

# Relayers

This documentation provides a comprehensive guide to Relayers within the Wormhole network, describing their role, types, and benefits in facilitating cross-chain processes.

Relayers in the Wormhole context are processes that deliver Verifiable Action Approvals ([VAAs](#)) to their destination, playing a crucial role in Wormhole's security model. They can't compromise security, only liveness, and act as delivery mechanisms for VAAs without the capacity to tamper with the outcome.

There are three primary types of relayers discussed:

- [Client-side Relaying](#)
  - : A cost-efficient, no-backend-infrastructure approach relying on user-facing front ends. It provides a simple solution, although it can complicate the user experience due to the manual steps involved.
- [Specialized Relayers](#)
  - : They are backend components that handle parts of the cross-chain process, offering a smoother user experience and allowing off-chain calculations to reduce gas costs. These relayers could operate through direct listening to the Guardian Network (Spy Relaying) or by providing a REST endpoint to accept VAAs to be relayed (REST Relaying).
- [Standard Relayers](#)
  - : A decentralized relayer network that can deliver arbitrary VAAs, reducing the developer's need to develop, host, or maintain relayers. However, they require all calculations to be done on-chain and might be less gas-efficient.
- 

## Client-side Relaying

Client-side relaying relies on user-facing front ends, such as a webpage or a wallet, to carry out the complete cross-chain process.

### Key Features

1. Cost-Efficiency:
2. Users only pay for the transaction fee for the second transaction, eliminating any additional costs.
3. No Backend Infrastructure:
4. The process is completely client-based, eliminating the need for a backend relaying infrastructure.
- 5.

## Implementation

Users themselves carry out the three steps of the cross-chain process:

1. Perform an action on chain A.
2. Retrieve the resulting VAA from the Guardian Network.
3. Perform an action on chain B using the VAA.
- 4.

## Considerations

Though simple, this type of relaying is generally not recommended if your aim is a highly-polished user experience. It can, however, be useful for getting a Minimum Viable Product (MVP) up and running. \* Users must sign all required transactions with their own wallet. \* Users must have funds to pay the transaction fees on every chain involved. \* The user experience may be cumbersome due to the manual steps involved. \*

## Specialized Relayers

Specialized relayers are purpose-built components within the Wormhole protocol, designed to relay messages for specific applications. They are capable of performing off-chain computations and can be customized to suit a variety of use-cases.

### Key Features

1. Optimization:
2. Capable of performing untrusted off-chain computations which can optimize gas costs.
3. Customizability:
4. Allows for specific strategies like batching, conditional delivery, multi-chain deliveries, and more.
5. Incentive Structure:
6. Developers have the freedom to design an incentive structure suitable for their application.
7. Enhanced UX:
8. The ability to perform steps 2 and 3 of the cross-chain process on behalf of the user can simplify the user experience.
- 9.

## Implementation

To make the development of specialized relayers easier, a plugin relayer is available in the [main Wormhole repository](#). This sets up the basic infrastructure for relaying, allowing developers to focus on implementing the specific logic for their application. There are two main methods of setting up a specialized relayer:

- Spy Relaying:
  - Involves listening directly to the Guardian Network via a spy.
- REST Relaying:
  - Provides a REST endpoint to accept a VAA that should be relayed.
- 

## Considerations

Remember, despite their name, specialized relayers are still considered untrusted. VAAs are public and can be submitted by anyone, so developers should not rely on off-chain relayers to perform any computation which is considered "trusted".

- Development work and hosting of relayers are required.
- The fee-modeling can become complex, as relayers are responsible for paying target chain fees.
- Relayers are responsible for liveness, adding an additional dependency for the cross chain application.
- 

## Standard Relayers

Standard relayers are a component of a decentralized network in the Wormhole protocol, facilitating the delivery of Verifiable Action Approvals (VAAs) to recipient contracts compatible with the standard relayer API.

## Key Features

1. Lower Operational Costs:
2. No need to develop, host, or maintain individual relayers.
3. Simplified Integration:
4. Because there is no need to run a relayer, integration is as simple as calling a function and implementing an interface.
- 5.

## Implementation

The standard relayer integration involves two key steps:

1. Delivery Request:
2. Request delivery from the Wormhole Relay Ecosystem Contract.
3. Relay Reception:
4. Implement a [receiveWormholeMessages](#)
5. function within their contracts. This function is invoked upon successful relay of the VAA.
- 6.

## Considerations

Developers should note that the choice of relayers depends on the specific requirements and constraints of their project. Standard relayers offer simplicity and convenience but might limit customization and optimization opportunities compared to specialized relayers. \* All computations are performed on-chain. \* Potentially less gas-efficient compared to specialized relayers. \* Optimization features like conditional delivery, batching, off-chain calculations might be restricted. \* Support may not be available for all chains. \*

## Fundamentals

This section highlights the crucial principles underpinning the operation and handling of relayers within the Wormhole network.

Relayers are fundamentally untrusted entities within the network. This means that while they don't require your trust, they shouldn't be trusted implicitly. They function as delivery mechanisms, transporting Verifiable Action Approvals (VAAs) from their source to their destination.

Key characteristics of VAAs include:

- Public emission from the Guardian Network
- Authentication through signatures from the Guardian Network
- Verifiability by any entity or any Wormhole Core Contract
- As a result of these characteristics, anyone can pick up a VAA and deliver it anywhere, but no one can alter the VAA content without invalidating the signatures.

•

When designing contracts, it's crucial to only trust information contained within your contract or a VAA. Relying on information from a relayer could expose you to untrusted input attacks.

Advanced strategies can involve having relayers perform untrusted off-chain computation, which is passed into the destination contract. These strategies can optimize gas costs but can also create attack vectors if not used correctly.

In summary, the design of a relayer should ensure there's a single, deterministic way that messages in your protocol can be processed. In an optimally designed protocol, relayers should have a 'correct' implementation, mirroring "crank turner" processes used elsewhere in blockchain.

Last updated 1 month ago

On this page \* [Client-side Relaying](#) \* [Specialized Relayers](#) \* [Standard Relayers](#) \* [Fundamentals](#)

Was this helpful? [Edit on GitHub](#)