

Mars Protocol Code Review Breakdown — Session #3 & #4, Safety Fund and Governance Modules

AUGUST 3, 2022



In the first two code review sessions, Mars contributor @larry0x went over the Airdrop and Vesting contracts. In these tutorials, the focus was on CosmWasm smart contract development. The third and fourth session will address the Cosmos SDK and the app-chain itself. More specifically, these sessions cover the “Safety Fund” and “Governance” modules.

This article will serve as a brief introduction to both modules. For a more detailed analysis of the code, check out the recordings:

- [Session #3 Recording](#) (Safety Fund)
- [Session #4 Recording](#) (Governance)

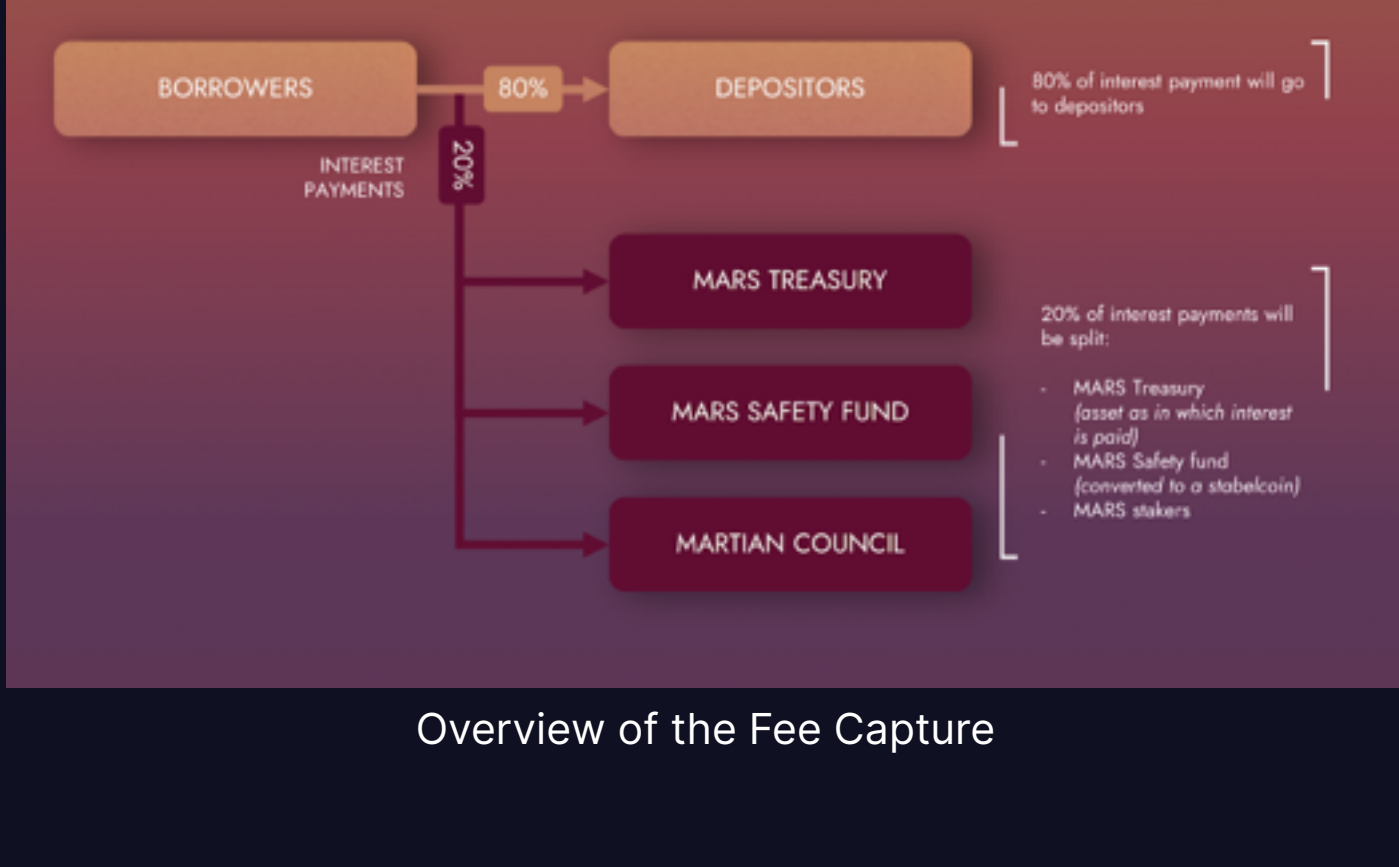
Introduction to the Safety Fund Module

The Cosmos SDK is made up of composable modules that together can be used to create application-specific blockchains. For example, the auth module stores accounts, the bank module stores balances, and the CosmWasm module enables smart contract development. Anyone can build and add new modules into the SDK, and integrating an existing module into your blockchain is as simple as importing it in.

Under the [Mars Hub github repo](#), there is an “x” folder. This is where custom modules for Mars Hub are hosted. So far, you should see: “gov”, “incentives”, and “safetyfund”.

The Safety Fund Module is the simplest module in the Mars Hub application which makes it a great place to start. Currently, the safety fund is simply a module account holding funds, which can be spent upon a successful governance proposal. That’s it. Here’s the idea behind it:

When a user “borrows” assets from a Mars Outpost, they pay an “interest” fee. 80% of those fees go to Depositors. The other 20% gets split between the Treasury, the Safety Fund, and MARS stakers. For the Safety Fund, fees are converted into a stablecoin.



Overview of the Fee Capture

In the case of a Shortfall Event, which occurs whenever the value of a borrower’s debt exceeds the value of his collateral (resulting in a deficit for lenders known as “bad debt”), governance can deploy the safety fund to protect lenders. This is distinct from an Illiquidity Event, where utilisation rates are at 100% and lenders are unable to withdraw. In the latter case, LPs are illiquid but solvent, whereas in a Shortfall Event the LPs are actually insolvent.

Shortfall Events can be caused by various risks such as smart contract exploits, untimely liquidations and/or oracle attacks. To be clear, they should never happen under normal conditions and can be mitigated by good risk management. To date, Shortfall Events in credit protocols have been limited, and, when they have occurred, resulted from exploits/economic attacks. However, when Shortfall Events do occur and are attributable to a governance failure of the Martian Council, the Martian Council should be able to respond by tapping funds available to compensate the injured users. Ultimately, compensation cannot be guaranteed and is subject to the discretion of the Martian Council.

In the long term, the module could be updated (upon governance approval) to automatically detect bad debt incurred in an outpost, and distribute appropriate amounts of funds to cover the shortfall without having to go through governance.

If you’re ready to dive into the code, check out the [recording](#).

Introduction to the Governance Module

The “customgov” module is a wrapper around Cosmos SDK’s vanilla gov module, inheriting most of its functionalities, but replacing the vote tallying logic with a custom implementation. Namely, tokens locked in the [lockup/vesting contract](#) count towards governance voting power.

These tokens are either “locked” (non-transferable), “unvested” (not yet fully earned), or both.

NOTE: Mars builders “vest” in (earn) MARS when they have contributed to the Mars protocol over a certain period of time. MARS may “unlock” (become transferable) later than they become vested, depending on when the particular person started working on Mars. The unlocking schedule for MARS has not yet been finalized and will be announced before launch.

In the first code review session, @larry0x went over the lockup/vesting contract, which performs token vesting and unlocking for Mars builders. Tokens locked in the vesting contract shouldn’t be eligible for staking rewards (actual users should be staking MARS and earning those rewards), but builders should be able to vote. The question is how to implement this in a custom module for the Mars Hub blockchain?

For the most part, the vanilla gov module that comes with the Cosmos SDK works for these purposes. But the vote tallying logic that the module uses needs to be modified so that tokens locked in the lockup/vesting contract get counted. In the original module, voting power only comes from one source: the amount of staked tokens in the staking module. For the custom module, there are now two sources: the staked tokens (as before), plus the amount of tokens in the lockup/vesting contract.

For staked tokens in the staking contract, here’s how voting power works: If I stake with a validator, I am delegating my governance voting power to that validator. In the absence of my vote on a specific proposal, the validator I delegate to can vote on my behalf. That is, if the validator votes on a proposal that I don’t, and I have delegated my staked tokens to the validator, my voting power will count towards the option that the validator has voted for. In the case that I choose to not only vote, but my vote is different from my validators vote, my vote will override the validator’s . This is the original implementation in the Cosmos SDK and remains the same for the Mars Hub chain.

For the tokens in the lockup/vesting contract, the delegation logic is new. When a typical user stakes their (unlocked, fully vested) tokens with a validator, they delegate their voting power to the validator. In the case of tokens in the lockup/vesting contract, no delegation occurs. This means if a builder doesn’t vote on a proposal, their voting corresponding to the tokens locked in the lockup/vesting contract will simply not be registered at all. Here’s an example:

Consider a blockchain with only one validator, as well as two users, Alice and Bob, who have the following amounts of tokens:

user	staked	vesting
Alice	30	21
Bob	49	0

Assume the validator votes YES on a proposal, while neither Alice or Bob votes. In this case, the validator will vote on behalf of Alice and Bob’s staked tokens. The vote will pass with $30 + 49 = 79$ tokens voting YES and the rest of the 21 vested tokens not voting.

If Alice votes NO, this overrides the validator’s voting. Both Alice’s staked and vesting tokens will count towards the NO option. The proposal will be defeated by 49 tokens voting YES vs 51 tokens voting NO.

This is how the module is intended to work. The last introductory topic to cover is the “wrapper” component of the module. The Cosmos SDK repo could be forked to include the modified governance contract, but this would result in having to change various dependencies to execute (and especially test) the new module. Instead, a “wrapper” module allows the new module to inherit everything from the original module while preserving the ability to override specific methods. This works similar to inheritance in object-oriented programming. The “wrapper” module just changes the tallying logic to account for vesting tokens using the delegation logic explained above.

If you’re ready to dive into the code, check out the [recording](#).

What’s Next

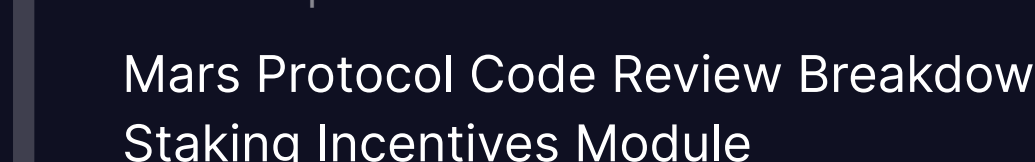
The next and fifth session will cover the staking incentives module. Listen, ask questions & help spot bugs in this live video transmission. Event details:

- August 06, 2022
- 11 AM EST | 3 PM UTC
- <http://Discord.gg/marsprotocol>

Please be advised all of the above reflects Mission Control’s current thinking, but is not guaranteed or promised. No contract is implied or duties assumed hereby. Do not make any financial decisions based on this announcement.

Follow [Mars on Twitter](#) and subscribe to [Mars’ email newsletter](#) for the latest updates from Mission Control.

Disclaimer: This article does not constitute investment advice and is subject to and qualified in its entirety by the Mars disclaimers [here](#).



Previous post
Mars Protocol Code Review Breakdown — Session #5, Staking Incentives Module

Next post
Mars Protocol Code Review Breakdown — Session #2 Airdrop Contract in Action & Overview of the Vesting Contract