

QANOVRA

DEVELOPER DOCUMENTATION v1.1

Developer, Operator and Verification Reference for the Qanovra Layer-1 Ecosystem

v1.1-FreezeFix / Final Freeze Candidate

24 August 2026

Document control	Canonical value
Status	FreezeFix / Final Freeze Candidate - external v1.1 re-audit PASS; delta confirmation pending
Protocol baseline	Qanovra L1 v3.7.5 - Frozen L1
Native asset	QNV
Primary language	English - canonical developer-documentation language
Public integration target	qanovra.org/docs
Design family	Aligned to Qanovra Whitepaper / Yellow Paper v2.4.2 publication layout
Supersedes	Developer Documentation v1.1 Re-Audit Candidate
Re-audited package SHA-256	27505fde3d29cfaa629f059f7378371011bcbb8610df1106050b61cfb714474
Audited v1.0 predecessor DOCX SHA-256	3ef8772e2178a563c40cc568b20f6d89663ff3f9dcbc1cd7d9fb5eca9b5d5b40

SCOPE This document explains how to learn, build, operate and verify Qanovra using only evidenced interfaces and controlled publication data. The external v1.1 re-audit returned PASS. This FreezeFix closes the two remaining N-level text findings; it does not convert software evidence into invite-testnet, production, legal or security approval.

0. Document Control

Purpose

A public, task-oriented technical handbook for developers, node operators, validator operators, auditors and machines. The document is intended to be converted into the website information architecture without creating a second source of truth.

Authority order	Source
1	Frozen implementation and canonical configuration
2	Canonical network/release registry
3	OpenAPI, schemas and conformance vectors
4	Generated Python and TypeScript SDKs
5	This Developer Documentation
6	Marketing/homepage summaries

NO GUESSING No production hostname, API field, system minimum, CLI flag, timeout, protocol value or SDK convenience method may be invented to make a page appear complete. Unknown or not-yet-published values are presented as explicit status.

Documentation contract

Every task page answers what to do, how to do it, what success looks like, how to verify it, and when to stop.

Every executable-looking block is labelled TESTED, ILLUSTRATIVE or PSEUDOCODE.

TESTED means executed in a stated controlled environment; it does not imply production-network qualification.

English is the canonical public developer-documentation language. Translations are derived and non-normative.

PASS, CLOSED and FROZEN always include the applicable scope or evidence line.

Revision history

Version	Date	Status	Change
v1.0	24 Aug 2026	External audit input	Initial documentation architecture; audited SHA-256 pinned above.
v1.1	24 Aug 2026	Re-Audit PASS	Path B remediation externally verified: 22/22 prior findings closed or fail-safe treated; no critical or medium findings.
v1.1-FreezeFix	24 Aug 2026	Final Freeze Candidate	Closes R11-N1/R11-N2; converts callout tables to semantic paragraph blocks; aligns visual system with Whitepaper/Yellow Paper v2.4.2.

1. Start Here

Choose the outcome you need. The same structure is used on the future web version.

- LEARN** Understand Qanovra, QNV, blocks, transactions, validators and finality. [Open section](#)
- BUILD** Use the public API and generated SDKs. [Open section](#)
- RUN** Operate node and validator infrastructure with fail-closed controls. [Open section](#)
- VERIFY** Check network identity, releases, manifests and reproducibility independently. [Open section](#)
- REFERENCE** Read exact constants, API status, releases, errors and machine data. [Open section](#)

Fast answers

Question	Answer
What network identity must I pin?	Chain ID qnv-78cefe355c7fb55c53dd173c and the full v3.7.5 genesis SHA-256.
Is there a public production API URL?	No public base URL is published in this candidate. Invite Testnet is HOLD / NOT AUTHORIZED and Public Mainnet is NO-GO.
Which application SDKs are evidenced?	Python and TypeScript. Go is not claimed as an application SDK.
Can I submit a transaction through the documented public surface?	No dedicated generated L1 write method is evidenced. The public documentation is read-oriented until a release-specific write contract is published.
Where do I report a vulnerability?	security@qanovra.org

2. Learn Qanovra

Qanovra in five minutes

Layer 1

Qanovra is an independent Layer-1 protocol. The current technical baseline is v3.7.5 Frozen L1.

Network identity

The network is bound to its genesis identity. Clients must pin both Chain ID and genesis SHA-256 before trusting an environment.

Ledger

The authoritative account state is the ledger. Protocol-controlled vaults are separated from ordinary accounts.

QNV

QNV is the native asset. It uses five decimal places; 1 QNV equals 100,000 qnv-atom.

Validators

Validators secure the network under protocol and operator-policy rules. Founding Validator admission requires 100,000 QNV operator-owned self-stake.

Verification

Releases are treated as evidence-bearing artifacts: hashes, manifests, provenance and reproducibility are part of the operational model.

Canonical constant	Value
L1 version	v3.7.5
Genesis SHA-256	78cefe355c7fb55c53dd173c9ca9d340306054a7fd5718e70d1d3193be7b93a3
Chain ID	qnv-78cefe355c7fb55c53dd173c
Asset	QNV
Base unit	qnv-atom
Decimals	5
Atoms per QNV	100,000
Maximum supply	10,000,000,000 QNV / 1,000,000,000,000,000 qnv-atom

DERIVATION NOTE The canonical papers state that Chain ID is derived from the genesis hash. This document does not invent the exact truncation/formatting algorithm. Until that rule is explicitly proven from the normative implementation/specification, clients compare the published Chain ID and genesis SHA-256 as two pinned trust inputs.

3. Environment & Network Identity

Current environment status

Environment	Status	API base URL	What developers may assume
Controlled documentation fixture	AVAILABLE	Local file / local test harness only	Suitable for TESTED documentation examples; not a network qualification.
Invite Testnet	HOLD / NOT AUTHORIZED	NOT PUBLISHED	Do not build production dependencies around an assumed hostname.
Public Mainnet	NO-GO	NOT PUBLISHED	No public-mainnet availability claim is made.

Task 1 — Verify network identity out of band

ANSWER Compare the machine-readable network record against the pinned v3.7.5 Chain ID and genesis SHA-256 before accepting an environment as Qanovra.

Prerequisites

A copy of the publication-controlled network.json generated from the canonical registry.

Python 3, jq, or another JSON parser.

DO

1. Fetch or obtain the publication-controlled network.json through an authenticated documentation/release channel.
2. Compare network.chainId with the pinned Chain ID.
3. Compare network.genesisSha256 with the full pinned genesis SHA-256.

4. Fail closed on either mismatch; do not enable signing or validator activation.

EXAMPLE STATUS TESTED — controlled fixture, 24 Aug 2026

EXPECT

Both values match exactly. A successful controlled-fixture run prints: PASS: Qanovra network identity matches the pinned v3.7.5 baseline.

VERIFY

The v1.1 build executed the supplied `verify_network_identity.py` against the publication-candidate `network.json` and received exit code 0. This tests the comparison procedure, not a live endpoint.

STOP CONDITIONS

Chain ID mismatch.

Genesis SHA-256 mismatch.

`network.json` is unavailable, malformed, unsigned/untrusted in its delivery context, or not bound to the current docs release.

REFERENCE

Canonical L1 v3.7.5 baseline.

Machine record format in section 14.

External audit finding DD-K1 remediation.

```
python verify_network_identity.py network.json
# PASS: Qanovra network identity matches the pinned v3.7.5 baseline
```

TESTED — controlled fixture, exit code 0

WHY OUT OF BAND The audited public API family does not currently evidence a dedicated network-identity endpoint. Therefore the v1.1 trust bootstrap uses the canonical `registry/network.json` path instead of inventing `GET /v1/network` or a similar endpoint.

4. Build — First API Read

CURRENT PUBLIC SCOPE The currently evidenced developer surface is read-oriented. Generated SDK evidence retains zero dedicated L1 write methods. A generic `/v1/rpc` relay exists in the audited SDK/OpenAPI line but is server-permission-controlled and must not be documented as a public transaction-submit contract without release-specific evidence.

Known operation families

Family	Evidenced path or behavior	Publication treatment
Explorer blocks	<code>GET /v1/explorer/blocks</code>	Public read reference.
Explorer block lookup	<code>GET /v1/explorer/blocks/{identifier}</code>	Public read reference.
Explorer transaction lookup	<code>GET /v1/explorer/transactions/{tx_hash}</code>	Public read reference.
Developer webhooks	Subscription endpoints, including DELETE by <code>subscription_id</code>	Developer-authenticated; schema generated from OpenAPI.
Credential lifecycle	<code>/v1/developer/rotate</code> and <code>/v1/developer/redeem</code>	Security-sensitive; no secret echoing.
Generic relay	<code>/v1/rpc</code>	Permission-controlled relay; not a documented public L1-write method.

Task 2 — Make the first read request

ANSWER Read the latest block list from an authorized Qanovra API environment after network identity has been pinned.

Prerequisites

An API base URL published by the specific environment/release.

Network identity verification completed.

Credential if required by that environment.

DO

1. Set QANOVRA_API_URL only from the current release/environment record.
2. Call GET /v1/explorer/blocks with an optional limit parameter.
3. Check transport status and Content-Type.
4. Validate the body against the success response schema in the matching public OpenAPI.

EXAMPLE STATUS ILLUSTRATIVE — no public network URL is currently authorized

EXPECT

A successful request returns HTTP 2xx with application/json and a body conforming to the release-matched OpenAPI schema. Exact fields are generated from OpenAPI rather than duplicated here.

VERIFY

Validate the endpoint path and response schema against public OpenAPI SHA-256 604d9bbd5948b330c9d0c7add41affcab396b30b545dea3c15f0b942b63f17f3 for the RC12 v3.7.5 repin line.

STOP CONDITIONS

No release-published API base URL exists.

Network identity has not been verified.

TLS validation fails.

Response schema differs from the matching OpenAPI.

REFERENCE

Public OpenAPI: docs/public-openapi-v0.13.0-rc12.json.

OpenAPI SHA-256 above.

SDK section 06.

```
export QANOVRA_API_URL="<<release-published-base-url>"
curl --fail --silent --show-error "$QANOVRA_API_URL/v1/explorer/blocks?limit=1"
```

ILLUSTRATIVE — execute only after a base URL is published

5. API Reference & Errors

Official terminology

API is the canonical documentation term for the public HTTP/OpenAPI surface. “RPC” is retained only as a search alias and for the specifically named generic /v1/rpc relay.

Reference source	SHA-256
Full OpenAPI v0.13.0 RC12 / v3.7.5 repin	5416f3abeaca08f6178dbAAF79c0fe226f2e895d889f039aa555c5988982b4b7
Public OpenAPI v0.13.0 RC12 / v3.7.5 repin	604d9bbd5948b330c9d0c7add41affcab396b30b545dea3c15f0b942b63f17f3

Endpoint-page contract

Purpose in one sentence.
 Method and stable path.
 Authentication and scope.
 Typed path/query parameters.
 Request body schema when applicable.
 Success response schema.
 Structured error responses and retry guidance.
 Copyable HTTP plus generated SDK examples.
 Applicable API/protocol versions.
 OpenAPI source digest.

Transport and structured-error behavior

Condition	Required client behavior
Non-HTTPS remote base URL	Reject by default. Localhost HTTP requires explicit opt-in in the audited SDK line.
Redirect	Refuse; do not forward x-api-key to redirect target.
Missing required path parameter	Reject locally before malformed network request.
Timeout	Audited SDK line: 10-second default.
Response size	Audited SDK line: 4 MiB maximum body.
Success content type	Require JSON for JSON success responses.
API error	Preserve HTTP status, Qanovra machine code and raw/detail body when supplied.
Array query values	Serialize as repeated query values according to the API contract.

HTTP class	Meaning	Default response
2xx	Accepted by HTTP/API layer	Still validate payload semantics/schema.
401	Authentication missing/invalid	Do not retry unchanged credentials.
403	Authenticated but unauthorized	Check role/scope; never bypass policy.
429	Quota/abuse control	Respect retry guidance and backoff.
5xx	Server/dependency failure	Retry only when operation safety/idempotency permits.

ERROR-CODE SOURCE Qanovra-specific machine error codes must be generated from the canonical API/implementation registry. v1.1 does not fabricate an error-code catalogue where a current canonical registry was not supplied.

6. SDKs

Supported application SDKs

Language	Status	Notes
Python	Evidenced generated SDK	Deterministic generation from public OpenAPI; package contains typing metadata.
TypeScript	Evidenced generated SDK	Deterministic generation; compiled package/declarations evidenced.
Go	Not claimed as an application SDK	Go is evidenced as qnv-verify, a standalone verifier.

Audited generated Python signatures

```
get_v1_explorer_blocks(*, limit: int | None = None, api_key: str | None = None) -> Any
get_v1_explorer_blocks_identifier(identifier: str, *, api_key: str | None = None) -> Any
```

ILLUSTRATIVE — exact signatures independently audited in RC8; current generated package remains authoritative

SDK PINNING A website page must show the exact SDK release and OpenAPI digest used to generate it. Do not copy method names between versions by hand.

Task 3 — Pin SDK and OpenAPI together

ANSWER Use an SDK only with the OpenAPI digest and release subject it was generated from.

Prerequisites

Release-matched SDK package.
 GENERATION-MANIFEST.json from the same package.
 Matching public OpenAPI document.

DO

1. Hash the OpenAPI file.
2. Compare the digest to the SDK generation manifest.
3. Confirm the manifest binds the current v3.7.5 subject.
4. Confirm generated L1 write methods remain exactly as stated by that release.

EXAMPLE STATUS ILLUSTRATIVE — exact release package required for execution

EXPECT

The OpenAPI digest, SDK manifest digest and protocol binding agree. If any one differs, the integration is not release-pinned.

VERIFY

For the current RC12 v3.7.5 repin line, the public OpenAPI digest recorded for documentation is 604d9bbd5948b330c9d0c7add41affcab396b30b545dea3c15f0b942b63f17f3.

STOP CONDITIONS

OpenAPI hash mismatch.

SDK was regenerated from a different schema.

Protocol binding is not v3.7.5.

An application write method appears without release evidence.

REFERENCE

sdk/GENERATION-MANIFEST.json

tools/generate_sdks.py

Public OpenAPI digest table above.

7. Transactions & QNV

Transaction lifecycle

Stage	Developer meaning
Create	Construct intended operation under the applicable protocol/schema.
Canonicalize	Serialize according to the canonical contract.
Sign	Authorize with the appropriate account/validator key outside documentation.
Submit	Send only through a release-authorized write surface.
Validate	Protocol/API checks admission.
Include	Transaction appears in a block candidate.
Finalize	Use the normative finality condition before treating settlement as final.

CURRENT SCOPE The current public developer documentation does not publish a dedicated transaction-submit API. The lifecycle is taught to explain the protocol path; submit instructions will be added only when a release-specific public write contract and authorization model are evidenced.

Working with amounts

```
# TESTED conversion rule
qnv_atoms = qnv_amount * 100000
max_supply_atoms = 100000000000 * 100000
assert max_supply_atoms == 10000000000000000
```

TESTED — arithmetic invariant

AMOUNTS Protocol amounts must be represented with integer base units where exactness matters. Do not use binary floating-point for consensus-critical or accounting-critical conversions.

8. Run — Node Operations

Operator sequence

Phase	Required evidence before next phase
1 Verify release	Artifact hash and release subject match.

Phase	Required evidence before next phase
2 Verify network	Chain ID + genesis pinned out of band.
3 Configure	Configuration is release-qualified and role-appropriate.
4 Start	Process/service starts without bypassing security controls.
5 Sync	Node converges to expected chain/state.
6 Monitor	Health, peers, storage and finality telemetry are observed.
7 Upgrade	Versioned artifact + rollback point + post-start verification.
8 Recover	Restore plus state-root/identity verification before service re-entry.

Task 4 — Operate the evidenced validator service unit

ANSWER The current operator package evidences deployment/systemd/qanovra-validator.service. This section does not invent a separate qanovra-node.service.

Prerequisites

- Qualified operator package installed.
- Service file reviewed against the exact release.
- Secrets supplied through approved credential mechanisms rather than embedded in the unit.

DO

1. Inspect the unit before installation.
2. Install using the operator package procedure.
3. Start only after network identity and key custody checks pass.
4. Read service status and logs.
5. Keep signing disabled until validator-specific prerequisites are complete.

EXAMPLE STATUS ILLUSTRATIVE — unit is evidenced; command execution depends on operator host

EXPECT

The service is active, connected to the intended network, and produces the expected release-specific health/telemetry evidence.

VERIFY

Verify service configuration against deployment/systemd/qanovra-validator.service and deployment/env/validator.env.example from the same operator package.

STOP CONDITIONS

- Wrong network identity.
- Release digest mismatch.
- Missing HSM/KMS custody dependency for a signing role.
- State-root or recovery mismatch.

REFERENCE

deployment/systemd/qanovra-validator.service

deployment/env/validator.env.example

P1-04 incident runbook line.

```
sudo systemctl daemon-reload
sudo systemctl start qanovra-validator
sudo systemctl status qanovra-validator
journalctl -u qanovra-validator --no-pager -n 200
```

ILLUSTRATIVE — standard systemd interaction with evidenced service name

Public hardware minima

Resource	v1.1 public status
CPU	NOT PUBLISHED — no canonical numeric minimum located in the supplied evidence.
RAM	NOT PUBLISHED — no canonical numeric minimum located in the supplied evidence.
Storage	NOT PUBLISHED — capacity/performance minimum must come from operator qualification evidence.
Bandwidth	NOT PUBLISHED — must come from qualified topology/operator evidence.
OS/runtime	Use only release-qualified operator/runtime matrices; do not infer a production minimum from documentation examples.

ADMISSION RULE Missing public hardware minima are not filled with industry defaults. Production validator admission remains controlled and evidence-bound; 0/5 founding operators are admitted in the current MAINBOARD status.

9. Validators

Policy item	Current rule
Founding cohort target	5 independent Founding Validator Operators.
Production target	At least 7 independent validators.
Minimum operator-owned self-stake	100,000 QNV per validator.
Bootstrap delegation	Disabled for the Founding Validator phase.
Qanovra-operated founding validator	None.
Key custody	Operator controls signing infrastructure and must retain exclusive signing authority.

Task 5 — Enable validator signing safely

ANSWER Enable signing only after release, network identity, operator admission, key custody, synchronization and monitoring evidence are all satisfied.

Prerequisites

Exact release verified.

Chain ID and genesis SHA-256 verified.

Operator-controlled key generation/custody complete.

Validator synchronized and connectivity verified.

Monitoring and recovery prerequisites satisfied.

Applicable admission state permits activation.

DO

1. Confirm every prerequisite in the operator admission/evidence package.
2. Confirm HSM/KMS/provider identity and key handle on operator-controlled infrastructure.
3. Confirm no Qanovra-held private key or master secret exists.
4. Confirm chain/state synchronization.
5. Record the activation evidence and only then enable signing according to the operator package.

EXAMPLE STATUS ILLUSTRATIVE — production activation remains an external gate

EXPECT

Signing is active only on the intended network and under the intended operator-controlled key. Evidence identifies release, network identity and activation event.

VERIFY

Review operator ceremony record, service state, current network identity, telemetry and admission state as one evidence set.

STOP CONDITIONS

Any network identity mismatch.

HSM/KMS unavailable or key handle differs.

Validator not synchronized.

Recovery/state-root mismatch.

Operator admission or production gate not satisfied.

REFERENCE

Founding Validator Policy v1.4.

P0-04/P0-05 live HSM and ceremony gates.

Current MAINBOARD.

Maintenance

Founding Validator policy requires planned maintenance notice and coordination so two founding validators are not intentionally taken down together. Operator-specific execution must follow the current controlled agreement/runbook.

10. Security & Key Custody

Security domain	Protects	Do not confuse with
Protocol security	Consensus/state invariants	Application permissions.
Application authorization	API scopes and credentials	Validator/private-key custody.
Operational custody	HSM/KMS/validator signing authority	Software simulation or dry run.
Build integrity	Artifact provenance/reproducibility	Runtime correctness after deployment.

Security domain	Protects	Do not confuse with
Evidence integrity	Hashes, manifests, attestations	Proof that an external real-world event actually happened.

KEY BOUNDARY Qanovra must not generate, receive, transmit, store or possess a Founding Validator operator's private keys, recovery seeds, HSM master secrets, KMS master credentials or material sufficient to reconstruct signing authority.

Provider examples in operator material

PKCS#11 configuration example

AWS CloudHSM example

Azure Cloud HSM example

Google Cloud HSM example

SoftHSM configuration explicitly for test-only use

REAL-WORLD BOUNDARY Provider example files, dry runs and software PASS states are not evidence of a live non-exportable production key. P0-04 and the production P0-05 ceremony remain separate gates until executed and attested.

Report a vulnerability

Security reports: security@qanovra.org

Describe the affected component and exact version.

Include reproducible steps and impact without exposing third-party secrets.

Do not publish active credentials, private keys or customer data.

Use coordinated disclosure; publication timing is agreed with the security contact.

PUBLICATION TARGET The website publication set must expose `/.well-known/security.txt`. No PGP key is claimed here because no canonical public key was supplied for this documentation release.

11. Monitoring & Incident Response

Monitor the conditions that answer operational questions, not only process liveness.

Area	What must be observed
Network	Peer/connectivity health and divergence indicators.
Chain	Block progress, finality/state progression and expected network identity.
Host	CPU, memory, storage and I/O against the qualified operator profile.
API	Availability, latency, quota responses and structured errors.
Certificates	mTLS/certificate validity, revocation and expiry.
Backup	Last successful backup, integrity verification and restore readiness.
Validator	Signing availability, synchronization and policy/admission state.

Task 6 — Diagnose a node that is not progressing

ANSWER Determine whether the failure is identity, connectivity, synchronization, storage, certificate or release-related before restarting blindly.

Prerequisites

Current release identity known.
Read access to service logs and monitoring.

DO

1. Re-check Chain ID and genesis pin.
2. Confirm service process and release digest.
3. Inspect peer/connectivity state.
4. Inspect block/state progression and divergence indicators.
5. Check storage capacity/I/O and certificate status.
6. Escalate using the current incident runbook if state is uncertain.

EXAMPLE STATUS ILLUSTRATIVE — exact metric names remain generated from release definitions

EXPECT

A single cause class is identified or the node is safely isolated pending deeper investigation.

VERIFY

Post-remediation, repeat network identity, synchronization and health checks before returning the node/validator to service.

STOP CONDITIONS

State/root mismatch.
Unknown release binary.
Certificate trust failure.
Evidence is insufficient to establish the intended chain.

REFERENCE

P1-04 incident runbooks.
Observability and SLO definitions from the operator package.

12. Backup & Recovery

Task 7 — Restore and re-enter service

ANSWER A restore is successful only when archive integrity, network identity, restored state and post-start synchronization all verify.

Prerequisites

Approved backup/archive.
Expected digest/manifest.
Fresh or isolated recovery host.
Release-matched configuration.

DO

1. Verify backup digest and manifest before extraction.
2. Restore using the current operator recovery procedure.
3. Start without enabling validator signing.
4. Verify Chain ID and genesis identity.
5. Verify restored state root/checkpoint according to the recovery evidence.
6. Allow synchronization and compare expected chain progression.
7. Enable service/signing only after all checks pass.

EXAMPLE STATUS ILLUSTRATIVE — real geographic DR remains an external gate

EXPECT

The recovered node matches the intended network identity, state verification succeeds, synchronization resumes and no unexplained divergence remains.

VERIFY

Record archive digest, release digest, restored state/root evidence and post-start network identity as one recovery record.

STOP CONDITIONS

Backup hash mismatch.
 State-root/checkpoint mismatch.
 Wrong Chain ID/genesis.
 Unexplained divergence after synchronization.

REFERENCE

Qanovra recovery orchestration.
 P0-07 multi-region/DR gate.
 P1-04 incident runbooks.

FAIL CLOSED Process started is not equivalent to recovered. State verification and network identity are mandatory before return to service.

13. Verify Releases Independently

SHA-256 verification

```
# Linux
sha256sum <artifact>

# macOS
shasum -a 256 <artifact>

# Windows PowerShell
Get-FileHash <artifact> -Algorithm SHA256
```

TESTED — command forms exercised in the documentation build; compare output to the release manifest

Task 8 — Verify a release artifact

ANSWER Hash the exact downloaded bytes and compare the result with the release SHA256SUMS before execution or installation.

Prerequisites

Artifact and release SHA256SUMS obtained through the intended publication channel.

DO

1. Compute SHA-256 locally.
2. Compare all 64 hexadecimal characters with the manifest entry.
3. Verify the manifest belongs to the same release and artifact name.
4. Reject on mismatch; redownload only from the intended channel.

EXAMPLE STATUS TESTED — hashing workflow validated; cited release digest comes from canonical repin evidence

EXPECT

Local digest equals the release manifest digest exactly.

VERIFY

For the Parallel Ecosystem RC12 v3.7.5 repin closure archive, the recorded SHA-256 is e7ee09ad5377a316ef9650c991c69418dfb117aa4f53679d0b5068449083820c.

STOP CONDITIONS

Any digest mismatch.
 Manifest/version mismatch.
 Artifact changed after manifest generation.

REFERENCE

Release SHA256SUMS.
 Build manifest/provenance.
 Reproducible-build evidence where applicable.

Standalone verifier

qnv-verify is the evidenced Go standalone verification implementation. It is not an application SDK. Its release-specific command syntax must be published from the exact verifier build; this document does not invent flags.

RC12 software-freeze carry-forward binary	SHA-256
qnv-verify-linux-amd64	db49e222dfda6f4161fcb229f9f41aa5e82b5f477cb972fbc0868f4b3336957b
qnv-verify-linux-arm64	df888c1f23e5c45b1fec668a367380cd6a70aca442c52c73e0d38c0564c227ef
qnv-verify-windows-amd64.exe	eab58219554fd7f5de5e590fa9623d588a325432ef38dbea9fba5660d3765c1b

CHECKSUM SCOPE These qnv-verify digests are explicitly labelled as RC12 software-freeze carry-forward evidence. The website must bind downloadable verifier binaries to the exact current release manifest before presenting them as current.

Reproducible builds

Result class	Meaning
MATCH	Independent build outputs/defined trees match.
DIFFERENCE EXPLAINED	Difference exists and has a documented reproducible cause.
UNEXPLAINED DIFFERENCE	Fail closed; no reproducibility claim.
TOOLCHAIN MISMATCH	Evidence does not qualify the intended build environment.

CURRENT EVIDENCE The project records P1-03 as technical PASS with matching Windows/macOS runtime/vendor trees, while final observer attestation remains a separate formal gate in the current MAINBOARD. Do not generalize a build match into mainnet approval.

14. Machine Interface

Publication targets

Path	Purpose	Status in this document
/docs/reference/network.json	Canonical network/asset reference generated from registry.	Candidate content defined below.
/openapi.json	Public API contract or redirect to current canonical OpenAPI.	Must bind exact release digest.
/.well-known/qanovra.json	Discovery record for canonical docs/network/API metadata.	Publication target.
/.well-known/security.txt	Security reporting metadata.	Publication target.
/llms.txt	Concise canonical index for AI/retrieval systems.	Publication target.
/llms-full.txt	Expanded documentation index/content map.	Publication target.

Canonical network record

```
{
  "schema": "qanovra-network-reference-v1",
  "protocol": {
    "name": "Qanovra L1",
    "version": "3.7.5",
    "state": "Frozen L1"
  },
  "network": {
    "chainId": "qnv-78cefe355c7fb55c53dd173c",
    "genesisSha256": "78cefe355c7fb55c53dd173c9ca9d340306054a7fd5718e70d1d3193be7b93a3"
  },
  "asset": {
    "symbol": "QNV",
    "baseUnit": "qnv-atom",
    "decimals": 5,
    "atomsPerQnv": "100000",
    "maxSupplyAtoms": "10000000000000000",
    "displayMaxSupplyQnv": "10000000000"
  }
}
```

TESTED — parsed and identity-validated in controlled documentation fixture

UNIT SAFETY maxSupplyAtoms is explicit and unambiguous. displayMaxSupplyQnv is presentation data. Numeric quantities are strings where cross-language integer safety matters.

security.txt publication content

Contact: mailto:security@qanovra.org
 Expires: <RFC3339 timestamp; no later than publication date + 1 year>
 Preferred-Languages: en, de
 Policy: <publication-controlled-security-policy-url>
 Canonical: https://qanovra.org/.well-known/security.txt

ILLUSTRATIVE - RFC 9116 model; Expires and Policy must be bound to the actual publication date/route before deployment

llms.txt minimum content model

Qanovra
 Canonical developer docs: https://qanovra.org/docs
 Canonical network reference: https://qanovra.org/docs/reference/network.json
 Canonical API contract: https://qanovra.org/openapi.json
 Protocol baseline: Qanovra L1 v3.7.5
 Rule: machine values come from canonical registry/OpenAPI sources, not from prose summaries.
 Public network status: Invite Testnet HOLD / NOT AUTHORIZED; Mainnet NO-GO.

ILLUSTRATIVE — publication-ready content model; URLs become live evidence only after deployment

Task 9 — Consume machine-readable network data

ANSWER Read exact network/asset constants from network.json rather than scraping prose.

Prerequisites

Publication-controlled network.json.
 Schema version supported by the consumer.

DO

1. Parse schema and protocol.version.
2. Read network.chainId and network.genesisSha256.
3. Read asset base-unit fields.

4. Apply the same fail-closed identity comparison as Task 1.

EXAMPLE STATUS TESTED — controlled JSON fixture

EXPECT

The consumer obtains the same values shown in the human Network Reference with no hidden second data model.

VERIFY

The documentation build parses the candidate JSON and validates both identity values.

STOP CONDITIONS

Unknown schema version.

Missing identity field.

Mismatch against pinned release metadata.

REFERENCE

Task 1.

Network Reference.

Docs Gate.

15. Troubleshooting

Symptom	First checks	Stop/escalate when
API 401	Credential presence, release/environment, clock only if auth scheme requires it	Do not retry unchanged secret.
API 403	Role/scope and endpoint policy	Do not bypass authorization.
API 429	Retry guidance/quota	Do not create retry storms.
Node will not sync	Network identity, peers, release, storage, certificate	Identity/state divergence.
Validator not signing	Admission, HSM/KMS, sync, policy state	Never “fix” by bypassing custody or fail-closed guards.
Hash mismatch	Source channel, artifact name, manifest version	Discard artifact.
Restore starts but state differs	Checkpoint/state root and network identity	Recovery failed; keep isolated.
SDK behaves differently from docs	SDK version + OpenAPI digest	Treat docs drift as release defect.

16. Releases, Compatibility & Deprecation

Current reference lines

Component	Current referenced status
Qanovra L1	v3.7.5 Frozen L1

Component	Current referenced status
Wallet	v0.30.17 Final Freeze Evidence Closure — documentation reference line
Parallel Ecosystem	v0.13.0 RC12 software freeze; operative v3.7.5 repin
Founding Validator Policy	v1.4 freeze-attested document line
Developer Documentation	v1.1-FreezeFix Final Freeze Candidate - re-audit PASS, delta confirmation pending

SEPARATION Moving product release lines here keeps Network Reference restricted to network/protocol constants.

API lifecycle policy

The /v1 path denotes the current major public API contract where present in the canonical OpenAPI.

Deprecation must be announced in versioned release notes and endpoint/reference metadata before removal.

A deprecation notice must name the replacement or explicitly state that no replacement exists.

The exact sunset date/window is release-specific and must be published before removal; this v1.1 document does not invent a global duration that has not been approved.

Removal requires a versioned release, updated OpenAPI, regenerated SDKs, migration notes and a passing docs gate.

A breaking change may not be hidden behind an unchanged canonical OpenAPI digest.

Compatibility matrix status values

Value	Meaning
QUALIFIED	Exact versions were tested/evidenced together.
SUPPORTED	Contract states support but no stronger qualification is implied.
NOT QUALIFIED	No claim is made; do not infer compatibility.
DEPRECATED	Still present under a published deprecation notice.
REMOVED	No longer present in the referenced release.

17. Documentation Gate

Documentation is part of the release. The gate fails when executable documentation drifts from canonical machine sources.

Gate check	Fail condition
Network constants	Displayed values differ from registry/network.json.
API paths	Executable path is not in the release-matched OpenAPI.
SDK methods	Displayed method is not generated/evidenced for the selected SDK release.
Example marker	Executable-looking code lacks TESTED / ILLUSTRATIVE / PSEUDOCODE.

Gate check	Fail condition
Base URLs	Placeholder/fabricated production hostname appears as usable.
Machine record	Units/schema are ambiguous or differ from human reference.
Security	security@qanovra.org / security.txt publication contract missing.
Metadata	DOCX/PDF metadata does not match the publication build.
Links/bookmarks	Internal or public navigation target is broken.
Scope status	Documentation implies Invite Testnet/Mainnet approval contrary to MAINBOARD.

Task 10 — Run the docs publication gate

ANSWER Validate this documentation candidate before publication and again after any source/API/version change.

Prerequisites

Candidate DOCX/PDF and generated machine records.
Pinned current registry/OpenAPI digests.

DO

1. Check canonical values and scope flags.
2. Check API/path terms against the known contract.
3. Check every code block for an example label.
4. Inspect package metadata and internal links.
5. Render the document and visually inspect every page.
6. Hash only the final byte versions and then generate SHA256SUMS.

EXAMPLE STATUS TESTED — local docs gate is executed on the final candidate

EXPECT

No critical/high gate failure; any accepted residual is explicit in the gate report and FreezeFix delta scope.

VERIFY

This FreezeFix delivery includes QANOVRA-DEVELOPER-DOCS-v1.1-FREEZEFIX-DOCS-GATE-REPORT.md generated after the final byte build.

STOP CONDITIONS

Any canonical-value mismatch.
Unlabelled executable example.
Wrong publication status.
Hash manifest generated before a later artifact mutation.

REFERENCE

External v1.0 audit.

Appendix C remediation matrix.
SHA256SUMS in the delivery package.

18. Reference

Network constants only

Field	Value	Type / unit
protocol.version	3.7.5	string
network.chainId	qnv-78cefe355c7fb55c53dd173c	string
network.genesisSha256	78cefe355c7fb55c53dd173c9ca9d340306054a7fd5718e70d1d3193be7b93a3	64-hex SHA-256
asset.symbol	QNV	string
asset.baseUnit	qnv-atom	string
asset.decimals	5	integer
asset.atomsPerQnv	100000	integer string
asset.maxSupplyAtoms	1000000000000000	integer string

Key evidence digests

Artifact/reference	SHA-256
v3.7.5 L1 candidate	1c41b8e5e7f757de2a79937ab5e408021718a456e2eb29f9b1c596a6ccab2414
Parallel RC12 v3.7.5 repin closure ZIP	e7ee09ad5377a316ef9650c991c69418dfb117aa4f53679d0b5068449083820c
Public OpenAPI RC12 v3.7.5 repin	604d9bbd5948b330c9d0c7add41affcab396b30b545dea3c15f0b942b63f17f3
Full OpenAPI RC12 v3.7.5 repin	5416f3abeaca08f6178dbaaf79c0fe226f2e895d889f039aa555c5988982b4b7

19. Glossary

Term	Definition
API	The canonical term in these docs for the HTTP/OpenAPI developer surface.
RPC	Search alias/general protocol term; specifically retained for the permission-controlled /v1/rpc relay where named by the API.
Canonicalization	Deterministic representation rules used so equivalent inputs have one protocol representation.
Chain ID	Published network identifier bound to the Qanovra genesis identity.

Term	Definition
Evidence chain	Hash-/manifest-/attestation-linked records supporting a scoped claim.
Finality	The protocol condition under which a block/state transition is treated as finalized; normative semantics belong to the Yellow Paper/implementation.
Genesis	The canonical initial network configuration/state whose hash participates in network identity.
Operator package	Controlled files, configuration, runbooks and evidence used by an operator for a specific release.
qnv-atom	Smallest native QNV unit; 100,000 qnv-atom = 1 QNV.
qnv-verify	Standalone Go verification implementation; not a Qanovra application SDK.
Registry	Canonical structured source for network/release values used to generate docs and machine records.
SDK	Generated Python or TypeScript application client derived from the canonical OpenAPI.
State root	Commitment to a state snapshot/checkpoint used in verification/recovery flows.
Validator	Operator-controlled consensus participant subject to protocol and admission/economic policies.

20. Appendix A — Source & Evidence Lineage

Source	Role
Qanovra Whitepaper v2.4.2	Publication-layout design reference and public technical/strategic baseline.
Qanovra Yellow Paper v2.4.2	Publication-layout design reference and normative protocol specification; implementation remains authoritative for exact frozen runtime behavior.
Qanovra L1 v3.7.5 candidate/baseline	Frozen technical subject.
Parallel Ecosystem v0.13.0 RC12 + v3.7.5 repin	Current developer/API/SDK/operator ecosystem evidence line.
Founding Validator Economics & Operator Independence Policy v1.4	Founding validator admission/economics/independence policy.
Execution Board v2.3 MAINBOARD	Current execution/gate status source used by this candidate.
External Developer Documentation v1.0 audit	Remediation basis for v1.1.
External Developer Documentation v1.1 re-audit	PASS basis for this FreezeFix; two N-level micro-corrections only.

STATUS PRECEDENCE When a volatile gate status conflicts with older prose, the current authoritative MAINBOARD/evidence register controls the status. Protocol constants still require their canonical protocol/configuration source.

21. Appendix B — Publication Companion Content

These are content contracts to be emitted by the future qanovra.org publication build. Their presence in this DOCX is not evidence that the URLs are already deployed.

network.json

```
{
  "schema": "qanovra-network-reference-v1",
  "protocol": {
    "name": "Qanovra L1",
    "version": "3.7.5",
    "state": "Frozen L1"
  },
  "network": {
    "chainId": "qnv-78cefe355c7fb55c53dd173c",
    "genesisSha256": "78cefe355c7fb55c53dd173c9ca9d340306054a7fd5718e70d1d3193be7b93a3"
  },
  "asset": {
    "symbol": "QNV",
    "baseUnit": "qnv-atom",
    "decimals": 5,
    "atomsPerQnv": "100000",
    "maxSupplyAtoms": "10000000000000000",
    "displayMaxSupplyQnv": "10000000000"
  }
}
```

TESTED — schema content parsed and identity checked

security.txt

Contact: mailto:security@qanovra.org
 Expires: <RFC3339 timestamp; no later than publication date + 1 year>
 Preferred-Languages: en, de
 Policy: <bind-to-published-security-policy>
 Canonical: https://qanovra.org/.well-known/security.txt

ILLUSTRATIVE - RFC 9116 model; bind Expires and Policy to the live publication release

Page metadata minimum

```
title
description
category
version
protocol_version
last_updated
canonical_url
source
source_digest
status
```

PSEUDOCODE — required metadata fields, not a concrete serialization format

22. Appendix C — v1.0 Audit Remediation Matrix

Finding	FreezeFix status	Remediation
DD-K1	VERIFIED CLOSED	Executable out-of-band network identity task + tested network.json verifier; no invented endpoint. External v1.1 re-audit: PASS.

Finding	FreezeFix status	Remediation
DD-K2	VERIFIED CLOSED FOR PUBLIC SCOPE	Ten concrete task pages; unavailable network/host/hardware values remain explicit status, not hidden gaps. External v1.1 re-audit: PASS.
DD-M1	CLOSED	Every executable-looking block labelled TESTED / ILLUSTRATIVE / PSEUDOCODE.
DD-M2	CLOSED	Transaction lifecycle explicitly separates conceptual submit stage from current read-oriented public surface.
DD-M3	CLOSED	API is canonical term; RPC retained only as alias/specific relay name.
DD-M4	CLOSED	Network Reference contains constants only; Wallet/Parallel versions moved to Releases.
DD-M5	CLOSED	Machine record uses explicit maxSupplyAtoms and displayMaxSupplyQnv.
DD-M6	CLOSED BY BUILD GATE	Publication metadata set to 24 Aug 2026 and validated after rendering.
DD-M7	CLOSED IN CONTENT	security@qanovra.org and security.txt publication contract added.
DD-M8	CLOSED	Version/deprecation/removal policy defined without inventing an unevidenced global duration.
DD-N1	CLOSED	No ambiguous QANOVRA_CHAIN_ID SDK variable claim; identity pin is an explicit verification input.
DD-N2	CLOSED	Generic node docs explicitly state only qanovra-validator.service is evidenced.
DD-N3	CLOSED	Language-neutral repeated-query terminology used.
DD-N4	CLOSED	Environment status table names fixture, Invite Testnet and Mainnet with current authorization state.
DD-N5	CLOSED	Revision history and v1.0 audited input hash included.
DD-N6	CLOSED FAIL-SAFELY	Derived relationship acknowledged; exact derivation algorithm not invented without normative proof.
DD-N7	CLOSED	Glossary expanded.
DD-N8	CLOSED	English canonical; translations derived/non-normative.

Finding	FreezeFix status	Remediation
DD-N9	PUBLICATION LEGAL BINDING	No permissive license invented; published site terms/imprint must supply final usage terms before freeze.
DD-N10	CLOSED BY REBUILD	v1.1 rebuilt as a clean DOCX rather than carrying the old template package forward.
R11-N1	CLOSED IN FREEZEFIX	security.txt models now include the mandatory RFC 9116 Expires field with publication-time binding rule.
R11-N2	CLOSED IN FREEZEFIX	Ambiguous b339... evidence-digest row removed rather than guessing an artifact identity.
R11-A1	CLOSED BY LAYOUT	One-row callout tables replaced by semantic shaded paragraphs; website build remains required to use <aside>/admonition components.

RE-AUDIT STATUS External v1.1 re-audit: PASS. No critical or medium findings remain. This FreezeFix applies only the two recommended N-level text corrections and a non-substantive paper-family layout alignment. A short delta confirmation is the remaining document-freeze attestation step.

23. Appendix D — Publication Checklist

- ☐ Canonical network values match registry.
- ☐ Public OpenAPI digest matches the published file.
- ☐ API base URL is either real release metadata or explicitly NOT PUBLISHED.
- ☐ Every code block has an example status.
- ☐ Machine record passes schema/value checks.
- ☐ security.txt includes Contact, Expires, Preferred-Languages, Policy and Canonical fields before deployment.
- ☐ English canonical page is linked from translations.
- ☐ SDK method pages are generated, not manually duplicated.
- ☐ Release compatibility/deprecation status is current.
- ☐ DOCX and PDF rendered and visually inspected.
- ☐ Metadata date/title/version are correct.
- ☐ DOCX + PDF hashes are generated only after final byte freeze.
- ☐ SHA256SUMS, docs-gate report, FreezeFix delta note and handoff are included in the final-delta package.
- ☐ No post-hash content-credential or metadata mutation occurs.

QANOVRA

Developer Documentation v1.1-FreezeFix | Final Freeze Candidate | 24 August 2026