Achieving the FAIR vision in imaging workshop

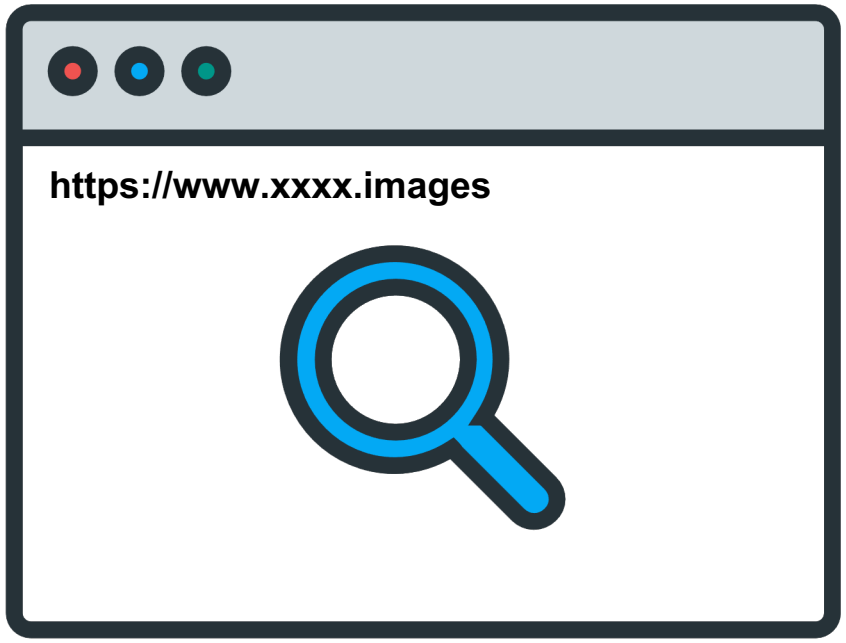
ELMI 2022

OME Team

HORIZON 2020
UKRI Biotechnology and Biological Sciences Research Council
University of Dundee
OME
wellcome
EMBL-EBI
EURO BIOIMAGING
GLOBAL BIOIMAGING growing collaboration

Motivation

Have you ever wanted to find existing data to reuse ?



The Image Data Resource (IDR)

- **Public access**
- **Reference datasets** - complete datasets containing molecular and functional annotations, associated with an existing or upcoming publication.
- **Study integration** - integrating studies or datasets with other datasets via **genes**, **compounds** or **phenotypes**.
- **Curated metadata**
- **Cloud re-analysis**

Workshop content

- Explore images and metadata **in Image Data Resource (IDR)**
- **Fetch and re-analyze** programatically using:
  - Proprietary file format and OMERO API
  - Cloud-optimized format (OME-NGFF)

How to reanalyze IDR data ?

- Download … (?)
  - Large data
  - Metadata not included
- Use Application Programming Interface (API)
  - Analysis close to data
  - Only what is needed is fetched
  - Image file format agnostic
  - All metadata available, easy to query
  - **Example workflows and analysis envs**

Suggested workflow

- **Find data of interest in IDR**
- **Set-up analysis environment**
- **Run examples to access IDR data**
- Edit the code to add your own analysis

YouTube **openmicroscopy** channel:

1. **Exploring IDR**
2. **IDR Analytical workflows** playlists

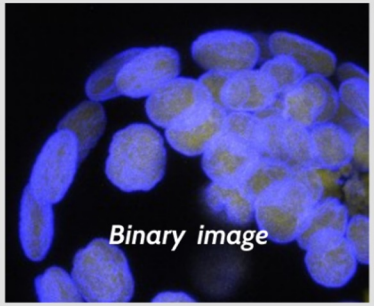
Materials

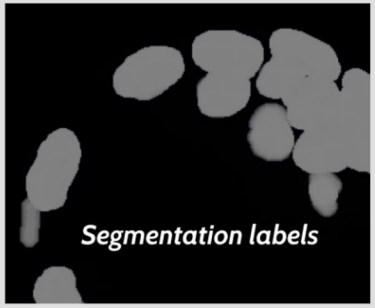
- Link to the walkthrough pdf https://downloads.openmicroscopy.org/presentations/2022/ELMI/

- Videos
- **openmicroscopy** YouTube channel
- IDR homepage
- I2K video (minute 28) for NGFF

Analysis using OMERO API

Data stored in IDR as .tiff with OMERO ROIs





Python-based analysis environment

Notebook with StarDist

Load Image with labels from IDR, analyze using StarDist and compare results

The notebook shows how to load an IDR image with labels.

The image is referenced in the paper "NesSys: a novel method for accurate nuclear segmentation in 3D" published August 2019 in PLOS Biology: https://doi.org/10.1371/journal.pbio.3000388 and can be viewed online in the Image Data Resource.

In this notebook, the image is loaded together with the labels and analyzed using StarDist. The StarDist analysis produces a segmentation, which is then viewed side-by-side with the original segmentations produced by the authors of the paper obtained via the loaded labels.

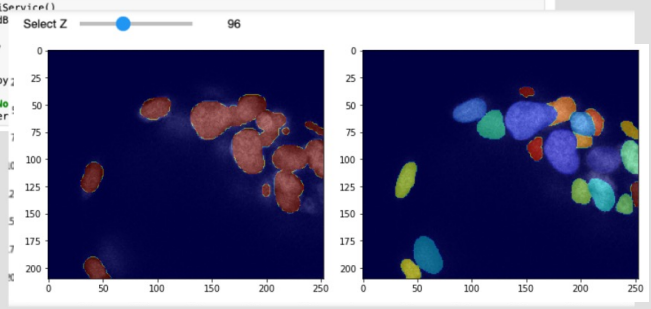
Labels have been saved as mask.

```
In [9]: from omero_zarr import masks

In [10]: roi_service = conn.getRoiService()
         result = roi_service.findB

         dims = (image.getSizeT(),
         shapes = []
         for roi in result.rois:
             shapes.append(roi.copy

         saver = masks.MaskSaver(No
         labels, fillColors, proper
```



Segmentation labels from original

Segmented by StarDist in the Notebook

Notebook segments images using StarDist and produces new segmentation labels

Notebook compares the labels

Build analysis environments

- Go to https://github.com/ome/omero-guide-python

- 1. Cloud-based services: Click on the badge
  - MyBinder 

  - Colab 

Build analysis environments

- 2. Locally on your computer
  - 2a) **repo2docker**
  - 2b) **conda** (and Jupyter).

Details in the README of omero-guide-python

What is OME-NGFF ?

Brief Communication | Open Access | Published: 29 November 2021

OME-NGFF: a next-generation file format for expanding bioimaging data-access strategies

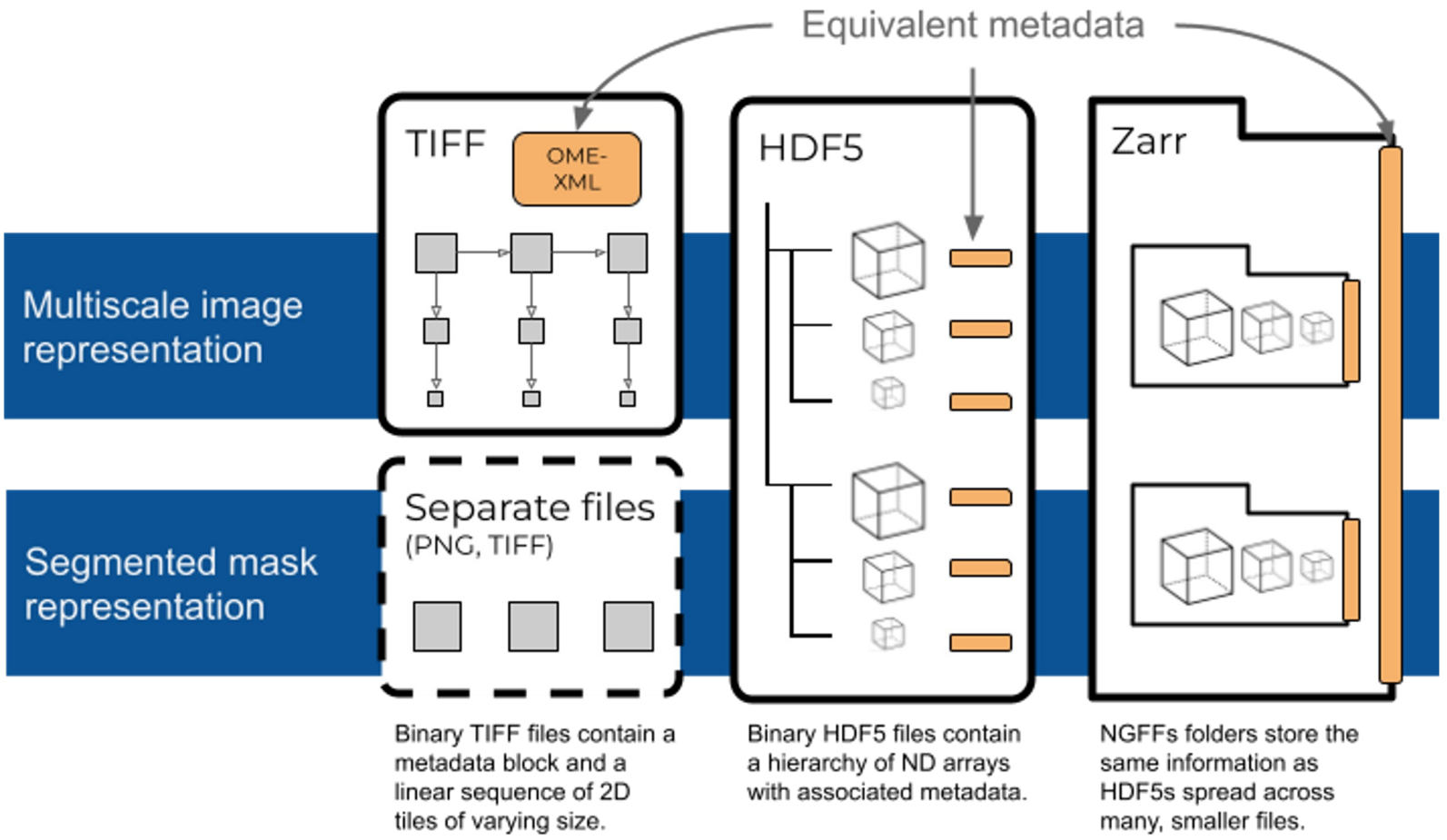
Josh Moore, Chris Allan, Sébastien Besson, Jean-Marie Burel, Erin Diel, David Gault, Kevin Kozlowski, Dominik Lindner, Melissa Linkert, Trevor Manz, Will Moore, Constantin Pape, Christian Tischer & Jason R. Swedlow

Nature Methods **18**, 1496–1498 (2021) | Cite this article

6045 Accesses | **4** Citations | **80** Altmetric | Metrics

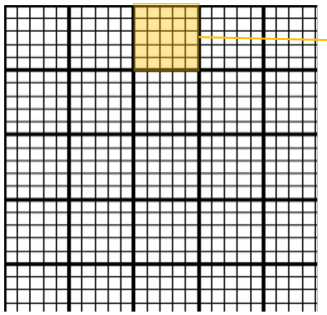
https://github.com/ome/ngff

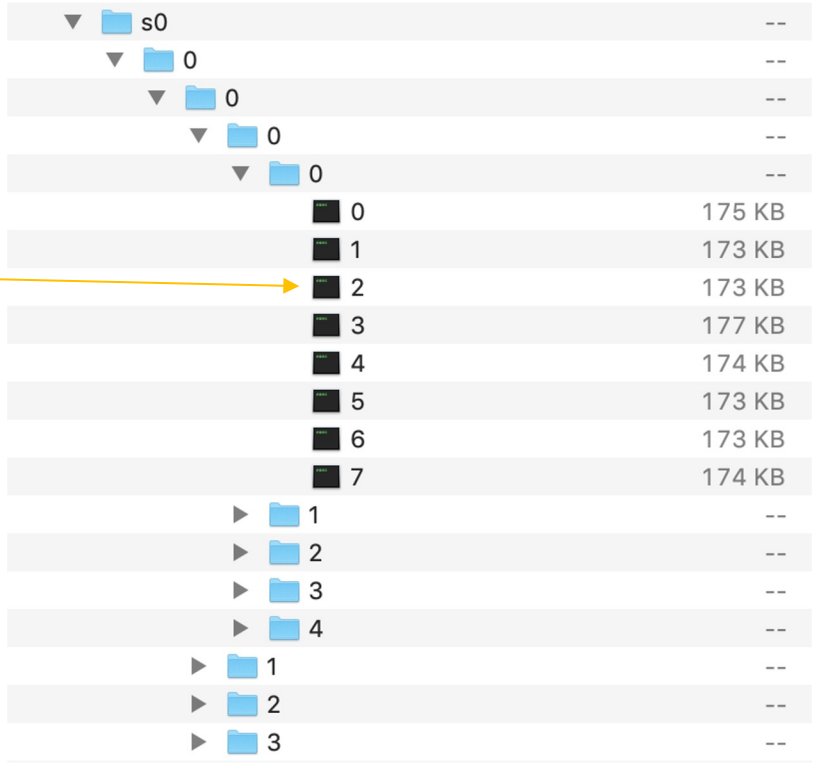
OME-NGFF uses Zarr



Zarr

chunk = file = object





Remote cloud storage

- See [https://github.com/joshmoore/NGFF-GBI-2022-Workshop/blob/main/workshop.ipynb]

Parallel analysis

- Natively scales Python
- Advanced parallelism for analytics
- See https://dask.org/index.html