arguments) do not understand this subtlety.

My Development Work:

In my own development work, I've been doing a variety of projects:

First, I extracted important BCH functionality from bitcoind into an application library called libbitcoincash. We have bindings for this library in C++, Python and Kotlin, and use it in my

"txunami" transaction generator (scalability testing) and in our Python QA tests.

The Kotlin library has been expanded into a fully-featured library "libbitcoincashkotlin" that also included a Electrum and P2P client layer. While it is true that there exists other client libraries, this one has an advantage that it directly uses much of the trickiest functionality in the full node. This reduces the likelihood of forking bugs and decreases maintenance cost (since its the same code).

I've also just announced an Android wallet named WallyWallet built on top of libbitcoincashkotlin. Although its still in "beta", I've been using it personally for about a year now. It supports all the basic P2P functionality such as bitpay payment protocols, etc. Its not yet released because I would like to build some interesting new ideas on top of it first. You can read one of them here: http://www.bitcoinunlimited.net/delegated_payment_protocol.

I am also experimenting with a variety of new opcodes and advanced technologies over at www.nextchain.cash. These include large number integer math, miner validated tokens, transaction introspection, and script subroutines. This website itself is generated via a wiki engine called "cwik" (pronounced quick) which I wrote. Its a wiki that uses the BCHidentity (http://www.bitcoinunlimited.net/identity__protocol) protocol for authorization, and git and .md files as source material. The purpose is to allow easy access and redisplay of the information created as part of the Bitcoin Cash Specification effort. But I also intend to use it as a venue to show how cryptographic identity can help solve the huge internet misinformation problem that is confronting the entire world. In essence, passages will be able to be +1 or -1 via your persistent cryptographic identity (which may or may not be connected to a real identity) and these identities can be affirmed by other sources, some of which may be considered an authority. For example, colleges might affirm the identity of graduates as having a degree in certain fields. Users may filter comments and other activity based on these affirmations, and correlate activity across the web or different topics. By looking at +1 or -1 votes across multiple topics, correlated with identity, analysis tools would be able to determine significant bifurcated opinions and correlate opinions across topics. Ultimately such a system could be used to resolve the "wiki edit wars" found in the BCH page for example, by noting a major difference of opinion and bifurcating the page. This system of credibility is likely to work much better than today's "proof-of-no-other-value" (i.e. the person who has the most time to devote to edits, re-edits, and arguments wins) that dominates wikipedia, and "the-most-sensational-wins" system that dominates social media.

Thanks for reading, and I hope that you will vote for me for another term as your Developer!