ICT for Civic Data — Crash Course 2026

Enrichment and Combination

Self-Paced Review — Section E

civicliteraci.es

Why This Matters

> Combine datasets to build evidence

One dataset is information. Two datasets combined tell a **story**. Enrichment is the act of combining datasets to produce insight that neither could provide alone.

This is still part of the **Verify/Clean/Analyse** arc. You are not adding new pipeline steps; you are repeating the existing ones with richer data. Each new dataset you bring in needs its own Get, its own Verify, and its own Clean before you can combine it with what you already have.

Before You Start

> What makes datasets combinable

Two datasets can be combined when they share a **common dimension**: something that links records across the two sources.

Common dimensions include:

- **Geography** — same country, same coordinate system, overlapping regions
- **Time** — overlapping periods (a 2020 flood dataset can combine with 2023 facility data if facilities have not moved)
- **Category** — same type of entity (both describe health facilities, or both describe administrative districts)

Without a shared dimension, there is no link between the datasets. You can display them side by side, but you cannot combine them analytically. Combining means: for each record in Dataset A, find the related records in Dataset B.

> Granularity must match

Combining a **daily** dataset with a **yearly** one, or a **city-level** dataset with a **regional** one, does not work directly. You must align to the **less precise** level.

This usually means **aggregating** the more granular dataset:

- Daily flood events → count per year per region → now combinable with yearly population data
- Point-level facility locations → count per district → now combinable with district-level statistics

Aggregation always loses information. You are trading detail for compatibility. This is a conscious design choice. Document it and explain why you made it.

The question is always: **what is the smallest unit I can meaningfully analyse?** That determines the granularity of the combined dataset.

Walkthrough

> Each new dataset repeats the pipeline

Enrichment is not a single operation. It is a second pass through the pipeline for the new dataset:

1. **Get** — retrieve health facilities from the Overpass API
2. **Verify** — put them on the map, check counts and locations
3. **Clean** — standardise fields, remove duplicates

Only after this second pass can you combine the new data with the existing dataset. Skipping Verify on the second dataset is the most common mistake: you end up combining clean data with dirty data, and errors become invisible.

> Six steps determine which facilities are in flood zones

Case study: which health facilities are in flood risk zones?

Step	What you do	Pipeline stage
1	Filter flood data to one country	**Clean**
2	Build a map of flood events	**Verify**
3	Get health facilities from Overpass	**Get**
4	Add facilities to the map	**Verify**
5	Separate at-risk facilities (within 1km)	**Clean/Analyse**
6	Add interactive filter for risk categories	**Present**

Each step produces a verifiable output. Each uses a separate prompt.

> Prompt: separate at-risk facilities

« Read the health facilities GeoJSON and the flood events CSV. For each facility, calculate distance to nearest flood event. Mark facilities within 1km as "at-risk." 1. Load both files. 2. For each facility, find nearest flood event using Haversine. 3. Add "risk_status" property ("at-risk" or "safe"). 4. Save two GeoJSON files: at-risk and safe. Two GeoJSON files with all original properties plus risk_status and distance_to_nearest_flood. Two files lets us style them differently on the map. Haversine accounts for Earth's curvature. 1km is a reasonable proximity threshold for flood impact. »

Objective | Steps | Output shape | Reasoning

All five prompt elements: objective, steps, output shape, reasoning. The fifth, human verification, is visual: put the results on the map and check.

> Prompt: add a filter to the map

« Add the at-risk and safe facility layers to the existing Leaflet map. 1. Load both GeoJSON files. 2. Style at-risk facilities in red and safe facilities in green. 3. Add a layer control so the user can toggle each category on and off. Update the existing map HTML file. The map should show flood events, at-risk facilities, and safe facilities as separate toggleable layers. Keep the map simple — use circle markers, not custom icons. The layer control lets the user focus on what matters. »

Objective Steps Output shape Reasoning

After this step, you have a map that is both a **verification tool** and a **presentation tool**.

Behind the Approach

> The map is the verification tool

For geospatial data, the fastest check is **visual**. If data looks wrong on the map, it probably is:

- Facilities in the ocean → coordinate error or wrong coordinate system
- All facilities clustered in one point → duplicate coordinates or a default value (0,0)
- Flood events in the wrong country → filtering error
- At-risk facilities far from any flood marker → distance calculation error

You do not need to read the data file to spot these problems. The map shows them immediately. This is why every Get step is followed by a Verify step that puts data on a map.

Visual verification does not replace checking the data, but it catches the most common errors in seconds. When the map looks right, then you dig into the numbers.

> Case study → angle → proposal

The enrichment exercise IS the proposal demonstration:

Case study: Indonesia flood monitoring: remote communities, scattered health facilities, no systematic risk mapping.

Angle: Prevention + staff safety + emergency response. If field staff know which facilities are at risk *before* a flood, they can prepare.

Proposal: A facility risk dashboard for field staff, overlaying flood history, facility locations, and population data. The organisation gains a planning tool it currently lacks.

The map you just built is the **"show don't tell"** artifact from Section A. Instead of describing what you would do, you demonstrate it. The funder sees evidence, not promises.

FAQ

> What if two sources show different numbers?

Different sources may have different results for the same question. The FloodArchive and EM-DAT may report different flood counts for the same country in the same year.

This does not mean one is wrong. It means they have:

- **Different objectives** — FloodArchive tracks events by location; EM-DAT tracks events by impact
- **Different definitions** — what counts as a "flood event" varies
- **Different time periods** — coverage start dates differ
- **Different collection methods** — satellite detection vs government reports

Both can be true. Your job is to **document the discrepancy** and explain your choice. "I used FloodArchive because it has coordinates, which EM-DAT does not" is a valid, transparent reason.

> How do I add population context to my map?

A student suggested adding **population density** to contextualise facility risk. A health facility in a flood zone serving 50,000 people is more critical than one serving 500.

Two sources for this:

- **Meta HRSL** (High Resolution Settlement Layer) — ML-derived, 30m resolution, available on HDX
- **WorldPop** — modelled population grids, available by country and year

Adding a population layer turns a binary risk map (at-risk / safe) into a **prioritisation tool**: which at-risk facilities should be addressed first? This is exactly the kind of enrichment that makes a proposal more compelling.

The prompt pattern is the same: Get the data → Verify on the map → Clean/Analyse to combine → Present with the existing layers.