L2ToL1Message

Classes

L2ToL1Message

Base functionality for L2->L1 messages

Extended by

- L2ToL1MessageReader

Methods

fromEvent()

static

fromEvent < T

(l1SignerOrProvider :

T , event : L2ToL1TransactionEvent , l1Provider ? : Provider) : L2ToL1MessageReaderOrWriter < T

Instantiates a newL2ToL1MessageWriter orL2ToL1MessageReader object.

Type parameters

Type parameter T extends SignerOrProvider

Parameters

Parameter Type Description l1SignerOrProvider T Signer or provider to be used for executing or reading the L2-to-L1 message. event L2ToL1TransactionEvent The event containing the data of the L2-to-L1 message. l1Provider ? Provider Optional. Used to override the Provider which is attached tol1SignerOrProvider in case you need more control. This will be a required parameter in a future major version update.

Returns

L2ToL1MessageReaderOrWriter <T

Source

message/L2ToL1Message.ts:73

getL2ToL1Events()

static

getL2ToL1Events (l2Provider : Provider , filter : object , position ? : BigNumber , destination ? :

string , hash ? : BigNumber , indexInBatch ? : BigNumber) :

Promise < L2ToL1TransactionEvent & object [ ]

Get event logs for L2ToL1 transactions.

Parameters

Parameter Type Description l2Provider Provider filter object Block range filter filter.fromBlock BlockTag - filter.toBlock ? BlockTag - position ? BigNumber The batchnumber indexed field was removed in nitro and a position indexed field was added. For pre-nitro events the value passed in here will be used to find events with the same batchnumber. For post nitro events it will be used to find events with the same position. destination ? string The L1 destination of the L2ToL1 message hash ? BigNumber The uniqueId indexed field was removed in nitro and a hash indexed field was added. For pre-nitro events the value passed in here will be used to find events with the same uniqueId. For post nitro events it will be used to find events with the same hash. indexInBatch ? BigNumber The index in the batch, only valid for pre-nitro events. This

parameter is ignored post-nitro

Returns

Promise

Any classic and nitro events that match the provided filters.

Source

message/L2ToL1Message.ts:102

L2ToL1MessageReader

Provides read-only access for l2-to-l1-messages

Extends

- L2ToL1Message

Extended by

- L2ToL1MessageWriter

Methods

getFirstExecutableBlock()

getFirstExecutableBlock (l2Provider : Provider) :

Promise < null

| BigNumber

Estimates the L1 block number in which this L2 to L1 tx will be available for execution. If the message can or already has been executed, this returns null

Parameters

Parameter Type Description l2Provider Provider

Returns

Promise <null |BigNumber

expected L1 block number where the L2 to L1 message will be executable. Returns null if the message can or already has been executed

Source

message/L2ToL1Message.ts:258

status()

status (l2Provider : Provider) :

Promise < L2ToL1MessageStatus

Get the status of this message In order to check if the message has been executed proof info must be provided.

Parameters

Parameter Type l2Provider Provider

Returns

Promise <L2ToL1MessageStatus

Source

message/L2ToL1Message.ts:226

waitUntilReadyToExecute()

waitUntilReadyToExecute (l2Provider : Provider , retryDelay :

number) :

Promise < CONFIRMED

|

EXECUTED

> Waits until the outbox entry has been created, and will not return until it has been. WARNING: Outbox entries are only created when the corresponding node is confirmed. Which can take 1 week+, so waiting here could be a very long operation.

Parameters

Parameter Type Default value Description l2Provider Provider undefined - retryDelay number 500

Returns

Promise <CONFIRMED |EXECUTED

outbox entry status (either executed or confirmed but not pending)

Source

message/L2ToL1Message.ts:239

fromEvent()

static

fromEvent < T

> (l1SignerOrProvider :

T , event : L2ToL1TransactionEvent , l1Provider ? : Provider) : L2ToL1MessageReaderOrWriter < T

> Instantiates a newL2ToL1MessageWriter orL2ToL1MessageReader object.

Type parameters

Type parameter T extends SignerOrProvider

Parameters

Parameter Type Description l1SignerOrProvider T Signer or provider to be used for executing or reading the L2-to-L1 message. event L2ToL1TransactionEvent The event containing the data of the L2-to-L1 message. l1Provider ? Provider Optional. Used to override the Provider which is attached tol1SignerOrProvider in case you need more control. This will be a required parameter in a future major version update.

Returns

L2ToL1MessageReaderOrWriter <T

Inherited from

L2ToL1Message .fromEvent

Source

message/L2ToL1Message.ts:73

getL2ToL1Events()

static

getL2ToL1Events (l2Provider : Provider , filter : object , position ? : BigNumber , destination ? :

string , hash ? : BigNumber , indexInBatch ? : BigNumber) :

Promise < L2ToL1TransactionEvent & object [ ]

Get event logs for L2ToL1 transactions.

Parameters

Parameter Type Description l2Provider Provider filter object Block range filter filter.fromBlock BlockTag - filter.toBlock ? BlockTag - position ? BigNumber The batchnumber indexed field was removed in nitro and a position indexed field was added. For pre-nitro events the value passed in here will be used to find events with the same batchnumber. For post nitro events it will be used to find events with the same position. destination ? string The L1 destination of the L2ToL1 message hash ? BigNumber The uniqueId indexed field was removed in nitro and a hash indexed field was added. For pre-nitro events the value passed in here will be used to find events with the same uniqueId. For post nitro events it will be used to find events with the same hash. indexInBatch ? BigNumber The index in the batch, only valid for pre-nitro events. This parameter is ignored post-nitro

Returns

Promise

Any classic and nitro events that match the provided filters.

Inherited from

L2ToL1Message .getL2ToL1Events

Source

message/L2ToL1Message.ts:102

L2ToL1MessageWriter

Provides read and write access for l2-to-l1-messages

Extends

- L2ToL1MessageReader

Constructors

new L2ToL1MessageWriter()

new

L2ToL1MessageWriter (l1Signer : Signer , event : L2ToL1TransactionEvent , l1Provider ? : Provider) : L2ToL1MessageWriter Instantiates a newL2ToL1MessageWriter object.

Parameters

Parameter Type Description l1Signer Signer The signer to be used for executing the L2-to-L1 message. event L2ToL1TransactionEvent The event containing the data of the L2-to-L1 message. l1Provider ? Provider Optional. Used to override the Provider which is attached tol1Signer in case you need more control. This will be a required parameter in a future major version update.

Returns

L2ToL1MessageWriter

Overrides

L2ToL1MessageReader.constructor

Source

message/L2ToL1Message.ts:281

Methods

execute()

execute (l2Provider : Provider , overrides ? : Overrides) :

Promise < ContractTransaction

> Executes the L2ToL1Message on L1. Will throw an error if the outbox entry has not been created, which happens when the corresponding assertion is confirmed.

Parameters

Parameter Type l2Provider Provider overrides ? Overrides

Returns

Promise <ContractTransaction

Source

message/L2ToL1Message.ts:310

getFirstExecutableBlock()

getFirstExecutableBlock (l2Provider : Provider) :

Promise < null

| BigNumber

> Estimates the L1 block number in which this L2 to L1 tx will be available for execution. If the message can or already has been executed, this returns null

Parameters

Parameter Type Description l2Provider Provider

Returns

Promise <null |BigNumber

expected L1 block number where the L2 to L1 message will be executable. Returns null if the message can or already has been executed

Inherited from

L2ToL1MessageReader .getFirstExecutableBlock

Source

message/L2ToL1Message.ts:258

status()

status (l2Provider : Provider) :

Promise < L2ToL1MessageStatus

> Get the status of this message In order to check if the message has been executed proof info must be provided.

Parameters

Parameter Type l2Provider Provider

Returns

Promise <L2ToL1MessageStatus

Inherited from

L2ToL1MessageReader .status

Source

message/L2ToL1Message.ts:226

waitUntilReadyToExecute()

waitUntilReadyToExecute (l2Provider : Provider , retryDelay :

number) :

Promise < CONFIRMED

|

EXECUTED

> Waits until the outbox entry has been created, and will not return until it has been. WARNING: Outbox entries are only created when the corresponding node is confirmed. Which can take 1 week+, so waiting here could be a very long operation.

Parameters

Parameter Type Default value Description l2Provider Provider undefined - retryDelay number 500

Returns

Promise <CONFIRMED |EXECUTED

outbox entry status (either executed or confirmed but not pending)

Inherited from

L2ToL1MessageReader .waitUntilReadyToExecute

Source

message/L2ToL1Message.ts:239

fromEvent()

static

fromEvent < T

> (l1SignerOrProvider :

T , event : L2ToL1TransactionEvent , l1Provider ? : Provider) : L2ToL1MessageReaderOrWriter < T

> Instantiates a newL2ToL1MessageWriter orL2ToL1MessageReader object.

Type parameters

Type parameter T extends SignerOrProvider

Parameters

Parameter Type Description l1SignerOrProvider T Signer or provider to be used for executing or reading the L2-to-L1 message. event L2ToL1TransactionEvent The event containing the data of the L2-to-L1 message. l1Provider ? Provider Optional. Used to override the Provider which is attached tol1SignerOrProvider in case you need more control. This will be a required parameter in a future major version update.

Returns

L2ToL1MessageReaderOrWriter <T

Inherited from

L2ToL1MessageReader .fromEvent

Source

message/L2ToL1Message.ts:73

getL2ToL1Events()

static

getL2ToL1Events (l2Provider : Provider , filter : object , position ? : BigNumber , destination ? :

string , hash ? : BigNumber , indexInBatch ? : BigNumber) :

Promise < L2ToL1TransactionEvent & object [ ]

Get event logs for L2ToL1 transactions.

Parameters

Parameter Type Description l2Provider Provider filter object Block range filter filter.fromBlock BlockTag - filter.toBlock ? BlockTag - position ? BigNumber The batchnumber indexed field was removed in nitro and a position indexed field was added. For pre-nitro events the value passed in here will be used to find events with the same batchnumber. For post nitro events it will be used to find events with the same position. destination ? string The L1 destination of the L2ToL1 message hash ? BigNumber The uniqueId indexed field was removed in nitro and a hash indexed field was added. For pre-nitro events the value passed in here will be used to find events with the same uniqueId. For post nitro events it will be used to find events with the same hash. indexInBatch ? BigNumber The index in the batch, only valid for pre-nitro events. This parameter is ignored post-nitro

Returns

Promise

Any classic and nitro events that match the provided filters.

Inherited from

L2ToL1MessageReader .getL2ToL1Events

Source

message/L2ToL1Message.ts:102

Type Aliases

L2ToL1MessageReaderOrWriter

type

L2ToL1MessageReaderOrWriter < T

:

T

extends

Provider

? L2ToL1MessageReader : L2ToL1MessageWriter ; Conditional type for Signer or Provider. If T is of type Provider then L2ToL1MessageReaderOrWriter will be of type L2ToL1MessageReader. If T is of type Signer then L2ToL1MessageReaderOrWriter will be of type L2ToL1MessageWriter.

Type parameters

Type parameter T extends SignerOrProvider

Source

message/L2ToL1Message.ts:51 Edit this page Previous L1Transaction Next L2ToL1MessageClassic