cosigo — satellite whitepaper (v1.2, text-only)

Tagline: value, not price — a silver-anchored, satellite-driven money and stewardship model.

Last updated: October 24, 2025

Author: cosigo project team / cosigo founder

Website: cosigo.io

Satellites directory: cosigo.io/Satellites

Reference contract source:

https://etherscan.io/address/0x0f1e8ee8a035270ed9952591d7dbdc600e2b4a49#code

0. note to reader

This is a consolidated text-only edition. It includes the original v1.0 content plus additional technical

details derived from the provided ABI and the latest CosigoSatellite implementation. It is

intentionally verbose to aid review, audit, and operations. You can paste this into any editor without

formatting symbols.

1. executive summary

Cosigo is a multi-layered, silver-anchored token ecosystem that aligns digital token units to physical

silver, governed by a modular Guardian to Satellite architecture. Each Cosigo token represents 1

mg of physical silver. The design supports everyday trade, deposit to mint to redemption flows, and

localized monetary stewardship via permissioned Satellites. The system prioritizes transparency,

custodial proofs, and local autonomy while preserving global safety through Guardian policy and

caps.

Key components

1) cosigo token. ERC-20 like. One token equals one milligram of silver.

2) Guardian. Global policy authority. Approves satellites. Sets global caps. Market gating and

emergency controls.

3) Satellite. Per town or region. Maintains deposit registry, computes mint headroom, enforces

redemption policy, routes fees locally, and settles an optional Guardian skim.

4) Sinks and roles. Burn sink, fee sink, treasury, custodian accounts. On-chain addresses are

published in the Satellites directory for verification and public transparency.

2. the problem

Price and value have drifted apart under elastic fiat systems and paper market abstractions.

Consumer goods and wages exhibit noise and manipulation. Many digital assets lack tangible

backing and operational accountability.

Cosigo is built to

1) Anchor value to a simple physical unit, the milligram of silver.

2) Empower local communities to run Satellites with accountable custodians.

3) Publish auditable on-chain trails for deposits, minting, fees, and redemptions.

4) Keep flows economical and gas-aware at everyday transaction sizes.

3. design principles

1) One mg anchor. One token equals one mg silver for mental math and durable accounting.

2) Local autonomy with global guardrails. Satellites control day to day, Guardians enforce global

caps.

3) Transparency first. SHA-256 deposit proofs, time stamped records, on-chain events for deposit,

mint, redeem, shipping and policy changes.

4) Plain language. Slugs use lowercase descriptive names like cosigo\_queretaro. Avoid acronyms.

5) Practical logistics. Explicit min and max redemption, shipping flat and per gram cents, fee sinks

for operations.

4. architecture overview

Guardian at the global layer approves and configures Satellites and publishes global caps and

market signals.

Satellite at the town layer runs deposits, headroom, minting, redemptions, local fee routing, and

optional Guardian skim.

Burn sink and treasury act as final destinations for destroyed supply and reserves as defined by

policy.

4.1 guardian responsibilities

1) Satellite approval registry and fee sink registry.

2) Global fee and premium caps including maxGuardianFeeBps, maxMaintenanceFeeBps,

maxRedemptionFeeBps, maxPremiumBps.

3) Market signals including isMarketOpen and eoaOnlyEnabled and any additional anti-bot stances

as exposed by GuardianAware on Satellites.

4) Emergency posture and freeze authority under published rules.

Satellites call Guardian-aware view functions at transfer and redeem time to verify approvals,

skims, and caps.

4.2 satellite responsibilities

1) Identify town with slug and slugHash while name and symbol follow cosigo\_slug.

2) Register deposits with gramsMilli and docHash and depositor identity. Compute headroom as

deposited mg minus totalSupply minus pending redemptions minus fulfilled redemptions.

3) Mint strictly against headroom. Emit MintedAgainstHeadroom after each mint for auditability.

4) Enforce redemption policy with two function shapes. One accepts bytes32 shipping reference.

One accepts string and hashes to bytes32. Burn the redemption fee together with the requested mg

to simplify accounting. Mark the mg as pending so burns do not resurrect headroom.

5) Fee routing. First route local maintenance fee to feeSinkLocal. Then apply Guardian skim if

configured. Transfer remaining amount.

6) Floors and latch. Enforce that spot times one plus premium is greater than or equal to minFloor

to protect against underpriced redemption.

7) Safety. Pausable, blacklist, reentrancy guard. Param timelock for sensitive setters.

8) Guardian change. Two step proposal and accept with optional delay. A helper view verifies

Guardian binding by slug registry and approval in the Guardian.

5. token economics

5.1 peg

One token equals one mg silver. Minting is unlocked only by authenticated deposits measured in

grams then converted to mg accounting units.

5.2 egg anchor

For human scale intuition we keep an egg anchor. One fresh egg is approximately 77 to 88 mg

silver depending on the edition. Use 77 mg for precise math and 80 mg for quick mental math.

5.3 fees and shipping

1) maintenanceFeeBps routes to the local fee sink to fund operations.

2) redemptionFeeBps applies to physical redemption to cover processing, risk, and handling.

Common range five to fifteen percent.

3) premiumBps is an optional local premium above spot for retail fulfillment.

4) shippingFlatCents and shippingPerGramCents represent logistics cost and are included in

redemption quotes and settlement policy outside the token math.

5.4 price and floor latch

The contract stores spotMicroCentsPerMg and minFloorMicroCentsPerMg. Redemption may

proceed only if spot multiplied by one plus premium is greater than or equal to the floor. This guards

against redemptions under the project floor while still allowing community price discovery above the

floor.

6. flows

6.1 deposit

1) Custodian receives silver. Weighs accurately. Captures photographs. Obtains witness signatures

and a sworn deposition.

2) Documents are hashed and uploaded to IPFS. The docHash is recorded on-chain.

3) The Satellite registers the deposit which increases mint headroom.

6.2 mint

1) The minter role mints to the depositor or operator wallet strictly up to the computed headroom.

Headroom equals deposited minus supply and pending and fulfilled.

2) Each mint emits MintedAgainstHeadroom with remaining room for public audit.

6.3 redemption

1) A holder calls redeemPhysicalSilver with a milligram amount and shipping reference. The

contract enforces minimum redemption, per address and global daily caps, blacklist checks, market

hours if enforced by Guardian, and the floor latch using spot and premium versus minFloor.

2) The contract burns the mg plus the fee. It records a deterministic id keyed by account, amount,

reference and block. The mg becomes pending to prevent headroom resurrection.

3) The custodian fulfills with a tracking hash. On success, pending decreases and fulfilled

increases. If canceled with a reason, the net mg is minted back to the requester and pending

decreases. All transitions are evented for public audit.

7. governance and safety posture

1) Roles include default admin, pauser, custodian multi signature, and minter.

2) Param timelock seconds may be set. Sensitive setters enqueue and require time to elapse

before taking effect. This makes cross checks easier for the community and reduces risk of rushed

parameter flips.

3) Emergency pause allows quick freeze of transfers and or redemptions depending on

enforcement points.

4) Blacklist mapping provides a last resort for known malicious actors under published procedures.

5) Guardian oversight includes denial or revocation of satellite approval, caps enforcement, and

skim settlement. GuardianAware ensures checks happen during transfers and redemption.

8. developer appendix overview

The system uses minimal interfaces and common OpenZeppelin building blocks. Satellites inherit

ERC20, ERC20Permit, Pausable, ReentrancyGuard, AccessControl, and GuardianAware. The ABI

of the main contract and the latest Satellite implementation provided by the user inform the

following indexes and references for builders and auditors.

9. operator quickstart

9.1 deploy

1) Deploy Guardian. Publish addresses and caps.

2) Deploy Satellite with town slug and Guardian address. Record slug hash.

3) Grant local roles. Admin retains timelock control. Pauser for emergency. Custodian multi

signature. Minter account or module.

4) Set fee sink local and burn sink. Publish addresses on the Satellites directory.

9.2 initialize policy

1) Set maintenanceFeeBps, redemptionFeeBps, premiumBps within both local and Guardian caps.

2) Set spotMicroCentsPerMg and minFloorMicroCentsPerMg. Optionally enable market hours and

set UTC offset.

3) Configure daily redemption caps per address and global. Configure min redemption mg if

desired.

4) Configure anti bot if the Guardian exposes EOA only, gas price ceilings, same block transfer

delay, and cooldowns.

9.3 perform a first deposit and mint

1) Register deposit with gramsMilli and docHash and depositor.

2) Verify headroom equals deposited mg minus supply and pending and fulfilled.

3) Mint to depositor address with mintAgainstHeadroom. Confirm event log values.

9.4 redemption and fulfillment

1) Request redemption with shipping reference. Check that a PhysicalRedemptionRequested event

has accurate net and fee figures if using quote view on main. In Satellite the fee is burned and mg

pending recorded as mg1e18 in the mapping.

2) Fulfill with tracking hash. Confirm PhysicalRedemptionFulfilled. Verify mgPending decreased and

mgFulfilled increased.

3) If cancellation is necessary, call cancel and verify net mg is minted back while fee remains

burned for accounting integrity.

10. parameter reference derived from latest Satellite

All names use milligrams at 18 decimals unless noted.

10.1 identifiers

slug string lower case letters and digits only between four and thirty two characters.

slugHash keccak256 of the slug string.

10.2 economics and fees

maintenanceFeeBps. Local maintenance fee in basis points.

redemptionFeeBps. Redemption fee in basis points.

premiumBps. Premium in basis points used in floor checks.

feeSinkLocal. Address where maintenance fees are routed.

burnSink. Address whose balance is excluded from effectiveSupply and may receive burns.

10.3 price and floor

spotMicroCentsPerMg. Price oracle input stored on chain for policy checks.

minFloorMicroCentsPerMg. Minimum allowed effective price for the latch.

10.4 deposits and headroom

totalGramsMilliDeposited. Accumulator of gramsMilli across Deposit records.

deposits array includes gramsMilli, docHash, depositor, timestamp.

headroomMg1e18 view equals deposited mg minus totalSupply minus mgPending minus

mgFulfilled.

effectiveSupply excludes balances held in feeSinkLocal and burnSink.

10.5 redemption policy

dailyCapPerAddrMg1e18.

dailyCapGlobalMg1e18.

minRedemptionMg1e18.

redemptions mapping keyed by deterministic id storing account, mg, ref, fulfilled, cancelled.

mgPending1e18 and mgFulfilled1e18 track lifecycle totals.

10.6 roles and safety

DEFAULT\_ADMIN\_ROLE. PAUSER\_ROLE. CUSTODIAN\_ROLE. MINTER\_ROLE.

paramTimelockSeconds and paramEta mapping for queued parameter changes.

blacklisted mapping with event codes.

pause and unpause controls.

10.7 guardian control

guardian address from GuardianAware base. Guardian skim bps and sink are applied during

transfer by hook functions exposed in GuardianAware. A helper method assertGuardianBinding

checks that the Guardian slug registry points back to this Satellite and that the Satellite is approved.

11. event glossary synthesized from ABI and Satellite

This section lists common events to support indexers and auditors.

AddressUpdated. Label and old and new address.

AntiBotConfigUpdated. EOA only flag, transfer delay flag, max transaction gas price, cooldown

seconds for transfers and redemptions.

Approval. Standard ERC20 allowance update.

BlacklistUpdated or Blacklisted. Account and status code details.

BurnSinkUpdated or BurnSinkSet. Changes to burn sink address.

CustodianChanged and CustodianPending. Pending and completed custodian transfers.

DailyRedemptionLimitUpdated and RedemptionCapsSet. Updates to daily limits.

DepositRegistered. Grams or gramsMilli, docHash, timestamp and depositor.

EmergencyStopActivated and EmergencyStopDeactivated in main. Pause events in Satellite.

FeeSinkUpdated and FeeSinkLocalSet. Local fee sink updates.

MaintenanceFeeUpdated or MaintenanceFeeSet. Basis points change.

MarketConfigUpdated. Market hours enforcement flag and UTC offset minutes.

MinFloorCentsUpdated and MinFloorUpdatedMicro or MinFloorSet. Floor policy changes.

OwnershipPending and OwnershipTransferred. Owner change lifecycle on main.

Paused and Unpaused. Pause lifecycle.

PhysicalRedemptionRequested Fulfilled and Canceled on main. On Satellite

RedemptionRequested Fulfilled and Cancelled plus tracking or reason.

PremiumSet. Premium basis points updated.

RedemptionBandUpdated on main where min and max net mg are adjustable.

RedemptionFeeUpdated or RedemptionFeeSet. Basis points change.

Rescue. Rescue of foreign tokens.

RolesSet. Owner and custodian roles set.

ShippingFeesUpdated. Flat and per gram cents updated.

SpotUpdated and SpotUpdatedMicro or SpotSet. Spot policy updates.

TokensMinted and MintedAgainstHeadroom. Mint lifecycle.

Transfer. Standard ERC20 transfer.

GuardianFeeSkim. Satellite emitted when Guardian skim is applied.

12. function index synthesized from provided ABI

This index is for operator reference and auditor scoping on the main contract whose ABI was

provided.

Constructor accepts custodian address, initial deposit grams, initial deposit document string, and a

boolean indicating whether to pre mint full supply.

Administrative acceptances include acceptOwnership and acceptCustodian and their pending

getters.

ERC20 surface includes name, symbol, decimals, totalSupply, balanceOf, allowance, approve,

transfer, transferFrom, increaseAllowance, decreaseAllowance.

Deposit and headroom functions include registerDeposit with grams and docHash and depositor in

main or depositor implied in Satellite, getDepositsCount, totalDepositedGrams, remainingMintable

or \_mintableHeadroom and effectiveSupply and maxSupply on main.

Redemption surface includes redeemPhysicalSilver with string reference and an older overload with

bytes32 on Satellite, cancelPhysicalRedemption, fulfillPhysicalRedemption, getRedemptionsCount,

totalFulfilledNetMg, lastRedemptionDay, dailyRedemptionLimit and dailyRedemptionUsed on main.

Satellite exposes caps and counters through per address and global mappings.

Pricing and floor functions include setSpotMicroCentsPerMg, setMinFloorMicroCentsPerMg and

their legacy cents based variants, getCurrentFloorCents and spot getters.

Fees include setMaintenanceFeeBps, setRedemptionFee, setPremiumBps, setShippingFees and

fee sinks and burn sinks updates.

Market configuration includes setMarketHours and setMarketUtcOffsetMinutes and enforce flags on

main. Satellite checks Guardian signals through guardianChecks hook and optional local pause

controls.

Safety includes pause, unpause, emergencyStopContract on main, blacklist updates, rescue

ERC20, and role updates such as updateCustodian and transferOwnership and two step pending

acceptance.

13. anti bot and market hours policy

The main contract includes an AntiBotConfigUpdated event and setter for multiple flags in one call.

The inputs include EOA only enforcement, transfer delay enable, maximum transaction gas price in

wei, cooldown seconds for transfers, and cooldown seconds for redemptions. Operators should

record a standard operating window and log utcOffsetMinutes where markets are enforced. The

Satellite respects Guardian checks via GuardianAware guardianChecks modifier and the local

pause and blacklist maps.

14. gas and cost guidance

1) Small transfers. Keep maintenance fee basis points low to avoid user-visible erosion and to

preserve utility for micro transactions.

2) Price updates. Calling setters for spot and floor may cost significantly more gas than toggles

because they update storage used in multiple checks. Batch operational changes and avoid

frequent churn.

3) Redemptions. Encourage users to batch mg requests above the min redemption to amortize gas

and shipping over useful amounts of physical silver.

15. security assumptions and risk controls

1) Silver custody is real world and off chain. Follow chain of custody procedures, multi party

witnesses, and repeatable weighing standards.

2) Multi signature custody and pauser roles are strongly recommended.

3) Parameter timelocks should be set non zero for production satellites to allow community review

before policy shifts take effect.

4) Always publish deposit doc hashes and redemption tracking hashes for auditability.

5) Keep Guardian keys under multi signature wherever possible.

16. compliance kit outline

1) Sworn deposit statement with ID references to photos and video and scales. English and

Spanish editions.

2) Shipping and insurance disclaimer and risk acceptance. English and Spanish.

3) Redemption affidavit acknowledging net mg, fees, and shipping costs at the time of request.

4) Public registry entries on cosigo.io including satellite slug, addresses, custody names, and links

to Etherscan views of relevant events.

17. deployment checklist

1) Prepare Guardian parameters and publish caps.

2) Deploy Satellite with chosen slug and Guardian address. Verify name and symbol are cosigo

underscore slug.

3) Grant roles for custodian and minter and pauser. Record role transaction hashes.

4) Configure fee sink local and burn sink. Record addresses publicly.

5) Configure timelock seconds for sensitive setters.

6) Configure daily caps and min redemption.

7) Configure floor and spot and optional premium. Document rationale.

8) Execute first deposit with docHash and mint against headroom. Publish proofs.

9) Perform a test redemption cycle end to end with shipping reference and tracking hash.

10) Post satellite page entry on cosigo.io with all links and proofs.

18. sample policy numbers for testing only

This section is illustrative, not prescriptive.

1) maintenanceFeeBps equals 50 which is zero point five percent.

2) redemptionFeeBps equals 700 which is seven percent.

3) premiumBps equals 1000 which is ten percent.

4) minRedemptionMg1e18 equals 33345000000000000000000 which is 33,345 mg net or about

one troy ounce when fees are included.

5) dailyCapPerAddrMg1e18 equals 50000000000000000000000 which is 50,000 mg.

6) dailyCapGlobalMg1e18 equals 250000000000000000000000 which is 250,000 mg.

19. glossary

Cosigo. The project and its units where one unit equals one mg silver.

Guardian. Global policy and registry contract family.

Satellite. Local operational token contract per town with deposit and redemption logic.

Headroom. Deposited mg minus total supply minus pending mg minus fulfilled mg.

Fee sink. Address that receives maintenance fees.

Burn sink. Address whose balance is excluded from effective supply and may be a destination for

burned amounts.

Param timelock. A delay between queuing and applying sensitive parameter changes.

20. references and links

Project home. cosigo.io

Satellites directory. cosigo.io/Satellites

Reference contract source and code verification page. etherscan link listed at top of this document.

21. appendices

Appendix A. Selected main contract surfaces from ABI

Constructor

acceptCustodian

acceptOwnership

allowance

approve

balanceOf

burnFromSink

burnSink

cancelPhysicalRedemption

coldAddress

cooldownSecRedemptions

cooldownSecTransfers

custodian

dailyRedemptionLimit

dailyRedemptionUsed

decimals

decreaseAllowance

deposits

effectiveSupply

emergencyStop

emergencyStopContract

enforceEoaOnly

feeSink

fulfillPhysicalRedemption

getCurrentFloorCents

getDepositsCount

getRedemptionsCount

hotAddress

increaseAllowance

lastRedemptionDay

maintenanceFeeBps

marketHoursEnforced

marketOpenNow

marketUtcOffsetMinutes

maxRedemptionNetMg

maxSupply

maxTxGasPrice

minFloorCents

minFloorMicroCentsPerMg

minRedemptionNetMg

mint

name

owner

pause

paused

pendingCustodian

pendingOwner

premiumBps

quoteRedeem

redeemPhysicalSilver

redemptionFeeBps

redemptions

registerDeposit

remainingMintable

rescueERC20

resumeContract

setAntiBotConfig

setBurnSink

setColdAddress

setDailyRedemptionLimit

setFeeSink

setHotAddress

setMaintenanceFeeBps

setMarketHours

setMarketUtcOffsetMinutes

setMinFloorCents

setMinFloorMicroCentsPerMg

setPremiumBps

setRedemptionBand

setRedemptionFee

setShippingFees

setSpotCentsPerToken

setSpotMicroCentsPerMg

shippingFlatCents

shippingPerGramCents

spotCentsPerToken

spotMicroCentsPerMg

symbol

totalDepositedGrams

totalFulfilledNetMg

totalSupply

transfer

transferDelayEnabled

transferFrom

transferOwnership

unpause

updateBlacklist

updateCustodian

Appendix B. Selected error set from ABI for audit scoping

AlreadyPaused

AmountZero

BadRedemptionId

CannotBlacklistPrivileged

ContractPaused

DailyLimitExceeded

DocHashEmpty

EmergencyActive

EmergencyAlreadyActive

EmergencyNotActive

EoaOnly

ErrAllowance

ErrBlacklisted

ExceedsHeadroom

FeeTooHigh

FloorBelowLatch

GasPriceTooHigh

GramsZero

InsufficientBalance

MarketClosed

NetOutOfBand

NoChange

NoPendingCustodian

NoPendingOwner

NotCustodian

NotOwner

NotOwnerOrCustodian

NotPaused

NotPendingCustodian

NotPendingOwner

PendingCustodianBlacklisted

PendingOwnerBlacklisted

PremiumCannotLower

RedemptionAlreadyFinalized

RedemptionTooSoon

Reentrant

RescueFailed

SameAddress

ShippingFlatTooHigh

ShippingPerGramTooHigh

SpotNotSet

TransferSameBlock

TransferTooSoon

UtcOffsetOutOfRange

ZeroAddress

Appendix C. Public state variables and setters in latest Satellite

maintenanceFeeBps and setMaintenanceFeeBps with local and Guardian caps

redemptionFeeBps and setRedemptionFeeBps with caps

premiumBps and setPremiumBps with caps

feeSinkLocal and setFeeSinkLocal

burnSink and setBurnSink

minRedemptionMg1e18 and setMinRedemption

spotMicroCentsPerMg and setSpotMicroCentsPerMg

minFloorMicroCentsPerMg and setMinFloorMicroCentsPerMg

dailyCapPerAddrMg1e18 and dailyCapGlobalMg1e18 with setRedemptionCaps

paramTimelockSeconds with setParamTimelock and paramEta queues

guardian change with proposeGuardian acceptGuardian setGuardianChangeDelay

grantLocalRoles for custodian multi signature and minter

recoverERC20 for foreign tokens

blacklist map with setBlacklist

pause and unpause

headroom view and effectiveSupply view

Appendix D. deployment transactions to record publicly

1) Contract creation transactions for Guardian and Satellite

2) Role grants with transaction hashes

3) Fee sink and burn sink set transactions

4) Timelock configuration transaction

5) Deposit registration transactions with docHash values

6) First mint transaction with MintedAgainstHeadroom event

7) First redemption request and fulfillment or cancel transactions

8) Link to your satellite entry on cosigo.io/Satellites with the above proofs

Appendix E. example human readable parameters

This section shows the shape of values only.

maintenanceFeeBps equals fifty

redemptionFeeBps equals seven hundred

premiumBps equals one thousand

minRedemptionMg1e18 equals 33345000000000000000000

dailyCapPerAddrMg1e18 equals 50000000000000000000000

dailyCapGlobalMg1e18 equals 250000000000000000000000

Appendix F. future work

1) External security review focused on headroom accounting and redemption lifecycle

2) Guardian transparency dashboard that surfaces satellite approvals, fee sinks, caps, and skim

events

3) Community owned Guardian via multisig and timelock parameters

4) Satellite factory with parameter presets and compliance checklist generation in English and

Spanish