# Initial Data Collection & Profiling

DAEN 459 – Fall 2026

**Alexander Abuabara**



Actual x Virtual | Practical x Ritual

## The Actual

What currently exists in physical reality. It is concrete, measurable, and historically realized.

## The Virtual

The realm of latent potential, mathematical abstraction, and simulations. Real forces or models that shape what can become actual.

## The Practical

The bridge of action, utility, and efficiency. It asks: *How do we apply ideas and data to solve problems and make decisions?*

## The Ritual

A structured sequence of actions focused on meaning, internal transformation, and alignment rather than external utility.

# Everyday Examples

*Actual:* Our conversation, it's live, it's action.

*Virtual:* This slides, a book, something that is stored.

# Everyday Examples

*Actual:* Our conversation, it's live, it's action.

*Virtual:* This slides, a book, something that is stored.

**Eating a Meal**

- *Practical:* Eating a protein bar at your desk solely to fuel your physical body.
- *Ritual:* Group family dinner, sharing gratitude, and passing specific dishes.

**Getting a Degree**

- *Practical:* Logging into an administrative portal to download a PDF diploma.
- *Ritual:* Walking across a stage in a cap and gown while your family cheers.

| Attribute | The Practical (Utility) | The Ritual (Meaning) |
|---|---|---|
| **Primary Goal** | Efficiency & Output | Transformation & Connection |
| **Focus** | The destination or product | The journey and execution |
| **Logic** | Instrumental (take the shortcut) | Symbolic (steps cannot be skipped) |
| **Outcome** | Changes physical environment | Changes you (psychological/social state) |

# Data Collection & Profiling

# What success looks like

## The goal is not "download a dataset"

Produce a **repeatable acquisition path**, a **measured quality baseline**, and a **decision about fitness for purpose**.

1. **Discover:** identify sources, owners, licenses, refresh cadence.
2. **Collect:** land a small but representative raw sample.
3. **Profile:** measure schema, missingness, keys, values, and time.
4. **Decide:** document risks and a go / investigate / reject call.

**Artifact:** one-page source inventory + raw sample + reproducible profiling report.

Core question: Can we trust this data enough to build on it?

# Example: Where do city bikes disappear?

**Project idea:** forecast bike availability at each station for the next hour.

| Source | Fields we want | Risk to test |
|---|---|---|
| Trips (CSV) | start/end station, times, trip ID | late uploads; duplicate trips |
| Station (JSON) | station ID, bikes, docks, timestamp | changing IDs; intermittent readings |
| Weather (API) | hourly rainfall, temperature | timezone and spatial mismatch |

## Think ...

A dataset with 10 million trip rows may still be **unfit** for hourly forecasting if it arrives three weeks late.

# Collect like an engineer, not a tourist

**Source / API $\longrightarrow$ Immutable raw $\longrightarrow$ Profile $\longrightarrow$ Curated tables**

**Capture provenance**

- source URL and access terms
- retrieval UTC timestamp
- requested period and filters
- file hash / batch ID
- API pagination and status

**Preserve reproducibility**

- keep raw bytes unchanged
- version the extraction code
- record schema and row count
- save failed requests safely
- separate secrets from notebooks

## Sample deliberately

Include weekdays *and* weekends, peak *and* quiet hours, old *and* recent batches. A convenient first 1,000 rows can mislead.

# Profile across six lenses

| Lens | Measure | Example red flag |
|---|---|---|
| Structure | fields, types, schema versions | switches type |
| Completeness | null %, gaps by source/date | 30% missing ends |
| Uniqueness | duplicate IDs, key collisions | repeated `trip_id` |
| Validity | ranges, allowed values, formats | negative trip duration |
| Consistency | cross-field / cross-source rules | empty docks $>$ capacity |
| Freshness | event vs arrival lag, missing hs | API data 8 hours stale |

## Do not stop at global percentages

Break every important metric down by **date, source, station, and batch**.
An error rate may hide a important broken information.

# A miniature profile tells a bigger story

**Illustrative sample:** 100,000 trip rows across 14 days.

| Check | Finding | Interpretation |
|---|---|---|
| Null `end_st_id` | 8.4% | could be valid free-floating returns |
| Duplicate `trip_id` | 0.7% | test ingestion replay |
| Duration $< 0$ | 0.2% | bad clock or timestamp parsing |
| Arrival lag $> 24$ h | 12.1% | unsuitable for live features? |

## Turn metrics into hypotheses

**Observation:** duplicates spike on Mondays.
**Hypothesis:** the weekend export is resent each Monday.
**Next test:** group duplicate IDs by *ingestion batch* before deduplicating.

# Time is a data quality dimension

**Three different clocks**

- **Event time:** when the trip occurred
- **Arrival time:** when our pipeline received it
- **Processing time:** when we transformed it

**Check explicitly**

- timezone and daylight-saving rules
- late-arriving records
- missing / repeated hourly buckets
- timestamps in the future

## Example

A trip at **08:15 local time** may be ingested at **11:40 UTC**. This is not necessarily a 3h25m delay: normalize timezones first.

**Capstone rule:** define whether each feature uses data *available at prediction time*, not merely data present today.

# A small, repeatable profiling script

```python
import pandas as pd

# Start with a representative raw sample.
df = pd.read_csv("raw/trips_sample.csv")
report = pd.DataFrame({
    "dtype": df.dtypes.astype(str),
    "null_pct": df.isna().mean().mul(100).round(2),
    "distinct": df.nunique(dropna=True),
})

print("rows:", len(df))
print("duplicate trip IDs:",
      df.duplicated(subset=["trip_id"]).sum())
print(report.sort_values("null_pct", ascending=False))

# Add domain checks, batch-level splits, and time lag tests.
```

**Next engineering step:** persist the report as a dated artifact and fail CI only on *agreed* thresholds, not arbitrary zero-null expectations.

# From finding to action: a quality decision log

| Finding | Decision | Evidence needed |
|---|---|---|
| 0.7% duplicate IDs | quarantine collisions; do not silently delete | compare source rows and batches |
| 8.4% missing end station | treat as "unknown" pending semantics | provider documentation + sample review |
| 12.1% arrive late | exclude from low-latency features | latency distribution over 30 days |

## Avoid two traps

**Trap 1:** "clean" means dropping anomalous records.
**Trap 2:** a dashboard of metrics is a substitute for a decision.

Record **owner, severity, proposed handling, and next verification** for each important issue.

# Capstone checkpoint: show your evidence!

**Submit (next week lab/Fri! in parallel with the project charter) four small artifacts:**

1. **Source inventory:** URLs, ownership, access, format, cadence, licensing.
2. **Raw landing proof:** file layout, timestamps, hashes, extraction command.
3. **Profiling notebook/report:** schema, 6 quality lenses, segmented metrics.
4. **Decision log:** top 3 risks, tests, mitigations, go/no-go judgment.

## 90-second project review

*"The most surprising data quality issue was _____. It matters because _____. Our next test is _____."*

Leave with a verified dataset assumption, not just a downloaded file.

# Plan for Wed and Fri

1. **Wed Oct 14** 2-Capco, 2-Navy, 3-LM
2. **Fri Oct 16** 1-Merlion, 2-EM, 1-HRRC, 1-ISEN, 1-Foundation

Let us meet with on the designated days to review team organization, discuss plans and next steps, facilitate coordination among teams working with the same sponsor, and provide guidance and support for the development of project charters.

Please come prepared to discuss your team's organizational structure, current progress, proposed project scope, and any questions or challenges related to the project charter.