# Inferring & Validating Datacenter Date-of-Operation

Marco Bova

September 10, 2025

# Project goal

- Infer or validate the **year of operation start** for each facility in `DataCenterMap`, last updated in july.
- Dataset size: **3,685** rows.
- Non-missing `year_operational`: **448** ($\sim$ 12%).
- Missing `year_operational`: **3,179** ($\sim$ 88%).

# Sources explored (signals for start-of-operations)

- **ENERGY STAR (facility lists)** → `Year Constructed` (upper bound for go-live).
- **EPA/ECHO** (ICIS-AIR, compliance) → permit/inspection/action dates (administrative).
- **State air-permit portals** → construction/operating permit issue/effective dates (upper bound).
- **Imagery/OSM history (OSHDB)** → first visible structure/footprint (construction onset).

# What we implemented (methods & concrete outcomes)

**ENERGY STAR integration**

- Exact name join ⇒ **12** hits.
- **Fuzzy join** (Jaro–Winkler $\leq 0.12$) + **exact state filter** + **one best per ID** ⇒ **64** total matches.
- Among missing `year_operational`: **39** usable years recovered.
- looked into "ENERGYSTARCertifiedLargeNetworkEquipment" unfortunately not relevant

**EPA/ECHO (ICIS-AIR)**: no reliable additions. **State portals**: No comprehensive dataset for land registry exists in the US.

Looked at "datacenter cooling demand total", "rexus dataset" but no luck, rexus only government owned land information.

# Satelite imagery methods

- Based on OSM dataset create a pipeline to get information on the dates of the building.
- Start from Overpass api to get unique id for each set of coordinates of the observations we have in the dataset "DataCenterMap"
- Use the ID to request information from the OSM historic data registry
- I have run various iterations from simpler to more complete.

# OSM Matching Pipeline — Versions & Selection

**V1** Find any `building=*` near provider coords; pick the *oldest mapped element* (first OSM version). *Selection:* history-only. *Signals:* `first_timestamp`.

**V2** Same search; choose by *proximity*; capture current tags. *Selection:* proximity + last-change emphasis (no start-date/op-year inference). *Signals:* `tags_after_change`, `is_datacenter_now`, `last_change_*`.

**V4** Add full element history; richer tag snapshot; parse `start_date` ⇒ `start_date_year`. *Selection:* proximity + richer tags (coverage still limited). *Signals:* `start_date_year`, temporal tags.

**V5.1** Hardened requests + history; *deterministic cascade*: current explicit DC → ever exact brand → current brand → generic shell. *Signals:* `dc_first_seen_explicit_year`, `operational_year_inferred` (= `start_date_year` else explicit-DC year), `selection_rule_used`.

**V5.2** *Radius escalation* (50/100/200 m); select current explicit DC else *generic fallback* (`building=yes`, etc.; brand matching not used). *Signals:* `search_radius_used`, `dc_first_seen_like_*`, `selection_rule_used`.

# V5.2 Pipeline Logic (and why we default to it)

**Steps**

1. Search by radius steps for *current explicit DCs* (`building=data_center` or `telecom=data_center`); if found, read history in order: `start_date` → first explicit DC tag → first DC-like tag.
2. If none, expand radius and *fall back to generic shells only*: accept `building=<allowed_generic>` (default `yes`); reject specific types (office/industrial/apartments). Optionally require a usable date signal before accepting.
3. Deterministic pick among candidates: use the most recent relevant-change timestamp as a stable tie-breaker.
4. Output inferred year + provenance; record the applied rule and the radius used (`selection_rule_used`, `search_radius_used`).

**Why V5.2 as default** Highest recall (more explicit DCs found; more rows with usable dates) and fully auditable decisions (rule + radius). Known trade-off: precision drops at larger radii; mitigate by restoring brand matching and adding acceptance checks beyond 50 m.

## Key Variables

- *Last Change* →timestamp of last significant change.
- *First timestamp* (first time the OSM element appears) → weak proxy for go-live.
- *start_date_year* (when present) → best single source but sparse; may reflect building opening, not DC go live, taken from start_date, opening_date, opened, construction_date, and start_date:edtf
- *dc_first_seen_explicit_year* → first time OSM explicitly tags it as a datacenter.
- *dc_first_seen_like_year* → first time a DC-like tag appears.
- *operational_year_inferred* → deterministic inference (prefer start-date; else first explicit DC year, else first dc like year).

# Accuracy on the Valid Sample (non-missing provider year)

**Metric:** alignment to provider `year_operational`. “Close” = absolute difference $\leq 1$ year.

| Model (n) | LastChg close | FirstTS close | StartDate cov / close | Inferred cov / close | FirstDC cov / close |
|---|---|---|---|---|---|
| V1 (37) | — | 10.8% | — | — | — |
| V2 (266) | 5.3% | 9.8% | — | — | — |
| V4 (408) | 6.4% | 10.5% | 7.8% / 15.6% | — | — |
| V5.1 (410) | 3.9% | 10.7% | 6.1% / 32.0% | 31.7% / 15.4% | 27.3% / 10.7% |
| V5.2 (448) | 4.2% | 10.0% | 4.2% / **36.8%** | **35.9%** / 13.0% | **33.0%** / 8.8% |

**Takeaways.**

- “First timestamp” is a weak proxy across all versions (close ≈ 10%).
- When present, `start_date_year` is the most accurate single source (V5.2 close **36.8%**), but it is sparse.
- `operational_year_inferred`: V5.2 trades a small accuracy drop (15.4% → 13.0%) for higher usable coverage (31.7% → **35.9%**).
- `dc_first_seen_explicit_year`: coverage improves (27.3% → **33.0%**); close is modest (10.7% → 8.8%).
- V5.2 delivers *more rows with usable dates* and finds more *current explicit DCs*. Accuracy on inferred op-year is slightly below V5.1, especially at larger radii or generic fallbacks.

# Coverage on Total Observations (rows with missing provider year)

**Population:** rows with missing provider `year_operational` ($n = 3179$). Entries show % available.

| Model | StartDate | FirstTS | LastChg | FirstDC | OpYear Inferred |
|---|---:|---:|---:|---:|---:|
| V2 | — | 46.40% | 46.40% | — | — |
| V4 | 3.15% | 69.14% | 69.14% | — | — |
| V5.1 | 2.6% | 69.0% | 69.0% | 28.5% | 30.1% |
| V5.2 | 2.0% | **80.60%** | **80.60%** | **36.70%** | **37.60%** |

**Availability structure in V5.2 (all rows):**

- At least one temporal field available: **83.3%**; all six fields: **0.2%**; none: **16.7%**.

**Interpretation.** V5.2 substantially increases coverage where it matters (rows lacking provider dates), especially for `FirstTS/LastChg`, `dc_first_seen_explicit_year`, and `operational_year_inferred`.

# Why select **V5.2** now (and how to use it responsibly)

**Why V5.2**

- **Best coverage**: highest share of usable inferences on the valid sample (`operational_year_inferred` **35.9%**; `FirstDC` **33.0%**) and on the missing-provider subset (**37.6%** and **36.7%**, respectively).
- **More explicit DC hits**: higher prevalence of current explicit DC selections (32.4% vs. 26.3% in V5.1).
- **Transparent QA**: `selection_rule_used` and `search_radius_used` allow confidence slicing.
- have thought of another viable option to support our method, that is to use a third party service called flypix.ai that allows to select via satellite images structures that have specific characteristics

**Bottom line.** V5.2 achieves the best *coverage* while keeping accuracy interpretable; its diagnostics let you dial precision via filtering without losing the recall gains.

# additional trials

- **V5.3**
  - *Find:* **radius escalation** 50→100→200m if no DC found
  - *Select:* **current explicit DC** else **generic fallback** (`building=yes`, etc.) *(brand matching not used)*
  - *Signals/Output:* `search_radius_used`, `dc_first_seen_like_*`, `selection_rule_used`