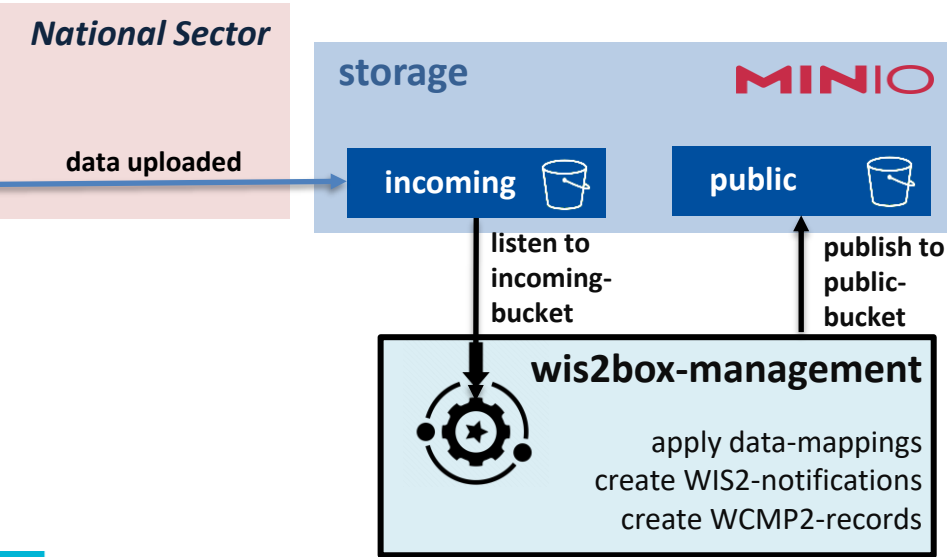


Ingesting and publishing data using wis2box

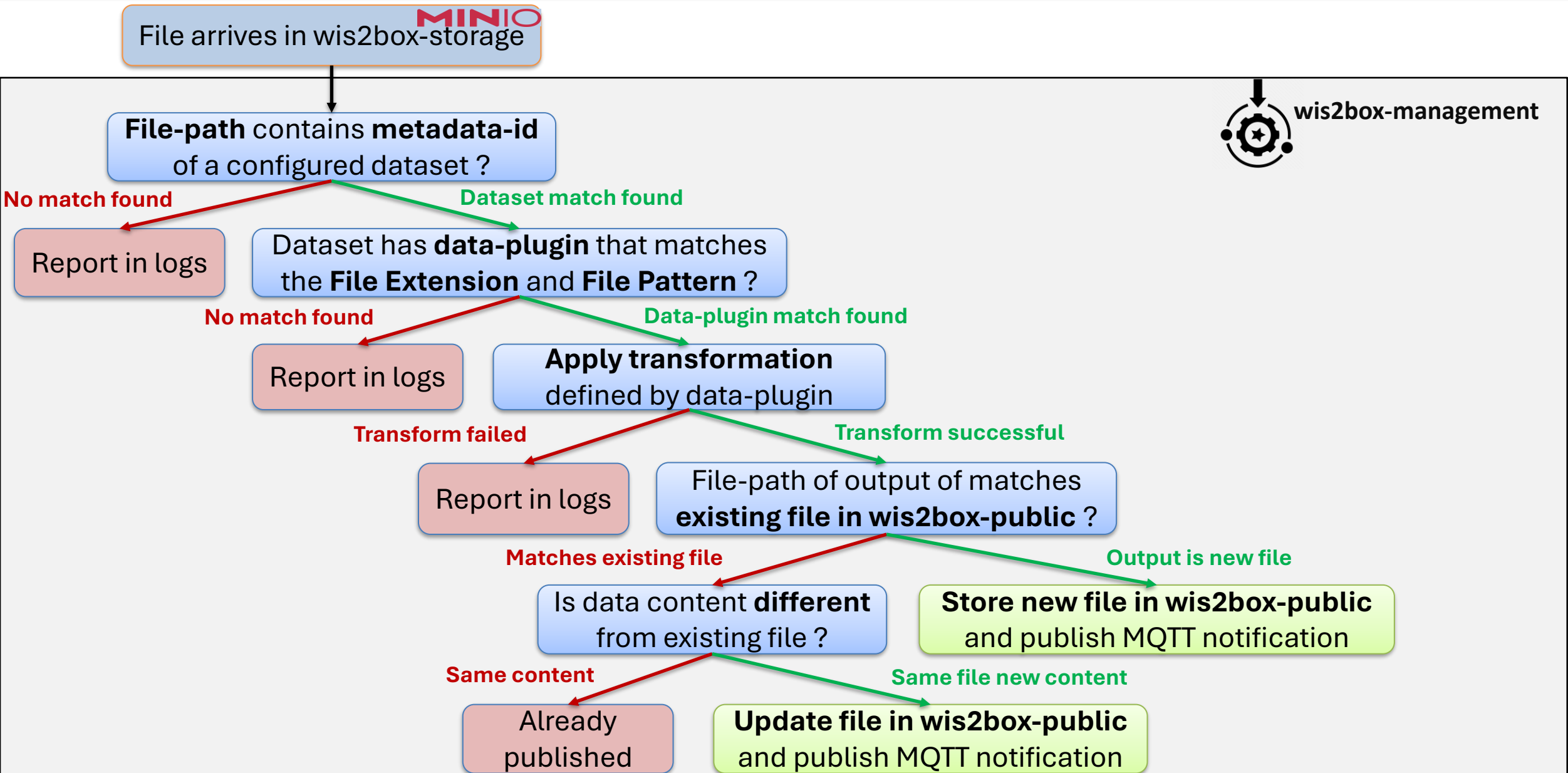
wis2box data ingest workflow



Dataset driven workflow triggered when wis2box-management receives a notification from the storage service

wis2box-management **matches incoming data** with datasets based on file-path
data plugins are applied to **transform & publish the data**

wis2box data mapping workflow



How to upload data to wis2box

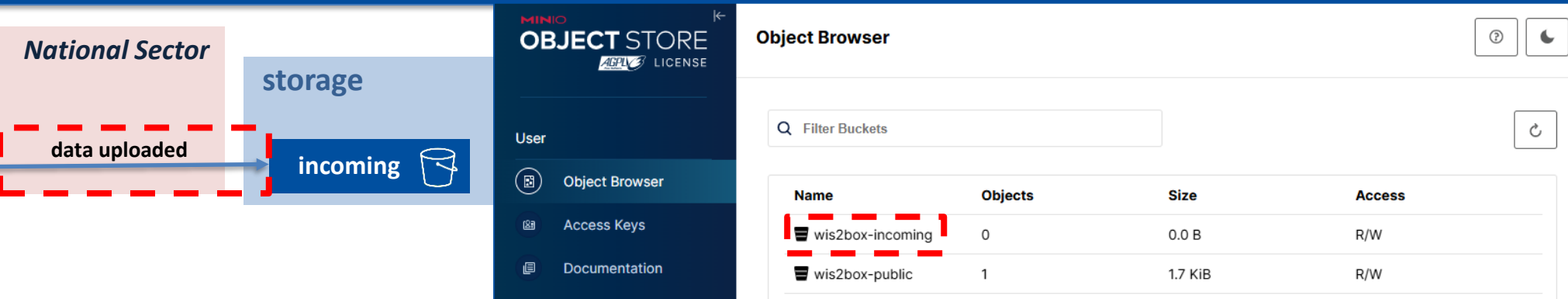


Data can be uploaded to wis2box-incoming in different ways
Optimal solution will depend on systems used in national sector

Options for ingesting data in wis2box:

- **Manual upload using [MinIO UI](#):** manually create path and upload data, useful for testing data-workflow
- **[Interfaces in wis2box-webapp](#):** manual submission using FM-12 form or manual file upload interface
- **[Scripts](#):** write code to upload data using MinIO client software
- **[Using SFTP protocol](#):** forward data to MinIO SFTP-endpoint

MinIO User Interface



The diagram on the left shows a flow from 'National Sector' to 'storage', with 'data uploaded' entering an 'incoming' bucket. The screenshot on the right shows the MinIO Object Browser interface. The 'wis2box-incoming' bucket is highlighted with a red dashed box in the table below.

Object Browser

Filter Buckets

Name	Objects	Size	Access
wis2box-incoming	0	0.0 B	R/W
wis2box-public	1	1.7 KiB	R/W

Data can be uploaded manually using the **MinIO web interface at `http://<your-host>:9001`**

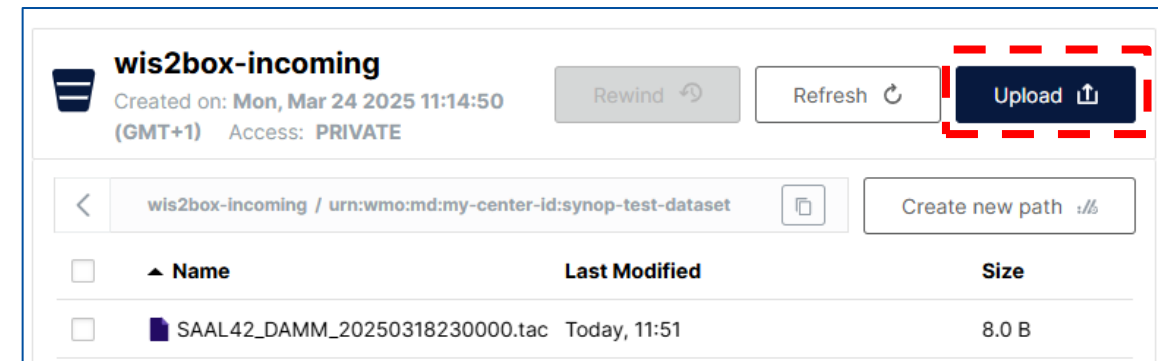
- **Useful for testing** your data mappings before setting up an automated workflow
- Login with `WIS2BOX_STORAGE_USERNAME` / `WIS2BOX_STORAGE_PASSWORD` defined in `wis2box.env`
- Create new path in `wis2box-incoming` bucket matching metadata-id and upload your file



Choose or create a new path

Current Path: wis2box-incoming

New Folder Path*



wis2box-incoming
Created on: Mon, Mar 24 2025 11:14:50 (GMT+1) Access: PRIVATE

< wis2box-incoming / urn:wmo:md:my-center-id:synop-test-dataset

<input type="checkbox"/>	Name	Last Modified	Size
<input type="checkbox"/>	SAAL42_DAMM_20250318230000.tac	Today, 11:51	8.0 B

For security do not enable external access on port 9001 nor share wis2box-storage credentials

Data ingestion script



Data can be uploaded using a minio client library

- Useful to setup custom data workflow:
 - Cronjob running Python script uploading files from local server
 - Query local database to extract CSV and upload result
- <https://min.io/docs/minio/linux/developers/minio-drivers.html>

```
import glob
import sys
from minio import Minio
```

Python example to upload a file using the MinIO client module:

```
filepath = '/home/wis2box-user/local-data/mydata.bin'
```

```
endpoint = 'http://wis2box-host:9000'
WIS2BOX_STORAGE_USERNAME = 'wis2box'
WIS2BOX_STORAGE_PASSWORD = 'XXXXXXXXXX'
```

```
client = Minio(
    endpoint=endpoint,
    access_key=WIS2BOX_STORAGE_USERNAME,
    secret_key=WIS2BOX_STORAGE_PASSWORD,
    secure=is_secure=False)
```

```
filename = filepath.split('/')[-1]
# path should match the metadata-id for corresponding dataset
minio_path = 'urn:wmo:md:my-center-id:synop-test-dataset'
client.fput_object('wis2box-incoming', minio_path+filename, filepath)
```

SFTP



Data can be uploaded using SFTP

- Useful for automating data-flow from vendors supporting SFTP

File protocol:
SFTP

Host name: mlimper.wis2.training Port number: 8022

User name: wis2box Password:

MinIO access over SFTP is enabled on port 8022

Login with WIS2BOX_STORAGE_USERNAME/WIS2BOX_STORAGE_PASSWORD

For security: restrict network access on port 8022 to known source-IP

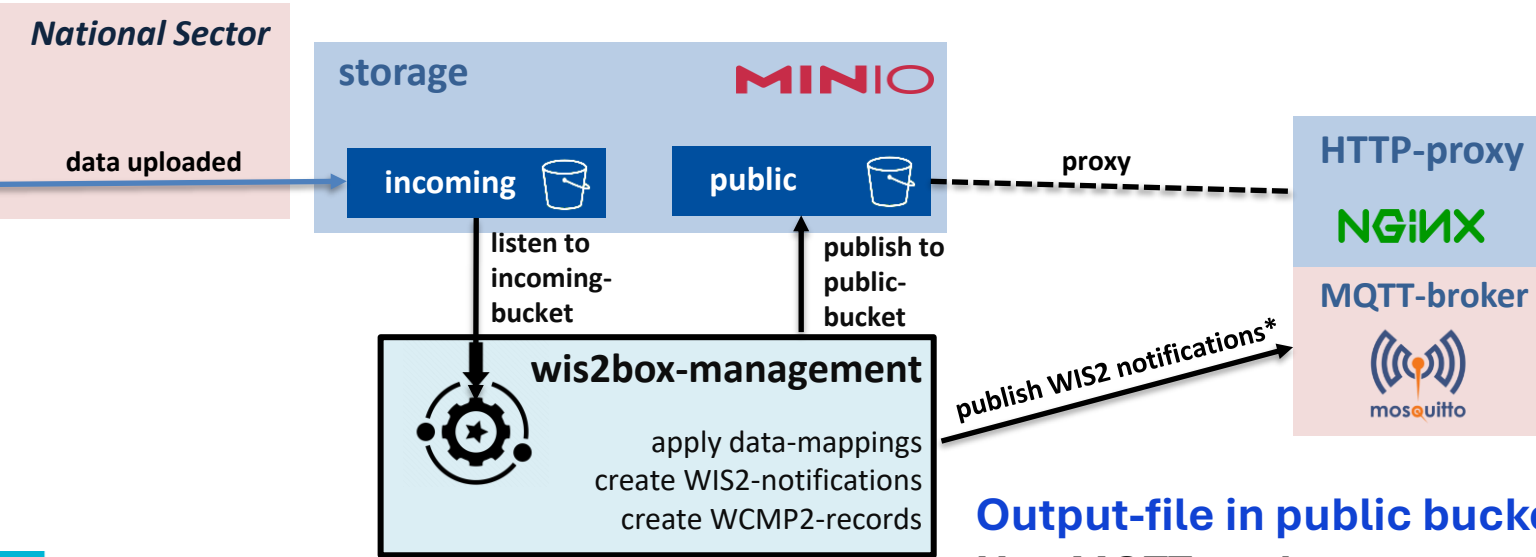
/				
Name	Size	Changed	Rig...	Ow...
..				
wis2box-public		3/24/2025 11:14:50 AM	---	0
wis2box-incoming		3/24/2025 11:14:50 AM	---	0

/wis2box-incoming/urn:wmo:md:my-center-id:synop-test-dataset/				
Name	Size	Changed	Rig...	Ow...
..				
SAAL42_DAMM_20250318230000.tac	1 KB	3/24/2025 11:51:36 AM	rw...	0

Use directory matching metadata-id or topic to trigger data ingestion workflow

Check published result

* Since the HTTP-proxy exposes wis2box-public bucket as '/data', WIS2 Notifications will use href="\$WIS2BOX_URL/data/..."



Output-file in public bucket results in WIS2-notification on MQTT
Use MQTT explorer to see the WIS2-notifications for your data

MQTT Connection mqtt://mlimper.wis2.training:1883/

Name: ☐ Validate certificate ☐ Encryption (tls)

Protocol: Host: Port:

Username: Password: ☐

MQTT Explorer Search...

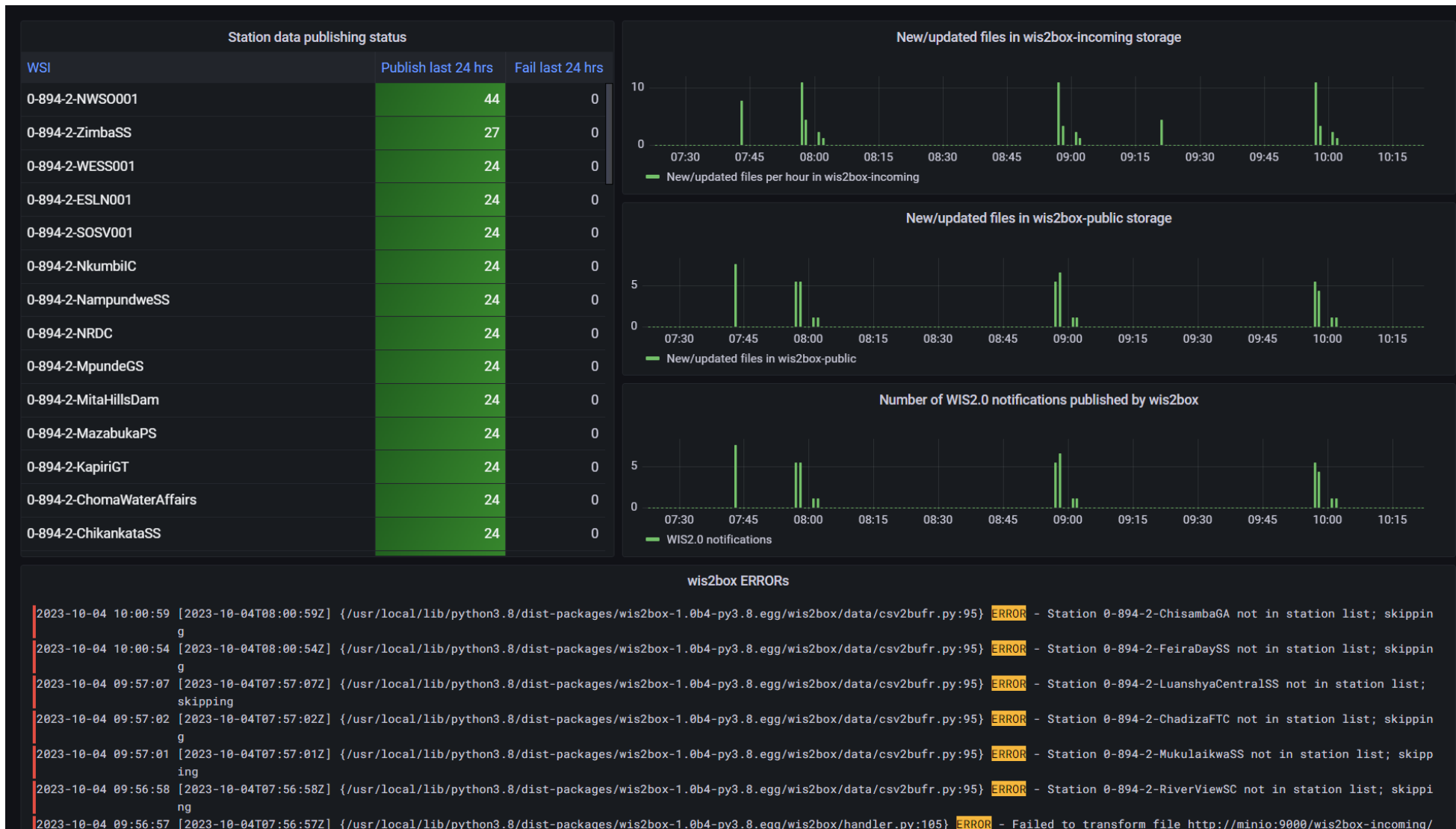
▼ mlimper.wis2.training

- ▼ origin
 - ▼ a
 - ▼ wis2
 - ▼ my-center-id
 - ▼ data
 - ▼ core
 - ▼ weather
 - ▼ surface-based-observations
 - synop = {"id": "89a4a332-8164-4afb-94a7-42ed3175c282"}

Monitoring wis2box data flow (and check errors)

Grafana home dashboard available at <http://<wis2box-host-address>:3000>

Grafana data-sources: **Prometheus** (time-series metrics) and **Loki** (logs)



'explore' option In Grafana sidebar

Exploring logs stored in Loki

The screenshot displays the Grafana Explore interface for querying logs in Loki. The interface is divided into several sections:

- Top Bar:** Includes the 'Explore' button, a dropdown menu showing 'wis2box-loki', and buttons for 'Split', 'Add to dashboard', 'Last 1 hour', 'Run query', and 'Live'.
- Query Patterns:** A section with a dropdown for 'Query patterns', a toggle for 'Raw query', and a 'Give feedback' link.
- Labels:** A section for filtering logs by labels. It shows 'container_name' set to 'wis2box-management'. A yellow box highlights this section, and a yellow arrow points to it with the text 'wis2box-loki as a data source'.
- Line contains:** A section for filtering logs by text. It has a 'Text to find' input field and a '+ Operations' button. A yellow arrow points to this section with the text 'select container'.
- Raw query:** A section for entering a raw query. It shows a query: `{container_name="wis2box-management"} |= ''`. A yellow arrow points to this section with the text 'run query'.
- Log volume:** A bar chart showing the volume of logs over time. The x-axis represents time from 13:00 to 13:55, and the y-axis represents the number of logs (0 to 500).
- Logs:** A section for displaying the raw log data. It includes a 'Time' toggle, a 'Unique labels' toggle, a 'Wrap lines' toggle, a 'Prettyfy JSON' toggle, a 'Dedup' toggle, and a 'Display results' button. The logs are sorted by 'Newest first'.

The logs displayed are as follows:

```
> 2023-10-04 13:57:09 [2023-10-04T11:57:09Z] {/usr/local/lib/python3.8/dist-packages/wis2box-1.0b4-py3.8.egg/wis2box/pubsub/subscribe.py:46} INFO - Data processed
> 2023-10-04 13:57:09 [2023-10-04T11:57:09Z] {/usr/local/lib/python3.8/dist-packages/wis2box-1.0b4-py3.8.egg/wis2box/data/base.py:195} INFO - Publishing output data
> 2023-10-04 13:57:09 [2023-10-04T11:57:09Z] {/usr/local/lib/python3.8/dist-packages/wis2box-1.0b4-py3.8.egg/wis2box/data/csv2bufr.py:95} ERROR - Station 0-894-2-LuanshyaCentralSS not in station list; skipping
> 2023-10-04 13:57:09 [2023-10-04T11:57:09Z] {/usr/local/lib/python3.8/dist-packages/wis2box-1.0b4-py3.8.egg/wis2box/metadata/station.py:257} INFO - Validating WIGOS Station Identifier: 0-894-2-LuanshyaCentralSS
> 2023-10-04 13:57:09 [2023-10-04T11:57:09Z] {/usr/local/lib/python3.8/dist-packages/csv2bufr/_init_.py:909} INFO - 2023-10-04T11:57:09.335450+00:00{'id': 'WIGOS_0-894-2-LuanshyaCentralSS_20231004T115400', 'geometry': {'type': 'Point', 'coordinates': [28.41, -13.14]}, 'properties': {'md5': '272c3fedeaee11d6463e34afb0abd5d1', 'wigos_station_identifier': '0-894-2-LuanshyaCentralSS', 'datetime': datetime.datetime(2023, 10, 4, 11, 54), 'origin_ating_centre': 0, 'data_category': 0}, 'result': {'code': 1, 'message': '', 'errors': []}}
> 2023-10-04 13:57:08 [2023-10-04T11:57:08Z] {/usr/local/lib/python3.8/dist-packages/wis2box-1.0b4-py3.8.egg/wis2box/pubsub/subscribe.py:43} INFO - Processing http://minio:9000/wis2box-incoming/zmb/zambia_met_service/data/core/weathe
r/surface-based-observations/synop/CR1000X_48229_20231004T135500Z.csv
> 2023-10-04 13:57:08 [2023-10-04T11:57:08Z] {/usr/local/lib/python3.8/dist-packages/wis2box-1.0b4-py3.8.egg/wis2box/pubsub/subscribe.py:49} INFO - Public filepath: wis2box-public/2023-10-04/wis/zmb/zambia_met_service/data/core/weathe
r/surface-based-observations/synop/WIGOS_0-894-2-NWS0001_20231004T115400-105.geojson
```

Data Ingestion and Monitoring

Data-driven workflow triggered when data is uploaded to the MinIO service:

- **MinIO-interface** to upload data manually for **testing** the workflow
- Interfaces in wis2box-webapp: FM-12 form, manual file upload
- **Automate** data upload using **scripts** or forwarding over **SFTP**

Status of the data ingestion can be **monitored** in the [Grafana Dashboard](#)

After configuration, **ensure the system is maintained and remains operational :**

- Check for any errors (view Grafana dashboard)
- Add/update/remove datasets
- Add/update/remove stations
- Review the data content being published

Practical exercises

PRACTICAL EXERCISES:

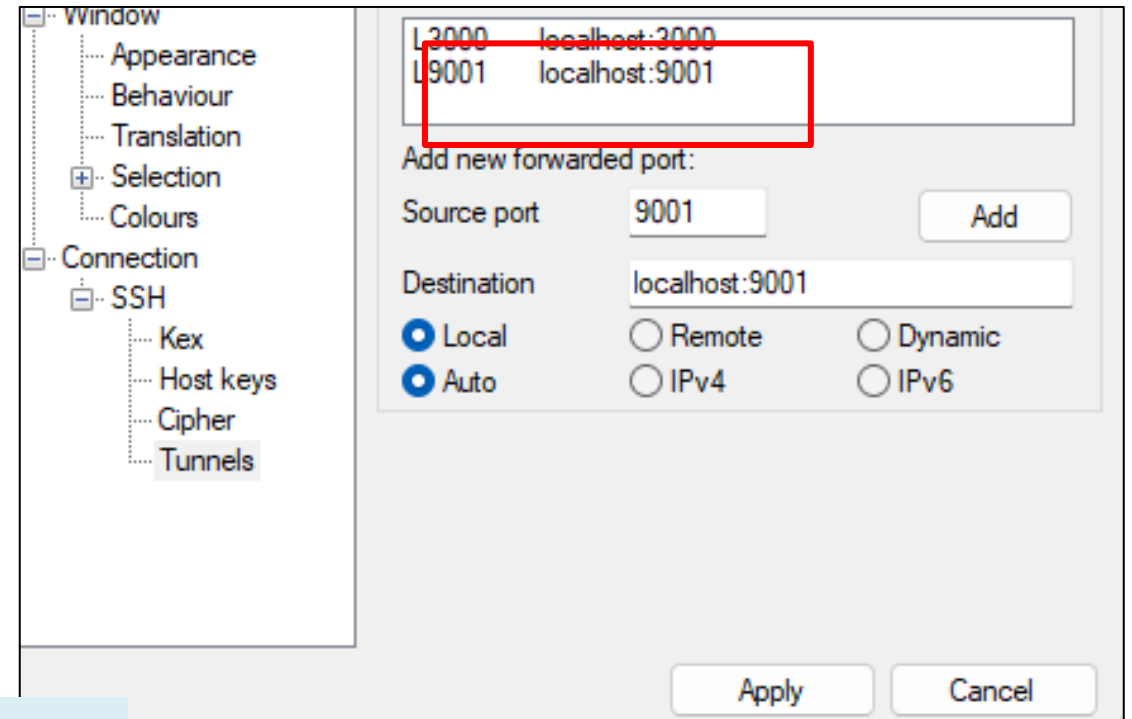
<https://training.wis2box.wis.wmo.int/practical-sessions/ingesting-data-for-publication/>

Ports 3000/9001 for Grafana and MinIO interface are NOT available externally on European Weather Cloud, **please use SSH tunnels to access Minio and Grafana**

How to setup SSH Tunnels in PuTTY:
right click on top of window and select
“Change settings”

Change Settings...

Go to Connection->SSH->Tunnels
Add tunnels for ports 3000 and 9001
click “Apply”
Access MinIO at <http://localhost:9001>
Access Grafana at <http://localhost:3000>



Thank you

wmo.int



WORLD
METEOROLOGICAL
ORGANIZATION

