

Sessione: Soluzioni tecnologiche per la tutela degli ecosistemi naturali: monitoraggio e analisi del rischio climatico e ambientale – 13 maggio 2026



ARGUS

Automatic Reporting Geodata UAS System

Automazione Python per la tracciabilità dei log e la visualizzazione in ambiente Esri

Francesca Di Nicola

Referente Unità Cartografia e GIS – ARPAE Emilia-Romagna





Chi siamo

ARPAE Emilia-Romagna

Arpae è l'ente tecnico-scientifico che supporta la **Regione Emilia-Romagna** in materia di tutela dell'ambiente.

La mission dell’Agenzia consiste nello svolgimento, per conto della Regione, delle attività inerenti alle autorizzazioni ambientali ed energetiche ed alle concessioni demaniali. Inoltre garantisce il presidio del territorio attraverso il monitoraggio di aria, acqua e suolo, l’attività di vigilanza e controllo e il supporto analitico specialistico fornito dal proprio laboratorio multisito.

Arpae è strutturata in più sedi per unità territoriale, per garantire un presidio diretto sul territorio.

La “P” della nostra sigla sta per prevenzione e indica, da parte della Regione Emilia-Romagna, una visione ampia attribuita alla complessità ambientale declinata affiancando, al presidio dei territori, anche la progettualità e il sostegno allo sviluppo sostenibile.

Cosa fa Arpae			
Monitoraggio ambientale	Vigilanza e controllo	Autorizzazioni e concessioni	Laboratori e attività analitica
Osservatorio clima e energia	Informazione e comunicazione	Servizio previsioni	Supporto alla protezione civile
Supporto tecnico progettuale	Pareri tecnici preventivi	Forniture dati	Studi, progetti e attività di ricerca
Promozione della sostenibilità	Progetti europei	Educazione alla sostenibilità	Autorizzazione integrata ambientale (AIA)

ARPAE Emilia-Romagna

Dalla Mission all'Azione: Il presidio del territorio e la prevenzione dei rischi richiedono una gestione d'avanguardia dei dati geospaziali.

Unità Cartografia e GIS all'interno del Servizio SITCG della Direzione

Tecnica presidia:

- Il monitoraggio del Consumo Suolo
- Gestione dei dataset geografici istituzionali.
- Standardizzazione dei flussi di lavoro geospaziali.
- Integrazione delle nuove tecnologie di rilievo (Droni e Sensori) nel sistema informativo dell'Agenzia.



L'ecosistema UAS in Arpae

Regia Unitaria

Governance Centrale: Coordinamento dell'intera attività UAS regionale, Gestione centralizzata dei registri di volo, pianificazione delle operazioni e monitoraggio della manutenzione dei droni

Operatività sul Territorio

- Network di piloti esperti distribuiti nelle sedi ARPAE
- Personale certificato per scenari Open e operazioni complesse in categoria Specific

Flusso di Lavoro

- Supporto alla pianificazione e gestione della missione
- Supporto e supervisione nell'elaborazione fotogrammetrica dei dati acquisiti



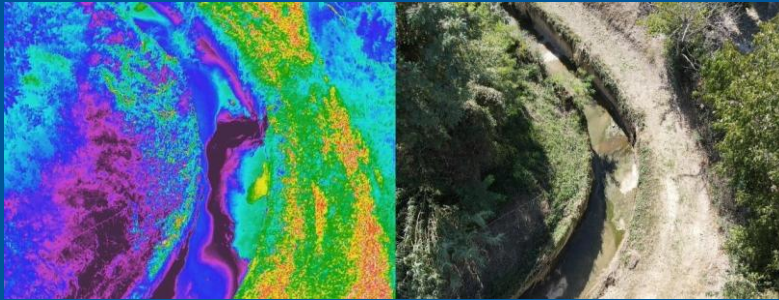
La nostra **sfida** è trasformare la complessa eterogeneità e lo mole di dati UAS in un **flusso di conoscenza organizzato, tracciabile** e pronto per l'**analisi geografica**.



L'esigenza

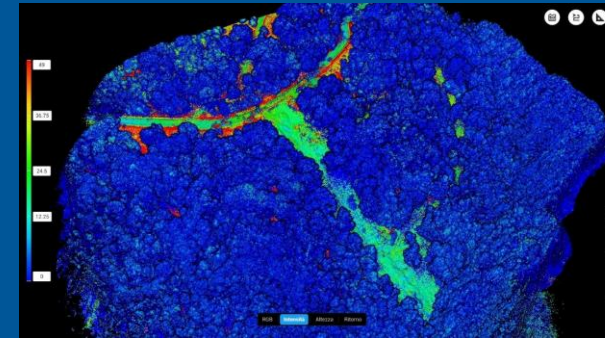
Esempi di operatività UAS in ARPAE

Attività di vigilanza e ispezione



Sversamenti di inquinanti nei corsi idrici
Elaborazione immagini IR

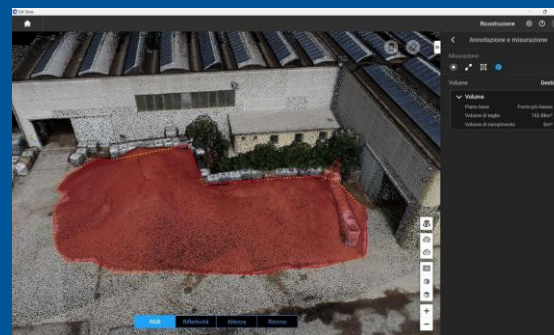
Attività di monitoraggio



Frana discarica in Toscana
Elaborazioni: Profili/Sezioni

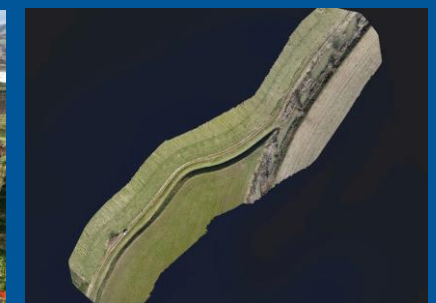


Cumuli effluenti zootecnici



Ispezioni in aziende

Misurazione di volumi con elevata precisione



Monitoraggio Metanodotto Sestino-Minerbio
Acquisizioni foto, Elaborazioni 2D e 3D

UAS Data Management: La sfida della complessità

Le Criticità del Monitoraggio UAS Massivo

- Frammentazione del Dato
- File LOG proprietari
- Gap rilievo - disponibilità dato



Esempi di "colli di bottiglia" nel workflow tradizionale

Rilievo post-alluvione: Il pilota scarica i dati su un disco locale, li elabora manualmente.

- lentezza e possibilità di errori, non immediata fruibilità a tutte le figure coinvolte.

Analisi volumetrica di una discarica: Impossibilità di risalire a sensori, quote e piloti.

- perdita di tempo in "indagini" d'archivio.

Analisi temporale di una zona costiera: I dati storici dei rilievi sono archiviati per missione.

- Impossibilità di risalire alle serie storiche e perdita di tempo nel "setacciare" i server aziendali.

Senza un **sistema di gestione**, il patrimonio informativo UAS rischia di rimanere un insieme di file isolati di conoscenza non condivisa, privi di tracciabilità e difficili da interrogare nel tempo.



La soluzione

Dalla Criticità alla Governance Digitale

Il progetto ARGUS nasce per rispondere puntualmente alle sfide del monitoraggio massivo:

● **Atomizzazione:** File isolati in cartelle sparse

● **Automazione:** Un unico database geografico strutturato

● **Log Proprietari:** Formati chiusi e illeggibili

● **Interoperabilità:** Dati GIS standard pronti all'uso

● **Latenza:** Settimane di attesa per i dati

● **Tempestività:** Flusso dato-mappa in tempo reale

● **Frammentazione:** Documenti scollegati tra loro

● **Fascicolo Digitale:** Telemetria e metadati uniti "By Design"

Caratteristiche Tecniche

Linguaggio: Python 3.12.13 con utilizzo di librerie geopandas e shapely.geometry.

Scalabilità: Il sistema è progettato per gestire flussi provenienti da sedi territoriali diverse in modo centralizzato.

Integrità: Riduzione dell'errore umano grazie all'eliminazione dei passaggi manuali di importazione e correzione dati.

Trasparenza e Condivisione: tutti i piloti hanno accesso alle missioni svolte

Roadmap di Sviluppo

Il progetto segue un'implementazione modulare per garantire solidità a ogni step:

1. **Ingestione Automatica:** Sviluppo della pipeline Python per estrarre lo strato informativo dai log grezzi.
2. **Fascicolo di volo:** Matching con i dati ripotati nel registro di volo
3. **Integrazione ESRI:** Sviluppo del WebGIS dedicato su piattaforma ArcGIS Enterprise e Vertigis Studio
4. **Scheduling & Backend:** Configurazione dell'ambiente di esecuzione e automazione dei processi (scheduling)
5. **Sharing dei Servizi:** Definizione dei workflow di pubblicazione automatica dei servizi mappa.
6. **Accessibilità ai Prodotti:** Integrazione diretta con i repository delle ortofoto e delle mesh 3D per la consultazione immediata dal WebGIS.
7. **Deployment:** Fase finale di collaudo operativo e messa in produzione.



La tecnologia

Tecnologie Esri utilizzate

ArcGIS Pro, ArcGIS Enterprise

Integrazioni con altri sistemi

Parser, Registro Voli Arpae, Vertigis studio

Standard e protocolli adottati

RNDT e Direttiva INSPIRE

Modalità di fruizione

WebGIS

Il "Motore" di ARGUS

Il Workflow di Ingestione (Step 1)

Parser Custom: Lo script Python agisce come interprete per i log proprietari, estraendo coordinate, quote, assetto (pitch/roll/yaw) e timestamp

```

"""**ID 09b31274**

Questa cella definisce la **funzione principale Python chiamata process_dji_logs_to_csv**. Il suo
obiettivo è gestire l'intero flusso di lavoro di elaborazione dei file di log DJI in modo
automatizzato. Trova i file di log .txt non ancora elaborati in una cartella specificata, chiama
funzione process_dji_log_file (definita nella cella 3f1b2781) per ciascuno di essi, verifica che
l'elaborazione sia andata a buon fine, aggiorna un file di stato per tenere traccia dei file già
elaborati e gestisce la copia locale temporanea dell'eseguibile dji-log-cli.
"""

# 3. Funzione principale
def process_dji_logs_to_csv(log_folder_drive, output_folder_drive, dji_log_cli_exe, desired_column):
    # Define the path to the temporary executable
    dji_log_cli_exe_temp = os.path.join("/tmp/", "dji-log-temp-exe")

    # --- NEW: Copy the executable from Drive to local /tmp/ *once* before processing files ---
    print(f"\nAttempting to copy executable from Drive ({dji_log_cli_exe}) to local temporary location ({dji_log_cli_exe_temp})...")
    try:
        if not os.path.exists(dji_log_cli_exe):
            print(f"Errore: L'eseguibile non esiste nel percorso di Drive specificato: {dji_log_cli_exe}. Impossibile procedere.")
            # Clean up the temporary executable if it somehow exists from a previous failed attempt
            if os.path.exists(dji_log_cli_exe_temp):
                os.remove(dji_log_cli_exe_temp)
            return # Exit if the executable isn't in the expected Drive location
    
```

```

# --- Convert to GeoDataFrame (Step 4) ---
print(f"Converting {file_name} to GeoDataFrame...")
# Ensure latitude and longitude columns exist and are numeric
lat_col = 'latitude' # Corrected to match the actual column name in the CSV
lon_col = 'longitude' # Corrected to match the actual column name in the CSV

if lat_col in current_df.columns and lon_col in current_df.columns:
    # Drop rows with NaN in lat/lon before creating geometry
    df_clean = current_df.dropna(subset=[lat_col, lon_col]).copy()

    # Ensure lat/lon are numeric types
    df_clean[lat_col] = pd.to_numeric(df_clean[lat_col], errors='coerce')
    df_clean[lon_col] = pd.to_numeric(df_clean[lon_col], errors='coerce')

    # Drop rows where conversion to numeric failed
    df_clean = df_clean.dropna(subset=[lat_col, lon_col])

    # NEW: Filter out rows with negative latitude or longitude
    initial_rows = len(df_clean)
    df_clean = df_clean[(df_clean[lat_col] >= 0) & (df_clean[lon_col] >= 0)]
    if len(df_clean) < initial_rows:
        print(f"Removed {initial_rows - len(df_clean)} rows with negative latitude or
        longitude from {file_name}.")

    if not df_clean.empty:
        # Create a list of Point objects from latitude and longitude columns
        geometry = [Point(xy) for xy in zip(df_clean[lon_col], df_clean[lat_col])]

        # Create the GeoDataFrame
        gdf = geopandas.GeoDataFrame(df_clean, geometry=geometry)
        print(f"Successfully created GeoDataFrame for {file_name}.")
    
```

Converte i dati grezzi in Feature Class, con generazione automatica della traccia di volo

Dal Log al Fascicolo di volo digitale

L'automazione del fascicolo di volo digitale (Step 2)

Grazie alla pipeline Python, il sistema integra automaticamente:

- ✓ Traccia di volo reale, quota media, velocità (estrapolati dai log).
- ✓ ID volo, pilota, drone, sensori utilizzati (estratti da registro di volo)

*Il Fascicolo di volo digitale non è un report,
è l'hub digitale della missione accessibile
via WebGIS*

```
##### ID 8jMPZI7qVh_b
Questa cella elabora e arricchisce i singoli shapefile filtrati. Riproietta i dati in un CRS standard
EPSG:25832 e aggiunge informazioni da un foglio Google. Il risultato sono nuovi shapefile
standardizzati (_E32.shp) pronti per un'ulteriore analisi.#####

import os
import pandas as pd
import geopandas
import math # For speed calculation

# --- Configuration ---
shapefile_input_folder = os.path.join(output_folder_drive, "shapefiles")
target_crs = "EPSG:25832"

# Define the final column order and properties for the shapefile
# (name, type, length, precision - for numeric type 'N')
final_schema_columns = [
    ("datetime", "C", 24, None),
    ("IDVolo", "N", 10, 0),
    ("latitude", "N", 19, 15),
    ("longitude", "N", 19, 15),
    ("height", "N", 19, 15),
    ("altitude", "N", 19, 15),
    ("speed", "N", 19, 15), # Horizontal speed
    ("xSpeed", "N", 19, 15),
    ("ySpeed", "N", 19, 15),
    ("zSpeed", "N", 19, 15),
    ("Drone", "C", 30, None),
    ("Luogo", "C", 50, None),
    ("Aborted", "C", 5, None),
    ("path", "C", 254, None),
    ("RGB", "C", 5, None),
    ("IR", "C", 5, None),
    ("LIDAR", "C", 5, None),
    ("MSP", "C", 5, None),
    ("Ortofoto", "C", 5, None),
    ("DEM", "C", 5, None),
    ("3D_model", "C", 5, None),
    ("Elab_Avanz", "C", 50, None),
    ("FilePianif", "C", 5, None)
]
```

Evoluzione del Sistema: WebGIS e Automazione

Dall'integrazione ESRI (Step 3) alla piena operatività (Step 4)



Step 3: Integrazione ESRI (In consolidamento)

- ✓ **ArcGIS Enterprise:** infrastruttura centrale che gestisce il database geografico, garantisce la sicurezza degli accessi e trasforma i dati in servizi web.
- ✓ **VertiGIS Studio:** framework applicativo che trasforma i servizi in interfacce utente avanzate.
- **Fascicolo di Volo Digitale: Il WebGIS diventa il punto di accesso unico.**

Step 4: Scheduling & Backend (in avvio)

- Automazione: Configurazione dell'ambiente di esecuzione server-side per lo scheduling degli script Python.
- Pipeline Ricorsiva: Passaggio a un sistema che monitora i repository e processa i nuovi log in background.
- Solidità di Sistema: Gestione centralizzata dei carichi di lavoro e monitoraggio automatico dei flussi di caricamento.



Verso ARGUS 2.0

Prossimi passi...



Step 5: Sharing dei Servizi

Definizione di workflow per la pubblicazione automatica di servizi mappa (REST/WMS).



Step 6: Accessibilità ai Prodotti

Accessibilità Totale: Integrazione dei prodotti fotogrammetrici nel portale:

- Mosaic Datasets: Per la consultazione di ortofoto ad altissima risoluzione.
- Scene Layers: Per l'esplorazione fluida di Mesh 3D e nuvole di punti da browser.

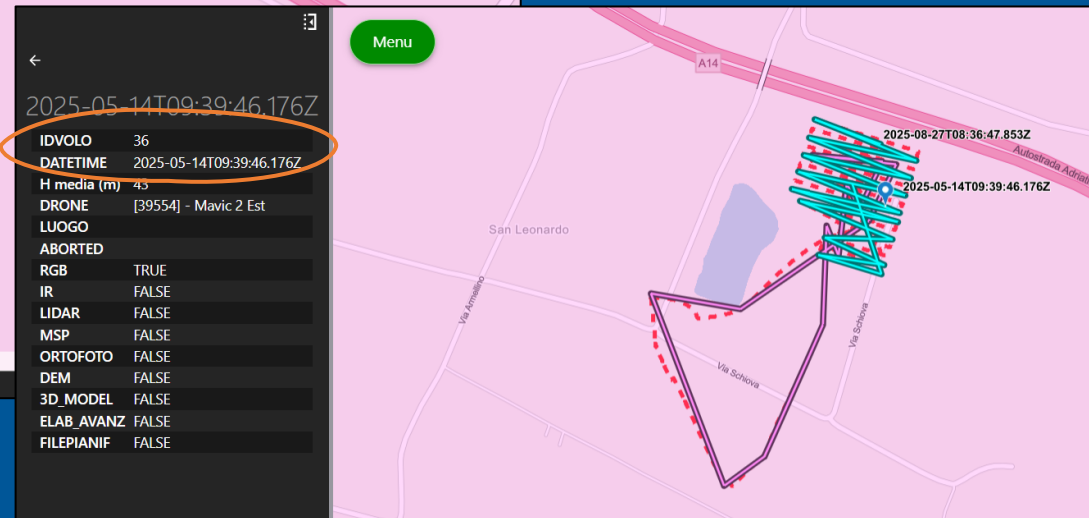
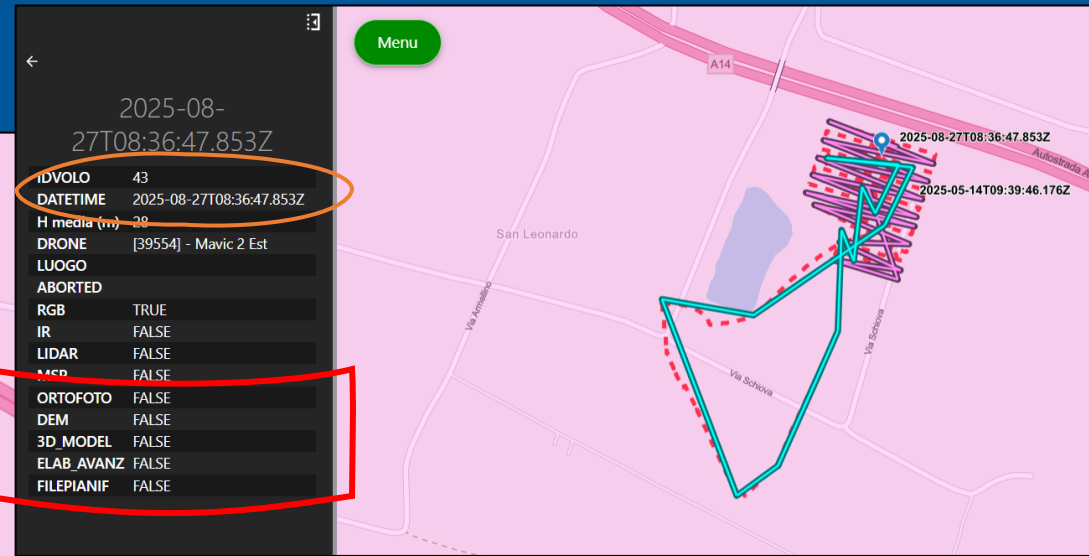
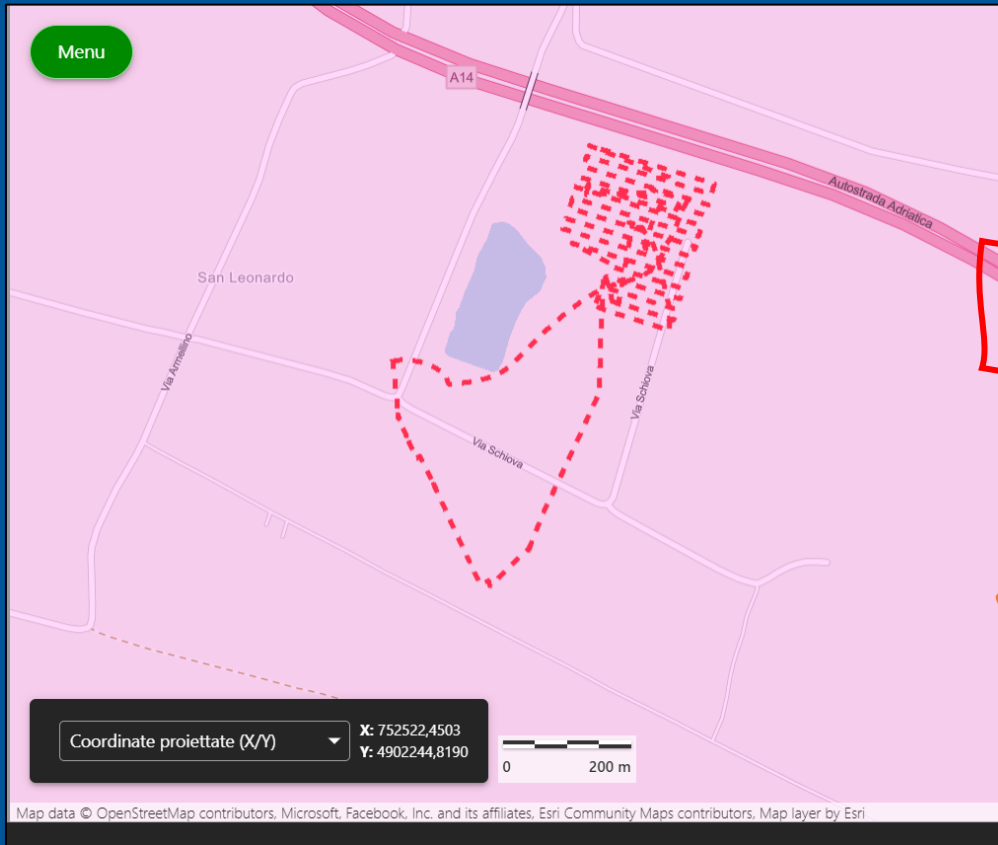


Step 7: Deployment

Collaudo finale e rilascio del sistema agli utenti.



Il risultato



ARGUS non è solo un WebGIS, ma è lo strumento di lavoro per l'analisi e la lettura del territorio

Evoluzioni future possono comprendere:

- **Integrazione AI** per il riconoscimento automatico di target ambientali.
- **Advanced Analytics**, ad esempio analisi predittiva dei cambiamenti morfologici basata sulle serie storiche.

Francesca Di Nicola

fdinicola@arpae.it

Agenzia regionale prevenzione, ambiente ed energia dell'Emilia-Romagna

Sede Legale Via Po, 5 | 40139 Bologna | tel. 051 6223811 - fax 051 541026 | e-mail: dirgen@pec.arpae.it | www.arpa.emr.it

Direzione Tecnica

Largo Caduti del Lavoro, 6 | 40122 Bologna | tel. 051 5281211 - fax 051 5281261

P.IVA e C.F. 04290860370 | e-mail: dirgen@pec.arpae.it |

