

AI-Assisted Data Pipeline

for Automated Mapping and Standardized XML Submission to EFSA

Shelake Ruturaj, Merboth Mathias, Schrapps Thorsten,
Sauter-Louis Carola, Staubach Christoph

KIDA-KON 2026

Flow of Presentation

01

Background & Motivation

EFSA data collection, SSD2, DCF

02

The Problem

Manual Mapping bottlenecks

03

Pipeline Architecture

Python, Airflow, modular design

04

ML Assisted Mapping

Fuzzy matching, learning, overrides

05

Validation & Database

Quality checks, PostgreSQL, traceability

06

Results & Conclusion

239,000+ records, lessons learned

Background

EFSA Data Collection

Data collection is a core task of EFSA under Articles 22 & 23 of Regulation (EC) No 178/2002.

The SIGMA project extended the SSD2 data model for harmonised collection of laboratory test results on animal diseases.

Reporting countries submit XML files via the Data Collection Framework (DCF).

Surveillance Project

FLI collects surveillance data from 16 federal states (Bundesländer) on multiple animal diseases

Diseases covered:

Avian Influenza (AI)

African Swine Fever (ASF)

West Nile Fever (WNF)

Q Fever, Hepatitis E

Data arrives on MSSQL database or as heterogeneous Excel files

From the Data Warehouse to EFSA Surveillance Reports

Submitted records feed EFSA's annual, disease-specific surveillance publications

```
<?xml version="1.0"?>
<records>
  <record>
    <id>1</id>
    <country>Spain</country>
    <date>2024-01-01</date>
    <pathogen>African swine fever</pathogen>
    <status>Suspected</status>
    <details>
      <location>Madrid</location>
      <age>Adult</age>
      <sex>Male</sex>
      <weight>150</weight>
      <height>180</height>
      <temperature>38.5</temperature>
      <heart_rate>100</heart_rate>
      <respiratory_rate>20</respiratory_rate>
      <oxygen_saturation>95</oxygen_saturation>
      <blood_pressure>120/80</blood_pressure>
      <glucose>100</glucose>
      <hemoglobin>15</hemoglobin>
      <hematocrit>45</hematocrit>
      <platelets>150</platelets>
      <white_blood_cells>10000</white_blood_cells>
      <neutrophils>80</neutrophils>
      <lymphocytes>15</lymphocytes>
      <monocytes>5</monocytes>
      <eosinophils>0</eosinophils>
      <basophils>0</basophils>
      <prothrombin_time>12</prothrombin_time>
      <partial_thromboplastin_time>35</partial_thromboplastin_time>
      <fibrinogen>300</fibrinogen>
      <fibrin_degradation_products>10</fibrin_degradation_products>
      <creatinine>1.2</creatinine>
      <urea_nitrogen>8</urea_nitrogen>
      <calcium>10</calcium>
      <phosphorus>3</phosphorus>
      <potassium>4</potassium>
      <sodium>140</sodium>
      <chloride>100</chloride>
      <magnesium>2</magnesium>
      <zinc>100</zinc>
      <iron>500</iron>
      <copper>100</copper>
      <manganese>100</manganese>
      <selenium>100</selenium>
      <chromium>100</chromium>
      <vanadium>100</vanadium>
      <cobalt>100</cobalt>
      <nickel>100</nickel>
      <silicon>100</silicon>
      <boron>100</boron>
      <fluorine>100</fluorine>
      <iodine>100</iodine>
      <bromine>100</bromine>
      <barium>100</barium>
      <strontium>100</strontium>
      <yttrium>100</yttrium>
      <zirconium>100</zirconium>
      <niobium>100</niobium>
      <hafnium>100</hafnium>
      <tantalum>100</tantalum>
      <tungsten>100</tungsten>
      <molybdenum>100</molybdenum>
      <rhenium>100</rhenium>
      <osmium>100</osmium>
      <iridium>100</iridium>
      <platinum>100</platinum>
      <gold>100</gold>
      <mercury>100</mercury>
      <lead>100</lead>
      <tin>100</tin>
      <antimony>100</antimony>
      <arsenic>100</arsenic>
      <phosphorus>100</phosphorus>
      <sulfur>100</sulfur>
      <silicon>100</silicon>
      <boron>100</boron>
      <fluorine>100</fluorine>
      <iodine>100</iodine>
      <bromine>100</bromine>
      <barium>100</barium>
      <strontium>100</strontium>
      <yttrium>100</yttrium>
      <zirconium>100</zirconium>
      <niobium>100</niobium>
      <hafnium>100</hafnium>
      <tantalum>100</tantalum>
      <tungsten>100</tungsten>
      <molybdenum>100</molybdenum>
      <rhenium>100</rhenium>
      <osmium>100</osmium>
      <iridium>100</iridium>
      <platinum>100</platinum>
      <gold>100</gold>
      <mercury>100</mercury>
      <lead>100</lead>
      <tin>100</tin>
      <antimony>100</antimony>
      <arsenic>100</arsenic>
      <phosphorus>100</phosphorus>
      <sulfur>100</sulfur>
    </details>
  </record>
  <record>
    <id>2</id>
    <country>France</country>
    <date>2024-02-01</date>
    <pathogen>Avian influenza</pathogen>
    <status>Confirmed</status>
    <details>
      <location>Paris</location>
      <age>Adult</age>
      <sex>Female</sex>
      <weight>60</weight>
      <height>160</height>
      <temperature>39.0</temperature>
      <heart_rate>110</heart_rate>
      <respiratory_rate>25</respiratory_rate>
      <oxygen_saturation>90</oxygen_saturation>
      <blood_pressure>130/90</blood_pressure>
      <glucose>110</glucose>
      <hemoglobin>14</hemoglobin>
      <hematocrit>40</hematocrit>
      <platelets>140</platelets>
      <white_blood_cells>11000</white_blood_cells>
      <neutrophils>75</neutrophils>
      <lymphocytes>10</lymphocytes>
      <monocytes>10</monocytes>
      <eosinophils>5</eosinophils>
      <basophils>0</basophils>
      <prothrombin_time>13</prothrombin_time>
      <partial_thromboplastin_time>38</partial_thromboplastin_time>
      <fibrinogen>250</fibrinogen>
      <fibrin_degradation_products>15</fibrin_degradation_products>
      <creatinine>1.5</creatinine>
      <urea_nitrogen>10</urea_nitrogen>
      <calcium>9</calcium>
      <phosphorus>2</phosphorus>
      <potassium>3</potassium>
      <sodium>135</sodium>
      <chloride>95</chloride>
      <magnesium>1</magnesium>
      <zinc>90</zinc>
      <iron>450</iron>
      <copper>90</copper>
      <manganese>90</manganese>
      <selenium>90</selenium>
      <chromium>90</chromium>
      <vanadium>90</vanadium>
      <cobalt>90</cobalt>
      <nickel>90</nickel>
      <silicon>90</silicon>
      <boron>90</boron>
      <fluorine>90</fluorine>
      <iodine>90</iodine>
      <bromine>90</bromine>
      <barium>90</barium>
      <strontium>90</strontium>
      <yttrium>90</yttrium>
      <zirconium>90</zirconium>
      <niobium>90</niobium>
      <hafnium>90</hafnium>
      <tantalum>90</tantalum>
      <tungsten>90</tungsten>
      <molybdenum>90</molybdenum>
      <rhenium>90</rhenium>
      <osmium>90</osmium>
      <iridium>90</iridium>
      <platinum>90</platinum>
      <gold>90</gold>
      <mercury>90</mercury>
      <lead>90</lead>
      <tin>90</tin>
      <antimony>90</antimony>
      <arsenic>90</arsenic>
      <phosphorus>90</phosphorus>
      <sulfur>90</sulfur>
    </details>
  </record>

```



Submitted XML

EFSA Data Warehouse

Our pipeline's output is part of the underlying surveillance dataset behind these EU-wide annual reports published by EC and EFSA.



EFSA Journal · Epidemiological analysis of African swine fever in the EU during 2024



EFSA Journal · Avian Influenza annual report 2024

The Problem

1

Inconsistent Source Formats

16 federal states send Excel files with different column names, languages, and structures

2

Manual Field Mappings

Species, locations and methods were mapped by hand – slow, error-prone and unrepeatable

3

No Traceability

Once data was transformed, the link to original source values was lost. Audits required starting over

4

Repeated Correction Cycles

Systematic mapping errors were discovered only after submission, triggering manual re-processing

Our Approach

A modular Python + Apache Airflow pipeline that transforms heterogeneous German surveillance data into EFSA-compliant XML and a normalized database.

R

Rule-Based
Transformation

Deterministic field
mapping, UUID generation,
disease-specific logic

M

ML-Assisted
Mapping

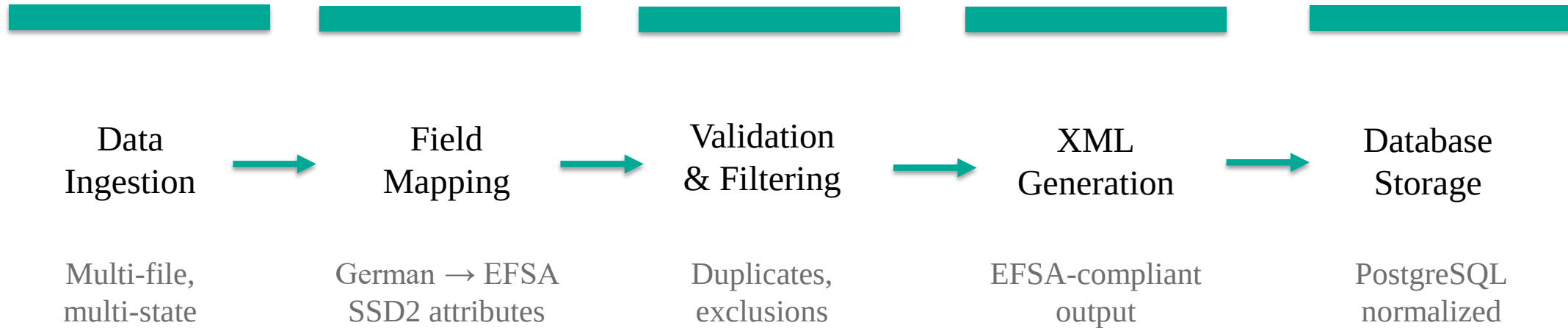
Fuzzy matching, TF-IDF
similarity, continuous
learning from validated
records

O

Manual
Overrides

Version-controlled config,
reviewed before
deployment, every use
logged for audit

Pipeline Architecture



Apache Airflow Orchestration · DAG-based scheduling · Task dependencies · Error handling & retries

Submitted XML

→ EFSA DCF

Excluded rows

→ Separate storage with reasons

Data Mapping: German Source → EFSA XML

German Source Field	EFSA XML Attribute	Mapping Method
Tierart (animal species)	sampMatCode	ML fuzzy + reference lookup
Lokalisation (location)	sampArea (NUTS code)	NUTS resolver + fuzzy match
Untersuchungsmethode	anMethCode	Rule-based (PCR / ELISA)
Resultat (pos/neg)	resQualValue	Direct mapping
Labordatum (lab date)	sampY / sampM / sampD	Date parsing + splitting
(generated)	resId (UUID)	Deterministic UUID generation

Each disease type (AI, ASF, WNF, Q Fever, Hepatitis E) has disease specific mapping rules

ML-Assisted Mapping

Species Resolution

1. Exact match against reference files (mammals, birds)
2. Partial / substring containment check
3. Fuzzy matching (SequenceMatcher, threshold>0.7)
4. Word-level splitting for compound names
5. Alternative translations via Google Translate library
6. Manual Override table for edge cases

Location / NUTS Resolution

NUTS 2021 · Level 3 (Kreis/Landkreis)

1. Direct NUTS code lookup from reference
2. Priority-based name matching
(exact > starts-with > contains > fuzzy)
3. Word splitting
4. Fuzzy matching with raised threshold (>0.8)
5. Abbreviation mapping
6. Fallback to manual override

Validation and Quality Control



Deterministic UUID Generation

resId is derived from stable identifying fields (disease, state, species, location, sample date) — not full content — so it persists across updates



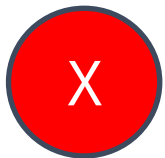
Duplicate Detection

Records with identical resId are compared field-by-field; differences trigger amType = Insert / Update / Delete (see slide 11)



Field Coverage Checks

Internal QC convention, not EFSA-mandated: coverage = non-null / total per critical field.
PASS $\geq 80\%$, WARNING $\geq 50\%$, FAIL $< 50\%$.



Exclusion Handling

Rejected records stored separately with detailed exclusion reasons. Source file and row number preserved for traceability

Normalized Database Model

dim_disease

disease_key
disease_name

fact_submission_xml

Res_id(UUID) · samp_y / samp_m
samp_mat_code · samp_area
an_meth_code · res_qual_value
am_type · is_deleted · timestamps

dim_species

species_code
name_en / name_de
animal_group

dim_source_file

source_filename
xml_filename
submission_year

dim_location

samp_area
nuts_label
lat / lon

fact_submission_raw

Original German values
source row

resid_changelog

old/new resid
change_type
timestamp

pipeline_run

run_timestamps
records_loaded

Update & Delete Handling

Previously submitted records must be updated or deleted if any discrepancy is found in data
The pipeline uses the amType attribute to manage record lifecycle:

Insert

amType = blank

New identifying key. Fresh resId assigned. Written to XML and database as a new submission.

Update

amType = 'U'

Same resId, but a non-key field (e.g. result value) differs from the stored version. Old values logged in resid_changelog.

Delete

amType = 'D'

Withdrawn record. is_deleted = true, deleted_at timestamp set. Record retained in database but excluded from active queries.

Results

239,000+

surveillance records processed and transmitted
From the 2025 reporting cycle into EFSA data warehouse

5+

Disease programs
(AI, ASF, WNF, Q Fever, Hep E)

16

Federal states
data integrated

0

Systematic mapping
errors in final submission

Based on internal mapping-verification checks and EFSA DCF acceptance without mapping-related rejections.

Conclusion

- ✓ An automated ML-supported data pipeline can reliably replace manual surveillance data processing at national scale.
- ✓ Rule-based transformations combined with ML-assisted mapping and manual overrides provide both precision and flexibility.
- ✓ Full traceability from EFSA XML back to original German source records enables auditing and quality assurance.
- ✓ The modular architecture supports extension to additional diseases, data sources and reporting years.

Acknowledgements



**Co-funded by
the European Union**

**OH4Surveillance is co-funded by the European Union
under Grant Agreement No 101132473.**

Disclaimer: Views and opinions expressed are those of the author(s) only and do not necessarily reflect those of the European Union or HaDEA. Neither the European Union nor the granting authority can be held responsible for them.

With thanks to:

The veterinary and surveillance authorities of the 16 German federal states, for providing the underlying disease surveillance data this pipeline processes; our colleagues at the Friedrich-Loeffler-Institut for their support and feedback throughout development; and the OH4Surveillance and EFSA OH Subgroup communities for valuable discussion.

THANK YOU FOR YOUR INTEREST & SUPPORT

QUESTIONS ?

Ruturaj Shelake · Friedrich-Loeffler-Institut
ruturaj.shelake@fli.de