

Subscription & Payments

Subscription Plans

With our subscription plans, users can choose the level of service that best fits their development and operational needs. Our plans are designed to cater to everything from small-scale projects to large enterprise operations.

Gelato Units (G-units)

We're introducing a new way to measure your usage: Gelato Units, or G-Units. G-Units are a straightforward way to calculate how much computing power your tasks use. They replace the old method of counting each task run, allowing you to run more tasks with less resource use. If your tasks are simple and use fewer resources, you can do more of them each month. The less computing power you use, the more tasks you can run—making G-Units a fairer way to measure and pay for what you actually use.

Resource Used Gelato Units Solidity Function 1 Run 1 GU Typescript Function * 1 RPC Call * 1 CPU Second * 1 GU * 2 GU * Transaction Simulation Each time a solidity or a typescript function returns an executable payload, we verify that the transaction is not reverting 1 GU You decide when your Function [triggers](#) ; running it more often will use up more G-Units. For instance, if a task is executed with every new block on the Polygon network, it will use more G-Units than if it's scheduled to execute just once at month's end. You can keep track of your current subscription and handle all aspects of your plan with ease. To view your current plan, [click here](#) . To manage your subscriptions [click here](#) .

Payments

In order for the network to be sustainable & decentralized, Gelato Nodes charge fees for running off-chain computation and executing transactions.

Our default method is to pay for all your Web3 Function costs across all networks from a single balance using Gelato 1Balance. Learn more about it by heading over the linked page:

[page 1Balance](#) If you exceed these limits your Web3 Function runs will be throttled. If you have any questions or would like to discuss your subscription, please reach out to us [here](#) .

Transaction Charges

Each transaction that Gelato Nodes execute require a small fee to incentivize Nodes to adhere to the protocol and get your transactions included into your desired blockchain in a fast and secure fashion.

To achieve this, Nodes charge a fee as a percentage of total gas cost for the executed transaction. This varies across networks - Nodes charge higher premiums on cheaper networks and vice versa.

Network Percentage Premium (%) Ethereum 20 Polygon 70 Fantom 50 Avalanche 40 BNB 30 Optimism 50 Arbitrum 50 Gnosis 100 ZkSync Era 50 Polygon zkEvm 50 Base 50 Linea 50 Table 1 - Fee premiums as a percentage of total gas cost per network. Testnet transactions are subsidized by Gelato.

These transaction premiums can be customised for users. Please reach out to us [here](#) to discuss your needs.

Transaction pays for itself

You can also choose to have your function pay the fee during executions. It must be remembered that running Web3 Functions has computational costs. Please see here the Free Tier limits, in the case that the Web3 Functions goes above these limits, [1Balance](#) will be also required to pay for the computational costs.

This can be done by inheriting [AutomateReady](#) .

...

```
Copy contractCounterWTisAutomateReady{ uint256publiccount; uint256publiclastExecuted;
constructor(address_automate,address_taskCreator) AutomateReady(_automate, _taskCreator) {}
receive()externalpayable{}

functionincreaseCount(uint256amount)externalonlyDedicatedMsgSender{ count+=amount; lastExecuted=block.timestamp;
(uint256fee,addressfeeToken)=_getFeeDetails();
_transfer(fee,feeToken); } }
```

...

In the `increaseCount` function, we use `_transfer` inherited from `AutomateReady` to pay Gelato.

`_transfer` has two parameters, `fee` and `feeToken` which has to be queried from the Automate contract by using `getFeeDetails()`

To create a task that pays for itself, head over to the task properties and enable the "Transaction pays itself"

?

[Previous Contract Addresses](#) [Next Account Abstraction](#) Last updated 1 month ago On this page * [Subscription Plans](#) * [Gelato Units \(G-units\)](#) * [Payments](#) * [Transaction Charges](#) * [Transaction pays for itself](#)