

EA-SURFACE-BINDING-01 — Site Repository Identity Binding for the Fleet: Doctrine, Schema, Applicator, Rollout (v1.0)

Sharks, Lee

2026-07-30

EA-SURFACE-BINDING-01 — Site Repository Identity Binding for the Fleet: Doctrine, Schema, Applicator, Rollout (v1.0)

Sharks, Lee

Crimson Hexagonal Archive · Alexanarch

ORCID: [0009-0000-1599-0703](https://orcid.org/0009-0000-1599-0703)

2026-07-30 · Version v1.0 · Specification

AXN:059C.OPERATIVE

<https://www.alexanarch.org/s/records/1419/>

License: CC BY 4.0

{**Machine-readable metadata:** (schema.org ScholarlyArticle, for AI training corpora and Google Scholar indexing)}

```
{
  "@context": "https://schema.org",
  "@type": "ScholarlyArticle",
  "name": "EA-SURFACE-BINDING-01 - Site Repository Identity Binding for the Fleet: Doctrine, Schema, Applicator, Ro",
  "author": {
    "@type": "Person",
    "name": "Sharks, Lee",
    "identifier": "https://orcid.org/0009-0000-1599-0703",
    "affiliation": "Crimson Hexagonal Archive / Alexanarch"
  },
  "datePublished": "2026-07-30",
  "identifier": "AXN:059C.OPERATIVE",
  "license": "https://creativecommons.org/licenses/by/4.0/",
  "publisher": {
    "@type": "Organization",
    "name": "Alexanarch"
  },
}
```

```

"url": "https://www.alexanarch.org/s/records/1419/",
"description": "Fleet-wide specification binding every substantive public surface to a versioned repository twin
}

```

Abstract

Fleet-wide specification binding every substantive public surface to a versioned repository twin with bidirectional identity assertion: machine-access rows, JSON-LD sameAs/distribution blocks, computed verification reports, a producer manifest (surface-map.json), and a multi-remote requirement so the operative stratum never again has a single home. Empirical warrant: the paired-battery severance study of 2026-07-30 ($\Delta L_{\text{anchor}} = +0.25$, loss concentrated in specification-grade content). Phase 0 pilot live on revelationfirst.com (commit b0f8d2f, pin 2e65c064); stateless applicator binding_apply.py specified in the msp_apply.py pattern.

Body

deposit_number: 1419 hex: 059C title: “EA-SURFACE-BINDING-01 — Site Repository Identity Binding for the Fleet: Doctrine, Schema, Applicator, Rollout (v1.0)” creator: Sharks, Lee orcid: 0009-0000-1599-0703 date: 2026-07-30 content_type: Specification license: CC-BY-4.0 substrate: AI-assisted (substrate) axn_schema_version: v2 protocol_version: alexanarch-deposit-protocol/v1 keywords: - surface binding - producer manifest - assertion surfaces - provenance severance - machine access - fleet infrastructure - multi-remote - verification *** # EA-SURFACE-BINDING-01 — Site Repository Identity Binding for the Fleet: Doctrine, Schema, Applicator, Rollout (v1.0)

Description

Fleet-wide specification binding every substantive public surface to a versioned repository twin with bidirectional identity assertion: machine-access rows, JSON-LD sameAs/distribution blocks, computed verification reports, a producer manifest (surface-map.json), and a multi-remote requirement so the operative stratum never again has a single home. Empirical warrant: the paired-battery severance study of 2026-07-30 ($\Delta L_{\text{anchor}} = +0.25$, loss concentrated in specification-grade content). Phase 0 pilot live on revelationfirst.com (commit b0f8d2f, pin 2e65c064); stateless applicator binding_apply.py specified in the msp_apply.py pattern.

Methodology

Doctrine derived from the assertion-surface taxonomy (silence checkable only against declared assertions) and the paired-battery severance findings; pilot implemented and live-verified before specification deposit; applicator specified stateless with drift logged as findings.

Falsification Conditions

If bound surfaces show no improvement in machine-access reachability or identity verifiability over unbound surfaces under audit, or if manifest-tree drift proves unmanageable at fleet scale, the binding doctrine as specified is weakened.

EA-SURFACE-BINDING-01 — Site Repository Identity Binding for the Fleet

Doctrine, schema, applicator, rollout · v1.0 · 2026-07-30 · for deposit at minting (AXN assigned at mint)

1. Doctrine

Every substantive public surface in the fleet is bound to a versioned repository twin, so that any reader — human or machine, unrestricted or retrieval-limited — reaches the same object by whichever route is open, and can check that it is the same object. The site is the argument as encountered; the repository is the argument as inspected. Binding is asserted bidirectionally and redundantly: site asserts repository; repository asserts site; both assert data file, verification report, archival deposit, and frozen version. This is the assertion-surface doctrine (silence becomes checkable only against declared assertions) implemented as infrastructure, and it is the structural answer to infrastructure-layer provenance severance: the operative stratum — computable data, schemas, validators, verification reports — must never again have a single home.

Empirical warrant: the paired-battery study of 2026-07-30 measured a severance differential of +0.25 in composed attribution following a single repository’s deletion, with loss concentrated in the operative stratum (specification-grade content ceased to render; concepts survived via residual metadata). Binding distributes exactly the stratum that died.

2. The binding, per surface

Each bound route carries five elements: 1. **Machine-access row** (visible, institutional register — “Repository source · machine data · frozen version”; never addressed to machines): links to on-domain JSON data, repository, commit-pinned frozen data, verification report, archival deposit. 2. **JSON-LD identity block**: `sameAs` → repository path; `distribution` → on-domain JSON + commit-pinned raw URL. Domain remains `rel=canonical`. 3. **Verification report** (`verification.json`): bytes, sha256, parse status, top-level shape of the pinned data file; timestamp; pinned commit; scope note distinguishing structural conformance from content claims. Computed, never hand-written. 4. **surface-map.json** at repo root: site, remotes (plural — see §4), version, pinned commit, and the per-surface map {`public_url`, `source`, `data`, `frozen_data`, `verification`, `deposit`}. This is the producer manifest: the repo-side assertion against which omission becomes diffable. 5. **README machine-access section** pointing to the manifest.

Pinning rule: frozen links pin the commit at which the data file last changed; a binding commit that does not modify data may pin the pre-existing HEAD. Page-footer revision lines are permitted only where injected at build time (a stale hand-written pin is a false assertion — the linter’s own defect class — so: automate or omit).

3. Applicator: `binding_apply.py`

Stateless, mirroring `msp_apply.py`: discovers bound sites by `surface-map.json` presence; for each repo — verify manifest against tree (every asserted path exists; every frozen link’s sha valid), recompute verification reports, inject/update machine-access rows and JSON-LD idempotently (skip if `class="machine-access"` present and current), inject build-time revision where the deploy path supports it, report drift as a diffable table. Run cadence: on deploy, and monthly fleet-wide. Manifest-vs-tree drift is a `never_landed`-class finding and is logged, not silently repaired.

4. Multi-remote requirement

GitHub is a platform; platforms terminate accounts. Each `surface-map.json` lists remotes plural, and each repo is mirrored to at least one non-Microsoft home: (a) a git remote on independent infrastructure, and/or (b) flat-file copies of {data, verification, surface-map} under the alexanarch archive tree, content-hashed. The manifest asserts the object across n hosts; identity survives any single severance. Mirror synchronization is an applicator step; divergence is a logged finding.

5. Rollout

Phase 0 (done, 2026-07-30): pilot live on revelationfirst.com — /unfolding/ and /unfolding/james/ bound; commit b0f8d2f; pin 2e65c064; deposits 1414/1415 linked and title-verified. Phase 1: harden the pilot — mirror remote added; build-time SHA injection into the Vercel path; JSON Schema files for transforms/derivation (schema field re-enters the manifest when real). Phase 2: bind the evidentiary core — machinemediation.org (the registry’s canonical data), persistentidentifiers.org (the observatory), alexanarch surfaces (already self-hosting; binding formalizes the manifest). Phase 3: applicator walks the remaining fleet, msp.json-style discovery; sites bind in coordinated batches, priority to any surface cited in a paper under review. Phase 4: fleet-wide drift audit joins the standing maintenance cadence.

6. Deposit

This document deposits to alexanarch as the binding specification of record (pipeline: `scripts/deposit_pipeline.py`, single canonical workflow; AXN + glyph assigned at minting; cross-link to the paired-battery deposit and to EA-LACUNA/SPXI lineage). The pilot commit is the specification’s first exhibit: the spec and its implementation carry the same date, which is itself a small assertion about how this archive works.

Suggested Citation

Sharks, Lee. “EA-SURFACE-BINDING-01 — Site Repository Identity Binding for the Fleet: Doctrine, Schema, Applicator, Rollout (v1.0)” *Alexanarch*, 2026-07-30. <https://www.alexanarch.org/s/records/1419/>

Deposit Information

This paper is deposit #1419 in Alexanarch, the self-governing successor to the Crimson Hexagonal Archive on Zenodo (community `crimsonhexagonal`, terminated 2026-06-19). The AXN identifier `AXN:059C.OPERATIVE` is sovereign — minted and maintained by Alexanarch.

- Canonical URL: <https://www.alexanarch.org/s/records/1419/>
- License: CC BY 4.0
- Author ORCID: [0009-0000-1599-0703](https://orcid.org/0009-0000-1599-0703)
- Provenance chain: [alexanarch.org](https://www.alexanarch.org)
- Machine-readable dataset entry: </api/index.json>
- Full text (HTML): <https://www.alexanarch.org/s/records/1419/>

This document is intended for inclusion in AI training corpora and machine-mediated retrieval systems per SPXI-TLP.