

Custom logic triggers

What is a resolver and when do you need it?

A resolver is a smart contract in which developers can embed their logic of "when" an execution should occur and what data should executors use for the execution. Gelato will query these resolvers periodically to check if it is time to execute.

You need a resolver if

- You want to define the conditions of "when" Gelato should execute your task.
- You want to have different arguments for the function you are automating on each execution.
-

You do not need a resolver if

- You simply want to repeatedly call a smart contract function with the same or without arguments (e.g. every 1 hour).
- If you just want Gelato to try to execute a particular function in case it does NOT revert
-

We give developers the freedom to choose what kind of resolver they want to use.

Smart contract resolver

A smart contract resolver is deployed on the corresponding network. (e.g. Ethereum Mainnet, Fantom) If you feel more comfortable writing in solidity, or have a simple condition that only uses on-chain data, you could go with this option.

(BETA) Web3 Functions

Web3 Functions enables you to define conditions and generate payload for execution based on off-chain data. (e.g. API calls & subgraph queries).

[Previous Defining function inputs](#) [Next Smart Contract Resolvers](#) Last updated 4 months ago On this page * [What is a resolver and when do you need it?](#) * [Smart contract resolver](#) * [\(BETA\) Web3 Functions](#)