

00:00

Pre planning survey and these presentations. And I hope that by the conclusion of this sprint, we'll really have a clear sense about top level planning and.

00:17

Have the project board all golden, looking.

00:21

Great, everyone having access to it and so on and so forth. So hopefully moving forward, it'll just be very much about jumping into the board. Can we turn on our videos and say a quick hi before it came out? Yeah, absolutely.

00:43

I like seeing everyone's happy faces. If you're happy. Yeah, say hi because we're a lot of us and it's been a while.

00:51

I know. Are you strong? Can you see me? I've got my video on now, but I've got sort of limited screen real estate today.

01:02

Yeah, we see you.

01:04

Great.

01:05

Thanks to everyone.

01:05

Cool faces.

01:10

All right, so like I said, we received the technical specification documents, the project requirement documents. OKRs. Just really taken a little bit of I mean, just so much good content and it's going to take a little while to digest it. This is pass number one going through that really went through and made sure.

01:36

To pull out the section on the deployment plan.

01:42

And really obviously Nakamoto SBTC planning and.

01:50

Are a bit further ahead.

01:52

Clarity VM is still pretty new, and then testing and Hardening is sort of like this meta layer that we want.

01:59

To always make sure that we're paying attention to.

02:02

But it's more difficult to talk about the specifics with that because in addition to the testing infrastructure, it's about applied testing and responding to the demands of the other projects. So we'll probably spend a good amount of time talking about Nakamoto SBTC. Bryce is out today, but I've got a slide to talk about Clarity just so that we can get back, make.

02:30

Sure that everyone's getting on the same page with again.

02:36

You know, really the goal is we're going to define some terms anytime that we feel like we are not speaking the same language. We want to make sure that we have a common vocabulary. So hopefully we'll talk about defining terms, defining scopes, really that goes hand in hand with terms like just when we're talking about a certain deliverable or epic or release or whatever. We want to make sure that we're all consistent in our understanding in terms of what's included in that and not included it. And then talk about the deployment plan, as you probably saw, getting more and more detail onto the project board. I'm not going to go through item by item in this call. That's what we're going to use the working group calls for this week, just FYI. So really the goals of this call, we want to make sure that we're identifying any action items.

03:38

So if you see something comes up and it's actionable right now, I want to make sure that we get it recorded in the audio. That is a big help for me when I'm going through the transcripts and employing all those things out. So please be sure to unmute and say, hey, this is an action item I've got, or hey, this is an.

04:02

Action item for so and so that.

04:05

They can work on. Right now we want to identify any dependencies like this part of this project is dependent on this other part of.

04:14

This project, or this part of this.

04:17

Project is dependent on this part of this other project. So internal to the project and interproject. We want to be able to pick that up and then also identify any efficiencies where are we doing work one project that can be applied to a couple? How can the efforts around Nakamoto help SBTC, help clarity vice versa all the way around, identify any risks on the.

04:47

Horizon and how to mitigate those and.

04:50

Then set some realistic deadlines.

04:54

And also as we go be calling.

04:58

Out action items in and around, testing and hardware. I think anytime that we can do.

05:04

That, that's going to be very helpful. So we'll jump in first with Nakamoto and Jude.

05:15

I think you're going to be sort of instrumental to this conversation.

05:21

I wanted to first start by just.

05:23

Talking about for myself, I like to.

05:28

Take the technique of.

05:33

Explaining it to.

05:33

Me like I'm a five year old.

05:36

So we talk about Stacker DB, we.

05:39

Talk about producer and signer. I think that there's been some language.

05:44

In there, producer and stacker signer and.

05:48

The idea of a stacker versus a signer. And then also you can unpack what's a producer versus a miner.

06:00

We could easily spend the whole hour doing this. Are you sure you want to?

06:04

Well, maybe if you could just kind.

06:07

Of give us the top level so.

06:12

That we can keep that in mind as we drill into the details of the project.

06:20

I see Aaron's Chat sorry, one sentence is not going to stop me from giving you a mouthful. Let's see stacker DB, it's an extension to the peer network that allows the Stacks network to store state on behalf of a smart contract like off chain state and mutable state. So Alice and Bob can share messages with each other as long as the smart contract that they mutually agree to use allows them to do so. The information is not written to a chain, it's not part of a transaction. So it gets propagated at line rate. So as fast as possible, the message that Alice writes can change later. The state in a Stacker DB is soft, it's not part of the blockchain, it can disappear, it can get deleted. It's best effort following.

07:13

Yeah, I am good.

07:15

Okay.

07:16

The reason it's a distinct unit in this is because it is a building block by which the two actually three sets of signing entities in the system. These being the block producers, the stackers who sign, the block producers blocks and the stackers who separately sign SBTC, peg ins and peg outs. This is the shared medium by which these three signing sets will coordinate with each other across the Internet in order to carry out the execution of threshold signatures via Thrust.

07:50

Got it.

07:51

So you cannot utilize the term when you're talking about Producer and signer, and then occasionally you'll utilize producer and Stacker signer. Are you using those interchangeably? No, they're not. Okay, so they're always distinct and separate from one another.

08:15

Correct. A signer in the Frost parlance is simply a term for one distinct share of the key. So a frost is an M of N . Threshold signature. If you control one out of the M or sorry, one out of the N , then you have one signer. The text describes an implementation detail of how we achieve weighted threshold signatures by assigning one or more signers to a single entity. So if I'm a block producer with 30% of the mining power, I would receive 30% of the 100 signers that are allotted to block producers during a tenure.

08:53

Excellent.

08:54

Thank you very much.

08:56

And then if we jump from there, and I think the next slides are going to help illuminate on all of these things again if people aren't following along. But could you just maybe touch a little bit about the BDF and how that relates to the Noble Gas releases that you identified? A VDF. I was researching it this morning. I guess the function that it could play in a protocol like this is creating some type of randomness at the consensus layer, ensuring fairness during the block mining invalidation, and then also just like adding some delay to protect against Precomp attacks. Is the intention of this media all of the above, or is it more pointed along a certain direction?

09:51

It's a mixture of right and wrong. What you just said. A VDF is like a stopwatch. It's a stopwatch, but in a world where I can control how fast time goes for myself. The high level problem we're trying to solve here is we have finite block capacity by design. If we did not have that, then I could send I the attacker could send a stream of transactions that take arbitrarily long to execute and stall the network indefinitely for doing so on the cheap. For this reason, every single blockchain that works has some sort of block limit, which you can just think of as like, I do no more than X work for some useful meaning of work per unit of time in our system or in the system proposed here. We have the notion of a tenure which is ten bitcoin blocks during which a single producer set makes all the blocks.

10:42

That producer set has a finite budget of how much computing work it's allowed to do in processing transactions. Now, the problem with using bitcoin blocks as a means of trying to measure how to spend my time during my tenure producing blocks is that we don't know when bitcoin blocks arrive. We know statistically they arrive every ten minutes, but it could be the next two minutes, it could be the next hour somewhere in between there. So if I'm a producer set, like, if I knew for a fact that it was going to be ten minutes, I could calculate pretty easily, oh, well, I'm going to release one block every X seconds so that by the end of my tenure, I will have run out of budget. But the next tenure is starting right away next anyway, so there's no downtime. We can't do that if we use just bitcoin to look at to determine how long I should wait, because I could be waiting anywhere between a few minutes to a few hours, but on average, 100 minutes.

11:39

So to avoid the bad case, which is where I end up using all my budgets at the 100 minutes mark, but then waiting another 100 minutes for the next bitcoin block to arrive, which could happen, leading to a terrible user experience. The system instead implements a distributed stopwatch via this VDF, where I, as the producer and my contemporaries, my other producer friends, can each prove that, hey, we actually did wait 100 minutes. Can we have some more compute budget, please? So we can keep the chain going until our tenure ends. If they can prove this, which the VDF allows them to do, then they can continue to produce blocks and keep the chain working until the next tenure arrives.

12:24

Awesome.

12:25

And then this may just be new to me, but this is the first time I've seen this nomenclature of the noble gases in your documentation. Is this something that you've decided to use for this particular effort, or is this something you've used before? And could you talk a little bit.

12:46

About.

12:48

The different phases?

12:50

This is all Ludo's idea. When we're releasing 2.0, this was his code names for the various milestones. I'm just reusing them.

12:57

Okay, cool.

13:00

So jumping from definitions of terms to.

13:05

Definitions of scope.

13:11

Your particular, I guess, deployment plan provided us with like an X and a Y. So we've got set levels of scope, feature sets, if you will, and then different releases for each one of those that's going to roll out. So you can think about that across time and we'll get to that later on.

13:37

But I thought it would be first.

13:39

Very helpful to just talk about what each of these are. I put the bold white outline around the V one of the blockchain rollout, because this is something that you identified as a pretty high priority item. And then you've also identified, I think, in a separate document with me. Like the VDF one is something that I believe you mentioned Jose is going to be working on here pretty shortly. So could we maybe just talk about going horizontally across all the V ones? Set aside the green column, but amongst the VDF, the stacker DB, the producer and signer and the blockchain rollouts. How do we get this ball rolling in terms of all of these different pieces and components? Is this something that is easy or not easy, but that you're able to delegate to others? Or are you still sort of in the architectural phase where it's best kept with your hand?

14:52

So each of these columns, besides the green one, can be thought of as independent work streams that largely do not depend on each other. With a few exceptions, the stacker DB is necessary to be operational before we can start end to end, testing the producer and the signer because they rely on the stacker DB to coordinate. However, we can get a long way into making the producers and the stackers sorry, the stacker signers to a workable state. And moreover, most of the implementation is already done. It's the SBTC signer. We're just going to modify it slightly so it works for blocks instead of for SBTC transactions. And we already know it works pretty well because we already use it on the SBTC Alpha. The VDF is something we can get off the shelf. There's already multiple implementations in other blockchains that we can just borrow.

15:39

The hard part with the VDF is the calibration system that allows us to make a distributed version of the VDF, but that could be done separately from the rest of the blockchain work. Blockchain changes. And rollout is probably the source of the biggest unknown. Unknowns because this is the task that captures all of the new consensus rules that eliminates forking changes the structure of blocks so that we just have a stream of blocks instead of blocks and microblocks changes the way in which we capture and relate stacks blocks to the underlying bitcoin blocks in order to support clarity operations against them, dealing with bitcoin forks and so on and so forth. The single biggest task, I think, in all of this here, before the end of this quarter, is going to be getting that work stream done, because that has to be done in a good enough form so we can have a very simple, very primitive version of the system running by the start of Q Three.

16:33

Sorry.

16:34

By the start of Q four. That system for my OKR is simply the consensus rules work well enough that we can produce a stream of blocks and those blocks can still read bitcoin state. There only needs to be one producer and one stacker signer that's actually signing. We don't have to worry necessarily about making sure they can aggregate sign just yet. But it's of paramount importance that we get the blockchain work done and out of the way so we can spend the remainder of our time Q Four making the system much more robust with an actual, increasingly complex rollouts until we reach main net.

17:11

Okay, awesome. So the item that you're focusing on here is the one with the bold outlines, the V one of the blockchain changes and then Martin or Jacinta. Jose, is there anything that's jumping out here in terms of items that your ears are perking up and it wasn't on your radar or areas where you're thinking, hey, we can jump in right now and lend a hand? It sounds like there's some overlap between the signer work that's happening with SBTC, the Clarity work that's happening with the.

17:56

SBTC.

17:59

Help me understand the amount of collaboration that's happening behind the scenes on this.

18:10

So in terms of collaboration, we're currently at the stage of syncing up the current state of things. So I am meeting mostly with Morten to go over potential design for Miniv One for the signer setup because it's not been finalized and there are multiple approaches that have kind of vastly different implementations. But in terms of overlap, there is considerable overlap with the Clarity working group, and potentially, depending on what design we go with, it might be even more so. So a lot of the coordination, the signer coordinator, a lot of that functionality could potentially be within the smart contract itself. And if we go that route, there would be even more overlap. So I think Marvin is the lead of that working group and is away in Japan at the moment. So I'm hoping in next week to meet with combo of Jose and Jesus to go over where our overlap kind of is and what design we think is best moving forward.

19:17

So that's kind of where we're at.

19:19

But you're talking about overlap with the Nakamoto release.

19:23

Oh, sorry, I missed that, honestly, because.

19:27

I guess this is a separate deployment of the signer, like within better blocks. Like the concept of using a VDF and signers to have a pool of block producers produce faster blocks, essentially. Right? And as far as I know, we haven't had that dialogue. So I'm pretty excited about finding synergies between, especially we as Cinta said, we're ironing out the design and how designer fits in SPDC to just get that written down. I think at that point it's going to be much easier for us to break out and scope reusable pieces of designer depending on how Nakamoto is designed and how we can find synergies in between there.

20:17

Great.

20:18

So with that in know we're just.

20:22

Doing some formatting and some hyperlinking or whatever with pushing the technical specification docs to GitHub, but I shared it as a skiff yesterday. If anyone on SPTC side wants to get in and sort of read through what Jude wrote up for Nakamoto and.

20:45

Begin to build some familiarity with the system that's being proposed there and begin.

20:52

Suggesting opportunities where work that's already been built for SPTC could land a hand.

21:00

On that side, that would be really great.

21:06

To speak just a bit more to that. I expect to get to a V one level by the end of this sprint. We have the bulk of it written already. It's just undergoing review and some refactorings along the way to perhaps make code coverage a bit better. The biggest thing I could use help with is actually both is this VDF implementation and the producer signer implementation. I can take the blockchain changes myself, and probably Aaron can take those, but those are the pink and the dark blue columns, the two biggest areas where there's a lot of opportunity for parallelization.

21:43

I assume that the producer designers would use Stackdb for internal communication. Right. And that's also something we want for SPTC. So that's also a dependency on the SPTC system, on the signer, and also a reason for correct.

21:59

But for V one, we don't need the Stacker DB, we just need to have one of them running.

22:05

Vivan Nakamoto or Vivan SPDC.

22:11

For Nakamoto specifically, the end of the quarter goal is actually that light blue column is done. Everything else is a stretch goal.

22:21

Okay, that's good. And also, I guess for SPC, there's no reason we can make a hard fork even without Stackerdb, so there's no extreme impression on that one.

22:32

Well, Stackerdb doesn't build ourselves in a corner.

22:35

Yeah, the Stackerdb is not a consensus breaking change, it's an add on. We can roll it out as soon as it's ready. In fact, I intend to do that so we can start testing it, like really testing it in prod as soon as we can.

22:46

That's good.

22:49

And then, Jude, is there anything here that jumps out in terms of getting on the radar of the testing and hardening team that they could be working? I see that you're pulling out, like, benchmarking and improved performance around Stackerdb, which you're hoping to sort of get out soon. Is there anything that could be done in anticipation of that by others to help support that rollout?

23:21

So, I did say that Stacker DB is not on the critical path to a V one per se. However, I do think that if I had to have two tasks on the critical path, it would because it's such a crucial component of the full system operation. Hence my working on it back in Spring, right before I went on leave, and hence my finishing it right now before even starting the blue work. If I can get that to a state where it works and we have a fully functioning artifact, then yes, we can start running it in production, we can start running it on testnet, we can start hammering it, we can start working on it with the SBTC mini. It's not necessary for it to be at a place where we're happy using it for block production, but I think that it would behoove us to try to get it out the door as soon as possible so that the rest of the ecosystem can start playing around with it and reporting bugs.

24:07

Awesome.

24:13

So thinking about this in terms of time, this is, I guess, reflective of the working document that I shared with you and the other engineering managers the.

24:27

Other day.

24:31

It's still pretty top level getting this added to the project board to follow this pattern. And I thought just being able to look at it as the minimum amount of line items across time before we get too detailed will help us understand if the broad strokes are consistent with what you're thinking, what is realistic, and what needs to be done sequentially.

25:08

If I could change this document. I hadn't tried to do it, but if I could make a change the light blue sections for the blockchain rollout, they would start no later than August for Sprint. Ideally, they would start that actually what I would start them. But they would continue to stretch until at least the end of September for the V One because that's going to be a substantial amount of work. The VDF can happen anytime. That can happen now, it can happen August. We don't need to tie it into the blockchain work until the blockchain work is almost done.

25:38

Okay.

25:39

The stacker DB stuff is almost done. We're probably going to finish the V one of that by the end of July. So that could be shrunk. V two and beyond for Stacker DB, I anticipate is going to be we found problems, we have to fix the problems. It's less about getting individual work things done and more about reacting to iterating on user feedback, users, in these case, being us, and any ambitious developers who want to give it a try for their own projects.

26:06

Okay, great.

26:08

So I will make the updates to this and share it with you. And you're obviously more than welcome to jump in there and drag and stretch things around yourself. I'm in present mode, so it's hard for me to make changes on the fly, unfortunately.

26:25

Yeah, no problem. I'm happy to do that.

26:28

I think one thing I want to highlight here is there's an important point here just looking at what Jude was saying about the VDF. The VDF in my mind is an example of something that people can just make progress on that in. I think I think we should try to identify as many things like the VDF as possible right now and then try to resource those things outside of this critical bus factor that we have.

26:55

On the blockchain side, I think that's.

26:56

Actually going to be very important. So more engineers, folks outside of Jude, Aaron Bryce, like, they can make progress there and that would immensely help us as we go down the road. So I think we should flag this thing. I think the first exercise is like, how many such things can be identified and then those components can be separated out. And then we basically look at our resource pool and try to figure out, like, hey, who are the folks who can make progress on it? And it's not like they're just going to pick it up and start making progress. We'll still have to scope it out, have the protocol specs, have a sign off that hey, this is exactly what's needed. But I do think a lot of parallel work can happen here MANIT if.

27:38

You have a chance to even just skim the work plan document that I had given to Will. One of the very first sets of tasks that ought to be completed before anything else on the Anakamoto work is the implementations of traits that abstract away the bulk of the blockchain works so that progress on the new pieces can be made without having to become masters in the old pieces specifically to enable.

28:03

This sort of thing.

28:04

Awesome.

28:08

And then one thing I want to.

28:09

Highlight from all of the technical spec.

28:13

Documents, but particularly Jude, is there's quite 20 different variations of testing laid out in great detail in there. So if anyone from the testing and hardening work stream wants to get in there and read up on that and begin anticipating what infrastructure could be built.

28:45

To help expedite that process when it's ready, that seems like something that could.

28:55

Help speed us along. Also cool as I noted, I will come back to this and share with you revised one and we'll make sure that we get something that is reflective of your thinking and feels manageable by the end of the day today and get this shared with everyone. So we'll jump from there to SBTC.

29:25

So.

29:30

I guess as alluded to in the conversation yesterday thinking about SBTC as these almost like tiers of product, primary, secondary, tertiary and really there's the asset, there's the protocol, there's the developer tooling that goes along with that and then there's a lot of the user experience, user interface, touch points that are also being done in concert. A lot of those by critical bounty recipients.

30:11

And so I guess I just want.

30:14

To make sure that we're always clear on what the core engineering team is working on and prioritizing and also make.

30:28

Sure that.

30:31

The shared vocabulary between what Andre and Rena do on the product side aligns with the way that the technical engineering team is working. Just so that everyone is kind of speaking the same language, so to speak. So I kind of took some liberties. Martin, I took what you identified in your document alpha Many MVP 1.0 monitoring.

30:59

Strategy and then I kept this kind.

31:02

Of.

31:05

Type of sip 21 end to end on the sheet so you can ignore that green one. But thinking about these primary Issuances finishing up alpha, really the main focus is on Many. So if there's anything that you wanted to call out in terms of the delineation at the primary level of how we're thinking about these different Issuances, that.

31:38

Would be a good place to start.

31:40

And then whenever you're ready we can jump and start thinking about this over the course of time. And this is again just a first pass at taking the notes and making sure that we're converging on a realistic deployment plan.

32:02

I have some questions clarifications here. I think I understand what SPDC mini is, but MVP and 1.0? If folks can sort of call out what the differences are there, that would help.

32:18

Yeah, I could clarify. So the intention has been communicated and of course everything is up for debate. But SPDC MVP is the first, like the minimal set of features that are upholding the final SPTC, but maybe compromising a bit on a lot of details on the products. That doesn't make it like 1.0, but it's the first consensus breaking release. Whereas SPTC 1.0, then we're seeing it like, then we're having everything, like recovery mode, the ability to vote for what is it called, the liveness factor, like all of those details that we can potentially cut out for an early.

33:02

Got it.

33:04

Can I suggest a different thing here? Because I saw a conclusion about this in this meeting as well. I think if we call SBDC MVP Nakamoto release and we call the 1.01.11 .1, sort of, like, gives the idea that, hey, you launched something on main net and now you're improving on it with the new version versus 1.0 feels like, hey, that's the main release, like, on main net, and nothing else is on main net before. I think MVP is likely what people generally think of as knockout, where the consensus algorithms are going live and everything's going live on the main net.

33:45

Yeah, it's in the chat that Andre proposed changing mini to MVP. I think some of these things we need to think about, like, especially how we are targeting the consensus breaking release. I think that's a very good discussion to have. We could break it out to Sptz 1.0 or 2.01.01 .1. Mini is substantially different from the consensus breaking release and we're sort of discovering how different that is. But it's still like interface changes. So I think we still want to keep that name a bit different.

34:26

Yeah, a little bit of clarification there because right now Clarity Working Group has three separate versions of mini that were supposed to be released. And so I'm kind of confused if other NVP counts as one of these or whether okay, so mini one, two and three goes inside mini. And for those aware, I'm talking to the Stacks 3.0 roadmap, which I just dropped the Google sheet here. But yeah, if you guys look at columns KL and M, we definitely had three versions of mini planned out, whether that is fixed and we end up releasing all three or only release two within the next coming months. I just want to raise some visibility to that because that threw me off for a second.

35:15

Yeah, I think that's a very good observation. The way I'm reading this again, this is my interpretation of this, is that SPDC mini is SPD mini three, right? And one. And two are early versions that are leading up until SPDC mini, but until we have SPDC mini three, as defined in the spreadsheet that you linked, that is like the final release of SPDC minia.

35:40

Perfect.

35:41

Yeah.

35:41

As long as that's very clear to everyone else here because that confused me for a second.

35:45

Cool.

35:46

Yeah. I think one thing to keep in mind is whenever the Nakamoto Consensus breaking changes happen, that's the launch that the world will be watching. Right, like even from a marketing perspective and other stuff. So I think that should be highlighted as that's the thing that's the big launch for all practical purposes, but we know that there are certain missing features like recovery mode or voting and other things that will be launched in a version after. So I think just making that clear to people because right now my eyes just go to SBDC 1.0 as the main big launch, which I don't think is the intention.

36:33

One observation on that spreadsheet is I was wondering if we could simplify the rollout of mini to have two release versions. And the thinking with that, just looking at the overall timeline that we have leading up to Nakamoto, is if we actually need that third release and maybe we could simplify the development by just kind of folding releases two and three. And then the third version is essentially what would go live on Nakamoto. So that's one thing that I think is maybe open for discussion for the broader group, if that would be helpful to kind of simplify those releases. And yeah, just generally just on the nomenclature, was thinking that I've kind of been viewing mini as the MVP. By definition, it is the minimum viable release where we'd be able to have the deposit withdrawal functionality within open stacking pool. And so just kind of like it's MVP because it's not consensus breaking, it doesn't entail all the additional functionality that we have coming later.

37:33

So might just want to kind of.

37:35

Open that up with the broader group.

37:36

See if we can refine those outstanding items there.

37:43

Yeah, sure.

37:43

And Andre, I know we have a call later to talk about basically I'm sure it's going to include that. I would say from there, I'll go back, talk to Marvin Friedger, the group there and see what we can cut out and or what dates we can shift. And on that note, I also do want to bring up a design that Marvin laid down. Martin, you're aware of, I'm happy to talk about that as well, but I'm not sure if right now is the time to talk about that.

38:09

Is it the peg outs on stacks instead of yeah, I think that's not a planning discussion, it's a super interesting idea. Probably some strong opinions out there on that. So we need to make it visible and see what it entails. But not in this forum right now, if that's okay for you.

38:32

Martin or Stefan, can you speak to how and.

38:36

Where the SDK would fall into this?

38:45

Stefan, you do want to no, you go first. Yeah, no, exactly. I mean, the SDK is essentially the fundamental piece in building a lot of the I would say it's tetrarchy products. Right. If you're building tools dashboard. Well, actually the bridge is a secondary product here within there. So I would say that's also very important. It's not part of the core protocol, but it's the reference implementation of the protocol and also complements the documentation. The documentation of the protocol I would say is actually primary. So that's something we can figure out if we want to. Yeah, let's not bike shedding too much. But it's like since we're delivering a protocol, the major delivery, of course we need to integrate it in the stacks blockchain and have a working system and have working tooling around it. But if it's not documented, then people can't build on it.

39:38

If it's not defined, then it's not a system. So it's not released if it's not defined. And the documentation will eventually up to updates of the Sip, which is getting more and more updated as we continue working on this. But it's better to have a more living documentation that we can consolidate into the stack exchanges into a sip.

40:04

Great.

40:04

Yeah.

40:07

Go ahead, Steph.

40:09

Yeah. Completely agree with Martin. The only thing I wanted to add is that the SDK might allow us to kind of keep everything organized as we go through like Alpha, Mini MVP and everything with a single implementation. From our kind of experience with the Alpha, if we don't have a canonical implementation, we will end up either having multiple implementations, multiple tests, like basically duplicated code that will be hard to either maintain or fix. So if we invest in this early, we will have a reusable library that we can use in all of these dashboards mentioned at the bottom of the slide, like even the bridge. I'm not really sure what goes into that, but yeah, it's not a primary thing whatsoever, but it might help with.

41:03

Everything else if we kind of build.

41:06

It in a nice way.

41:08

It's like a catalyst that enables us to move fast. For example, the things Alpha has taken a lot of time, a lot of bugs that wouldn't exist if we had gone the library first approach previously. We could have saved months of work on Alpha and months of debugging by doing it that way. So this is just about accelerating our development going forward.

41:32

Sounds highly valuable.

41:36

Yeah, it's extremely valuable. And also that's the internal egoistic motivation behind it. And we also have the external motivation, like having an SDK is going to accelerate. It's going to make it more attractive for the community to build on SPDC. So, yeah, there's a lot of benefit to it, but yeah, it's not directly tied into any of these deliveries. But it's a very strong supporting function. I would say almost the same as documentation. Documentation is a supporting function in the deliveries. Like we can't deliver without docs. But since this is a protocol that's also one of the delivers, I'm also going to try call out to your AI action point to have a discussion on rebranding mini as MVP or something else to reduce confusing because I'm seeing a lot of activity in the chat about whether or not we should call it MVP or not.

42:32

And there are some good questions to do. Does it make sense if anyone drives calls to that? Or maybe we can just do a GitHub discussion where we can settle the arguments.

42:49

Yeah, if you want to take I say we take a few minutes right now and try and settle it if we can. Can you just reaffirm what the debate is? Basically rebrand mini as MVP. Rebrand MVP as Nakamoto and 1.0 as 1.1.

43:20

I think there are a few points here, but yeah, personally MVP as 1.0 is not super controversial as I understand it. Mini as MVP, there's a lot of opinions on whether or not it should be and I think there's some other igor had a proposal on renaming everything. Jacinta raises your hand.

43:40

This is just my personal opinion, but I would steer away from unofficial channels ever referring to a release as MVP just because what someone might consider MVP is not the same as someone else. For example, tokenomics to make a system viable from a tokenomic perspective, I would consider minimal viable product. Like if there's an incentive to do the work, then why even have it? It's more a proof of concept, if anything else than an MVP. So I would probably just sure you can call it that within internal channels, but in terms of marketing and releasing to the broader public, I wouldn't call it that. I would come up with some other name even 0.0. Like anything like that I think would better.

44:32

Then we need to brainstorm a bit. Let's play with the idea. I'm all in favor of dropping MVP completely, like no MVP. And I think there's feelings like people want to the reason to put MVP on mini is that we want to get away from the mini name as well. Right? So like what Igor proposed, SPDC 1234. Just version numbers. Yeah, that's something people can stand behind.

44:58

I think the reality is that as we will get closer to the launch, there is no way that stuff is going on mainnet and tens of millions of dollars are coming into the SBDC contract without us feeling good about this is good enough. So I think it's totally okay to have a placeholder that here's the next version. We can call the next version, whatever it is to avoid feature creep what Jay Valley was talking about. Because otherwise all the features just make it into the main thing. We're saying that we're going to maintain a minimal set of features doesn't mean that it's not complete, doesn't mean it's not secure, because we're responsible folks, right? We're not going to launch something with tens of millions of dollars coming on it, maybe hundreds of millions coming on it that we don't feel good about. So I think it's not MVP.

45:45

It's the main launch. I think people are already calling it the Nakamura launch. We can probably just use that name, SBDC is Going Live.

45:53

Right?

45:53

We can just call it. SBDC is going live. It's the first version, SBDC is Going Live.

45:58

And then there's the next version that will 1.0.

46:01

Yeah, then there's a fast upgrade on that. And I think most folks, when looking from outside, they're not going to go into the details. They just want to know when the main launch is happening, and they want to learn about that, and then they want to learn about what is the path towards it. So in my conversation with outside folks just yesterday, there were some people who wanted to learn. And I sort of indicated to them that watching mini, going live as a smart contract, and then watching the testnet of Nakamoto are probably the two most important things you can watch to gauge how far along you are. Because people are going to look at the Nakamoto testnet and the mini on sort of like the Stacks mainnet to see, like, okay, this is moving forward. And then they know that from testnet, they just need to go to and obviously there's a ton of work in between, but that's one way to look at it.

46:52

Yeah.

46:53

One quick note on the mini nomenclature. I was doing some research on this past week, and one industry framing that I've seen come up is called basically a Chaos Net, which is an unaudited and unrefined version that would allow developers to really focus on testing in live environments. So we could consider that or something similar, if you think that would be appropriate.

47:19

I like Hesses proposal that he wrote. SPD stacking pool. Maybe we could call it Spdzp or something for short, for mini, because that's actually what it is. It's a variant that's based around the Stacking Pool. And then we have 1.01.1.

47:36

Cool.

47:39

We can finish this async and then Martin, could you speak a little bit about the monitoring strategy that you flagged and when that can start and how.

47:51

Others can get involved?

47:56

Yes.

47:57

Did I flag this? I mean, it's quite important to monitor the state of SPDZ, right, and make that visible to people around the ecosystem and having a strategy for understanding. Well, I mean, there are two things that we need to monitor for SPDC. One is our interest in releasing SPC. Want to monitor activity.

48:17

Right.

48:17

We want to see and do analysis on who is participating in protocol, like, what's the locked up capital in SPDC, all of that. Then, of course, as developers. Well, that's a bit more since it's a distributed protocol. It's not really the same thing when we're talking about monitoring that, so.

48:39

Yeah.

48:41

Okay, great.

48:44

Sorella, any thoughts jumping out at you with this whole conversation? And I guess you're always really great at being able to pull out actionable things that folks could jump on.

49:06

Actionable things? Well, we're still quite siloed in the different working groups. Right? So the Clarity work stream has the best knowledge of the actual Clarity work that goes on. Jacinta has the best context on actual designer work. We have Stefan who is scaffolding the new stacks, SDK repo and things like that. And that will, in time, quite soon, hopefully have a lot of issues where ICS can contribute more. Right now, as far as I know, all the ISIS have their sort of primary source of places to contribute to. We're closing off Alpha and still have a lot of some tickets there to run off. But as we're going with the documentation, getting things sorted out and dragging in more things there, we will get more clarity and also migrating to the board. It's going to be much easier for ICS to find the place to contribute.

50:06

To.

50:09

If that sort of answers your question.

50:12

Yes, that's great. As noted, we'll definitely be spending more and more time in the board, beginning the working group calls this week and.

50:22

Then from there on out and also.

50:26

Make sure to sort of review this with you, Martin. And we can make sure that everything is all the naming and the delineations are reflective of what was discussed here and afterward. So Bryce is out today, but I.

50:44

Wanted to quickly kind of highlight month by month the items that Bryce flagged in his document and just make sure.

50:57

That people.

51:00

Have their eyes on this.

51:01

And Aaron, Jude, I don't know if this is anything that you could speak to or Ludo or Hugo is on the call and Bryce is absent. If not, we can certainly just circle back to this when he's back next week.

51:25

Yeah, I can just leave some high level comments here. I think this work stream, I'm having a feeling that it's both very important for the performance gains, but it also seems like a ton of work and something that can potentially delay things. So the logical conclusion sort of is I think we should resource this more. Right? And if for some reason it's not getting resourced, then I think we should have a backup path that, hey, maybe the WASM version will go live in an X iteration and not on the.

52:00

Launch, but I would love to see it live.

52:03

I think this would be amazing before then.

52:09

From Hero's point of view, we are actively hiring for a Clarity VM specific focused role. Monique and I might have some good news. I don't want to jinx it, but very actively on it. And we have some potential prospects within the current team. I'm also trying to pull in Hugo Ludo is already part of conversations so there'll be more people added to this working stream.

52:40

Mean, just to reiterate something that touched on and many of you could probably speak to this better than anyone, but the scope of the work and the functionality improvements, the features, the performance enhancements that's outlined in the technical documents.

53:01

That came in this week, I think.

53:04

It was noted in there. This is as big of an undertaking as 2.0 and this is some real attention grabbing work and improvements that.

53:21

I.

53:22

Don'T want to dwell on this too much, but you can't deny the fact that this is all lining up with a having coming up and there's going to be a lot of excitement and intention around bitcoin naturally. And I think having all of this good news and the wind at our sales due to the having is something that we should all be really excited about and just plotting, of course, to.

53:49

Make sure that we sync up with.

53:52

That as best as we can.

54:00

But I think that's all of the slides for today and really the gist of what I wanted to make sure that we covered as a team again.

54:15

This is going to be much more actionable going forward.

54:20

Taking the transcripts of this people probably saw a lot of action items being added to the project board yesterday. Working with Jesse and Mark just trying to get there's some people that were not able to tag as an assignee just due to permission. So we're just getting all of that sorted out. And then we'll be jumping into the project board in all the working group calls and talking through that and then all the sprint planning calls moving forward. So that think that we're nearly aligned on what the major boundaries are of what we're working on, and everything's just going to be able to get much.

55:06

More tactical from here on out.

55:12

Any parting thoughts?

55:19

Cool.

55:21

I will see you online and yeah Martin if you want to let me know where you land, where you and the team land on a final decision with naming, I'll just make sure to adhere to that moving forward.

55:37

Sounds good. I'll start a discussion.

55:39

Okay. Thank you.

55:42

Thanks everyone.

55:43

This was a great meeting.

55:44

Excited.

55:45

Thank you.

55:45

Thank you.

55:46

It's been great.

55:47

Excited to have everything on the SPTC board or the Stacks Builders board.

55:52

It's stacks core engine board.

55:55

Stacks core engine board. See it's getting late for me.

56:03

Take care. Bye.