HashConsensus

- Source code
- Deployed instance for AccountingOracle
- Deployed instance for ValidatorsExitBusOracle

info It's advised to readWhat is Lido Oracle mechanism before

What is HashConsensus

HashConsensus is a contract responsible for managing oracle members committee and allowing the members to reach consensus on a data hash for each reporting frame.

Time is divided in frames of equal length, each having reference slot and processing deadline. Report data must be gathered by looking at the world state (both Ethereum Consensus and Execution Layers) at the moment of the frame's reference slot (including any state changes made in that slot), and must be processed before the frame's processing deadline.

Frame length is defined in Ethereum Consensus Layer epochs. Reference slot for each frame is set to the last slot of the epoch preceding the frame's first epoch. The processing deadline is set to the last slot of the last epoch of the frame.

Note that all state changes a report processing could entail are guaranteed to be observed while gathering data for the next frame's report. This is an essential property given that oracle reports sometimes have to contain diffs instead of the entire state, which might be impractical or even impossible to transmit and process.

Consensus members rotate within one time into two subsets:

- Non-fast-lane members
- Fast-lane members

Once the consensus is gathered, aReport processor would allow submitting and processing the actual report data. The latter is a part of thephased Oracle report flow .

Report processor (IReportAsyncProcessor

)

IReportAsyncProcessor defines the interface for a contract that gets consensus reports (i.e. hashes) pushed to and processes them asynchronously.HashConsensus doesn't expect any specific behavior from a report processor, and guarantees the following:

1. HashConsensus
2. won't submit reports viaIReportAsyncProcessor.submitConsensusReport
3. or ask to discard
4. reports viaIReportAsyncProcessor.discardConsensusReport
5. for any slot up to (and including)
6. the slot returned fromIReportAsyncProcessor.getLastProcessingRefSlot
7. .
8. HashConsensus
9. won't accept member reports (and thus won't include such reports in calculating the consensus)
10. that haveconsensusVersion
11. argument of theHashConsensus.submitReport
12. call holding a diff.
13. value than the one returned fromIReportAsyncProcessor.getConsensusVersion
14. at the moment of theHashConsensus.submitReport
15. call.

There are two core protocol contracts that implements this interface:

- AccountingOracle
- ValidatorsExitBusOracle

Fast-lane members

Fast lane members is a subset of all members that changes each reporting frame. These members can, and are expected to, submit a report during the first part of the frame called the "fast lane interval" and defined viasetFrameConfig orsetFastLaneLengthSlots . The calculation of the Fast-lane members subset depends onframeIndex ,totalMembers

andquorum . Under regular circumstances, all other members are only allowed to submit a report after the fast lane interval passes. This is done to encourage each oracle from the full set to participate in reporting on a regular basis, and identify any malfunctioning members.

The fast lane subset consists of quorum members; selection is implemented as a sliding window of the quorum width over member indices (mod total members). The window advances by one index each reporting frame.

With the fast lane mechanism active, it's sufficient for the monitoring to check that consensus is consistently reached during the fast lane part of each frame to conclude that all members are active and share the same consensus rules.

note There is no guarantee that, at any given time, it holds true that only the current fast lane members can or were able to report during the currently-configured fast lane interval of the current frame.

In particular, this assumption can be violated in any frame during which the members set, initial epoch, or the quorum number was changed, or the fast lane interval length was increased.

Therefore, the fast lane mechanism should not be used for any purpose other than monitoring of the members liveness, and monitoring tools should take into consideration the potential irregularities within frames with any configuration changes.

View methods

getChainConfig()

Returns the immutable chain parameters required to calculate epoch and slot given a timestamp.

function

getChainConfig ( )

external

view

returns

(uint256 slotsPerEpoch , uint256 secondsPerSlot , uint256 genesisTime)

Returns

Name Type Description slotsPerEpoch uint256 Number of slots per epoch,32 by default secondsPerSlot uint256 The time allocated for each slot,12 by default genesisTime uint256 Consensus Layer genesis time,1606824023 onMainnet

getFrameConfig()

Returns the time-related configuration.

function

getFrameConfig ( )

external

view

returns

(uint256 initialEpoch , uint256 epochsPerFrame , uint256 fastLaneLengthSlots)

Returns

Name Type Description initialEpoch uint256 Epoch of the frame with zero index epochsPerFrame uint256 Length of a frame in epochs fastLaneLengthSlots uint256 Length of the fast lane interval in slots; seegetIsFastLaneMember

getCurrentFrame()

Returns the current reporting frame.

function

getCurrentFrame ( )

external

view

returns

(uint256 refSlot , uint256 reportProcessingDeadlineSlot)

Returns

Name Type Description refSlot uint256 The frame's reference slot: if the data the consensus is being reached upon includes or depends on any onchain state, this state should be queried at the reference slot. If the slot contains a block, the state should include all changes from that block. reportProcessingDeadlineSlot uint256 The last slot at which the report can be processed by the report processor contract.

getInitialRefSlot()

Returns the earliest possible reference slot, i.e. the reference slot of the reporting frame with zero index.

function

getInitialRefSlot ( )

external

view

returns

(uint256)

getIsMember()

Returns whether the given address is currently a member of the consensus.

function

getIsMember (address addr)

external

view

returns

(bool)

getIsFastLaneMember()

Returns whether the given address is a fast lane member for the current reporting frame.

function

getIsFastLaneMember (address addr)

external

view

returns

(bool)

getMembers()

Returns all current members, together with the last reference slot each member submitted a report for.

function

getMembers ( )

external

view

returns

(address [ ]

memory addresses , uint256 [ ]

memory lastReportedRefSlots)

getFastLaneMembers()

Returns the subset of the oracle committee members (consisting ofquorum items) that changes each frame.

function

getFastLaneMembers ( )

external

view

returns

(address [ ]

memory addresses , uint256 [ ]

memory lastReportedRefSlots)

getQuorum()

Returns quorum number

function

getQuorum ( )

external

view

returns

(uint256)

getReportProcessor()

Returns report processor address, i.e oracle address

function

getReportProcessor ( )

external

view

returns

(address)

getConsensusState()

Returns info about the current frame and consensus state in that frame.

function

getConsensusState ( )

external

view

returns

(uint256 refSlot , bytes32 consensusReport , bool isReportProcessing)

Returns

Returns info about the current frame and consensus state in that frame.

Name Type Description refSlot uint256 Reference slot of the current reporting frame. consensusReport bytes32 Consensus report for the current frame, if any. Zero bytes otherwise. isReportProcessing bool If consensus report for the current frame is already being processed. Consensus can be changed before the processing starts.

getReportVariants()

Returns report variants and their support for the current reference slot.

function

getReportVariants ( )

external

view

returns

(bytes32 [ ]

memory variants , uint256 [ ]

memory support)

getConsensusStateForMember()

Returns the extended information related to an oracle committee member with the given address and the current consensus state. Provides all the information needed for an oracle daemon to decide if it needs to submit a report.

function

getConsensusStateForMember (address addr)

external

view

returns

(MemberConsensusState memory result)

Parameters

Name Type Description addr address The member address.

Returns

Returns a new typeMemberConsensusState

struct

MemberConsensusState

{ /// @notice Current frame's reference slot. uint256 currentFrameRefSlot ;

/// @notice Consensus report for the current frame, if any. Zero bytes otherwise. bytes32 currentFrameConsensusReport ;

/// @notice Whether the provided address is a member of the oracle committee. bool isMember ;

/// @notice Whether the oracle committee member is in the fast lane members subset /// of the current reporting frame. See getIsFastLaneMember. bool isFastLane ;

/// @notice Whether the oracle committee member is allowed to submit a report at /// the moment of the call. bool canReport ;

/// @notice The last reference slot for which the member submitted a report. uint256 lastMemberReportRefSlot ;

/// @notice The hash reported by the member for the current frame, if any. /// Zero bytes otherwise. bytes32 currentFrameMemberReport ; }

Methods

updateInitialEpoch()

Sets a new initial epoch given that the current initial epoch is in the future.

Can only be called by users withDEFAULT_ADMIN_ROLE .

function

updateInitialEpoch (uint256 initialEpoch)

external * Reverts withInitialEpochAlreadyArrived() * if current epoch more or equal initial epoch from current frame config. * Reverts withInitialEpochRefSlotCannotBeEarlierThanProcessingSlot() * if initial frame refSlot less than last processing refSlot. * Reverts withEpochsPerFrameCannotBeZero() * ifepochsPerFrame * from frame config is zero. * Reverts withFastLanePeriodCannotBeLongerThanFrame() * iffastLaneLengthSlots * from config more than frame length.

setFrameConfig()

Updates the time-related configuration.

Can only be called by users withMANAGE_FRAME_CONFIG_ROLE .

function

setFrameConfig (uint256 epochsPerFrame ,

uint256 fastLaneLengthSlots)

external * Reverts withEpochsPerFrameCannotBeZero() * ifepochsPerFrame * is zero. * Reverts withFastLanePeriodCannotBeLongerThanFrame() * iffastLaneLengthSlots * more than frame length.

Parameters

Name Type Description epochsPerFrame uint256 ALength of a frame in epochs. fastLaneLengthSlots uint256 Length of the fast lane interval in slots; seegetIsFastLaneMember .

setFastLaneLengthSlots()

Sets the duration of the fast lane interval of the reporting frame.

Can only be called by users withMANAGE_FAST_LANE_CONFIG_ROLE .

function

setFastLaneLengthSlots (uint256 fastLaneLengthSlots)

external * Reverts withFastLanePeriodCannotBeLongerThanFrame() * iffastLaneLengthSlots * more than frame length.

Parameters

Name Type Description fastLaneLengthSlots uint256 The length of the fast lane reporting interval in slots. Setting it to zero disables the fast lane subset, allowing any oracle to report starting from the first slot of a frame and until the frame's reporting deadline.

addMember()

Add a new member of the consensus.

Can only be called by users withDISABLE_CONSENSUS_ROLE role ifquorum set as UINT256_MAX.

Can only be called by users withMANAGE_MEMBERS_AND_QUORUM_ROLE role ifquorum not set as UINT256_MAX.

function

addMember (address addr ,

uint256 quorum)

external * Reverts withDuplicateMember() * ifaddr * address is already the member of consensus. * Reverts withAddressCannotBeZero() * ifaddr * address is zero. * Reverts withQuorumTooSmall(uint256 minQuorum, uint256 receivedQuorum) * ifquorum * less or equal than total members of consensus divided by 2 (quorum <= total members / 2 *)

removeMember()

Remove a member from the consensus.

Can only be called by users withDISABLE_CONSENSUS_ROLE role ifquorum set as UINT256_MAX.

Can only be called by users withMANAGE_MEMBERS_AND_QUORUM_ROLE role ifquorum not set as UINT256_MAX.

function

removeMember (address addr ,

uint256 quorum)

external * Reverts withNonMember() * ifaddr * address doesn't exists * Reverts withQuorumTooSmall(uint256 minQuorum, uint256 receivedQuorum) * ifquorum * less or equal than total members of consensus divided by 2 (quorum <= total members / 2 *)

setQuorum()

Update consensus quorum

Can only be called by users withDISABLE_CONSENSUS_ROLE role ifquorum set as UINT256_MAX.

Can only be called by users withMANAGE_MEMBERS_AND_QUORUM_ROLE role ifquorum not set as UINT256_MAX.

function

setQuorum (uint256 quorum)

external * Reverts withQuorumTooSmall(uint256 minQuorum, uint256 receivedQuorum) * ifquorum * less or equal than total members of consensus divided by 2 (quorum <= total members / 2 *)

disableConsensus()

Disable consensus quorum, i.e set quorum asUINT256_MAX (UNREACHABLE_QUORUM)

Can only be called by users withDISABLE_CONSENSUS_ROLE

function

disableConsensus ( )

external * Reverts withQuorumTooSmall(uint256 minQuorum, uint256 receivedQuorum) * ifquorum * less or equal than total members of consensus divided by 2 (quorum <= total members / 2 *)

setReportProcessor()

Set report processor address, i.e oracle address

Can only be called by users withMANAGE_REPORT_PROCESSOR_ROLE .

function

setReportProcessor (address newProcessor)

external * Reverts withReportProcessorCannotBeZero() * ifnewProcessor * address is zero. * Reverts withNewProcessorCannotBeTheSame() * ifnewProcessor * address is equal to the previous processor address.

submitReport()

Used by oracle members to submit hash of the data calculated for the given reference slot.

function

submitReport (uint256 slot ,

bytes32 report ,

uint256 consensusVersion)

external

Parameters

Name Type Description slot uint256 The reference slot the data was calculated for. Reverts if doesn't match the current reference slot. report bytes32 Hash of the data calculated for the given reference slot. consensusVersion uint256 Version of the oracle consensus rules. Reverts if doesn't match the version returned by the currently set consensus report processor, or zero if no report processor is set.

Reverts

- Reverts withInvalidSlot()
- ifslot
- is zero.
- Reverts withInvalidSlot()
- ifslot
- is not equal current frame refSlot.
- Reverts withNumericOverflow()
- ifslot
- is more thanUINT64_MAX
- Reverts withEmptyReport()
- ifreports
- is zero hash (bytes32(0)
-)
- Reverts withNonMember()
- if caller address doesn't exists in members array
- Reverts withUnexpectedConsensusVersion(uint256 expected, uint256 received)
- ifconsensusVersion
- is not equal report processor consensus version.
- Reverts withStaleReport()
- if the current frame slot is more than the frame report processing deadline slot.
- Reverts withNonFastLaneMemberCannotReportWithinFastLaneInterval()
- if the current frame slot is less or equal frame ref slot plus fastlane length AND the member who submits the report is not fastlane member.
- Reverts withConsensusReportAlreadyProcessing()
- if the member sends a report for the same slot.
- Reverts withDuplicateReport()
- if the member already sends the report.

Events

FrameConfigSet()

Emits when a new frame config set viasetFrameConfig .

event

FrameConfigSet (uint256 newInitialEpoch ,

uint256 newEpochsPerFrame)

FastLaneConfigSet()

Emits when fast lane length changed (i.e., length defined in slots).

event

FastLaneConfigSet (uint256 fastLaneLengthSlots)

MemberAdded()

Emits when a new member of consensus is added.

event

MemberAdded (address

indexed addr ,

uint256 newTotalMembers ,

uint256 newQuorum)

MemberRemoved()

Emits when an existing member of consensus is removed.

event

MemberRemoved (address

indexed addr ,

uint256 newTotalMembers ,

uint256 newQuorum)

QuorumSet()

Emits when a quorum of consensus members is changed.

event

QuorumSet (uint256 newQuorum ,

uint256 totalMembers ,

uint256 prevQuorum)

ReportReceived()

Emits when a new report received for the providedrefSlot bymember containing thereport hash.

event

ReportReceived (uint256

indexed refSlot ,

address

indexed member ,

bytes32 report)

ConsensusReached()

Emits when a consensus reached for the providedrefSlot containing thereport hash.

event

ConsensusReached (uint256

indexed refSlot ,

bytes32 report ,

uint256 support)

ConsensusLost()

Emits when the previously established consensus for the providedrefSlot is disbanded.

event

ConsensusLost (uint256

indexed refSlot)

ReportProcessorSet()

Emits when the report processor is changed fromprevProcessor toprocessor . Both addresses must comply with theIReportAsyncProcessor interface.

event

ReportProcessorSet (address

indexed processor ,

address

indexed prevProcessor) Edit this page Previous ValidatorsExitBusOracle Next LegacyOracle