

ODNCA-STD-008 · Registration Data (WHOIS)

The public lookup surface, and its RDAP mapping

Status: v1.0 — Adopted

Normative reference implementation: ORDnet Registry v16.2x (`/whois/<domain>` , `/api/owner/<address>` , reverse lookup via the resolver).

1. Scope

Registration data answers "tell me about this name" for humans and tools: status, holder, on-chain anchors, dates. The primary format is the deployed JSON below — native, honest, and chain-anchored. An RDAP field mapping (§4) is provided so existing domain tooling can consume the same data in the shape it expects.

The lookup is public and unauthenticated; it exposes only what the chain and the coordinated lists already make public. There is no tiered or gated WHOIS: the same answer for everyone.

2. Statuses

Every name is in exactly one lookup state:

STATUS	MEANING	EXTRA FIELDS
available	not registered	current <code>price_usd</code> (and <code>free: true</code> when zero), registration URL
reserved	held on the registrar's own list (POL-002 §2.2.5)	<code>reserved_at</code>
flagged	under a POL-003 measure	<code>flagged_at</code> ; display per POL-003 §2 (honest, never hidden)
registered	claimed and live	full record (§3)

Consistency rules: `reserved` and `flagged` answer before availability — a reserved name is never advertised as buyable; a flagged name is never denied to exist. Retired-TLD names answer as not-registrable while existing registrations under them keep their full records (POL-001).

3. The registered record

For a registered name the answer carries: the canonical name; status; the **holder address** (`owner_bsv`); the **content coupling** (`target_txid` , `target_vout` — the content channel of ODNCA-STD-006 §4); and registration/update timestamps. The on-chain anchors (origin and current outpoint) are served by the resolver's signed answer (ODNCA-STD-001 §6) — WHOIS states, the resolver proves; tools wanting proof follow the outpoint, not the WHOIS text.

Reverse lookup completes the surface: address → all names held (`/api/owner/<address>` at the registry; `GET /reverse/<address>` with pagination at the resolver). Forward tells you who holds a name; reverse tells you what a wallet holds — both are public chain facts, exposed rather than pretended away.

4. RDAP mapping

For tooling that speaks RDAP (the JSON successor of port-43 WHOIS), the same data maps as:

RDAP	THIS STANDARD
<code>objectClassName: "domain"</code>	fixed
<code>ldhName / unicodeName</code>	canonical name (non-ASCII names: <code>unicodeName</code> , ODNCA-STD-003 §6)
<code>status[]</code>	§2 status (<code>active</code> for registered; <code>reserved</code> ; measures per POL-003 as <code>locked</code> + remark)
<code>entities[]</code> (registrant)	holder address as the registrant handle — the key <i>is</i> the identity
<code>events[]</code> (<code>registration</code> , <code>last changed</code>)	created/updated timestamps
<code>links[]</code>	resolver answer URL, <code>ord://</code> content URL (ODNCA-STD-005), certificate URL (ODNCA-STD-004 §6)
<code>remarks[]</code>	honest measure disclosure (POL-003 reason code) where applicable

Notes: there are no expiry events — names do not expire, they are held or transferred (renewal-shaped fields are absent by design, not omission). There is no privacy/redaction layer: the registrant is a public address on a public chain; RDAP's redaction machinery is unused. A conformant RDAP façade is a pure translation of §2–3 — it introduces no data of its own.

5. Freshness

WHOIS follows the same pipeline as everything else (chain → indexer → owner-sync → registry) and can therefore lag the chain by minutes. It never pretends otherwise: for the live, provable answer, the resolver's signed answer with the outpoint check is the authority; WHOIS is the human-friendly ledger view. Where they briefly differ, the chain is right — as always.

ODNCA — the ORDnet Decentralized Names Coordination Authority. Don't trust us — recompute us.