CUSF - "Core Untouched Soft Fork"
or: "Soft Forks, without a Soft Fork",
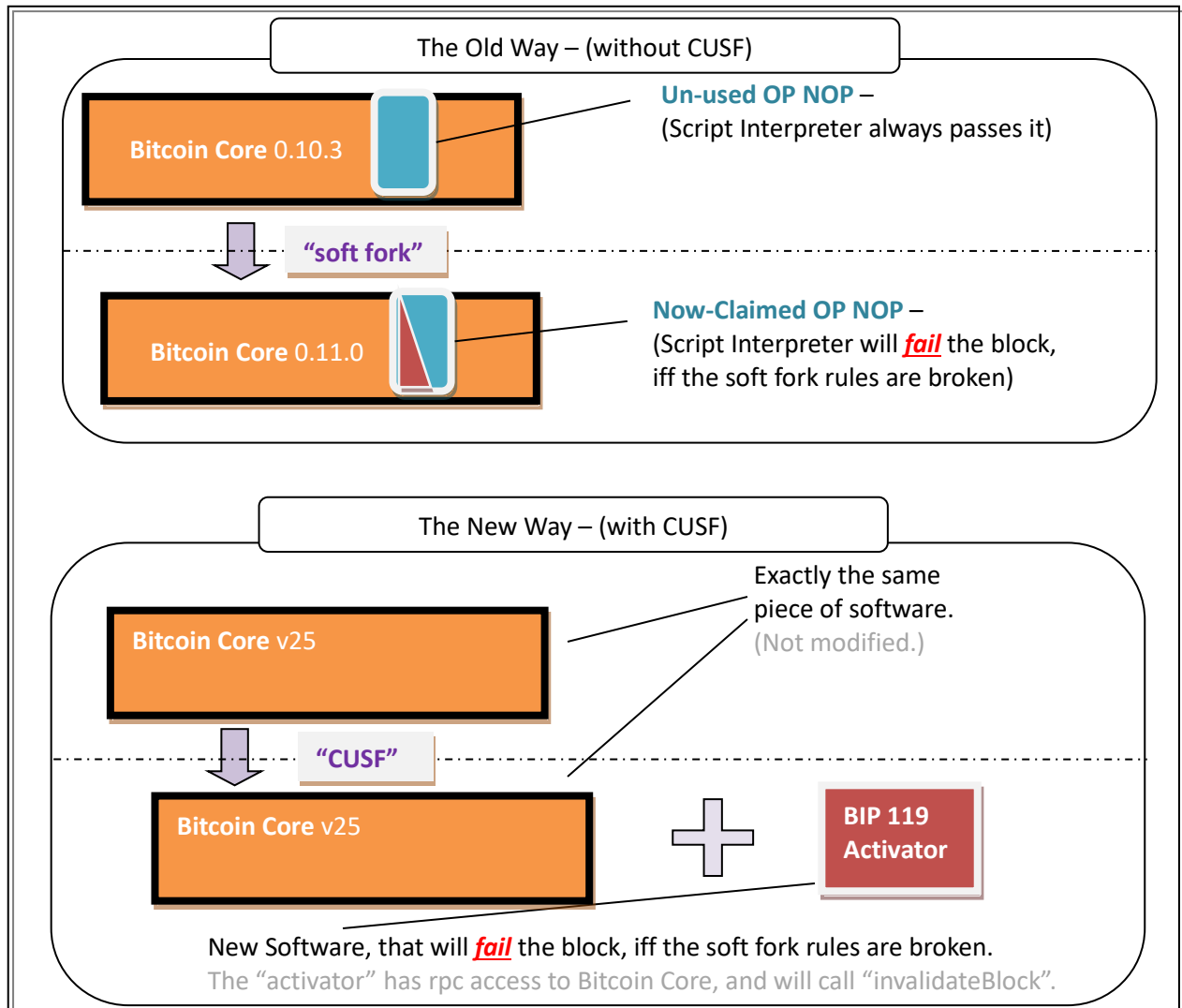*or "The Ordinal-ization of Soft Forks"*

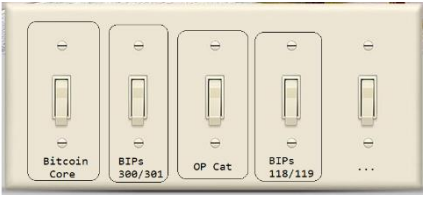**Paul Sztorc**
v0.4.1 -- 6/23/2024

Summary

Each new soft fork (SF) should be a separate, standalone piece of software, "piloting" Bitcoin Core via the "invalidateblock" rpc. This makes soft forks _faster, safer, and easier to understand_ -- ushering in a new age of Bitcoin Development.

The Idea

The current soft fork process is so vague that arguably _no one knows what it is_ -- but it certainly involves opening a GitHub pull request. Here, I present an alternative process: put new soft fork validation rules in their own, separate piece(s) of software. This software can use "getblock" and "invalidateblock" (via rpc access to Bitcoin Core) to enforce new consensus rules. This has many advantages.

This _tiny_ change, has enormous implications:

	Before CUSF	After CUSF
How are SFs perceived by the layperson?	 SFs are _surgery,_ on our beloved only child. One software (Bitcoin Core) that is "changed" in a permanent, and poorly-understood way.	 SFs are just other apps "on top" of Bitcoin L1 – similar to ordinals. We turn these apps on/off, the same way we'd turn anything else on/off. They are modular and safe.
How are SFs activated?	Follow these steps: 1. Think of the idea. 2. Discuss on bitcoin-dev (mailing list). 3. Write code for testnet/regtest version. 4. Test on Inquisition / similar. 5. ??? Get feedback from users / Twitter 6. Spend 20+ hours rebasing your SF to the latest version of Bitcoin Core. 7. Open pull request. 8. Reply to PR-Feedback on GitHub. 9. Repeat steps 5-8, every 3 months for 2 years. 10. Pull request is merged. (?) (Or not.) 11. Activation logic is merged. 12. Debates about activation, Bip9/8, Speedy Trial, LoT=true, Hashrate Thresholds, UASF -- virtue signaling on Twitter -- 13. Speedy Trial (or whatever), is yolo'ed by someone. 14. Months later, 90% hashrate finally upgrades -- even though they don't really understand what the SF is or what it does. 15. People start using the feature.	Follow these steps: 1. Think of an idea. 2. Write the code. 3. Write a document, explaining how your idea boosts miner profits. (Either via a higher BTC price, or via more txn fees.) 4. Miners (ie Pools) run your software, alongside their existing software. (They can stop running it at any time.) 5. Users also run your software, and start using the feature.
How do you de-activate the fork?	This is so difficult, that it has never happened. It involves: * A hard fork (ie, a disaster), OR * A new soft fork, that censors the 1st SF at the txn level (ie, bikeshedding & authority).	Very easy – people stop running the Activator software. The SF just naturally de-activates.
Speed / ease of Innovation?	SFs are always SLOW and academic. "Like replacing an aircraft engine, while the plane is in	SFs can be FAST and experimental – they can be like startups. They can

	the air".	fail without bothering anyone.
How is each SF justified to the layperson?	We need _to explain to people_ why the SF is safe.	It is _obvious_ that SFs are safe. No existing users can even _detect_ a CUSF. SFs are pushed to the mining side where they belong.
Who must agree to run the SF?	Users of the New Feature, + + 51% Hashrate, + All BCMs, + All who rely on Bitcoin Core	Users of the New Feature, + + 51% Hashrate
What is the Guiding Principle of the Yes/No Activation Decision?	Does this SF "have consensus"? (This is an unfalsifiable theory in practice – it also defeats the original purpose of the hard/SF distinction. At best, it is very hard to measure – at worst it is an unfalsifiable theory.) Will the code be easy to merge/maintain/run ?	Will running this software _increase miner profits?_
Who can be negatively affected by a fork? (In a way other than a reorg.)	BCMs: they must… …evaluate the SF-code. …maintain the SF-code in perpetuity (if merged). …release an emergency fix if something goes wrong.	Only those who _choose_ to opt-in to the new feature. (Note: this includes 51% hashrate, since –in order to have reached this point— they must have opted-in.)
What are today's Developer Incentives?	Bad – we must trust today's BCMs. (Trust them to only make the "right" changes.) Low oversight (or even understanding). BCMs are hard to fire or replace. Each change makes the software code harder for a newbie to learn.	Good – anyone can become a Bitcoin developer at any time. Or leave. Devs compete _against_ each other – (competition keeps developers honest). Developers are accountable to _a neutral external metric_ (mining profits), not a corrupt USSR-style bureaucracy "popularity contest".
Effect on "job security" of BCMs?	Enormous "job security" for BCMs.	Job security decreases. SF innovators can do whatever they like, without needing permission from BCMs. BCM role fades into irrelevance as they become more replaceable "maintenance" workers.
What form can the new code take?	The SF must be in C++. It must be a GitHub/Bitcoin pull request. It must obey the style guides & naming conventions & code architecture of GitHub/Bitcoin.	The SF can take any form. It can be written in any programming language. It can use any style/naming convention.
How might we port the SF to an Altcoin?	The SF would have to be rewritten. A new set of _CMs will be inconvenienced.	Can be freely reused by _any_ L1. (So, Litecoin, Monero, whatever, they can have their own BIP 119/118, without changing their own code.)
How important is code review?	Review is essential.	Review is unnecessary.
Can anyone	Core devs have a veto (incl. silent veto & pocket	Core devs do not necessarily need to

obstruct the process, and get away with it (without accountability)?	veto) , can demand changes in style, formate, language, readability – these can be time-wasting filibuster changes.	be consulted. (Note: miners may _voluntarily consult_ 3rd party expert advisors, and _choose_ to follow their advice.)
Toxic Incentives	The high 90% Activation Threshold results in "toxic limbo": where 2 (or more) 11%-hashrate-coalitions can emerge, and make mutually inconsistent demands – resulting in minority gridlock. This gridlock is an Achille's Heel of Bitcoin that can be exploited by other enemies.	The 50% hashrate threshold is simple, logical, and internally consistent. No 3rd parties have a veto.
Does anyone _suffer_ by acting too slowly?	No – and this is bad! It invites laziness! Today, Bitcoin moves along at a slow, academic, bureaucratic pace. No one feels nervous about being "left behind".	Yes – and this is a good thing! If 49%-hashrate are too-slow to join the 51%-hashrate, then they are at risk. Their blocks may be invalidated by pranksters, causing loss of ALL their revenues (whilst paying 100% costs). This means that miners are obligated to stay "at the cutting edge" of SFs, just as they are obligated to stay at the cutting edge re: electricity, ASICs, cooling, etc.

FAQ

Q: Why is this idea important?

A: Because today's Bitcoiners misunderstand the SF. To the potential ruin of us all.

Q: Why do you want to change Bitcoin???

A: SFs are _simultaneously_ a change, and not-a-change. Like how a car can go both forward and in reverse. Anyone who believes "soft fork" = "change", has _missed the whole point_ of the SF.

Q: Some people [such as Luke Dashjr] told me that SFs are mandatory, and that it isn't "Bitcoin" unless I run the latest version.

A: They are wrong. In a different universe, Wladimir clicks a button to release the next "Bitcoin" release on github, and immediately a sphere of light expands, from his index finger, outward in all directions, traveling through all matter, through even the core of the planet. It destroys old versions of Bitcoin on contact (or magically uninstalls them, whatever). We don't live in that universe. But in that one, Luke is correct.

Q: Bitcoin is just fine as it is! Our success is inevitable.

A: Defeating the USD is hard enough – but that is only the tip of the iceberg. Bitcoin must also defeat <u>every rival version of itself</u>, since these will all be released as Altcoins sooner or later. We must be proactive about every deficiency, and every use-case. We must grow as quickly and rapidly as possible, or we will be replaced by something else that does. We have been lucky so far, but this luck will not last. The "we are fine" crowd, are paving the road to $0 per BTC. This idea is about defeating those people.

Q: What's so great about these new SFs?

A: Bitcoin is underperforming. The UX should not have addresses anymore –the privacy should be complete – we should be able to scale to 8 billion users without LN – miners should be making billions of dollars a year in txn revenue. If the SF bottleneck is solved, we will achieve all of this and more. Otherwise, it will be "Custodial LN" and other anti-P2P services such as Liquid or Fedimint.

…

Some Historical Background on the "Soft Fork"
to shed light on the current irrationality

The **<u>Soft Fork</u>** (SF) is underappreciated in Bitcoin, to a degree that boggles the mind. Never in the history of technology, has a thing --so useful and safe-- been so completely misunderstood. The SF can do almost anything, for free and at the lowest risk. It gives Bitcoiners an <u>ossified</u> protocol that is also <u>extensible</u> – the best of both worlds.

SFs are so safe, they are actually <u>safer</u> than routine code-maintenance. For example: code refactoring, CVE-fixes, dependencies / OS -compatibility – these are all riskier than a SF. The SF is so magical, that it <u>reduces</u> overall code risks, just by being used (for example, it utterly discredits its evil brother – the hard fork) and fosters a climate of modularity and liberalism. SFs are also <u>optional</u>, and even <u>reversible</u> (though the latter has not yet been tried).

For example, Bip300 (my own SF) would grant the protocol the following: planetary scale throughput, zCash level privacy, and limitless extensibility. It is so good of an idea, that it is likely to bankrupt all the rival cryptocurrencies, the rival BTC L2s, and perhaps even many exchanges, podcasts, and other middlemen. Other SFs are equally powerful in their own way, such as OP Vault or OP cat.

All that is true. Yet the common Bitcoiner of 2024, has a strange reluctance to embrace the SF. We now hear this curious phrase: "can be done on Bitcoin without a soft fork", as if this were a good thing. (Hardly!) Some prominent voices –who should know better— mistakenly describe SFs as "a change to Bitcoin", and stir up false prejudice against them, insinuating that they may "screw up the base layer". (As if a library's card catalog could be ruined, if never-before-seen books were placed into an empty closet[1].)

Despite their nearly-divine perfection, SFs do have one victim in earnest: the Bitcoin Core Maintainers (BCMs). BCMs do the thankless (and de facto unpaid) job of maintaining Bitcoin Core on Github. They build each new release. They have their own "way of doing things". They have a culture, norms, etc. They are held responsible if the code breaks. (Thus, they err on the paranoid side; and certainly on the <u>uncontroversial</u> side.) For you or I, to go to <u>github.com/bitcoin/bitcoin</u> and move the code around – that

---

[1] Meanwhile, the Refactor Gang comes into the library every Tuesday, copies over the card catalogue by hand, onto blank cards, and throws out the old cards, ad hoc -- with no complaints from the Ossify Gang. Truly the lunatics have taken over the asylum.

would be as rude, as if they walked into our house and started moving around our furniture. They also have a _prestige economy_ with Tribal Rules about who is allowed to touch the Sacred Code -- and when, and why. This is partly political (and thus, partly corrupt and evil), but also partly rational. For example, consider the case of CVEs – there must be specialist devs _who are warned first_ of severe bugs, so as to patch them secretly _before_ the public learns of them.

SFs have a 2nd weakness: their history. The Blocksize War led to the politicization of –first— the Blocksize increase, and –second— the "SegWit" SF. SegWit activation was plagued by obstructionist Largeblockers. This "descended into politics", and ruined the technical dialogue for (probably) an entire generation. In particular, the word "consensus" took on a new meaning ("all humans agreeing" – something which never happens). Today, SFs are treated to a dysfunctional, USSR-style committee discussion. BIPs 118/119, for example, are "finished" – they are helpful, harmless, coded, and tested. But BCMs refuse to even _discuss_ activation. Thus 118 has languished, un-activated and unused. For 119, this pattern repeated, with the variation that Jeremy Rubin released an activation client (and also an anti-activation client), and was then immediately criticized by "the community" to the point that he quit -- not only quitting 119 but also quitting Bitcoin Development and Bitcoin itself. Years later, his critics would be the very same people cheering for "ARK" and "CTV" -- the very same idea they hated years before. It is the _softfork process_ that is broken – not 119. So it is with Bip300, and LNHance, and OP CAT, and every other SF. It is time to bring them all back.

Appendix 1. Can corrupt CoreDevs block CUSF, in any way?
(This question asked for Theoretical / Political Wargaming purposes only.)

No.

First -- they may try to remove the "invalidateblock" rpc. (After all, it is already a "hidden rpc" not shown under the help menu.).

But probably, they would not dare to try even this. First of all, "invalidateblock" is very useful, especially during times of crisis. Second, _today's software_ has "invalidateblock" –devs can only remove it from future versions. But miners/users may never run those. (Ironically, this would be akin to a hard fork.)

But second -- enforcer does not literally require invalidateblock. Instead, it could just _repeatedly delete_ the offending block, off of the user's computer. And then restart the node (or call -rescan). The enforcer's delete operation will be faster than the re-download – the enforcer will be able to [parse/delete/force-rescan] a 2 MB block in about 2/10th of a second. But to connect/download/rescan/validate a 2 MB block, takes +100x as long .

The Enforcer has rpc access, and can therefore do all sorts of things. It can _peers.dat_ or _banlist.dat,_ it can parse blocks and corrupt them in some fine-tuned way (after which Core will see them as invalid). In a theoretically extreme case, the Enforcer can conjure up a 2nd instance of BitcoinD -- in regtest mode, where the enforcer is the only peer. Thus it would only ever see blocks that are soft-fork-valid.

Appendix 2: A Brief Timeline of Bitcoin Liberalism

Different people use the blockchain in different ways. This is a timeline of various attempts – successful and unsuccessful – to resolve the issue of disagreement.

Year	Invention	Enabled…	…But:
2010	BitDNS (aka "Namecoin")	…a new blockchain, with a new feature (human-readable name ownership).	Rival PoW system, + rival coin-unit.
2010	Merged Mining	…mine many blockchains at once, for free, "increasing total strength".	Still a rivalrous coin unit.
2012	"Soft" Change ("Soft fork")	…deploy new features to the whole network, without needing everyone to upgrade their software. (51% hashrate must upgrade, only.)	Only works for some features.
2014-2017	Sidechain Research	…discuss "sidechains" in the public consciousness. Bitcoiners can utilize *any* existing feature.	No software – research only.
2017	Major Hard Fork	A new feature (8 MB blocks) is deployed (via BCH). BTC-UTXO-owners automatically own BCH, so they are not impoverished if the new feature succeeds.	Competitive struggle over network effects, and the "Bitcoin" name. Enormous advantage to the status-quo coin.
2021	zCash Regtest Demo	Software demo of a P2P sidechain – cloning the Altcoin zCash.	Regtest/testnet -- not real BTC.
2023	Bip300 Pull Request	"Real BTC" sidechains.	Corrupt/dysfunctional Bitcoin Core monopoly is run via prestige economy, does not prioritize what is best for the network.
2024	CUSF	Activate via a 2nd daemon -- without modifying the 1st daemon (ie Bitcoin Core). Allows BIP-Authors & Bitcoin Core to ignore each other.	Requires 51% hashrate. Miner collective action problem – early upgraders take most risk, and later miners free-ride off them.
2024	SHAD	Miners upgrade independently of what other miners do. (In fact, miners now prefer their rivals be slow to upgrade.)	Somewhat disruptive for a few months.