Introducing WIS2 in a box (wis2box)

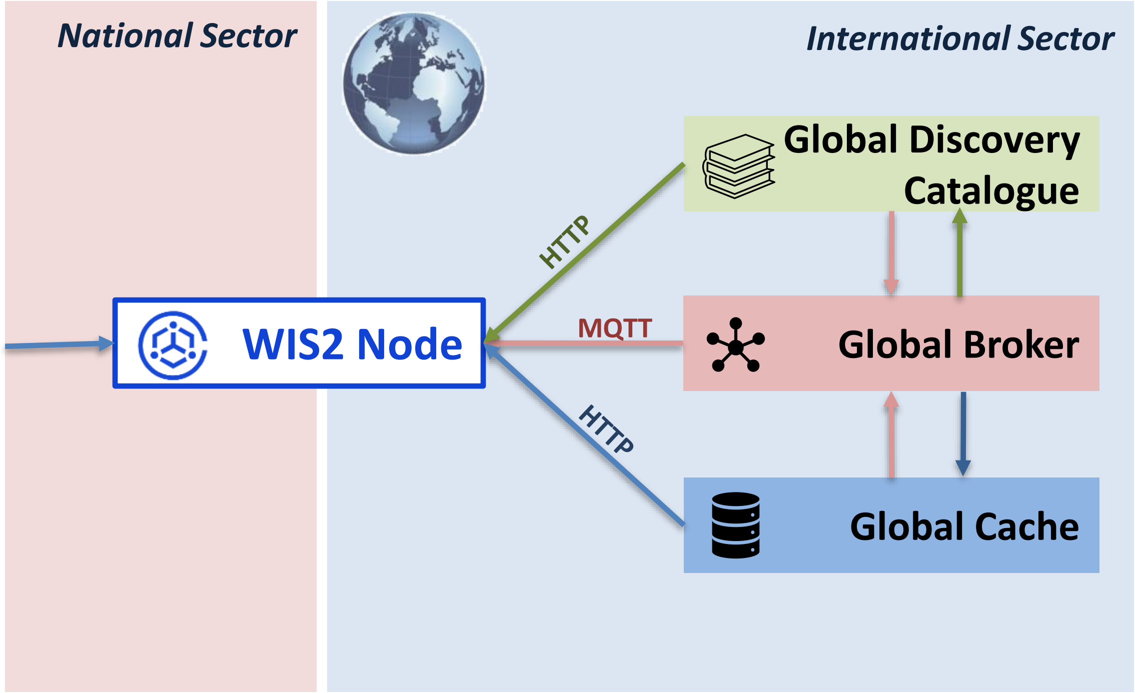
WORLD
METEOROLOGICAL
ORGANIZATION

1950-2025
SCIENCE *for* ACTION

Reminder: What is a WIS2 Node?

A WIS2 node is composed of 2 endpoints that need to be exposed over the public internet:

- ***MQTT broker***: to publish WIS2 notifications for metadata and data
- ***HTTP data server***: to enable the download of data files and metadata records



Global Discovery Catalogues download all valid **WCMP2 records** from the HTTP-endpoint for notifications on topic=*origin/a/wis2/+/metadata*

Global Brokers subscribe to topic=*origin/a/wis2/<centre-id>/#* on the WIS2 Node MQTT broker, **and republishes** all valid WIS2-notifications

Global Caches download data from topic=*origin/a/wis2/+/data/core/#* **and republish** *on* topic=***cache/a/wis2/+/data/core/#***

MQTT channel defined by the **WIS2 Topic Hierarchy (WTH)** standard

Discovery Metadata records defined by **WCMP2** standard

MQTT payload defined by the **WIS2 Notification Message (WNM)** standard

WORLD METEOROLOGICAL ORGANIZATION

What is WIS2 in a box?

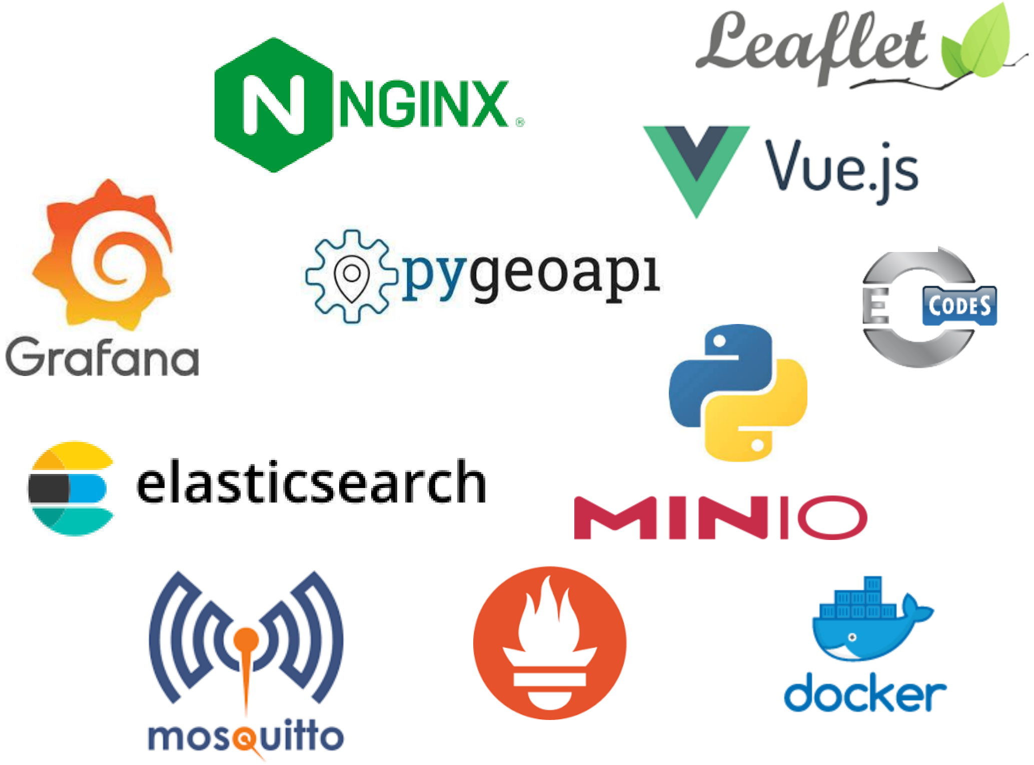
- WIS2 in a box (wis2box) is a **Reference Implementation of a WIS2 Node**
- Designed to be **cost-effective** and **low-barrier** to operate*
- Developed as Docker Compose stack using existing **Free and Open Source** software and wis2box-specific components
  - wis2box source code at: https://github.com/World-Meteorological-Organization/wis2box
- Developed by WMO together with Canada to **accelerate the WIS2 implementation**
- Currently over 50 WMO-Members are using wis2box to share data on WIS2

*wis2box hosting requirements:

- minimum 2 vCPUs with 4GB Memory and 24GB of local storage
- requires Python, Docker and Docker Compose pre-installed
- HTTP and MQTT ports routed to a publicly accessible address
- See documentation at https://docs.wis2box.wis.wmo.int

WIS2 in a box is Free and Open

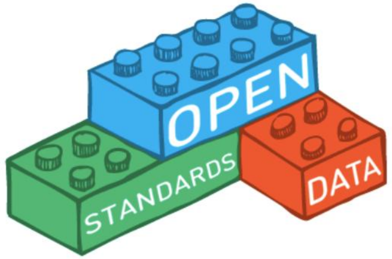
Free and Open Source Software



Open Standards

W3C®
ISO
I E T F®

- MQTT
- GeoJSON
- OGC APIs

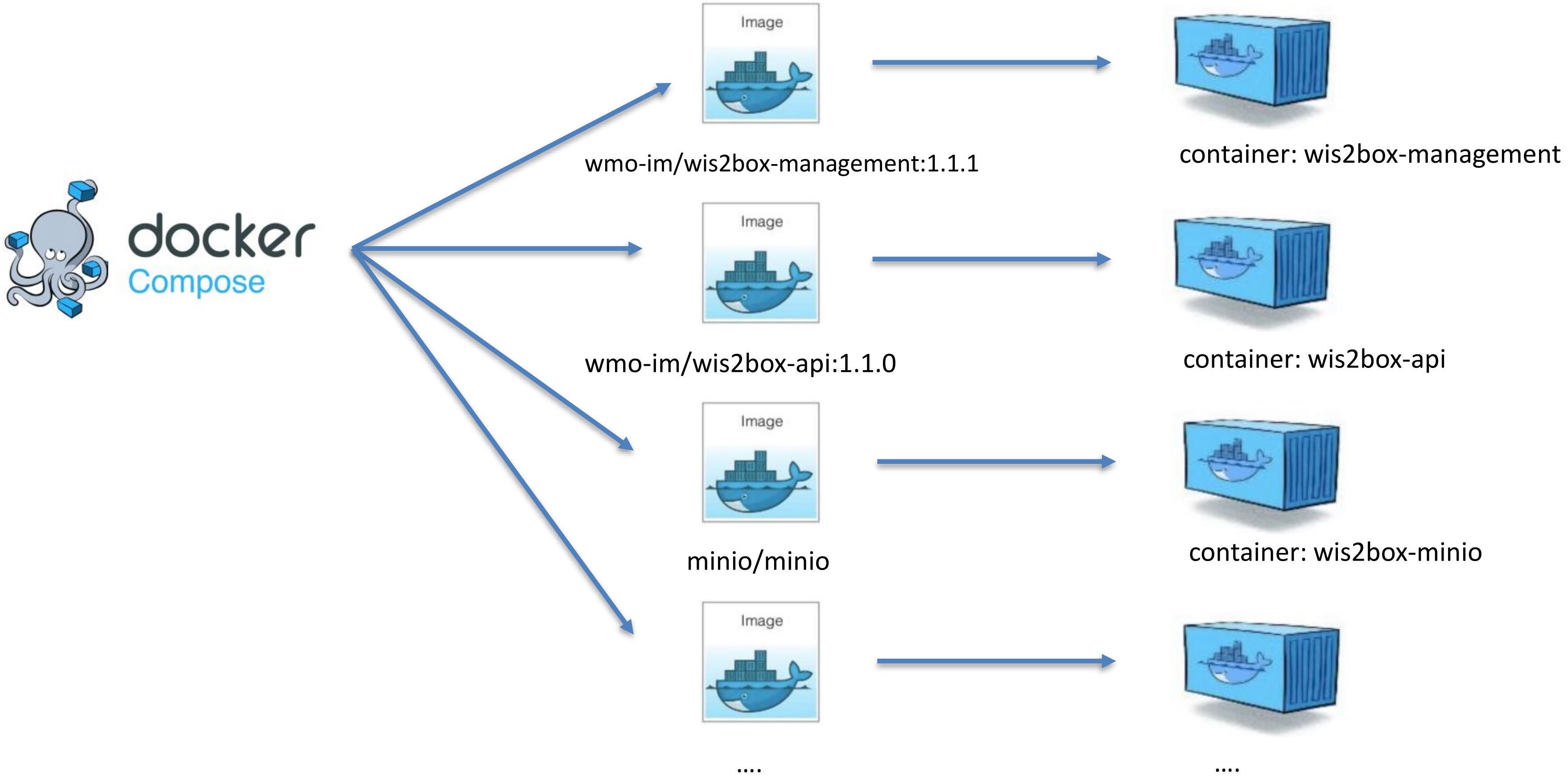


Docker and Docker Compose

wis2box uses Docker and Docker Compose to define a set of services

Using pre-built Docker images to create containers providing a specific service

NOTE: you do not need to be familiar with Docker to run wis2box



wmo-im/wis2box-management:1.1.1

container: wis2box-management

wmo-im/wis2box-api:1.1.0

container: wis2box-api

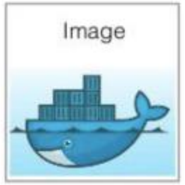
minio/minio

container: wis2box-minio

....

....

Docker and Docker Compose



Why is wis2box composed as a set of Docker containers?

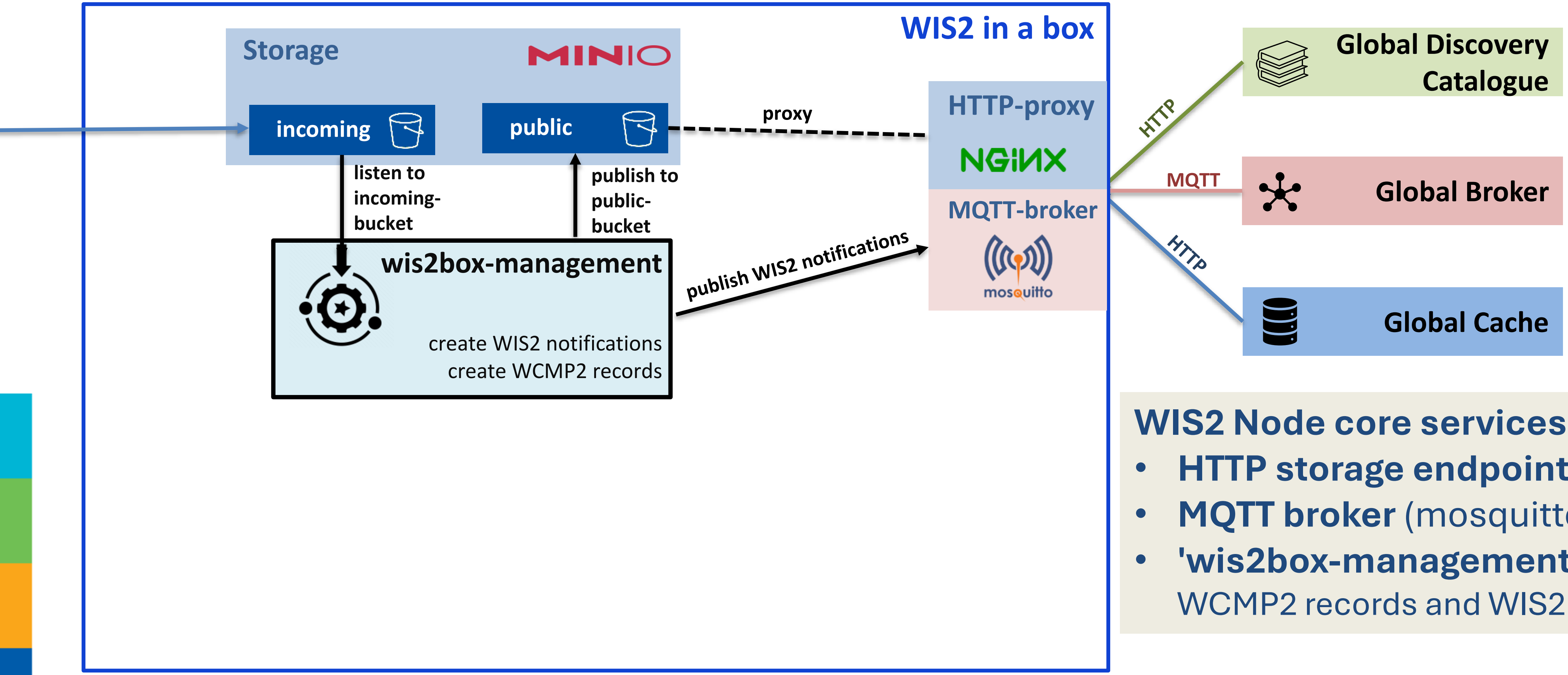
- Docker containers contain all necessary dependencies, libraries, and binaries required to run the service
- Docker containers run on any system with Docker installed, regardless of underlying hardware or operating system
- Docker containers provide process and resource isolation, enhancing security

The Python script ‘wis2box-ctl.py’ provides a wrapper around Docker Compose commands to interact with wis2box

Software required on the host to run wis2box:

- **Python**
- **Docker**
- **Docker Compose**

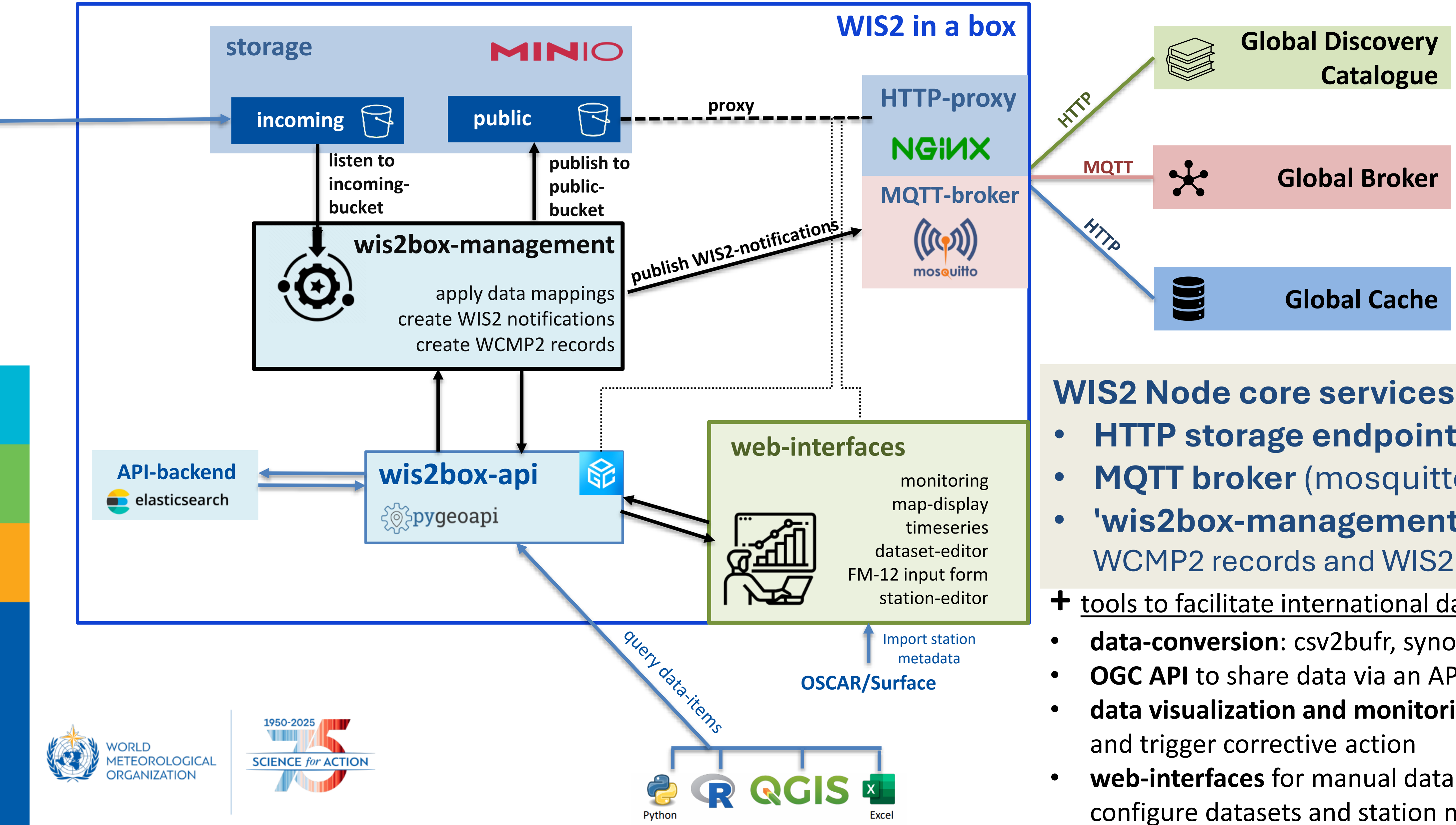
WIS2 in a box core services



WIS2 Node core services:

- **HTTP storage endpoint** (MinIO)
- **MQTT broker** (mosquitto)
- **'wis2box-management':** create WCMP2 records and WIS2 notifications

WIS2 in a box services



WIS2 Node core services:

- **HTTP storage endpoint** (MinIO)
- **MQTT broker** (mosquitto)
- **'wis2box-management':** create WCMP2 records and WIS2 notifications

+ tools to facilitate international data sharing:

- **data-conversion**: csv2bufr, synop2bufr, bufr2bufr
- **OGC API** to share data via an API
- **data visualization and monitoring** to detect issues and trigger corrective action
- **web-interfaces** for manual data input (FM-12/CSV), configure datasets and station metadata

WORLD METEOROLOGICAL ORGANIZATION

1950-2025 75 SCIENCE for ACTION

wis2box-api (Geospatial Web API)

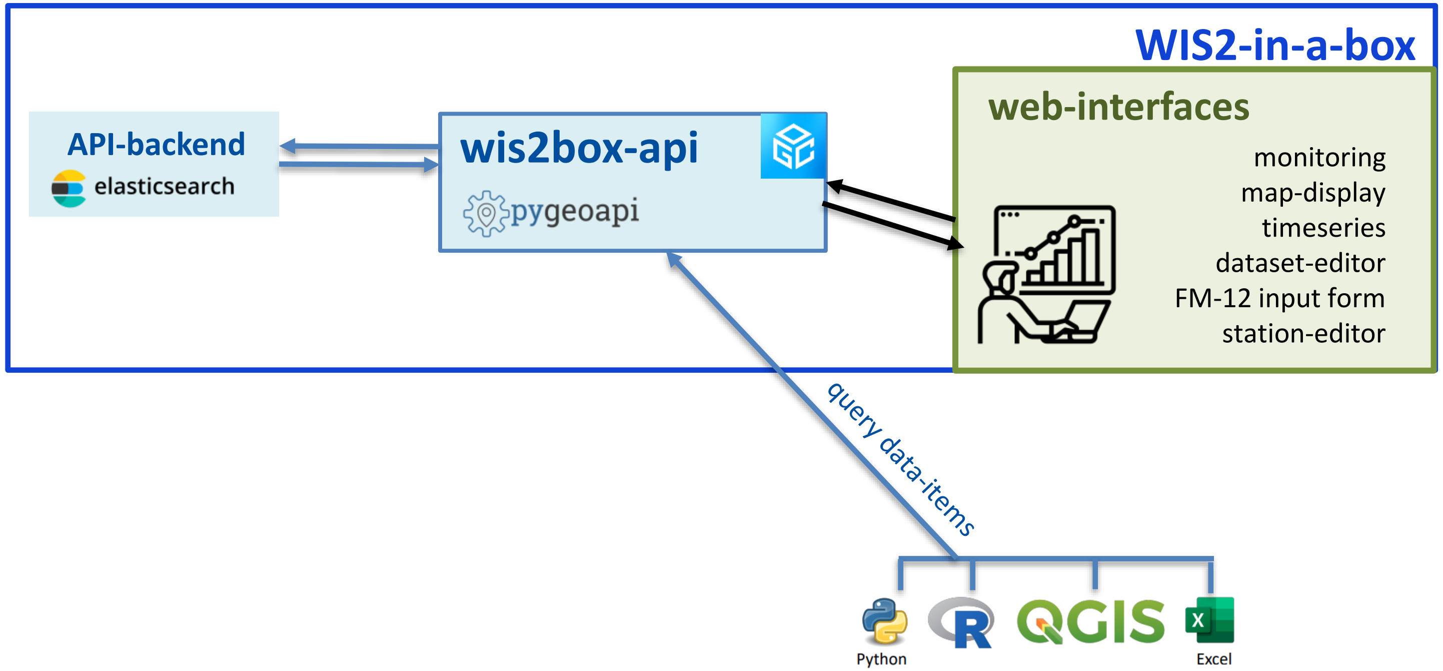
wis2box-api provides a **Application Programming Interface** (API) within wis2box

built using **pygeoapi**: a Free and Open Source project that provides an OGC Reference Implementation of the **OGC API** standards:

- OGC API – Processes to provide additional data processing functionality
- OGC API – Features to datasets/collections stored in backend
- OGC API – Records for WCMP2 records stored in backend

Open Geospatial Consortium.

wis2box-api enables users and web-interfaces to query data items programmatically using a Geospatial Web API



Summary

wis2box is a Free and Open Source Reference Implementation of a WIS2 Node

- Developers can freely use components used inside of wis2box to adapt existing systems to be WIS 2.0 compliant
- Source code: https://github.com/World-Meteorological-Organization/wis2box
- **Feedback by the community is appreciated to help improve wis2box**

wis2box is software not hardware

- To be deployed on a host with Python, Docker and Docker Compose pre-installed
- Documentation: https://docs.wis2box.wis.wmo.int

Setting up a WIS2 Node using wis2box-software

Host requirements for running wis2box

Running a WIS2 Node using the wis2box-software requires a suitable host

Since wis2box was developed to be run as a set Docker-containers, wis2box can be run on any server with Python, Docker and Docker Compose installed

The recommended OS is Ubuntu (20.04 LTS, 22.04 LTS or 24.04 LTS)

https://docs.wis2box.wis.wmo.int/en/latest/user/getting-started.html

Software dependencies:

- Python 3.8 or higher
- Docker Engine 20.10.14 or higher
- Docker Compose 2.0 or higher

Minimum Hardware requirements:

- 2 vCPUs, x86_64 architecture
- 4GB Memory
- 24GB storage

When choosing the environment for the wis2box, consider the following:

- The wis2box-host should have **good Internet connectivity** to download the required Docker images used in the wis2box stack and to enable reliable data download from subscribers
- For the wis2box instance to function as a WIS2 Node, HTTP and MQTT ports on the instance need to be accessible over the public internet -> network traffic to be routed over a **public IP**
- Ensure that the system providing input data can access the wis2box instance

Initializing wis2box on your student VM

For the purpose of this training, host-instances have been provided on the European Weather Cloud

Use PuTTy (or any other SSH client) to login to your instance using the instructions received by e-mail

PRACTICAL EXERCISE:
Proceed with the following exercise to start the initialize wis2box
https://training.wis2box.wis.wmo.int/practical-sessions/initializing-wis2box/

Thank you

wmo.int

WORLD
METEOROLOGICAL
ORGANIZATION

1950-2025
SCIENCE *for* ACTION