We’ll be talking tonight about a blockchain system called EOS.

But first the disclaimer. We are NOT part of block.one, the company developing EOS code, and we are NOT in any way representing them. We’re a separate entity.

Before we jump into EOS, I’m going to give you a little bit of context, to help you realize how EOS fits into a bigger picture.

Back in 2008 we started with a simple payment system. Which then inspired the creation of Ethereum, an amazing smart-contract system. Which now is about to be challenged by a full-blown self-governed ecosystem, not just for smart-contracts, but for entire blockchain-based applications.

OK, let’s start with the concept of a decentralized application. So what is a decentralized app?

To make it really simple, and to give you a concrete example, we might say it’s something like an Uber, but without the Uber company running it.

Building decentralized apps is not an easy task. A lot of things we take for granted when creating conventional web apps, are not easily available in a decentralized environment.

Let’s take a moment to consider what a decentralized app needs in order to be a successful business enterprise.

* Firstly, your app needs a platform that’s cheap to run. This basically means transaction fees being low & predictable.
* Your users need to be able to interact with your app without paying anything.
* Your app needs to be available via mobile phones & web interfaces.
* Your users need account names, no public keys, no fancy cryptographic stuff.
* Your users need a procedure for account recovery, in case they are hacked, or just lose their password.
* If you build a financial app, your users will expect privacy & confidentiality of transactions.
* Your app needs to be able to talk to other apps, including those on other blockchains.
* Your app needs to be upgradeable, so that you can constantly improve it.
* And finally, your app needs bug recovery. Software development is an imperfect art, so no matter how much effort you put into checking the code, there will always be bugs happening.

Is this list complete? I don’t think so. We didn’t mention the single most important thing. High performance.

If we want to host real-life businesses on the blockchain, we need high performance.

And what is high performance?

* You need about 20k transactions per second, just to do Visa & MasterCard, not to mention all the other payment systems out there.
* Facebook does 50k likes per second, not including all the posts, and voting, and other actions.
* And for currency trading, it’s sometimes a hundred thousand transactions per second, and this is just for one currency pair.

So, imagine putting all of these on a single platform, so they can inter-operate. Well, that would require millions of transactions per second. This is a massive scale.

-- And what can the blockchain currently offer? Bitcoin does less than 4 transactions per second. Ethereum, something between 15 and 30.

-- And those two systems are extremely expensive to run. By means of inflation we are currently spending something like 6 billion USD a year to operate those two systems.

-- You might say it’s just a matter of time before scaling solutions are introduced. But actually, there are only two ways to go.

* We can go off-chain, which is basically the idea behind state channels. If it works, it will only solve the problem for basic payments, not smart-contracts. So we need more than that.
* The other option is creating several sub-domains within one blockchain, or a hierarchy of multiple blockchains. This helps a bit, if you want to run a lot of small apps within the same ecosystem. However, if you want to compete with big businesses like PayPal, Uber or eBay, or if you’re into currency trading, this solves nothing. For those systems you need enormous processing power on a single blockchain, and oftentimes on a single CPU thread.

OK, it’s time to introduce EOS, and find out what it can do about all those issues.

What is EOS? The simplest way to put it would be this. EOS is a general-purpose smart-contract platform, similar to Ethereum.

That’s a fair definition.

-- But I prefer this one, as it perfectly captures the whole purpose behind EOS.

What it says is that EOS makes decentralized apps look and behave like conventional apps.

How does EOS work?

Let’s start with Ethereum. This is what Ethereum actually looks like. We have a bare-bone computer (in this case the EVM) and we have some apps trying to run on top of it.

They have a tough job to do, because apart from some basic stuff, like the ability to send funds & maybe store some data, they need to take care of pretty much everything else.

And because everything is being implemented in the app layer, it’s being done within relatively inefficient scripting environment. Which makes it really hard.

This might explain why most of what we have right now in the crypto-space are just prototypes with quite horrible UX. Developers don’t get to spend their time on building business logic and user interfaces. Instead, they are stuck on figuring out the low-level stuff, things like data storage, account permissions or inter-app communication.

Furthermore, even if some generic solutions eventually emerge as a result of this effort, they will be expensive to use, as every single line of their code will consume gas.

Actually, the whole thing reminds me of trying to build a mobile app on a smart-phone without having Android or iOS at your disposal.

So what are we missing here?

-- Right, we need a blockchain operating system.

And this is what EOS brings to the table. EOS is an operating system for running decentralized applications. Actually, it’s both. A computer and an operating system on top of it.

OK, so now let’s go through the main features of EOS.

There are lots of them, so let’s divide them into 6 categories.

**#1. Processing power**

We will start with a very bold claim. On day one EOS will be capable of at least 50 thousand transactions per second. Maybe even more.

-- Let’s take a look how this fits into our landscape. As you can see, it’s already on par with what the biggest businesses need.

-- Now, how is EOS going to achieve that?

It’s gonna do it in two dimensions.

* Firstly, via enormous speed of sequential processing. EOS uses a solution borrowed from a currency exchange engine called LMAX, which is capable of millions of transactions per second on a single thread.
* On top of that, we’ll have parallel processing, made possible thanks to clear separation between the things which can be done in parallel, and things which cannot. This is an important distinction, because not everything can be subjected to horizontal scaling, e.g. managing an order book of a decentralized exchange always needs to be run on a single thread. Whereas things like validation or account authentication can be done in parallel.

To give you a better picture of what it means, imagine this. Not so long time ago all our computers had single-core processors, which can only do one thing at a time. This is sequential processing. And this is how all blockchains operate right now.

Nowadays all our computers, and even smartphones, have multi-core processors, which can do several things at the same time. This is parallel processing. And this is what EOS will be capable of. Right now, no other blockchain can do this, as it requires a very different architecture.

**#2. Built-in governance**

* In the heart of EOS there is a consensus mechanism called DPOS, which stands for Delegated Proof of Stake. It’s a sophisticated system, but here is a quick overview.

In DPOS block producers are elected by the token holders, each voting according to their stake. So it’s not the hash power that determines who has the right to produce a block. Instead, token holders take a vote and elect 20 unique block producers.

What’s important, they can switch their vote any time they want. So the whole thing is powered by reputation. It’s hard to get elected, but very easy to lose the job.

* Back to EOS again. EOS implements DPOS, but extends it even further. In EOS block producers do much more than just produce blocks.

For example, block producers have the ability to freeze & fix broken apps.

Had the DAO incident happened on EOS, it would have been easily contained within a few hours.

* What else have we got? There is a complete governance structure, including a legally binding constitution, an arbitration system for resolving disputes, and an entire system of shareholders voting, to decide on things like the level of inflation, and the way the income from inflation is to be distributed.

Generally, EOS itself acts like a giant, fully autonomous, self-governed decentralized app, whose business is hosting & running other decentralized apps.

**#3. Infrastructure for apps**

* In EOS developers only need to write code for what’s unique for their application. All common features, including low-level stuff, are provided by the blockchain.
* Those features are listed here. Account permissions, account recovery, scheduling, authentication, inter-app communication, biometric 2nd factor validation.

Account recovery might be especially interesting here. It works like this. If you get hacked, and as a result lose control of your account, you can get it back, provided you know the previous password, before it was changed by the hacker. And it gets even better. If no hacker was involved but you just lost your password, you can still regain access to your account using EOS social network feature.

* On top of that, the system has a built-in data storage solution based on IPFS and maintained by the block producers, as part of their job.

**#4. No transaction fees**

* This is my favorite. Your EOS tokens are never consumed, because in EOS there is no such thing as a transaction fee or gas. If you’re an EOS token holder, you can use the system for free.
* The deal is very simple. If you own 1% of the tokens, you own 1% of the network, including all its bandwidth and all its resources.

The absence of transaction fees means one thing. You can monetize your app any way you want. By default your users can interact with your app without paying anything.

**#5. Upgradeable apps**

* In EOS you publish on the blockchain the source code of your smart-contract, not the compiled machine code, as is the case in Ethereum. Does it make any difference? Actually, it does.
* Firstly, it means that your intentions conveyed in the code become part of the blockchain consensus. This might matter if there are any disputes around your smart-contracts.
* Secondly, it means is that the source code of your app can be recompiled in the future. This way, you can make your smart-contracts fully upgradable, if you opt to retain permission to do so.

So it’s up to you to decide. If you opt in, you’ll be able to improve your app, and also fix bugs in it. If you opt out, your app becomes fully autonomous.

**#6. Asynchronous communication**

* EOS works like a giant email server, as everything is based on sending and receiving messages. Which means that the entire communication in EOS is asynchronous, which, by the way, is the exact opposite of Ethereum’s architecture.
* And it just happens that asynchronous communication is the only way to arrange communication between blockchains. As a result, EOS apps can talk to each other NOT only within the same blockchain, but also across blockchains.

So, if we had multiple EOS blockchains, both public and private, everything could be fully interconnected.

Here are EOS main features again, listed together. As you can see, everything revolves around one purpose. To help you build & run a decentralized business.

* EOS processing power makes your app fast & scalable.
* Built-in governance gives you a safety net in case things go seriously wrong, or if there are any disputes around your smart-contracts.
* Built-in infrastructure saves you a lot of development work. It also saves you a lot of operating costs, as you can rely on features which are free to use.
* The absence of transaction fees gives you full flexibility about the way you can monetize your app.
* Upgradeability lets you constantly improve your app & fix bugs.
* And finally, asynchronous communication opens you up to interactions with other blockchains.

And now I’d like to tell you a little bit about EOS origin.

Generally, there are two ways of building a smart-contract platform.

* You can start by creating the whole thing, and then attempt to build some concrete apps on top of it. Most probably you’ll be confronted with tough reality when you try to make those apps actually useful. However, this is the prevailing approach in the crypto-space.
* Or you can start with building a couple of concrete, non-trivial apps, make some mistakes while doing this, learn from those mistakes, and as a next step figure out what’s common between your apps, and only then start building the abstraction layer. This is how EOS was born. The team of developers behind EOS have already built some of the most successful blockchain apps in the crypto-space, and now they have set out to leverage their experience by building a general-purpose platform.

And when I say they’ve built successful apps, I don’t necessarily mean market capitalization. What I mean is, how useful those apps are in the eyes of users. Not investors. Actual users.

This is what it looks like. Those two apps, BitShares & Steem, were created by the same developers who are now behind EOS.

And those two apps are very different. BitShares is a decentralized exchange, whereas Steem is a social media platform.

As you can see, Steem is the most used blockchain app in the entire space. Its users perform approximately 8 transactions per second, which is close to a million transactions per day, and which by the way is less than 1% of its total capacity.

And this is what their UIs look like. As you can see, those are quite complex apps. They still have their deficiencies, but they work smoothly enough to trick you into thinking that they are just normal web apps.

They have a couple of quite useful features, e.g. account recovery was first introduced in Steem. Generally, a lot of EOS features originate from those systems.

And one more interesting thing. While it’s evident that BitShares & Steem have valid business cases, in reality none of them could be hosted on any existing smart-contract platform, including Ethereum.

Why is that?

Firstly, because those two little apps alone would eat up all of Ethereum processing power.

The second reason is even more significant. Steem business model requires that all user actions are free of charge, as otherwise nobody would post or upvote anything. And guess what, Ethereum with its concept of gas, is unable to offer that, and most probably never will.

There is an ongoing controversy regarding DPOS, the consensus mechanism, which is going to be applied in EOS. Some people dismiss EOS, claiming that its reliance on DPOS makes it not decentralized enough.

I’m going to present you some data that sheds some light onto this issue. This chart shows how diversified the block production is in various systems, including POW and DPOS.

As you can see, DPOS is doing pretty well. Actually, it looks like the most decentralized system out there, if we take into account the top 80% of the blocks. Why top 80%? Because 80% is what actually matters, if you're a tiny player in this game your impact is zero.

Of course, you might say it’s not fair, as those are just mining pools, and there are many miners behind each of them. Correct, but in the same way there are many people voting for each block producer in DPOS. The difference is that in POW you vote with your hash power, while in DPOS with your stake. However, in both cases we have the same underlying principle. You delegate your power to some bigger third party who produces blocks on your behalf.

Nevertheless, we need to admit that DPOS has its own deficiencies. Surely, it’s not perfect. But it’s arguably the best trade-off we can make.

EOS strong points.

* EOS has a very efficient team with a proven track-record. The best example is Steem. It was conceived, created and launched in just four months. And it's a complex app.
* EOS is using Web Assembly for running smart-contracts. Web Assembly is a big deal, it’s an emerging industry standard for web applications, supported by Google, Microsoft, and Apple. Web Assembly is a universal compile target for several programming languages, including C++, Java and JavaScript.
* Regarding EOS financial supporters, we have some big names here. Interestingly, both of those guys were also early investors of Ethereum, and they still are.
* And last but not least, we have a declaration from EOS founders to spend about 1 billion USD on supporting the emerging ecosystem around EOS.

-- Apparently, this is where most funds from the currently running ICO will go. Not on EOS code development (as this is already fully funded), but on those of you who will try to do something useful with EOS.

Unsurprisingly, EOS has also got its weak points.

* The ecosystem around EOS has not really emerged yet. It might be hard for EOS to catch up with the high number of developers currently working on Ethereum-based projects.
* Using C++ is right now the only way to create smart-contracts on EOS. This will surely change, as Web Assembly gets upgraded, but unfortunately, at this early stage, this is the only way.
* EOS source code might be a bit of a challenge, as it’s quite hard to work with. As one of our friends put it: it’s an act of constant balancing between *cutting edge* and *bleeding edge*. Unfortunately, oftentimes it’s just bleeding too much.

But this doesn’t affect those of you who just want to use EOS for writing smart-contracts. It affects entities like us who try to integrate with EOS on a deeper level.

* Track-record of poor documentation. Unfortunately, this is true. Hopefully this will change this time around.
* And the system is not live yet, but it will be in June next year.

Which brings us to the roadmap. Currently we are at the MVP stage.

In a couple of weeks there’ll be a subsequent release, called Dawn 3.0, with all major functionalities deployed, including multithreading.

And now let’s revisit our list of demands required by decentralized apps. As you can see, EOS is doing a pretty good job here, as it covers pretty much every aspect we can think of.

The only exception might be privacy protection. At this stage, I’m not aware of any special features in EOS regarding zero-knowledge cryptography or untraceable transactions. This might change in the future.

To wrap it up. I think that what sets EOS aside, is not just the technology, it’s actually the unique way it’s going to operate.

* When you consider EOS features, all this might look like some sort of revolution, but actually it’s an evolution. EOS builds on solutions which have already been battle-tested in real-life blockchain apps.
* We must admit that Ethereum is an excellent playing field for smart-contract prototypes. Whereas EOS takes the notion of smart-contracts to the next level, and makes the whole thing extremely business oriented.
* Can other systems copy EOS solutions? Not really. EOS is quite different from anything else on a very deep level. To follow EOS path, other platforms would have to backtrack a lot regarding some of the fundamental elements of their technology. So it’s quite unlikely.
* Will EOS replace Ethereum? I don’t think so, at least in the predictable future. But what will probably happen is EOS playing a similar role for Ethereum, as Ethereum is playing for Bitcoin.

And now very briefly about our company.

* Our focus is on blockchain-based fundraising and digital asset management solutions.
* We are in the process of setting up a software house for building dApps, both on Ethereum and EOS.
* And we are aiming to be elected as EOS witness, which is just another name for block producer in EOS. We have some experience with that, as one of our advisors is already a successful witness for Steem.

And what we need.

* Not looking for funding, we are already fully funded.
* Always looking for good ideas which could be turned into EOS apps.
* And we’re hiring. If you have solid experience with C++ or Ethereum smart-contracts, we definitely look forward to talking to you.

Thank you. If you have any questions we’ll be happy to answer them, and this is how you can reach out to us.