

Hey all,

We've been experimenting with building an intent-centric Order Flow Auction (OFA) on Suave and have come up with the following structure to define an intent:

```
{ "core": { "description": "Essential information about the intent. All properties are be mandatory.", "accounts": { "description": "The account that the intent is associated with.", "type": "array", [ "address": { "type": "string", "description": "The address of the account that the intent is on behalf of." }, "chainId": { "type": "number", }, }, "id": { "description": "The unique identifier for the intent. This can be any string.", "type": "string" } }, "constraints": { "description": "All are optional. Constraints allow you to specify limits on how this intent should behave.", "properties": { "permittedChains": { "description": "An array of permitted chains that this intent is allowed to use. An empty array would indicate any chain, and a single entry in the array would indicate one chain only.", "type": "array", }, "deadline": { "description": "The deadline for the intent specified as unix timestamp", "type": "number" }, "maxGas": { "description": "The maximum gas token spend for the intent. Specified as in wei.", "type": "string" }, "slippagePercentage": { "description": "The maximum percentage at which the gas cost should consume the total amount of the native asset in the account used to pay for gas. For example, setting this to 50% ensures that the total gas cost does not exceed 50% or more of the remaining gas token in the account.", "type": "number" }, "dispensableAssets": { "description": "An array of permitted assets that this intent is allowed to use.", "type": "array", }, "desiredAssets": { "description": "An array of desired assets that this intent is hoping to fulfil.", "type": "array", }, }, }
```

The solution/bid corresponding to this intent is defined as:

```
{ "bidId": { "description": "A unique hash identifying a solution", "type": "string", }, "steps": [ { "sequenceNo": { "type": "number", "description": "The sequence of execution of each the steps", }, "chainId": { "type": "number", "description": "The chainId where this txn should be executed", }, "validUntil": { "description": "Unix timestamp until when this transaction will be valid", "type": "number", }, "solution": { "description": "The transaction data that needs to be signed", "to": { "type": "string", "description": "The address of the account/smart contract to execute the txn", }, "calldata": { "type": "string", "description": "The calldata associated with the txn", }, "value": { "type": "string", "description": "Used in case of native transfers", }, }, ], }
```

Are there any teams that are building solvers on Suave? if you are building solvers on Suave, wanted to collab on defining a standard for cross-chain order flow. Some exciting stuff we are working on, so, please feel free to share your thoughts, or even DM me if that works for you.