TIMELINE AND MILESTONE

sBTC Clarity Contracts

Estimate Advice

The time estimate should include the time to deliver the full feature into main. Do not estimate based on how long you think it will take to write the code. Estimate how long it will take between beginning the task and completely closing it out. This includes unit tests, writing the PR description, understanding the actual needs, etc.

Estimates should be the number of working days. A 23 day estimate is a little over a month.

Resources

sBTC Clarity Design
Voting Mechanisms

Milestone 1 - Scaffold & Import

This milestone is about setting up the nakamoto-sBTC repo within the Clarity repo, importing all the reusable logic from previous contract versions & setting up tests. This does not include updating or adding new features, simply removing anything not needed & importing confirmed logic from the design document.

Without a doubt, every single time, testing with a PoX-2.5/3/etc… contract is excruciatingly difficult for a few reasons (such as requiring nodes to call private functions). Therefore, during this milestone, our completion goal will be to have the testing infrastructure setup (though not working)

Assumptions

1. A decision is made on where voting mechanisms go

Task Breakdown

#	Ticket	General Idea	Time Estimate - 30% variation	GH Ticket

1	Clarity Design document cleanup	There are still a few remaining comments on the general Clarity design document found here.	2 days	Not available
2	Voting document cleanup	While there are no comments in the vote mechanism doc, there are no supporting docs on the final decision for voting logic in this version of sBTC (mainly do we need an additional contract)	2 days	https://github.com/stacks-network/sbtc/issues/404
3	Import sbtc-token	There are two versions of sbtc-token (one in sbtc-mini & one in romeo). The goal here is import what we can keep & remove everything extraneous (no new features)	1 - 2 days	https://github.com/stacks-network/sbtc/issues/399
4	Import sbtc-deposit	There is a previous version of this in sbtc-mini	1 - 2 day	https://github.com/stacks-network/sbtc/issues/400
5	Import sbtc-withdraw	There is a previous version of this in sbtc-mini	1 - 2 day	https://github.com/stacks-network/sbtc/issues/401
6	Import sbtc-handoff	There is a previous version of this in sbtc-mini	1 - 2 day	https://github.com/stacks-network/sbtc/issues/402
7	Import sbtc-controller	There is a previous version of this in sbtc-mini. This is modeled off of the a lite derivative of the Executor DAO	1 day	https://github.com/stacks-network/sbtc/issues/403
8	Import latest version of PoX-3 / PoX-4	There is a draft of sbtc-db floating around in a branch, will make use of that as much as possible	1 - 2 days	
9	Update PoX-3 / PoX-4 data structures	There should be a few changes necessary to data structures (such as including a signer public key when delegating)		
10	Update PoX-3 / PoX-4 vote function			
11	Update Stacker registration function			
12	Enable Clarinet-SDK PoX testing			

After this milestone is complete the goal is to have all relevant contracts scrubbed/imported & to have a full testing environment (particularly supporting PoX-4).

Total Time Estimates	
Average	11 days
Maximum	~ 13 days
Assuming 40% Overhead for PRs, Meetings, etc.	
Average	15 days
Maximum	~ 18 days

Milestone 2 - New Clarity Storage Contracts

This milestone is about completing all Clarity contract setup by now focusing on starting new contracts / contracts previously not merged. This still does not include any additional / new updates to the imported contracts, this milestone is mainly to finish connecting all contracts in the protocol before updating to match the new logic dictated in sBTC Design - Clarity (likely everything surrounding a single UTXO consolidation or sBTC-BTC processor contracts) .

Assumptions

1. All tasks from milestone 1 are completed (by importing all

Task Breakdown

#	Ticket	General Idea	Time Estimate - 30% variation	GH Ticket
1	Create sbtc-db data structures	There is a draft of sbtc-db floating around in a branch, will make use of that as much as possible	1 day	https://github.com/stacks-network/sbtc/issues/405
2	UTXO setters & getters	The core functions for the single, consolidated-UTXO that the peg-wallet holds	1 - 2 days	https://github.com/stacks-network/sbtc/issues/406
3	Insert deposit challenge	Insert challenge logic for when a user believes their deposit is being ignored	3 - 5 days	https://github.com/stacks-network/sbtc/issues/407
4	Clear deposit challenge	Either proving that a deposit was processed (during a deposit consolidation) or the user re-claimed their BTC	3 - 5 day	
5	Insert pending	The core withdrawal logic initiated by a Stacks	1 - 2 days	

	withdrawal	user attempting to withdraw; locks up sBTC until it's either processed or re-claimed		
6	Clear pending withdrawal	The core withdrawal logic for clearing a pending withdrawal in either a successful withdrawal or by a reclaim	1 - 2 days	
7	Import stacker-db & replace terminology	While there's a draft PR that was closed, it needs	1 day	
8	Debug existing stacker-db setter	The imported stacker-db draft is incomplete in that it does not update anything. The correct behavior is to update the signer-slots-by-reward-cycle map for the current & previous signers during the prepare phase.	1 - 2 days	
9	Extend test infrastructure	Decide how we're going to test private functions that are only callable by nodes; can we setup where a node already has access to a contract private calls?	1 - 2 days	

With the end of this milestone, every sBTC Clarity contract protocol would be accounted for in setup though we have not reached code completion & testing.

Total Time Estimates	
Average	16 days
Maximum	~ 20 days
Assuming 40% Overhead for PRs, Meetings, etc.	
Average	22 days
Maximum	~ 28 days

Milestone 3 - PoX-4 & Test Setup

This milestone is about completing the contract setup & leveraging that into setting up our testing infrastructure. Without a doubt, every single time, testing with a PoX-2.5/3/etc… contract is excruciatingly difficult for a few reasons (such as requiring nodes to call private functions). Therefore, during this milestone, our goal is to finally import the latest version of PoX-3 & PoX-4 (which there are currently two versions, one in mini, & one in Jude's Nakamoto PR); afterwards,

we'll focus on standing up a testing infrastructure, as long as we're unblocked with the PoX contract, the rest of testing should sail smoothly.

Assumptions

1. All tasks from milestone 1 & 2 are completed
2. No additional contracts have been discovered during the previous milestones

Task Breakdown

#	Ticket	General Idea	Time Estimate - 30% variation	GH Ticket
1	Import latest version of PoX-3 / PoX-4	There is a draft of sbtc-db floating around in a branch, will make use of that as much as possible	1 - 2 days	
2	Update PoX-3 / PoX-4 data structures			
4	Update PoX-3 / PoX-4 vote function			
5	Update Stacker registration function			
6	Enable Clarinet-SDK PoX testing			

After this mile is complete not only will all contracts in the protocol be here but the testing infrastructure should be ready as well.

Total Time Estimates	
Average	12.5 days
Maximum	~ 16.25 days
Assuming 40% Overhead for PRs, Meetings, etc.	
Average	17 days
Maximum	~ 23 days

Milestone 4 - Update BTC Processors

This milestone is about completing the contract setup & leveraging that into setting up our testing infrastructure. Without a doubt, every single time, testing with a PoX-2.5/3/etc… contract is excruciatingly difficult for a few reasons (such as requiring nodes to call private functions). Therefore, during this milestone, our goal is to finally import the latest version of PoX-3 & PoX-4 (which there are currently two versions, one in mini, & one in Jude's Nakamoto PR); afterwards, we'll focus on standing up a testing infrastructure, as long as we're unblocked with the PoX contract, the rest of testing should sail smoothly.

Milestone 5 - Write Unit Tests

This milestone is about completing the contract setup & leveraging that into setting up our testing infrastructure. Without a doubt, every single time, testing with a PoX-2.5/3/etc… contract is excruciatingly difficult for a few reasons (such as requiring nodes to call private functions). Therefore, during this milestone, our goal is to finally import the latest version of PoX-3 & PoX-4 (which there are currently two versions, one in mini, & one in Jude's Nakamoto PR); afterwards, we'll focus on standing up a testing infrastructure, as long as we're unblocked with the PoX contract, the rest of testing should sail smoothly.