What is Flood Plain?

Flood Plain serves as the settlement contract within the Flood protocol. Although it plays a crucial role in the operation of Flood, its functionality on its own is limited. In real world applications, it works in conjunction with supporting infrastructure. It is specifically designed to manage Orders, Zones, and Fulfillers, focusing on a small core set of responsibilities.

Flood Plain has four responsabilities:

- Validate orders against the order Zone.
- Delegate signature verification and offer transfer toPermit2(opens in a new tab)
- . It does this by setting the order as the Witness in the Permit2 transfer.
- Ensure the recipient receives enough tokens.
- Execute pre and post trade hooks.

Let's first precisely define what an order is, and we'll then see how those responsabilities are take care of during order fulfillment.

What is an Order?

An order is defined as a struct with the following fields:

struct

Hook { address target; bytes data; }

struct

Item { address token; uint256 amount; }

struct

Order { /// The account providing the offer. address offerer; /// The Zone, determining wether the order is valid or not. address zone; /// The address receiving the consideration. address recipient; /// A list of tokens to sell, together with their respective amounts. Item[] offer; /// The token to receive for the offer. The amount here is the minimum amount the user is willing to receive. Item consideration; /// When the order expires. uint256 deadline; /// The permit2 nonce associated with the order. uint256 nonce; /// Pre-trade hooks. Hook[] preHooks; /// Post-trade hooks. Hook[] postHooks; } Owners create orders that users then sign, thereby explicitly agreeing to the conditions outlined in the order. For instance, a trading interface (Owner) codes the generation of orders based on user preferences, and these orders are subsequently signed by the users.

Owners then share those orders with Fulfillers, which fulfill them by using liquidity sources onchain or offchain.

Flood Plain, being the entry point to order fulfillment, ensures the order's semantics are upheld and that Zones and Fulfillers perform their roles correctly.

More on how to construct orders can be found in theReference .

Order Fulfillment Flow

Let's take a close look at each step of order fulfillment:

1. Start:
2. A Fulfiller-controlled account initiates the transaction by callingsourceConsideration
3. on FloodPlain.
4. Pre Hooks:
5. FloodPlain executes pre-trade hooks.
6. Order Validation:
7. FloodPlain callsvalidateOrder
8. on the order Zone. If the order is invalid, the transaction reverts.
9. Signature Verification and Offer Transfer:
10. APermitTransferFrom
11. struct is constructed and the order used as witness. Alongside the signature, this is passed to Permit2, which verifies the signature and transfers the offer tokens to the Fulfiller.
12. Trade Execution:
13. FloodPlain callssourceConsideration
14. on the Fulfiller. The Fulfiller returns the amount of tokens to be sent to the order recipient.
15. Amount Checks & Transfers:
16. If the amount of consideration of the previous step is less than the minimum set in the order, the transaction reverts, otherwise the tokens are pulled from the Fulfiller and sent to the order recipient.

17. Post Hooks:
18. FloodPlain executes post-trade hooks.

Last updated onFebruary 3, 2024 Overview Everything About Zones