

withdraw-funds-from-treasury)

- [Gelato Functions replacing Automate](#)
- [When should I migrate?](#)
- [Legacy Automate: Key Changes](#)
- [Cost changes](#)
- [Task Expiries](#)
- [Naming Updates](#)
- [How do I migrate a task from Legacy Automate to Functions?](#)
- [Advanced user migration](#)

Legacy Automate Migration Guide

Withdraw Funds from Treasury

For users looking to withdraw their remaining funds from the treasury during the migration from Legacy Automate to Gelato Functions, please follow the steps below. This is critical to ensure you can access your client funds across various networks before completing the migration:

- Check Tokens Available:
- Use the function:
...
• Copy
- `functiongetCreditTokensByUser(address_user)externalviewreturns(address[]memory)`
...
• This function will display all the tokens available in your account.
- Check Token Balance:
- Verify the token balance using:
...
• Copy
- `functionuserTokenBalance(address_user,address_token)externalviewreturns(uint256)`
...
• This will show the amount of each token you hold.
- Withdraw Funds:
- To withdraw your funds, call the following method:
...
• Copy
- `functionwithdrawFunds(addresspayable_receiver,address_token,uint256_amount)`
...
• Replace `_receiver`
- with your wallet address, `_token`
- with the token you wish to withdraw, and `_amount`
- with the desired amount to be transferred.

Treasury Addresses by Chain

ID Chain Wallet Address Treasury Avalanche Avalanche 0xe167199a490b04042414afd71830bbf165f44c09 Treasury BSC
Binance 0xbece6a2101ec94e817c072622671b399a3508ac1 Treasury Optimism Optimism
0xb3f5503f93d5ef84b06993a1975b9d21b962892f Treasury Arbitrum Arbitrum
0xb2f34fd4c16e656163dadfeeae4ae0c1f13b140a Treasury Polygon Polygon
0x32dc6700ac87f6300277a63b0a4fdf132a8392bd Treasury Polygon Polygon
0x32dc6700ac87f6300277a63b0a4fdf132a8392bd Treasury Ethereum Ethereum
0x2807b4ae232b624023f87d0e237a3b1bf200fd99 Treasury Ethereum Ethereum
0x2807b4ae232b624023f87d0e237a3b1bf200fd99 Treasury Fantom Fantom
0x0af13072280e10907911ce5d046c2dfa1b604d23 Treasury Gnosis Gnosis
0x04462c8ad55a3d970fd9b4944a2f4c7c15700883 Treasury Cronos/Moonbeam/Moonriver Cronos / Moonbeam / Moonriver
0xa5f9b728e9eb9a1f6fcc89dcc2efd810ba4dec41 TaskTreasury Ethereum Ethereum
0x66e2F69df68C8F56837142bE2E8C290EfE76DA9f TaskTreasury Ethereum Ethereum
0x66e2F69df68C8F56837142bE2E8C290EfE76DA9f TaskTreasury Arbitrum Arbitrum
0x527a819db1eb0e34426297b03bae11F2f8B3A19E TaskTreasury Optimism Optimism
0x527a819db1eb0e34426297b03bae11F2f8B3A19E TaskTreasury Binance Binance / BNB
0x63C51b1D80B209Cf336Bec5a3E17D3523B088cdb TaskTreasury Avalanche Avalanche
0x63C51b1D80B209Cf336Bec5a3E17D3523B088cdb TaskTreasury Polygon Polygon
0xA8a7BBE83960B29789d5CB06Dcd2e6C1DF20581C TaskTreasury Fantom Fantom
0x6c3224f9b3feE000A444681d5D45e4532D5BA531 The "Treasury" and "TaskTreasury" refer to v1 and v2 versions

respectively. If you're unsure which one you used, please check both addresses to ensure you retrieve your funds correctly.

Gelato Functions replacing Automate

With the launch of Gelato Functions we are deprecating our Automate service. The key benefits of Gelato Functions include:

- Access off-chain data & computation via Typescript Functions
- Unified payments across all networks via 1Balance
- Listen to on-chain Events to trigger your Functions
- Support for many more networks
- Much higher performance and lower latencies
- Enhanced UX, DevX and analytics
-

To get these benefits users need to migrate their tasks from Automate (now Legacy Automate) to Gelato Functions.

When should I migrate?

Legacy Automate will be deprecated 30th June 2024 so users should plan to migrate to Functions before this date.

Legacy Automate: Key Changes

- From 1st December 2023, some Legacy Automate tasks will be expired and won't be checked anymore [Check conditions for expired tasks](#)
-
- From 1st January 2024, Automate Legacy transactions will cost 50% higher than on new Gelato Functions [Check details on cost changes](#)
-
-

Cost changes

Gelato Functions unifies your transaction costs across all networks using our 1Balance system. For each execution of your target contract you will be billed the USDC equivalent of your gas costs plus a fee - the fee varies depending on the network and your subscription plan. Fees will be applied to all Automate Legacy transactions from 1st January 2024. These fees will be 50% higher than those on Gelato Functions.

Task Expiries

As part of the migration we are going to introduce task expiries from 1st December 2023 - any Legacy Automate tasks that meet the following conditions will be automatically cancelled.

Testnets Tasks expire after 30 days without any execution Mainnets Tasks expire after 120 days without any execution If you have any concerns about these limits please do not hesitate to reach out to us.

Naming Updates

We have updated what some things are called in Functions, but they are directly equivalent. The most notable one is that Resolvers are now called Solidity Functions - the smart contract code that determines if your target contract can be executed and with what payload. The flow for creating a new task has been changed to follow more of an "if this then that" pattern and allows you to also use Typescript functions that tap into off-chain APIs to also determine when target contracts should be executed and with what payloads.

How do I migrate a task from Legacy Automate to Functions?

The exact steps you need to go through will depend on the precise characteristics of your existing tasks. Here is some general guidance - if you have any questions or require further support please contact us for assistance.

1. Smart contract changes (optional)
2. If you are inheriting AutomateReady
3. or OpsReady
4. in your target contracts, upgrade them to use the new version of [AutomateReady](#)
5. If you are whitelisting Gelato as a restricted msg.sender
6. , update your whitelisting to use your new dedicated msg.sender
7. :
8.
 - Check your dedicated msg.sender address on each network here <https://app.gelato.network/settings>
- 9.

- 10. ?
- 11.

Security Considerations

1. Create your new tasks on app.gelato.network
2. Connect to app.gelato.network
3. to create your new tasks
4. If you were using "Resolver", they are now called "Solidity functions"
5. if you were using "Predefined Inputs", they are now called "Automated Transactions"
- 6.

?

1. Deposit into 1Balance
2. Deposit USDC into your 1Balance to pay for your transactions across all networks
- 3.

?

- Set up low balance e-mail alert
-

?

[1Balance Alerts Gelato Network](#) * Your tasks are now up and running on the new Gelato Functions✓ *

1. Cancel old tasks & withdraw remaining balance
2. Connect to [Automate Legacy app](#)
3. and check the modal listing all your current active tasks
4. Manually "Pause" all your old tasks
5. Withdraw your assets on each network where you still hold any balance on Gelato TaskTreasury
- 6.

Advanced user migration

Creating tasks via Smart Contract

- Upgrade your Smart Contracts to use the latest version [AutomateTaskCreator](#)
- If you were relying on Time Module, it has been replaced by Trigger Module, which allows similar configurations
-

...

Copy `function_timeTriggerModuleArg(uint128_start,uint128_interval)`

...

- Note that it's now required to use the Proxy Module on all your task creations, to enforce stricter security
-

Creating tasks via SDK

- Make sure to upgrade to use the latest version of the `automate-sdk`
- :
- - `yarn add @gelatonetwork/automate-sdk`
- *
- You can create task using the same `createTask`
- method
- if you were using `startTime`
- & `interval`
- , these options are now available inside the trigger configuration:
-

...

Copy trigger:{ type:TriggerType.TIME;// time interval trigger interval:number;// task interval in ms start:number;// task start timestamp, task will start immediately if undefined or 0 }

...

[Previous Subscription & Payments Next Account Abstraction](#) Last updated 13 days ago