

joel.mn |

The Blockchain Application Stack

Nov. 25, 2014

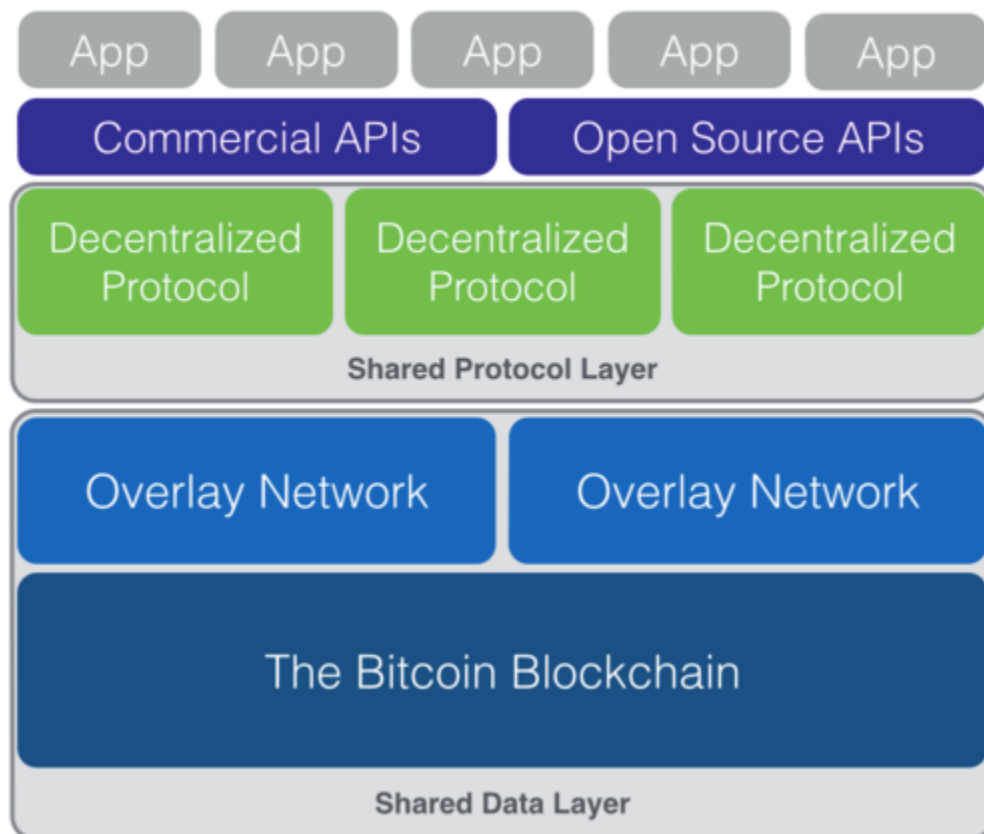
Last week I led a workshop at NYU's Bitcoin Hackathon, HackBit, where I talked about Bitcoin as a Protocol, alternative uses for the Blockchain, and a little bit about the challenges and opportunities that lie therein. I tried to share everything we've been learning about this space, but unfortunately time constraints prevented us from doing a deep dive. To compensate, I'll be writing a series of short blog posts explaining some of the ideas in that workshop, starting with a look at what I'm calling The Blockchain Application Stack.

The first thing Brad taught me when I joined Union Square Ventures is that one of the greatest things about working in the Venture Capital business is that you get to look at markets from a very different vantage point. Every day, we have the privilege to learn what the future is going to look like from the companies and entrepreneurs who are building it. It's thrilling, especially if you're a technology geek like we are.

We spend a lot of time looking at everything that's going on with Bitcoin and the Blockchain. This involves e-mailing, tweeting, texting, calling, skyping and meeting with teams all over the world who are building next-generation technologies and applications that leverage the blockchain to undo many of the

paradigms that dominate the software business today. Some are just ideas, some are products already in the market. Some have flopped, and some have gone on to raise millions of dollars in funding. Being exposed to all these has allowed us to identify certain patterns and trends that are helping us build an image of what's coming. I'd like to begin sharing with everyone, as well as invite you to help us think through this.

The Blockchain Application Stack



This is what I think the architecture of internet applications is going to look like in 10 years. This is just a simple illustration and it leaves a lot of important insights and issues out. I'll try my best to explain the thinking behind it below. To keep things short, we'll run through every part of the stack from the bottom

up, and do a deep dive on each in the posts that will follow.

The basic idea is that everything inside the gray rectangles is decentralized and open source. For now I'm calling these the Shared Data and Protocol Layers. Nobody controls these parts of the system, and they're accessible by any person or company. If we use Bitcoin as an example, the Blockchain is the shared data layer and the Bitcoin protocol is a Decentralized Protocol that's part of the Shared Protocol Layer.

You'll notice that each layer gets thinner the higher up you go. You'll also notice that the Shared Data and Protocol Layers cover about 80% of the entire stack. Internet applications today are built on top of open, decentralized technologies like TCP/IP and HTTP, but if you were to graph the current internet application stack like above, those open, decentralized protocols would probably only make up about 15% with everything on top being private and centralized.

1. Miners and the Blockchain.

If you know a little about how Bitcoin works, you know what miners are. In a nutshell, miners are the nodes in a network of computers who, together, verify all Bitcoin transactions. In exchange, the algorithm rewards them with Bitcoin. Because Bitcoin has real-world value, the operators of these machines are incentivized to keep them running. If you'd like to learn more about mining, [this is a great explanation of how they work.](#)

The Blockchain is the public ledger that holds a permanent record of all Bitcoin transactions, and is maintained by the miners. It's not controlled by a single

entity and it's accessible by everyone. You can read more about the Blockchain here.

2. Overlay Networks

This is where things start to get interesting. Developers are starting to build networks that work in parallel to the Bitcoin blockchain to perform tasks that the Bitcoin network can't, but that make use of the Bitcoin blockchain to, for instance, timestamp or validate their work.

One example is Counterparty. Another might be Side Chains. Whatever form these overlay networks take, the one thing they have in common is their connection to the Bitcoin blockchain, and how they benefit from its network effects to achieve liquidity without having to bootstrap their own alternative cryptocurrency and/or blockchain like alternative solutions such as Ethereum require.

3. Decentralized Protocols

Thanks to the Blockchain, for the first time we can develop open source, decentralized protocols with built-in data (thanks to Overlay Networks and The Blockchain), validation, and transactions that are not controlled by a single entity. This is where the traditional architecture of software businesses begins to break down. The best example of a decentralized protocol on top of a Shared Data Layer is Bitcoin, and we're already well aware of how it's affecting money and finance.

Companies like Ebay, Facebook and Uber are very valuable because they benefit tremendously from the network effects that come from keeping all user information in centralized in private silos and taking a cut of all the transactions. Decentralized protocols on top of the blockchain have the potential to undo every single part of the stacks that make these services valuable to consumers and investors. They can do this by, for example, creating common, decentralized data sets to which any one can plug into, and enabling peer to peer transactions powered by Bitcoin.

In fact, a number of promising teams have already begun working on the protocols that will disrupt the business models of the companies above. One example is Lazooz, a protocol for real-time ride sharing and another is OpenBazaar, a protocol for free, decentralized peer to peer marketplaces.

4. Open Source and Commercial APIs

Protocols are hard for the average developer to build on top of, so there's an opportunity in making it easy to connect to them. Whether it's a good business in the long term is up for debate, but I think it's a very important part of the stack. Making it quick and easy for developers of any skill set to quickly build an application and experiment on top of these decentralized protocols is paramount to their success.

These will be either commercial services or open source projects. Good examples of this trend are Chain.com's APIs and Coinbase's Toshi for Bitcoin. They both serve the same purpose, but Chain is a hosted, commercial service, and Toshi is Open Source.

5. Applications

This is the consumer-facing part of the stack. Applications built atop this architecture will, in most cases, work very similarly to the ones we have today – just like Coinbase works similarly to PayPal. The big difference to consumers, however, is that because they are built on decentralized protocols, they will be able to talk to each other, just like different e-mail applications and bitcoin wallets can interoperate.

One thing I like about this stack is that it's growing from the bottom up. First we had miners, the blockchain, and Bitcoin, and now we're building everything else on top. As far as I know, the most significant revolutions in technology have been built this way.

This imposes a very interesting set of challenges for developers, entrepreneurs, and investors as so much of the value in the current Internet stack will be commoditized by this architecture. But the best thing about this stack is that the user is better off thanks to lower or nonexistent take rates, switching costs, individual ownership of data, and consumer market power. This is the part I'd like to focus more on in the posts that follow.

If you liked this post, [you should follow me on twitter here](#) to be notified when the next one in the series is up.

(Thanks to [Fred Wilson](#), [Albert Wenger](#), and [Muneeb Ali](#) for helping me polish this post)

