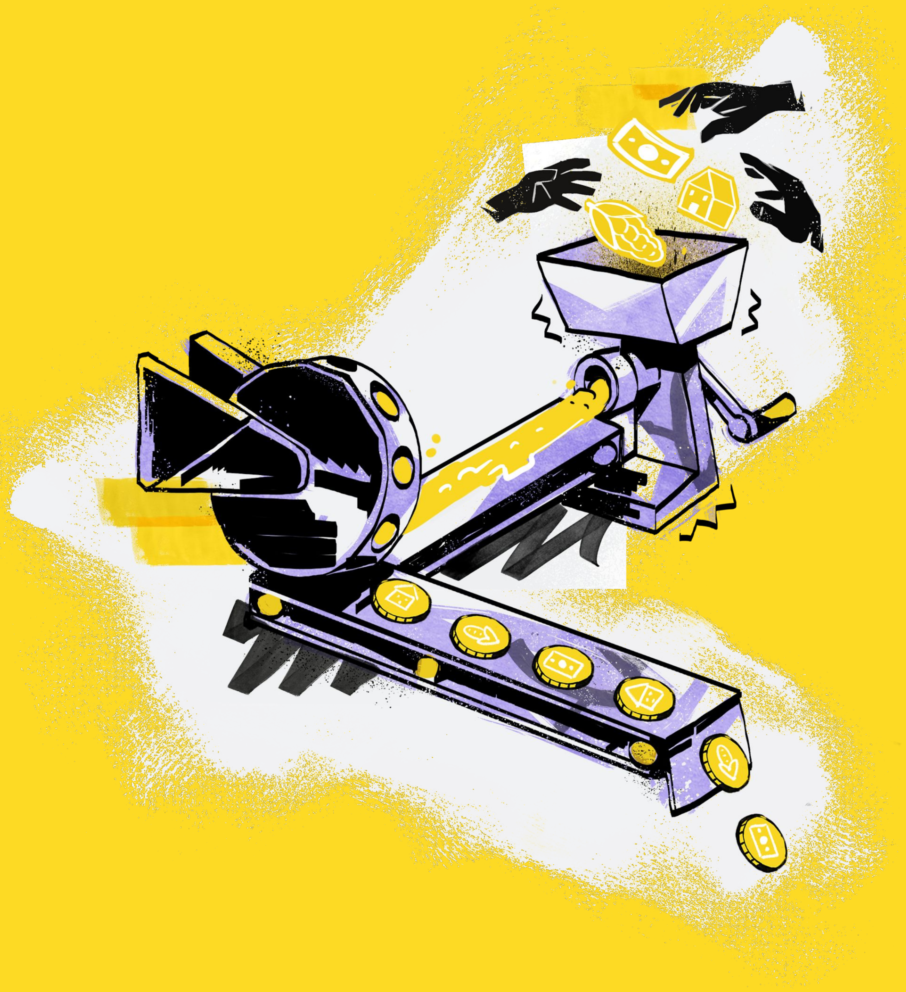
Stellar

Guidebook

# ASSET TOKENIZATION ON STELLAR



June, 2023

# EXECUTIVE SUMMARY

This document highlights how assets are issued on the Stellar blockchain network. Stellar is built for payments and the movement of value. It is designed to allow for the tokenization of assets, such as fiat currencies and securities, in a secure and easy way, among other things.

Asset Tokenization on Stellar entails four basic steps: (1) create an issuing account, (2) create a distribution account, (3) add a trustline for the asset to the distribution account, and (4) transfer the asset from the issuing account to the distribution account. It's that simple.

The ease at which one can issue assets is complemented by Stellar's built-in programmatic functions for asset control. These controls include the ability to limit how the asset is used and who can hold it by setting different configuration flags.

Stellar also has important account management features and built-in compliance capabilities. Stellar supports multi-signature signing schemes and omnibus accounts. It can also facilitate the execution of specific compliance obligations.

Stellar is a powerful, flexible asset issuance platform. Asset issuers have a number of customizable features depending on issuer preferences and the asset's use case.

This document complements Stellar's Asset Sandbox, which is available at  Users of the Stellar Asset Sandbox can explore asset issuance on the Stellar network. https://stellar.cheesecakelabs.com Users of the Stellar Asset Sandbox can explore asset issuance on the Stellar network.

# 1. Understanding Stellar Assets

***Stellar is an open-source blockchain network optimized for the issuance of digital assets. It allows entities to create digital representations of assets — from fiat currencies to securities — and to control how those assets can be used.***

## Asset Tokenization

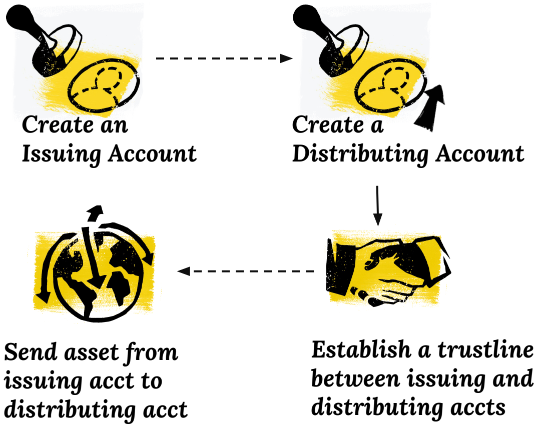
There is no dedicated “mint” or “issue” operation on Stellar. Assets are typically created on Stellar by using a payment operation. In Stellar, an asset is issued when an asset is transferred from an issuing account to a distribution account.[1]

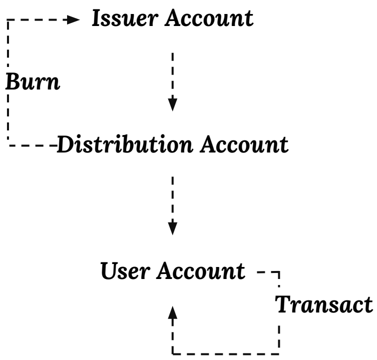
Asset tokenization is a straightforward process that involves four basic steps: (1) create an issuing account, (2) create a distribution account, (3) establish a trustline for the asset in the distribution account, and (4) transfer the asset from the issuing account to the distribution account.

Issuing and distribution accounts are standard Stellar accounts and are represented using a public key that can be shared with anyone and a private key used to sign transactions affecting the account. The issuing account is the origin of the asset and forever linked to the asset’s identity. The distribution account is the first recipient of the asset.

An asset issuer creates an asset by sending a “payment” from the issuing account to the distribution account. For non-payment instruments, a transfer of the asset is made from the issuing to the distribution account. To remove the asset from circulating supply, the opposite happens. The asset is transferred back to the issuing account.

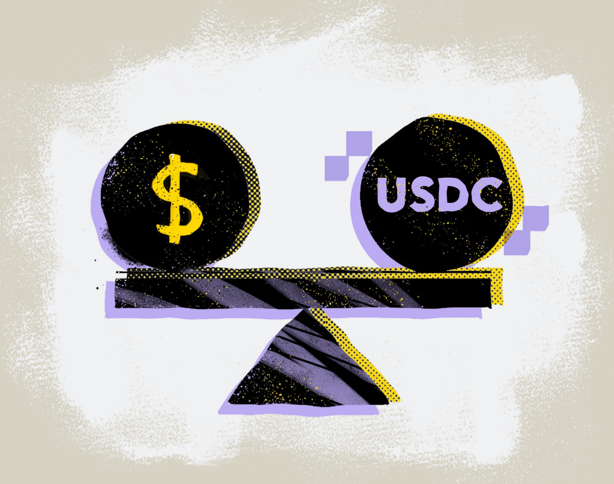
Before an account can hold a specific asset, the account holder must explicitly opt-in to hold a particular token, which is done by adding a trustline. A trustline is an explicit approval by an account owner to to hold and trade an asset. Only after the trustline is in place can an asset be sent to and from that account. The trustline specifies the asset code, the issuer, and amount of the asset that can be held in the account.


Create an Issuing Account
Create a Distributing Account
Establish a trustline between issuing and distributing accts
Send asset from issuing acct to distributing acct



[1] Other ways to issue an asset include selling the asset on Stellar’s decentralized exchange, sending the asset as a claimable balance, or depositing the asset into a liquidity pool. For the purpose of this document, we’ll focus on the use of the payment operation.

# WHAT IS A STABLECOIN?



The term “stablecoin” has been used to label different types of assets across the blockchain industry. Currently, there are debates in a number of jurisdictions regarding the definition of a stablecoin. For the purposes of this paper, we define a stablecoin as a digital asset issued on a blockchain whose value is backed fully by highly liquid assets (such as cash and short-term government bonds) in the currency of denomination.

When an entity issues a stablecoin, it must set up an off-chain reserve that holds the asset backing the stablecoin. When a stablecoin holder cashes out or redeems their stablecoin, they receive an equivalent amount of the stablecoin’s underlying asset from the stablecoin issuer when converted.

There are a number of questions to ask when analyzing stablecoins. We focus on four important ones: (1) how is the stablecoin collateralized, (2) how transparent are the reserve arrangements, (3) what are the local regulatory and licensing requirements for issuing the stablecoin in a jurisdiction, and (4) the use cases — for what purposes the stablecoin will be used.

Stablecoins are generally collateralized — with the underlying cash or with underlying cash and cash-equivalents like short-term government bonds. Attestations and audits should be performed regularly on the stablecoins’s reserves and should be public.

Why do these questions matter? The asset holder should know that the issuer can exchange the token for fiat at any point in time. Highly liquid assets give greater assurance that the issuer can do that.

While these are important considerations, asset issuers must pay close attention to the evolving discussions and lawmaking in various jurisdictions to ensure that they comply with applicable laws with respect to stablecoins.

# 2. Asset Design Considerations

The first step to issuing a Stellar asset is determining its use case(s), as this will guide its design. Stellar enables issuers to establish their own requirements and controls for their assets. Below are three design considerations for issuers: asset control, asset nomenclature, and asset information.

## Asset Control Features

There are a number of account-level controls or flags that can be applied to an asset when the asset is created. The flags are account-level controls that are applied to the asset by the issuer at the time of issuance.

### Limiting access to an asset

Flag Name:
AUTHORIZATION_REQUIRED

This flag allows an issuer to approve an account (specifically, an account's trustline) to hold its asset. This setting allows an issuer to prequalify account holders by requiring the collection of information necessary to run appropriate compliance checks, such as KYC ("know your customer"), AML ("antimoney laundering"), and CFT ("countering the financing of terrorism") before allowing the account holder to receive the asset.2 Trustlines created prior to enabling this flag will still be authorized. AUTHORIZATION_REQUIRED

### Locking the asset's configuration flags

Flag Name:
AUTHORIZATION_IMMUTABLE

This flag prevents an asset's configuration from being changed. It is used to signal to current and potential asset holders that the asset's configuration flags can never change.

### Revoking access to an asset

Flag Name:
AUTHORIZATION_REVOCABLE

This flag allows an issuer to withdraw the authorization level of an account. In cases where an account no longer meets the approved criteria of the issuer, the account holder's access to the asset can be revoked, effectively freezing the assets in the account. This flag takes effect immediately and can be used to revoke access from previously authorized trustlines.

### Clawing back an asset

Flag Name:
AUTHORIZATION_CLAWBACK_ENABLED

This flag allows an issuer to burn its assets in any account. This functionality could be useful in cases of fraud or lost account keys or simply to comply with local regulations. When enabled, the clawback flag will affect only the subsequently created trustlines. The issuer can not clawback funds from trustlines that were established prior to the flag being set.[3]

[2] An open loop asset is one that does not have an AUTHORIZATION_REQUIRED flag set. USD Coin, issued by Centre, is an example of such an asset.
[3] If an issuer seeks to transfer clawed assets from one account to another, the issuer will need to burn the original assets and issue new ones.

### Off-chain Authorization Server (SEP-8)

At times, it may be useful to require an issuer’s (or a delegated third party's) approval on a per-transaction basis (for example, where monitoring issued assets to enhance compliance). Stellar Ecosystem Proposal (SEP) 8 provides a standard interface between wallets and issuers to negotiate approvals.

SEP-8 or the authorization server feature uses trustlines to enforce asset control. When an end user is performing a transaction and a certain criteria is met, the authorization server acts as a signer and creates a temporary trustline for the movement of funds for that specific transaction. Once the transaction is complete, the trustline is revoked.

## The workflow for SEP-8 implementation is below:

01 Account creates and signs a transaction.

02 Account resolves asset information and detects that it’s an asset requiring authorization by checking the authorization flags of the asset issuer’s account

03 Account finds the approval server via the issuer’s SEP-1 stellar.toml, and sends the transaction to the approval Account finds the approval server via the issuer’s SEP-1 stellar.toml, and sends the transaction to the approval serverserver

04 The approval server determines whether the transaction should be approved with the below responses:.

**Success:** transaction has been approved and signed by the issuer. (Account should display a success message.)

**Revised:** transaction has been revised to meet compliance requirements and signed by the issuer. (When a transaction is revised, an approval service could respond with a different transaction. It is important that the account shows the changes to the user and asks them to sign the new transaction.)

**Pending:** The issuer can not determine approval at the moment. (Account can resend the same transaction to the approval server later.)

**Action Required:** Transaction re.uires a user action to be completed. (Account will present the re.uired action to the user, and an option to resubmit once the action has been taken)

**Rejected:** transaction has been rejected by the issuer. (Account should display an associated error message.)

It is important to note that the off-chain authorization server involves operational overhead and user complexity because every transaction must be approved externally before it can take place on the Stellar network.

## Asset Nomenclature

One of the first things that an issuer must do is create an identifying code to provide a unique name for its asset. Currently, there are two naming formats:

01 Alphanumeric 4-character maximum: Any characters from the set [a-z][A-Z][0-9] are allowed. The code can be shorter than 4 characters, but the trailing characters must all be empty.

02 Alphanumeric 12-character maximum: Any characters from the set [a-z][A-Z][0-9] are allowed. The code can be any number of characters from 5 to 12, but the trailing characters must all be empty.

It is worth noting that the asset name would ideally allow potential asset holders to understand quickly what the asset represents. The asset will be made known to the world via a TOML file.

## Published Asset Information - TOML File

An issuer providing clear information on what its asset represents empowers users with information they rely on in deciding whether to interact with a particular asset.

On Stellar, this information is provided by linking the issuing account to a domain owned by the issuer, publishing a Stellar TOML file on that domain, and making sure that file is complete and accurate. By linking the issuing account to the issuer's domain, applications and users can find out more information about the issuer and its assets.

The TOML file allows an issuer to give exchanges, wallets, potential asset holders, and the public more information about itself and its assets. At a minimum, a TOML file should include: General Information, Organization Documentation, Point of Contact Documentation, and Currency Documentation. linking the issuing account to a domain

## CENTRE'S TOML FILE FOR USD COIN (USDC)

https://centre.io/.well-known/stellar.toml

```
ACCOUNTS=
["GA5ZSEJYB37JRC5AVCIA5MOP4RHTM335X2KGX3IHOJAPP5RE34K4KZVN",
"GDEWOLMOPAVRTGNJVWOE6U6LHZVAWIJZVWM6PDLCFTUTJJEKSU32TO5W"]

 [DOCUMENTATION]

ORG_NAME="Centre Consortium LLC" ORG_DBA="Centre Consortium"
ORG_URL="https://www.centre.io"
ORG_LOGO="https://www.centre.io/images/logo-icon.png"
ORG_PHYSICAL_ADDRESS="San Francisco, CA"
ORG_OFFICIAL_EMAIL="usdc@centre.io" ORG_GITHUB="centrehq"
ORG_TWITTER="centre_io" ORG_DESCRIPTION="Centre Consortium is a joint
venture aimed at establishing a standard for fiat on the internet and providing a
governance framework and network for the global, mainstream adoption of fiat
stablecoins founded by Circle and Coinbase. USD Coin (USDC) is the first fiat
stablecoin implementation from Centre."

[[PRINCIPALS]]
name="David Puth" email="usdc@centre.io"

[[CURRENCIES]]
 code="USDC"
issuer="GA5ZSEJYB37JRC5AVCIA5MOP4RHTM335X2KGX3IHOJAPP5RE34K4KZ
VN" is_asset_anchored=true anchor_asset_type="fiat" anchor_asset="USD"
attestation_of_reserve="https://www.centre.io/usdc-transparency"
redemption_instructions="Redeemable through a Circle account at
https://circle.com" name="USD Coin" desc="USDC is a fully collateralized US
Dollar stablecoin, based on the open source fiat stablecoin framework developed
by
Centre."image="https://www.centre.io/images/usdc/usdcicon-86074d9d49.png"
```

# 3. ACCOUNTS AND KEY MANAGEMENT

***Since multiple accounts are involved in asset tokenization, it's important to understand the standard makeup of a Stellar account in addition to account management best practices.***

## Stellar Accounts

Like other blockchains, Stellar accounts use public key-private key cryptography. The public key for Stellar accounts is a 56-alphanumeric sequence that starts with "G." When a person initiates a transaction, their public key is broadcast as the "source" for that operation, and their public key is the address to which others can send an asset to them. The public key is safe to share with a counterparty.

### Multisignature Schemes

Stellar accounts can be managed using a single Stellar public and private key pair (by default, the key pair of the account); but for added security and controls, an issuer can explore multi-signature schemes, where multiple public and private key pairs must be used before a transaction can take place. In multi-signature schemes, each private key holder is referred to as a "signer" and a signer's use of their key is referred to as "approving" a transaction.

Configuring multi-signature schemes is done using the **Set Options operations**. Using this operation, issuers can add additional signers and set their signing **threshold** to control what actions can take place. Each signer has a weight, and the cumulative weight of the signers of a particular transaction must be equal to or greater than one of the account's thresholds (low, medium, and or high) in order for the transaction to be valid.

Say that an issuer sets the account medium threshold to 4 for a payment operation. This means that the signature weights of the account signers need to be 4 or greater in order for the operation to run. If the weight of the signers is less than 4, the operation does not run. By using multiple keys and threshold weights, issuers and asset holders can distribute trust and responsibility for approving transactions for an account between different users and systems, which gives them the flexibility to gate those signatures with their internal processes and controls.

See **Signatures & Multisig** for more information.

# SUPPORTING CUSTODIAL ACCOUNTS IN OMNIBUS ACCOUNTS

Issuers may benefit from understanding two features supporting omnibus custodial account management in Stellar. Custodians using an omnibus accounting system can use memos and muxed accounts to facilitate account management.

**Memos,** can be used to differentiate "individual" accounts in a pooled or omnibus account. Memos are an optional, unstructured data field that can be used to embed any additional identifying information about the transaction relevant to the sender or receiver. For example, a custodian can use memos to route inbound payments to its users. If the custodian uses an omnibus account, G**A7QYNF7SOWQ3GLR2BGMZEHXAVIRZA4KVWLTJJFC7MGXUA74P7UJ**VSGZ, to receive all inbound payments, they can use the memo field to identify which sub account to credit the payment.

**Muxed Accounts** are a way to do the same thing without requiring users to enter a memo manually, reducing the requirements and complexities associated with transactions in custodial accounts. Muxed accounts are created by merging an integer ID, a number associated with each custodial account holder, with the omnibus Stellar account. Muxed accounts (M-accounts) share many characteristics with the underlying address, but they start with an M rather than a G, and they are 69 characters long rather than 56. For example, using the Stellar account mentioned previously:

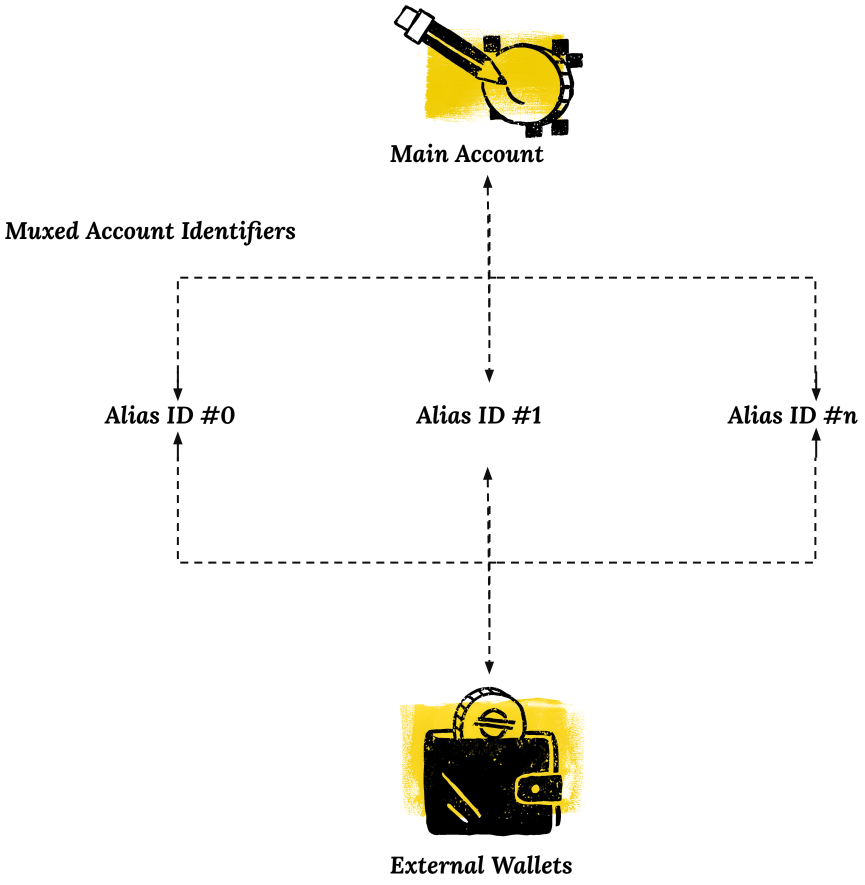
G**A7QYNF7SOWQ3GLR2BGMZEHXAVIRZA4KVWLTJJFC7MGXUA74P7UJ**VSGZ

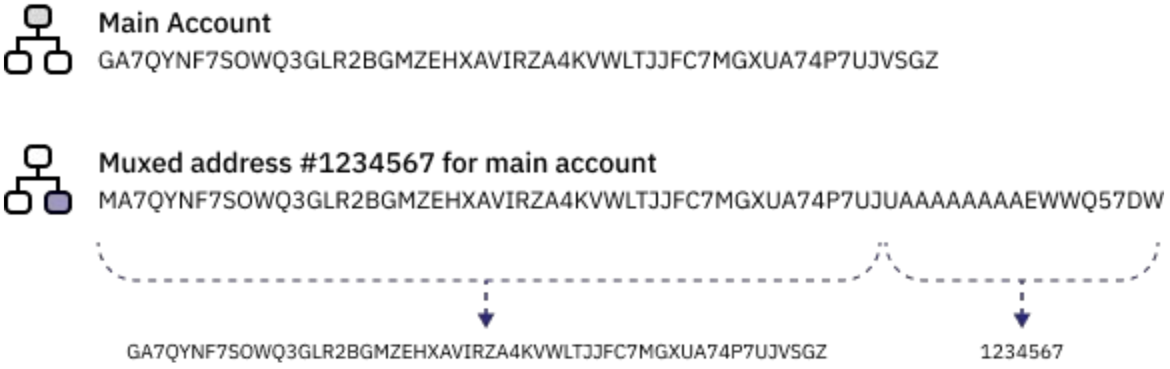
Its associated M-accounts would look like the address below. Please take note that a significant portion of the G-account is embedded in the M-account identifier.

M**A7QYNF7SOWQ3GLR2BGMZEHXAVIRZA4KVWLTJJFC7MGXUA74P7UJ**UAAAAAAAAAAAACJUQ has the ID 0, while

M**A7QYNF7SOWQ3GLR2BGMZEHXAVIRZA4KVWLTJJFC7MGXUA74P7UJ**UAAAAAAAAEWWQ57DW has the ID 1234567.

While using Muxed accounts to disambiguate a pooled account's users is considered a better practice than using memos, the ecosystem is still in the process of adding support for Muxed accounts — many centralized exchanges, for example, don't yet support them — so in the short term, it makes sense to support both.





Main Account
GA7QYNF7SOWQ3GLR2BGMZEHXAVIRZA4KVWLTJJFC7MGXUA74P7UJVSGZ

Muxed address #1234567 for main account
MA7QYNF7SOWQ3GLR2BGMZEHXAVIRZA4KVWLTJJFC7MGXUA74P7UJUAAAAAAAAEWWQ57DW

GA7QYNF7SOWQ3GLR2BGMZEHXAVIRZA4KVWLTJJFC7MGXUA74P7UJVSGZ

1234567

# 4. MANAGING RESERVES AND FEES ON STELLAR

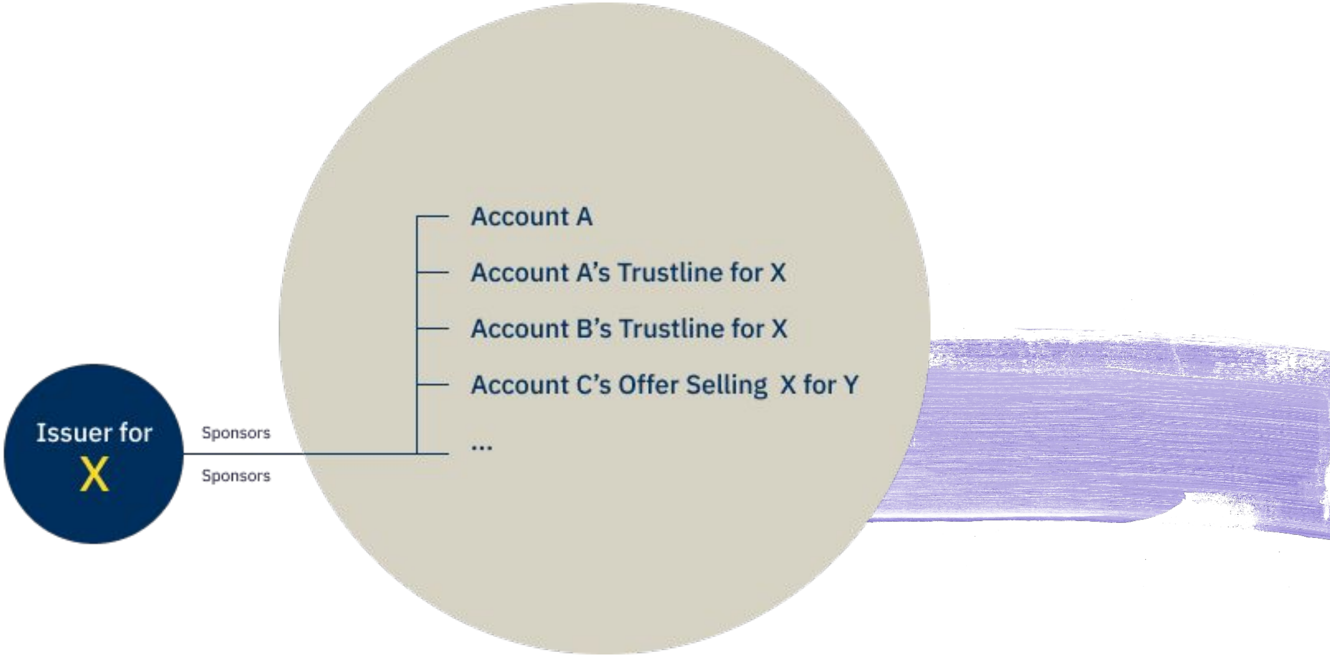
***In order for Stellar to operate efficiently, the network charges a fee for operations on the blockchain. Stellar requires a base reserve of its native currency–lumens (XLM)–for every account as well as a transaction fee for the movement of value. The reserves and transaction fees are intended to prevent spam on the network and to prioritize transactions.***

### Base Reserves

Stellar accounts must maintain a minimum balance to exist, which is calculated in a unit called “base reserves.” The current base reserve on Stellar is 0.5 XLM. And the minimum balance for every account balance is 2 base reserves (or 1 XLM). To transact with any non-XLM asset, like a stablecoin, a trustline must be established in the account. Each additional trustline to an account requires an additional base reserve until the trustline is removed. Learn more about base reserves

### Sponsored Reserves

Issuers may want to avoid the inconvenience of asset holders having to manage XLM base reserves. Issuers may want to consider sponsored reserves where a "sponsoring account" (typically owned by the issuer or a wallet provider) can hold the XLM reserves required for another “sponsored” account. With the proper design, an issuer can minimize an asset holder’s need to separately acquire XLM reserves to hold and transact in non-XLM Stellar assets.


Issuer for X
Sponsors
Sponsors
Account A
Account A’s Trustline for X
Account B’s Trustline for X
Account C’s Offer Selling X for Y
...

### Fee-Bump Transactions

Just like sponsored reserves, there is a mechanism for a "sponsoring account" to cover fees for other accounts making transactions, such as the accounts using an issuer's asset or wallet application. This feature is known as a **fee-bump transaction.**

The current base fee on a Stellar transaction is 100 stroops (0.00001 XLM) per operation. Stellar transactions usually contain one operation, but can contain up to 100. When network activity is below capacity, the cost of the transaction is the base fee. When the network is at or above capacity, Stellar uses **surge pricing**, raising the minimum fee based on demand for ledger space. The base fee and all surge pricing can be covered by the fee-bump sponsoring account.

### Covering Fees for Users

Using sponsored accounts gives issuers the ability to cover all network costs for their users. This approach can drastically improve the user experience as users do not need to manage an XLM balance

## 5. INFRASTRUCTURE CONSIDERATIONS

***Issuers may also want to consider whether it would be advantageous for them to maintain their own Stellar infrastructure for operational efficiency and resiliency.***

### Horizon

Horizon is an API that enables users to query data from and submit transactions to the network. This API serves as the gateway for apps to interact with the Stellar network. Issuers and other network participants such as wallet providers and exchanges use Horizon to submit transactions, query account balances, and read any other blockchain data. Users have the option of running their own Horizon instance, or utilizing available public instances.

In deciding whether to implement an instance of Horizon, users should consider the pros and cons of having a private instance. For users relying on third parties, there is no guarantee that whoever is hosting the Horizon instance will continue to do so in the future. Additionally, running one's own instance of Horizon allows the issuer to bypass rate limiting that come with public instances for guaranteed network access and gives the issuer full operational control on its own infrastructure.

### Validators

Submitting transactions to the Stellar network or querying network data only requires a user to have a Stellar account, which at its essence is a public-private key pair. In order to participate in the process of ratifying the transactions that account holders submit to the network, an entity must runa validator, which is a node that participates in the Stellar Consensus Protocol (SCP)–the consensus mechanism that powers the Stellar network.

Validators engage in SCP with other validators to reach consensus on the validity of each submitted transaction and, upon reaching consensus, add those transactions to the Stellar ledger. There are many benefits to being a validator, including being the authoritative source of truth governing who does or does not own their asset. Also known as Issuer Enforced Finality, there is never any ambiguity as to who owns an asset, because everyone knows which copy of the ledger the issuer will consult.[4]

## 6. TREASURY MANAGEMENT

### Reconciliation

An issuer of an asset-backed token will hold deposits or reserves of the underlying asset for which the token can be redeemed. A reconciliation process allows an issuer to work across two or more networks (for example, Stellar and fiat rails) to ensure that both the token and underlying reserve counts are in sync. Reconciliation is bringing an issuer's view of two networks into sync based on what it believes their states should be.

### Checking Reserves

When new tokens are created, the issuer ensures that its reserves equal (or exceed) the tokens in circulation after the issuance. In the reverse transaction where tokens are redeemed for fiat, the process is reversed, confirming that the issuer's reserve holdings match or exceed the number of tokens. This ensures that the issuer is not withdrawing funds that are reserved to back other issued tokens. It is common to have an external third party conduct regular audits of an issuer's reserves in order to ensure trust in the token as well as to comply with local regulations.

## 7. TESTNET FOR ASSET TOKENIZATIONS

***In order to support potential asset issuers on Stellar, Cheesecake Labs has developed an asset tokenization sandbox called the Stellar Asset Sandbox to allow prospective issuers to issue test assets on Stellar testnet.***

***Stellar testnet simplifies the implementation of an institutional asset by removing real-world constraints Below are some attributes of testnet to consider in deciding whether and when to see it:***

[4] Validators that run at least three nodes that are the "center" of the network are considered "Tier 1" validators. In this context, "center" is defined as validating a significant portion of the network by being in a number of quorum sets. The Tier 1 validators help to decentralize decision-making and add fault tolerance (meaning that the network can sustain more validators going down before halting).

## What is the Stellar testnet good for

- Creating test accounts (with finding sing Friendbot, XLM is provided to allow for transactions at no cost; sing Mainnet XLM has real-world costs).
- Developing applications and exploring tutorials on Stellar without the potential of losing any valuable assets.
- Testing existing applications against new releases or release candidates of SCP and Horizon. Performing data analysis on a smaller, non-trivial data set compared to the public network.
- Performing data analysis on a smaller, non-trivial data set compared to the public network.

## What is the Stellar Mainnet good for

- Utilizing real-world assets, including money.
- Developing applications and exploring tutorials on Stellar without the potential of losing any valuable **assets.**
- **High availability test infrastructure;** SDF makes no guarantees about the availability of the testnetp
- **Long term storage of data on the network –** the network is ethereal, and resets periodically.
- The ability to control the data reset frequency.

Stellar

# LEARN MORE

stellar.org