Hello

~Overview~
I was asked to come here, and speak about these topics: soft forks, governance, and culture.

I was assigned this topic. Believe it or not.

~agenda~

The agenda is :

some basics about soft forks, four comments I have.

Second, a definition of Bitcoin governance, that's actually useful. Three strategies for governance, and their trade-offs. In a big table.

Finally, Bitcoin culture over time (with respect to soft forks).
A simple explanation for why that may be, that ties the whole presentation together.

~Review~

First, let us review the basics. Hopefully all of you already know this.

Now, actually, what's up here on this slide is not perfectly accurate. As we will see. But it is the conventional wisdom. So it is a good starting point.

The conventional wisdom --- is that soft forks "tighten" the rules, and hard forks "loosen" the rules.

A second, competing piece of information, is that hard fork requires the user to update their software, whereas the soft fork does not. That's a lot closer to the truth.

Anyway, here are some soft forks you should be familiar with:

August 2010, Satoshi took a bunch of messages that were previously broadcast-able in Bitcoin's blockchain, and he banned them. This tightened the rules ... and also regular users did not need to update immediately. Right? Satoshi banned a bunch of stuff, but no one was using it. It is like if you ban skateboarding, before anyone in the neighborhood buys their skateboard. Very soft!

Sept 2010 -- Satoshi added the blocksize limit, of 1 Megabyte. Shrinking the blocksize, tightens the rules. Soft fork. --- Expanding it, is loosening the rules.

Thats the bottom.

Then we started adding new features. P2SH enabled multisig (among other things). Check Lock Time Verify allows you to post-date a check basically. SegWit helped lay the foundation for the Lightning Network.

~History~

Ok that was the basics.
Now we will discuss 4 things are NOT widely known:

Gavin called them "changes"
Why it is the case that Fork is kind of a strange term.
The origin of this soft fork term.
-- And why it actually makes sense after all.
Why it was used for upgrades

~Gavin~
Ok, Gavin Andresen, who of course was the "leader" of the Bitcoin project after Satoshi left in 2010. He wrote this here.

This is very important: Note -- there are soft rule changes and there are hard rule changes. ....

Now what I want to point out here is.

~Forbidden~

...when Gavin says "at this point", he is referring to June 29, 2012! Quite a long time ago. This will be important when we get to the end of the talk.

Now "impossible to roll out 'hard changes'". Ironically, Gavin would later attempt the hard fork blocksize increase change, and fail. Exactly as he predicted he would, here. But I want to convey this idea that something is *forbidden on grounds of impracticality*. It is as if a law of physics prevents it. Money has network effects, the participants don't want a split, and so hardfork changes just can't be done.

~Fork~

Now a tiny detour. Why is it called "fork"?

I wrote a blog post long ago, with this image. You can see the culinary fork, the tuning fork, and the fork-in-the-road. They all have a split. Yet a soft fork has no split. And a network that upgrades via hard fork, doesn't have a split either!

So why call it a fork, if there's no fork.

~Defintions~

Believe it or not, there's a very satisfying answer to this question. We go back to the origin of the term itself. This post, from Nov 2012. Five months after Gavin's remark (that I showed you on Github moments ago).

~The 4 Definitions~

Now, let me read these *at* you.

~slide~
First though I want to point out that even Adam Back and Luke Dashjr disagree on what a hard vs soft fork is, which is why I'm boring you with this! I hope it helps you.

~slide~

OK.

When there are a sufficient number of bitcoin clients on the network that disagree on the rules about how blocks are created and recorded in the blockchain. It leads to a split in the chain, one set of bitoin lients follow one branch and another set follows the other. To fix hard forks some action must be taken by us.

Orphaned Blocks -- whenever a soft fork or hard fork occurs, the blockchain is split into two paths. One of these chains will eventually be considered the valid one, and the other will be the invalid chain. Blocks that are in an invalid chain are called orphan blocks.

Soft Fork -- at the end, it says: *these kinds of forks will solve themselves without any intervention from us*.

So you see? Splitting is a problem --- --- it means that some blocks that are valid today, will later become invalidated, which is BAD! We want synchronization. Tight synchronization always.

Forking is a *problem.* The hard fork and the soft fork, are *problems to be solved.* The hard fork is hard problem, because we have to do something. -- --- The soft fork is not as big of a problem, since it will eventually work itself out.

That's why """fork""" is a good word, after all.

~slide~

Ok finally the logic.

A soft fork will resolve itself, as we just said.

Governance

~Definition~

Ok, I typed "governance" into Merriam Webster. It gives this garbage, which is not going to help us in any way whatsoever.

But I'll read it: "... overseeing the control and direction of ... something". That's what it says.

Now obviously this has many problems. Who is controlling who? After all, Bitcoin is supposed to be decentralized. Or "peer to peer", as *more often* Satoshi put it. Everyone an equal peer. So there is no separation of people into "governors" and "the governed".

~slide~

But this definition is bad in another way. We are here at MIT. -- So it would be better to have a very cut-and-dry, black-and-white *technical* definition of governance. -- How do we know that we are doing governance well? (?)When people don't complain? -- (?)When we think that we are doing it right, despite the complaints?

No objective function, in the MW definition.

~slide~

So here it is, I'll give you a better definition.

~Technical Governance~

Governance equals *finding today's node software.*

Governance is an answer to the question: *where* does the node software come from? What process? In that sense, it is more like an industrial process -- like how do we make steel. Or a recipe. How do we build this bridge?

Which code is fullnode-code? And which code is not.
How do we tell Bitcoin Nodes from non-nodes?

If there is a dispute, then who is correct? And who is incorrect?

We can restate this as the problem of meta-consensus: consensus about consensus. Your full node does consensus, yes... but only after you find that software and run it.
So, you can call it "pre-consensus" if you like. How to find the consensus software. If you didn't have a node, how would you get one?

~NodeFinding Strategies~

There are three main philosophies for solving this problem.

Number one -- go to bitcoin.org , and run the latest version.

Luke Dashjr proudly advocates this strategy, as does Mike Hearn believe it or not.

~slide~

Here he says "youre the decider".

There was an old opcode call OP VER, that literally made every upgrade mandatory. If you just published some software with a new version number, it put you on a completely different network basically. Satoshi added this, then removed it, so he seems to have thought it was in the wrong direction. This op ver is the opposite of strategy three. Which we will get to.

--- There's a lot of culture in this part of the talk, as well.

~back~

The second strategy is characterized by extreme anxiety. The adherents really like bitcoin consensus, and so they are very annoyed that meta-consensus spoils the fun. It rains on the parade. So these people say: find an old version, run that software, and then try *not to update*. *Resist* the updates as they are all ZERO benefit + potential problems. If it 'aint broke, don't fix it.

~slide~
Mircea Popescu was a famous adherent of this point of view. These people ran a version of Bitcoin that was ... 0.5.4.

And they have many followers today. Michael Saylor -- absolutely has this view, he believes that if we do nothing, Bitcoin will be worth a lot of money, every change is a risk, so resist the changes. Most of the "toxic" crowd, today, has this view.

~return~

The third strategy was the mainstream, I think until the SegWit War of 2017, and the failed SegWit2x revolt. Those events made soft forks much more ... polarizing. You're either with us or your against us!

But this view originally meant ...

~slide~

...that, we could all peacefully coexist. Since a line of soft forks can coexist, it kind of doesn't matter. Run whatever software you like, as long as it is on the line.

~return~

The other thing that SegWit2x did was. After it failed, Bitcoiners concluded that the reason for BTC's success was that it had successfully resisted change. Thus the second strategy rose in prominence.
...

~Governance Table~

Ok, now I will explain problems with each of those three strategies.

First, what is the problem with the strategy of always going to Bitcoin.org and downloading the software there?

Bitcoin.org might be hacked, or compromised.

It is similar to something called the YUU-thith-throw dilemma.

God tells us that something is good, or moral.

So, the metaphor is -- Bitcoin.org can tell us that some software is "Bitcoin". It is the software that we want, to protect our rights and freedoms (and finances). This whole thing is a metaphor.

So, God tell us something is good. But god-liking and goodness are different things. Either God is an advisor, on the left, who knows about morality. In which case morality *preceded God*. Or on the right, we have a weirder problem. God's opinion is the only opinion, called Divine Command Theory. If God tells you to murder children, you have to do it.

But God could change his mind. On Monday he might say kill all the children, on Tuesday he might say murdering children is bad. So it also seems flaky.

And in both cases, we might later learn that we misunderstood God's instructions.

This is all a metaphor, but it is called "fallibilism" -- the idea that knowing things is hard. We might be mistaken about anything we think we know.

~slide~

So, here's the obvious stuff. Blindly outsourcing your thinking to bitcoin.org, isn't great. Sticking with one piece of software is a little better. Best of all is when you can choose from a bunch viable soft-wares.

~slide~

Ok, the problem of expertise. I was going to get a great twitter screenshot of this but Luke was suspended from Twitter, like yesterday. Bad luck.

But Luke-Jr's position, is that you must run the latest version of the software. And you learn about what that version is, by *participating in the technical community.*

This has obvious problems: learning takes effort --- for some people prohibitive effort. (Not everyone can devote their life to the minutia of Bitcoin technical debate.). So laypeople are kicked out. And there will always be more laypeople than experts, by definition. So that's bad.

Another problem is that there is no accumulation of recognizability. You can't put out one thing, and allow the public to become more familiar with it over time. So like with cars we have turn signals, speed limits. Stop signs. It's a lot at first, but since it doesn't change, it is possible for each person to learn it -- in their own time.

~slide~

So a static protocol is easier for laypeople.
...

Why does soft fork coexistence only do OK?

Well,

~slide~

if you do two different soft forks, at the same time, then this actually equals a hard fork.

So... ~slide~

...the soft forks must be in a line.

~slide~

Furthermore, Bitcoiners disagree about everything, so even getting people to agree on this line is difficult.

Now, the problem of Innovation. Very straighforward

Well, there's no limit to *how good of an idea* people can come up with tomorrow.

So we want those innovations to make it into our project. If we can't, then those innovations will launch as altcoins, and potentially destroy the project.

Culture

What is culture?

Groups, causing their members to behave a certain way.

The thing about culture is, it changes.

~slide~

Ok, what I've done here, is I've taken every fork every done, according to BitMEX. Not me.

And the reason I did that, was so I could make this slide, with no accusations of bias, or whatever.

So here it is...

~slide~

Notes

- Segwit Trauma / PTSD
    - Unsolved mysteries of the Blocksize war
        - Why did people get hashrate support for a hard fork, when hashrate is irrelevant to a hard fork? I don’t know.

        - Miners signed a meaningless piece of paper backing the wrong side, but they never actually did anything. Yet still they feel guilty and unwilling to do further soft forks.

- Also it is harder to be an expert now
    - In 2012, you could “catch up” easily, by studying 2009-2012. Now , you need to catch up by studing the 2009-2023 lore, and new knowledge is growing all the time.
    - Harder to be an expert; and MORE non-experts. Both of these things are growing geometrically. So the experts:laypeople ratio is plummeting.
    - Although, it didn’t help – SegWit2x PTSD. And the increased power of the soft fork. Both bad.

Also psychology of sour grapes – people look at zCash zk-snarks or Eth DeFi , and people say “I never wanted that anyway!”. So there is less of an appetite.

~About me~