

# Class: Twap

ComposableCoW implementation of a TWAP order.

Author

mfw78[mfw78@rndlabs.xyz](mailto:mfw78@rndlabs.xyz)

## Hierarchy

- [ConditionalOrder](#)
- [<TwapData](#)
- [,TwapStruct](#)
- 
- ↳ Twap

## Constructors

### constructor

•new Twap (params ):[Twap](#)

### Parameters

Name Type params [ConditionalOrderArguments](#) [<TwapData](#)

### Returns

[Twap](#)

See

[ConditionalOrder.constructor](#)

Throws

If the TWAP order is invalid.

Throws

If the TWAP order is not ABI-encodable.

Throws

If the handler is not the TWAP address.

### Overrides

[ConditionalOrder .constructor](#)

### Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:172

## Properties

### data

•Readonly data [:TwapData](#)

### Inherited from

[ConditionalOrder .data](#)

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:37

## handler

• Readonly handler :string

### Inherited from

[ConditionalOrder](#) .handler

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:35

## hasOffChainInput

• Readonly hasOffChainInput :boolean

### Inherited from

[ConditionalOrder](#) .hasOffChainInput

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:39

## isSingleOrder

• isSingleOrder :boolean =true

### Overrides

[ConditionalOrder](#) .isSingleOrder

### Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:164

## salt

• Readonly salt :string

### Inherited from

[ConditionalOrder](#) .salt

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:36

## staticInput

• Readonly staticInput :[TwapStruct](#)

### Inherited from

[ConditionalOrder](#) .staticInput

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:38

## Accessors

### context

•get context ():undefined [ContextFactory](#)

Enforces that TWAPs will commence at the beginning of a block by use of theCurrentBlockTimestampFactory contract to provide the current block timestamp as the start time of the TWAP.

### Returns

undefined [ContextFactory](#)

### Overrides

ConditionalOrder.context

### Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:205

## createCalldata

•get createCalldata ():string

Get the calldata for creating the conditional order.

This will automatically determine whether or not to usecreate orcreateWithContext based on the order type's context dependency.

NOTE : By default, this will cause the create to emit theConditionalOrderCreated event.

### Returns

string

The calldata for creating the conditional order.

### Inherited from

ConditionalOrder.createCalldata

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:113

## ctx

•get ctx ():string

The context key of the order (bytes32(0) if a merkle tree is used, otherwise H(params)) with which to lookup the cabinet

The context, relates to the 'ctx' in the contract<https://github.com/cowprotocol/composable-cow/blob/c7fb85ab10c05e28a1632ba97a1749fb261fcd9b/src/interfaces/ConditionalOrder.sol#L38>

### Returns

string

### Inherited from

ConditionalOrder.ctx

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:166

## id

•get id ():string

Calculate the id of the conditional order (which also happens to be the key used forctx in the ComposableCoW contract).

This is akeccak256 hash of the serialized conditional order.

### Returns

string

The id of the conditional order.

### Inherited from

ConditionalOrder.id

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:157

### leaf

•get leaf ():[ConditionalOrderParams](#)

Get theleaf of the conditional order. This is the data that is used to create the merkle tree.

For the purposes of this library, theleaf is theConditionalOrderParams struct.

### Returns

[ConditionalOrderParams](#)

Theleaf of the conditional order.

See

ConditionalOrderParams

### Inherited from

ConditionalOrder.leaf

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:177

### offChainInput

•get offChainInput ():string

If the conditional order has off-chain input, return it!

NOTE : This should be overridden by any conditional order that has off-chain input.

### Returns

string

The off-chain input.

### Inherited from

ConditionalOrder.offChainInput

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:201

### orderType

•get orderType ():string

Get a descriptive name for the type of the conditional order (i.e twap, dca, etc).

### Returns

string

The concrete type of the conditional order.

### Overrides

ConditionalOrder.orderType

### Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:219

## removeCalldata

•get removeCalldata ():string

Get the calldata for removing a conditional order that was created as a single order.

### Returns

string

The calldata for removing the conditional order.

### Inherited from

ConditionalOrder.removeCalldata

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:145

## Methods

### assertIsValid

►assertIsValid ():void

### Returns

void

### Inherited from

[ConditionalOrder](#) .assertIsValid

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:95

## cabinet

►cabinet (params ):Promise <string

Checks the value in the cabinet for a given owner and chain

### Parameters

Name Type Description params [OwnerContext](#) owner context, to be able to check the cabinet

## Returns

Promise <string

## Inherited from

[ConditionalOrder](#) .[cabinet](#)

## Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:337

## encodeStaticInput

►encodeStaticInput ():string

Get the encoded static input for the TWAP order.

## Returns

string

The ABI-encoded TWAP order.

## Overrides

[ConditionalOrder](#) .[encodeStaticInput](#)

## Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:449

## encodeStaticInputHelper

►encodeStaticInputHelper (orderDataTypes ,staticInput ):string

A helper function for generically serializing a conditional order's static input.

## Parameters

Name	Type	Description
orderDataTypes	string []	ABI types for the order's data struct.
staticInput	<a href="#">TwapStruct</a>	-

## Returns

string

An ABI-encoded representation of the order's data struct.

## Inherited from

[ConditionalOrder](#) .[encodeStaticInputHelper](#)

## Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:234

## endTimeStamp

►endTimeStamp (startTimestamp ):number

Given the start timestamp of the TWAP, calculate the end timestamp.

## Parameters

Name	Type	Description
startTimestamp	number	The start timestamp of the TWAP.

## Returns

number

The timestamp at which the TWAP will end.

Dev

As usually the `endTimestamp` is used when determining a TWAP's validity, we don't do any lookup to the blockchain to determine the start timestamp, as this has likely already been done during the verification flow.

Dev

Beware to handle the case of `span != 0` ie. `durationOfPart.durationType !== DurationType.AUTO` .

### Defined in

`external/cow-sdk/src/composable/orderTypes/Twap.ts:302`

## handlePollFailedAlreadyPresent

► `handlePollFailedAlreadyPresent ( _orderId , _order , params )`: `Promise <undefined | PollResultErrors`

Handles the error when the order is already present in the orderbook.

Given the current part is in the book, it will signal to Watch Tower what to do:

- Wait until the next part starts
- Don't try again if current part is the last one

NOTE: The error messages will refer to the parts 1-indexed, so first part is 1, second part is 2, etc.

### Parameters

Name Type `_orderId` string `_order` `DataStruct` `params` [PollParams](#)

### Returns

`Promise <undefined | PollResultErrors`

### Overrides

[ConditionalOrder](#) `.handlePollFailedAlreadyPresent`

### Defined in

`external/cow-sdk/src/composable/orderTypes/Twap.ts:378`

## isAuthorized

► `isAuthorized ( params )`: `Promise <boolean`

Checks if the owner authorized the conditional order.

### Parameters

Name Type Description `params` [OwnerContext](#) owner context, to be able to check if the order is authorized

### Returns

`Promise <boolean`

true if the owner authorized the order, false otherwise.

### Inherited from

[ConditionalOrder](#) `.isAuthorized`

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:326

## isValid

► isValid (): [IsValidResult](#)

Validate the TWAP order.

### Returns

[IsValidResult](#)

Whether the TWAP order is valid.

Throws

If the TWAP order is invalid.

See

[TwapStruct](#) for the native struct.

### Overrides

[ConditionalOrder](#) .isValid

### Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:230

## poll

► poll (params ): Promise <[PollResult](#)

Poll a conditional order to see if it is tradeable.

### Parameters

Name Type params [PollParams](#)

### Returns

Promise <[PollResult](#)

The tradeableGPv2Order.Data struct and the signature for the conditional order.

Throws

If the conditional order is not tradeable.

### Inherited from

[ConditionalOrder](#) .poll

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:249

## pollValidate

► pollValidate (params ): Promise <undefined [PollResultErrors](#)

Checks if the owner authorized the conditional order.

### Parameters

Name Type params [PollParams](#)



## Returns

Promise <undefined | [PollResultErrors](#)

true if the owner authorized the order, false otherwise.

## Overrides

[ConditionalOrder](#) .pollValidate

## Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:320

## serialize

►serialize():string

Serialize the TWAP order into it's ABI-encoded form.

## Returns

string

The ABI-encoded TWAP order.

## Overrides

[ConditionalOrder](#) .serialize

## Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:441

## startTimestamp

►startTimestamp(params):Promise <number

## Parameters

Name Type params [OwnerContext](#)

## Returns

Promise <number

## Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:268

## toString

►toString():string

Create a human-readable string representation of the TWAP order.

## Returns

string

A human-readable string representation of the TWAP order.

## Overrides

[ConditionalOrder](#) .toString

## Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:476

## transformDataToStruct

►transformDataToStruct (data ): [TwapStruct](#)

Transform parameters into a native struct.

### Parameters

Name	Type	Description
data	<a href="#">TwapData</a>	As passed by the consumer of the API.

### Returns

[TwapStruct](#)

A formatted struct as expected by the smart contract.

### Overrides

[ConditionalOrder](#) .[transformDataToStruct](#)

### Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:517

## transformStructToData

►transformStructToData (struct ): [TwapData](#)

Transform parameters into a TWAP order struct.

### Parameters

Name	Type	struct
	<a href="#">TwapStruct</a>	

### Returns

[TwapData](#)

A formatted struct as expected by the smart contract.

### Overrides

[ConditionalOrder](#) .[transformStructToData](#)

### Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:527

## deserialize

►deserialize (twapSerialized ): [Twap](#)

Deserialize a TWAP order from it's ABI-encoded form.

### Parameters

Name	Type	Description
twapSerialized	string	ABI-encoded TWAP order to deserialize.

### Returns

[Twap](#)

A deserialized TWAP order.

### Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:458

## deserializeHelper

►deserializeHelper <T

(s ,handler ,orderDataTypes ,callback ):T

A helper function for generically deserializing a conditional order.

### Type parameters

Name T

### Parameters

Name	Type	Description
s	string	The ABI-encoded IConditionalOrder.Params struct to deserialize.
handler	string	Address of the handler for the conditional order.
orderDataTypes	string []	ABI types for the order's data struct.
callback	(d :any ,salt :string ) =>T	A callback function that takes the deserialized data struct and the salt and returns an instance of the class.

### Returns

T

An instance of the conditional order class.

### Inherited from

[ConditionalOrder](#) .[deserializeHelper](#)

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:400

## fromData

►fromData (data ,salt? ): [Twap](#)

Create a TWAP order with sound defaults.

### Parameters

Name	Type	Description
data	<a href="#">TwapData</a>	The TWAP order parameters in a more user-friendly format.
salt?	string	-

### Returns

[Twap](#)

An instance of the TWAP order.

### Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:187

## fromParams

►fromParams (params ): [Twap](#)

Create a TWAP order with sound defaults.

### Parameters

Name	Type	params
	<a href="#">ConditionalOrderParams</a>	

### Returns

[Twap](#)

An instance of the TWAP order.

### Defined in

external/cow-sdk/src/composable/orderTypes/Twap.ts:196

### leafTold

►leafTold (leaf ):string

Calculate the id of the conditional order.

### Parameters

Name	Type	Description
leaf	<a href="#">ConditionalOrderParams</a>	The leaf representing the conditional order.

### Returns

string

The id of the conditional order.

See

[ConditionalOrderParams](#)

### Inherited from

[ConditionalOrder](#) .[leafTold](#)

### Defined in

external/cow-sdk/src/composable/ConditionalOrder.ts:191 [Previous](#) [SubgraphApi](#) [Next](#) [ApiContext](#)