Esteemed Super Validators:

During the MainNet Staging rehearsal last week, while going through the acceptance criteria, the Super Validator operators, and rights owners, identified three failure cases that caused the software version and configuration proposed for MainNet to be rejected.

The Super Validator operators have proposed fixes for these issues.

The software version containing these is available now to all the Super Validator operators, and the operators have proposed a new MainNet launch date, June 11th, that would allow these fixes to be included in MainNet. The proposed changes have not yet taken effect on-chain.

Because these fixes modify the configuration of Canton Coin, and the configuration of decentralized processing on the Global Synchronizer, I'm proposing that we use the CIP process to vote to accept these changes, and to instruct our respective Super Validator operators to vote them into effect on chain.

Please find the CIP attached.

If possible please respond within 24 hours. We will consider the CIP vote to be closed in 48 hours.

For the record, Digital Asset votes in favor of CIP-0012

Thank You

Eric

CIP-0012

Minor adjustments to Canton Coin processing, and Operational configuration, on the Global Synchronizer

Background

During the MainNet Launch Staging rehearsal process we identified three minor issues or inconsistencies that we'd like to address. Because the resolution of these issues involves changes to Daml model code and changes to the on-chain configuration of the Daml models, we'd like to confirm the support of the Super Validator rights owners for these changes.

Because these three changes impact two different sets of Daml models, we will implement them on chain as two separate governance votes. We need to complete these votes this week so the changes can take effect on DevNet before being added to TestNet on Tuesday May 28th, 2024.

On chain Vote 1: Resolve holding fee error and Round stall fault

Vote 1, Part 1: . Correct a rounding error in holding fees.

Background

The setting used for calculating holding fees is a configurable value. This value is assigned to the variable `transferConfig.holdingFee.rate`.

The default value for this variable is actually set by a separate piece of code that computes the rate and populates this field. This is bootstrapping convenience: instead of requiring SV node operators to configure a value for every variable in all the Daml models, the default values are computed and entered by a bootstrapping program.
That bootstrapping program incorrectly populated the default value for the holding fee per round using floating point math, instead of the fixed decimal math used in all the Daml models, and using 360 days/year instead of 365 days/year.

Updating the value through an on chain "App Configuration Schedule" vote overrides the default value that was generated by the bootstrapping code.

The on chain vote proposal will insert the corrected value into all Daml models that rely on it.

Action:

Using the “App Configuration Schedule” vote proposal form, propose a vote to change the value of `transferConfig.holdingFee.rate` from

```
0.0000192901 USD
```

To

```
0.0000190259 USD
```

Vote 1, Part 2: Fix for stalled round processing

Background:

We identified a bug in the Daml models that stopped a given round from moving to its minting phase if the full allotment of Canton Coin is made available for minting in a given round. This happened because the code assumed there would always be some unminted Canton Coin in each round. While other rounds still advance independently of the stalled round, the Canton Coin Scan aggregate data, and the Canton Coin Scan UI only update if every round is fully closed. As a result, Scan data processing stalls, and the Scan UI stops displaying new data.

The fix corrects this assumption and allows rounds to proceed when all available Canton Coin is minted.

The fix has been completed by the Canton Network engineering team at Digital Asset, and made available to all Super Validators and Validators. However the fix will only take effect after this vote proposal is accepted by the Super Validator operators, and takes effect on chain.

Action:

Using the “App Configuration Schedule” form, propose a vote to change the effective version of these Daml model packages:

```
packageConfig.amulet
packageConfig.amuletNameService
packageConfig.dsoGovernance
packageConfig.wallet
```

`packageConfig.walletPayments`

from

0.1.0

to

0.1.1

Vote Proposal Details

The operators will implement a single vote proposal including both Part 1 and Part 2, above.
The proposal should expire within 24 hours of of being submitted to the SV operators
The proposal should take effect one hour after vote expiration.

On Chain Vote 2: Resolve stalled SV action confirmations

Background

All actions that impact the operation of the entire Global Synchronizer service take effect by a ⅔ vote of the Super Validators. This includes many automated actions. For these automated actions, the Super Validator app of each node creates a confirmation contract indicating that it endorses the action. Once enough confirmations have been produced, the active Super Validator delegate node automatically initiates the proposed action on chain.

These confirmation contracts have an expiration date. The default value for this expiration has been set to five (5) minutes. If a Super Validator node is not available during this five minute window, the confirmations of some nodes may expire before the action reaches the threshold of required confirmations. To be more robust against these cases, this proposal willi increase the expiration time on those contracts to a full hour.

Action

Using the “Set DsoRules configuration “ vote proposal form,

Change
`actionConfirmationTimeout.microseconds`

From

```
300000000

(300,000,000 microseconds)
```

To

```
3600000000
(3,600,000,000 microseconds)
```

Vote proposal details

The vote proposal should expire and take effect within 24 hours of being submitted to the SV operators.