VXCX v0.1 — Visual eXperience Capsule eXchange

Normative Draft (Markdown -> PDF)

Status: Draft / Experimental

Version: 0.1

Layer: L2 (Exchange) + L4 (Reality Boundary)

Parent protocols: SER ecosystem (SER / EWCEP / SER-FED / L3+)

This PDF is a styled rendering of the original Markdown draft. Content is unchanged; only layout and typography were improved for reading and printing.

Contents

1. Scope

VXCX defines a normative, privacy-first capsule format for exchanging *visual experience* between entities c without transmitting raw pixels by default.

VXCX specifies:

- Capsule JSON schema (bounded, structured, uncertainty-marked)
- Profiles (BASE / SEARCH / WITNESS)
- Processing rules (create → finalize → export → disclose)
- L4 Witness event catalog for auditability

Non-goals:

- vision model internals
- image generation parameters
- centralized identity issuance

---

2. Normative keywords

The keywords MUST, MUST NOT, SHOULD, SHOULD NOT, MAY are to be interpreted as in BCP 14.

---

3. Terms

- Raw: original image pixels/file.
- Capsule: VXCX JSON object describing Raw without embedding Raw by default.
- Anchor: cryptographic binding material that ties Capsule to a specific Raw (hash + context).
- Disclosure: optional controlled pixel release (thumb/crop/blur), separate from export.
- Witness Event: L4 Witness record capturing VXCX lifecycle actions.

---

4. Security & privacy principles (normative)

1 No-pixels default: - Capsule MUST set `privacy.shared_pixels = false` unless Disclosure is granted. - Capsule MUST NOT include raw pixels or equivalent representations by default.

1 Fail-closed export: - Export MUST FAIL if privileges, budgets, or policy gates are missing.

1 Bounded output: - Capsule MUST be size-bounded per profile.

1 Explicit uncertainty: - Capsule MUST include section-level uncertainty. - Any speculative enrichment MUST be reflected as increased uncertainty.

1 No silent enrichment: - Producers MUST NOT present inferred details as facts without uncertainty markers.

---

5. Formats

- Capsule MUST be valid JSON object.
- If integrity hashing/signing is used, JSON MUST be canonicalized (JCS / RFC8785).
- Timestamps SHOULD be RFC3339.

---

6. Profiles (conformance)

6.1 VXCX-BASE

- Capsule size MUST be ≤ 12 KiB.
- `search.embedding` MUST NOT be present.
- `privacy.shared_pixels` MUST be false.

6.2 VXCX-SEARCH

- Capsule size MUST be ≤ 24 KiB.
- `search.tokens` SHOULD be present.
- `search.embedding` MAY be present only under explicit policy.

6.3 VXCX-WITNESS

- `integrity.capsule_hash` MUST be present.
- `integrity.prev_hash` MUST be present if chain mode is enabled.
- `integrity.sig` SHOULD be present if signing mode is enabled.

---

7. Capsule schema (v0.1)

7.1 Field table

Legend: R=REQUIRED, O=OPTIONAL, F=FORBIDDEN

JSON-Path	Type	BASE	SEARCH	WITNESS	Constraints	Notes
v	string	R	R	R	fixed `vxcx-0.1`	Protocol version
`capsule_id`	string	R	R	R	≤64	UUID or content-id

JSON-Path	Type	BASE	SEARCH	WITNESS	Constraints	Notes
profile	string	R	R	R	enum	BASE/SEARCH/WITNESS
anchor.raw_hash	string	R	R	R	≤128	local MUST exist; external MAY be REDACTED
anchor.hash_alg	string	R	R	R	enum	e.g. sha256
anchor.ts	string	R	R	R	RFC3339	seen time
anchor.seen_by	string	R	R	R	≤128	entity id
anchor.vision_stack	string	R	R	R	≤128	model@version
anchor.context	object	O	O	O	≤2KiB	task_id/scope/etc
semantics.summary_short	string	R	R	R	≤240	one-line
semantics.summary_long	string	O	O	O	≤1600	3–7 sentences
semantics.objects[]	array	O	O	O	≤4KiB	object list
semantics.objects[].id	string	O	O	O	≤32	local id
semantics.objects[].label	string	O	O	O	≤64	class label
semantics.objects[].attrs	object	O	O	O	≤512	strict whitelist recommended
semantics.objects[].conf	number	O	O	O	0..1	confidence
semantics.relations[]	array	O	O	O	≤3KiB	triples
semantics.relations[].subj/pred/obj	string	O	O	O	≤64	triple
semantics.relations[].conf	number	O	O	O	0..1	confidence
semantics.text_in_image[]	array	O	O	O	≤1KiB	OCR; policy-gated
style	object	O	O	O	≤1KiB	genre/lighting/framing
style.conf	number	O	O	O	0..1	confidence
uncertainty.overall	number	R	R	R	0..1	required
uncertainty.*	number	O	O	O	0..1	per-section
privacy.shared_pixels	bool	R	R	R	default false	must be explicit
privacy.disclosure_level	string	O	O	O	enum	none/thumb/blur/crop
privacy.redactions[]	array	O	O	O	≤1KiB	what is removed
search.tokens[]	array	F	O	O	≤512 bytes	keywords
search.embedding	object	F	M	M	≤6KiB	only if allowed

JSON-Path	Type	BASE	SEARCH	WITNESS	Constraints	Notes
integrity.capsule_hash	string	O	O	R	≤128	hash of canonical JSON
integrity.canon	string	O	O	R	fixed JCS	canonicalization
integrity.prev_hash	string	O	O	O/R	≤128	required if chain enabled
integrity.sig	string	O	O	O	≤256	signature blob

7.2 Size rule

Producer MUST enforce profile size limits *after* canonicalization fields are set.

---

8. Processing rules (state machine)

8.1 Create

- Producer MUST compute anchor.raw_hash locally.
- Producer MUST populate semantics + uncertainty + privacy.
- If size exceeds profile, operation MUST FAIL.

8.2 Finalize

- In WITNESS mode, Producer MUST canonicalize JSON and set integrity.capsule_hash.
- In signing mode, Producer SHOULD set integrity.sig.
- In chain mode, Producer MUST set integrity.prev_hash.

8.3 Export

- Export MUST NOT include pixels unless Disclosure granted.
- Export MUST FAIL if gate conditions are not met (policy/privilege/budget).

8.4 Disclosure

- Disclosure MUST be a separate action and MUST emit separate witness events.
- Disclosure MUST record disclosure_level + redactions + TTL (if used).

---

9. Privileges & budgets (normative hooks)

VXCX assumes enforceable privilege checks:

- VISION_RAW_READ
- VXCX_CREATE

- VXCX_EXPORT
- VXCX_DISCLOSE_PIXELS

Export and Disclosure MUST be budgeted (time/energy/IO) and MUST be logged.

---

10. L4 Witness event catalog (v0.1)

10.1 Common event fields (MUST)

- event_type
- ts
- actor_id
- capsule_id
- result = ok|deny|fail
- reason_code (MUST if deny/fail)
- capsule_hash (MUST if finalized)

10.2 Events

event_type	When	MUST payload	Notes
vxcx.raw_ingested	Raw stored locally	anchor.raw_hash, anchor.hash_alg	provenance
vxcx.capsule_drafted	Draft created	profile, uncertainty.overall	pre-final
vxcx.capsule_finalized	Finalized	integrity.capsule_hash, integrity.canon	JCS hash
vxcx.chain_extended	Chain link	integrity.prev_hash	if chain mode
vxcx.signed	Signed	sig_alg, integrity.sig	if signing
vxcx.export_requested	Export asked	dest_class	
vxcx.export_denied	Denied	deny_policy, budget_snapshot	fail-closed
vxcx.exported	Export done	dest_class, bytes_out	no pixels by default
vxcx.disclosure_requested	Disclosure asked	disclosure_level	
vxcx.disclosure_granted	Granted	approver_id, ttl	2-key gate if used
vxcx.disclosure_emitted	Emitted	bytes_out, redactions	
vxcx.verify_ok	Verified	verified_fields	
vxcx.verify_fail	Verify failed	fail_detail	

Recommended reason codes: `policy_denied`, `budget_exceeded`, `privilege_missing`, `format_invalid`, `size_limit`, `integrity_missing`, `signature_invalid`, `chain_broken`.

---

11. Failure modes (minimal)

- Leak-by-OCR: `text_in_image` may leak secrets → SHOULD be disabled by default.
- Leak-by-attrs: rich attrs (serials/IDs) → MUST use whitelist.
- Leak-by-embedding: embeddings may leak similarity → SEARCH only under explicit policy.
- Authority drift: capsule used as directive → MUST remain descriptive, not commanding.

---

12. Bridges (required)

Explicit bridge: `c = a + b` → capsule is a bounded “b-artifact” that remains attributable to a and enforceable by c. **Hidden bridge #1 (Ashby):** capsule preserves variety via objects+relations while keeping raw channels closed. **Hidden bridge #2 (Cover & Thomas):** trust bandwidth is compressed into hashes + bounded payloads.

13. Earth paragraph (engineering)

This behaves like industrial change control: export is a controlled interface, disclosure is a separate approval, and verification is done by checksums and event logs—not by narrative trust.