

Develop data-centric web apps in Jupyter with Voilà and VOIS

Davide De Marchi – davide.de-marchi@ec.europa.eu

Pieter Kempeneers

Armin Burger

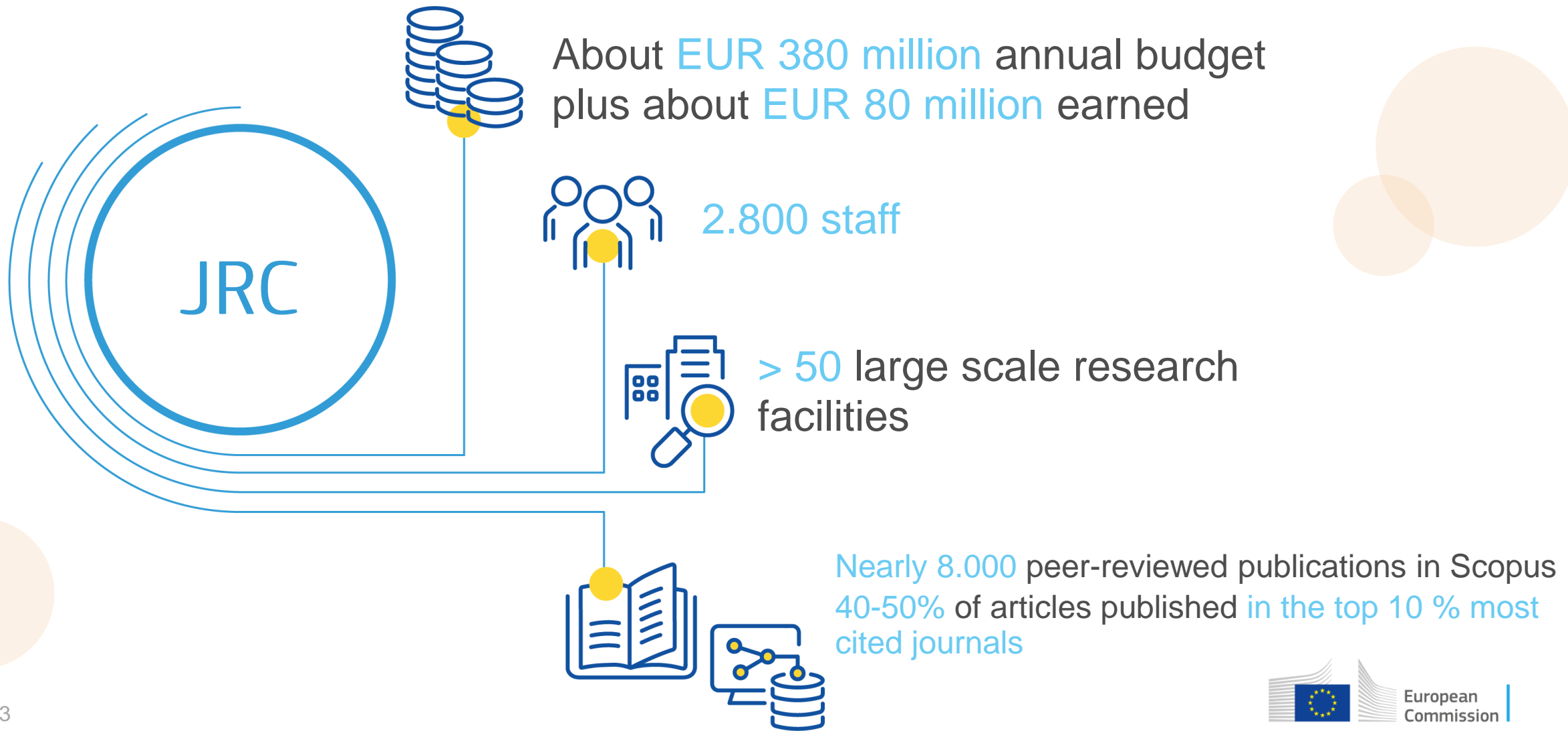
Pierre Soille

Joint Research Centre - T.4 Unit – CS3 Conference 12/03/2024

Agenda

- Voilà in JRC BDAP
- Vois library
- Development
- Applications
- Live Demo
- Deployment in free and commercial cloud platforms

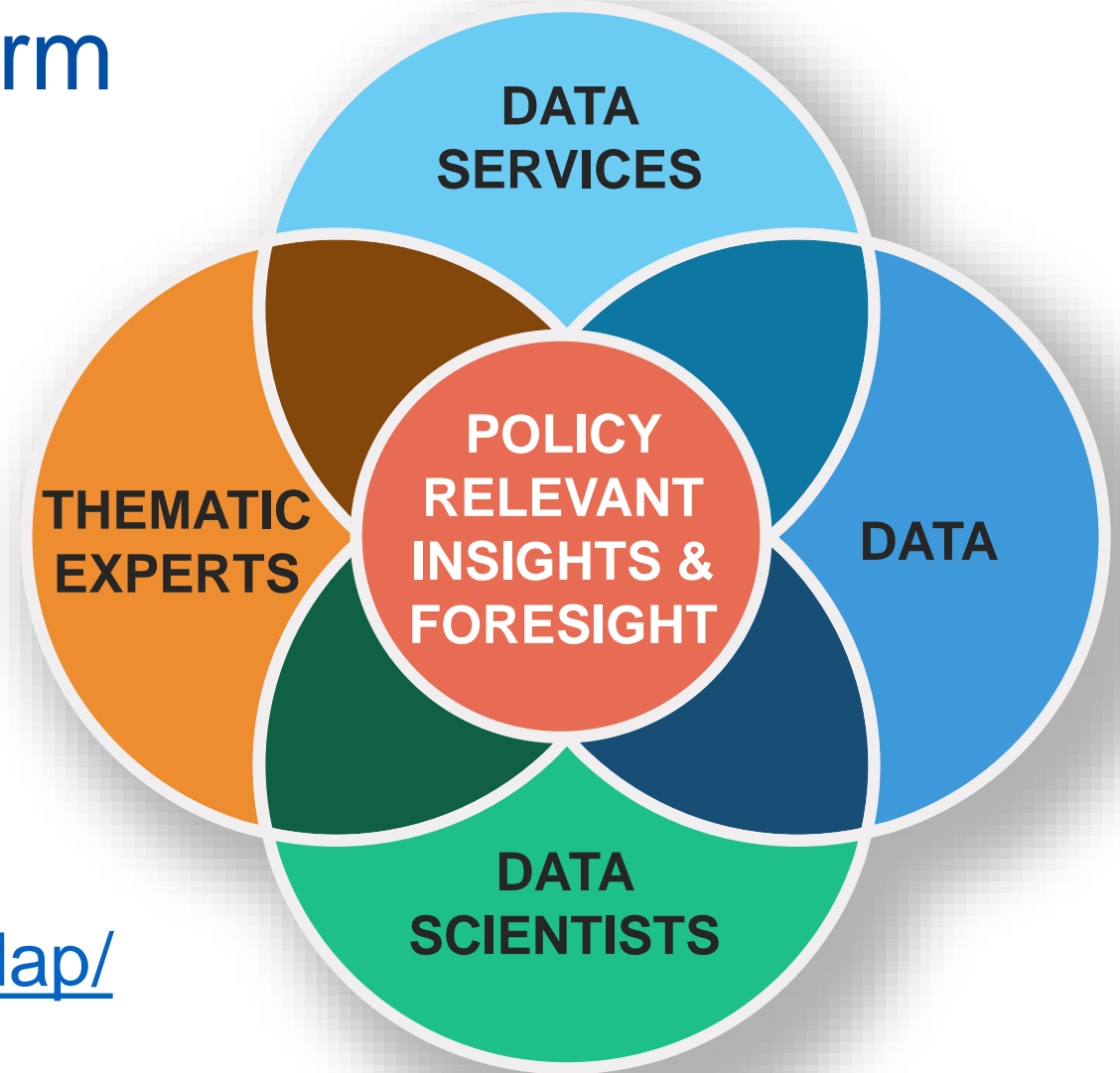
JRC – Facts and figures



Big Data Analytics Platform

The BDAP aims at linking data, data services, data scientists, and thematic experts to generate policy relevant insights and foresight

<https://jeodpp.jrc.ec.europa.eu/bdap/>



Voilà

voilà

- A Jupyter notebook extension to automatically create standalone data-centric applications and dashboards.
- Notebooks are rendered by showing only the output of the cells, while the code is hidden.
- Suitable for non-technical experts for communicating insights and foresight to a wider audience.
- Single environment for full data analytics workflows from research and innovation to outreach engaging policy makers and citizens.
- It is not a click&drag tool: programming capabilities are needed!

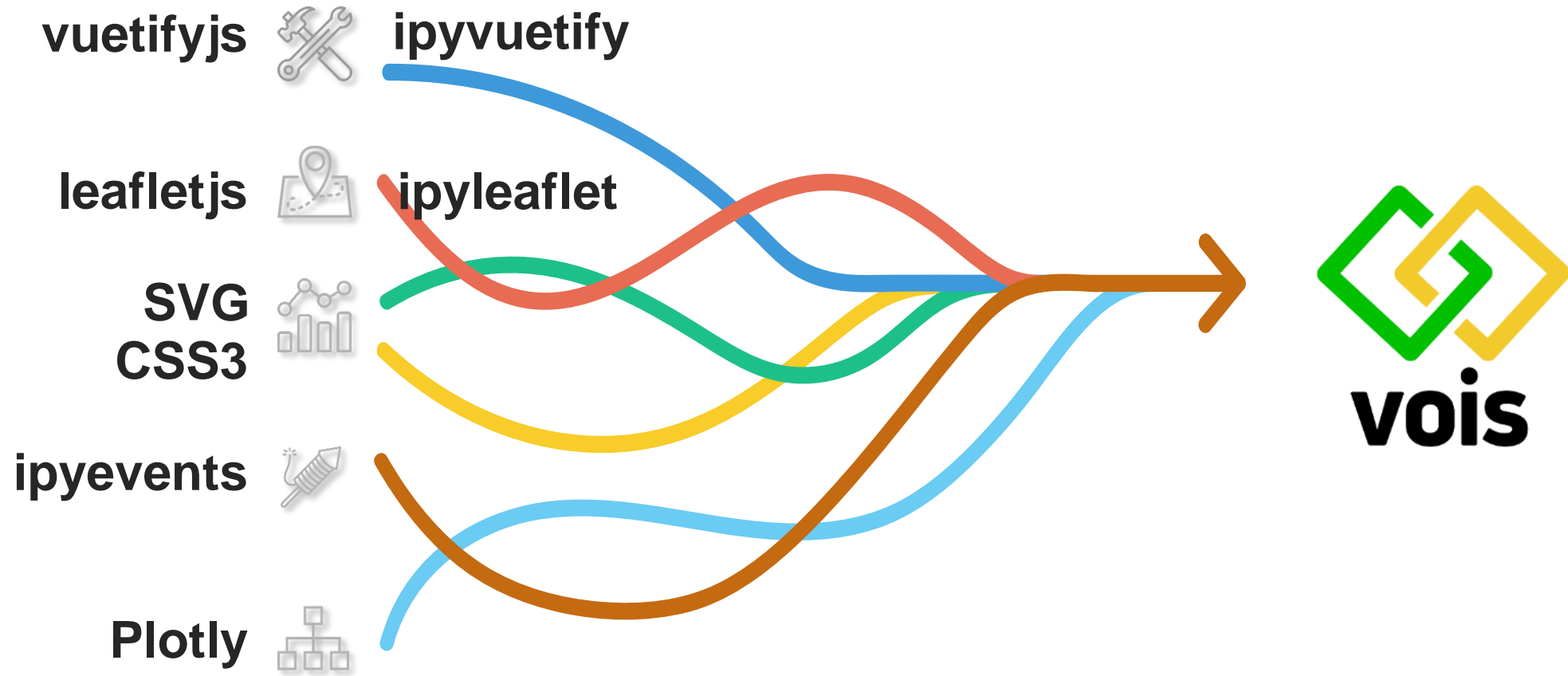
VOIS library



- **VOI**là Simplification library
- Its goal is to make life easier for the creation of impactful Voilà dashboards using Python
- Partially developed in the context of the CS3MESH4EOSC Horizon2020 project lead by CERN
- Published as open source on the <https://code.europa.eu> following European Commission Decision of 8 December 2021 on the open source licensing and reuse of Commission software 2021/C 495 I/01.
- Code: <https://code.europa.eu/jrc-bdap/vois>
- Documentation: <https://vois.readthedocs.io/>



Dependencies



Vois objectives

- **Facilitate** full exploitation of ipyvuetify/vuetify.js components with less code
- **Consistent** usage of widgets variants/colours/themes
- Allows for **full screen** applications, possibly avoiding scrollbars
- Support multiple screen resolutions: **responsiveness**
- **Compound elements** created from groups of widgets
- **Layered** components: popup-menus, dialog-boxes, etc.
- **Multipage** applications (!)
- Advanced **geo-spatial** visualizations
- Custom interactive **charts** with SVG drawings and events management
- **Cloud ↔ local system** data exchange

Easy start web application development

Vois helps developer in the creation of the application framework, for instance providing a multipage web-app with few lines of code:

```
from vois.vuetify import mainPage
import geoPage, textPage, resources

def onclickGeographicSearch():
    p = geoPage.geoPage(output)
    card = p.create(debug=False)
    p.open()

def onclickTextSearch():
    p = textPage.textPage(output)
    card = p.create(debug=False)
    p.open()

m = mainPage.mainPage(title='EMBAL validator',
                      subtitle='European Monitoring of Biodiversity in Agricultural Landscapes',
                      credits="D.5 Food security - T.4 Data Governance And Services",
                      applogo_url=resources.logoEMBAL,
                      applogo_widthpercent=7,
                      titlebox_toppercent=10, titlebox_widthpercent=46,
                      vois_show=False, vois_opacity=0.1,
                      buttonbox_toppercent=46, button_widthpercent=23,
                      background_image='https://JRC/EMBAL/Background1.jpg',
                      creditbox_toppercent=76, creditbox_opacity=0,
                      text_color='#222222')

m.addButton('Geographic search',
            subtitle='Search EMBAL plots from a overall EU map',
            tooltip='',
            image=resources.getLocalImageURL('./Graphics/geo.png'),
            onclick=onclickGeographicSearch)
```

Multipage web-apps

EMBAL validator

European Monitoring of Biodiversity in Agricultural Landscapes



Geographic search

Search EMBAL plots from a overall EU map



Textual search

Search EMBAL plots by country and textual description



Check and validate plots

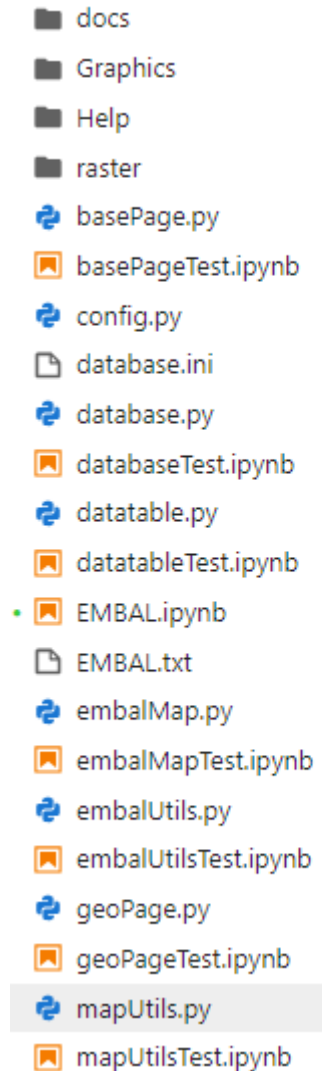
Validate EMBAL plot data by inserting annotations and comments



Development

Jupyter is not a classical framework for software development, but with some tricks, it can become a productive development environment. Here some tips:

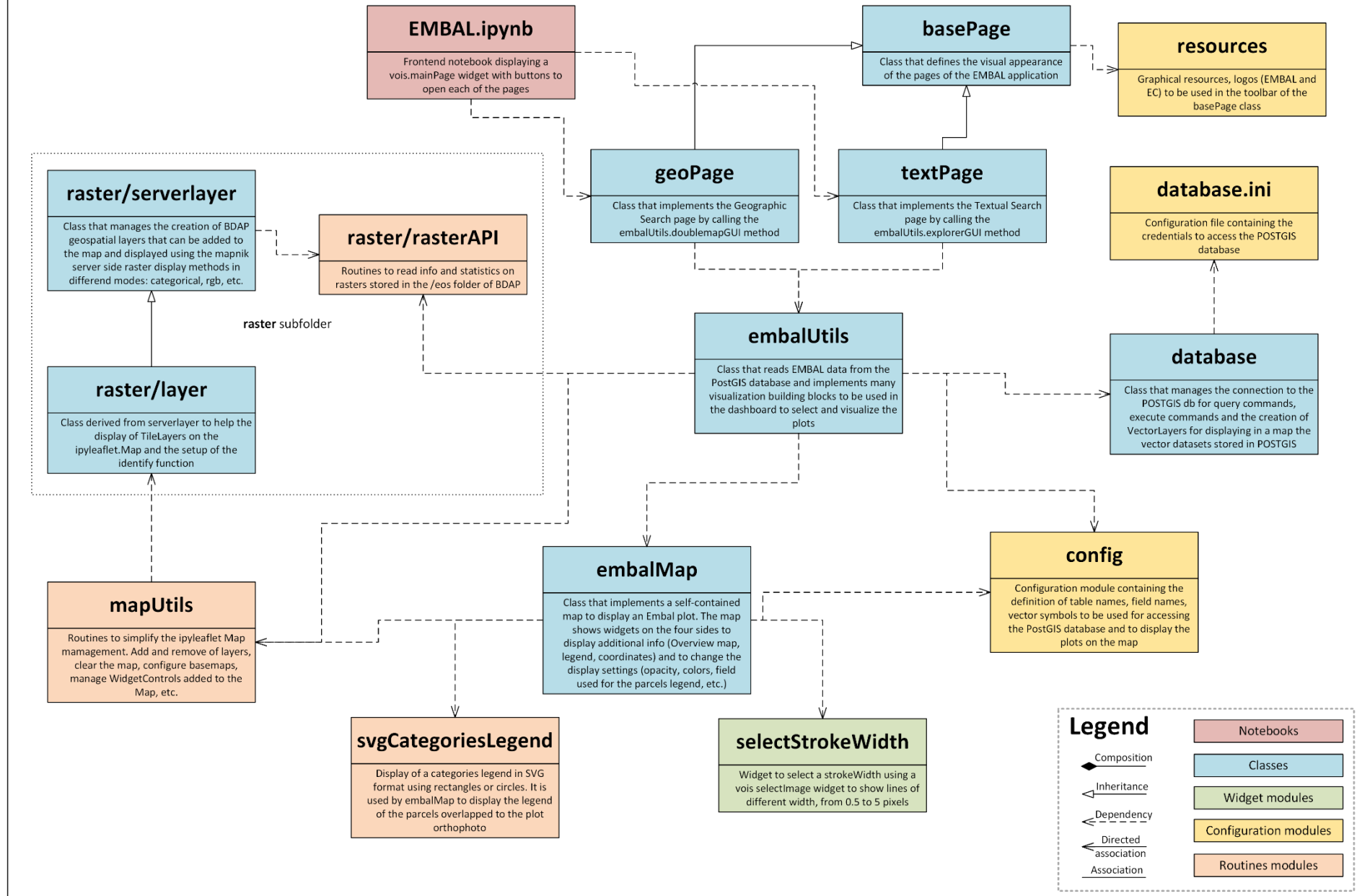
1. Modular development (divide code in self-contained modules)
2. Object-oriented programming
3. Less code possible in the main .ipynb file
4. Create a test notebook for each module/class
5. Document modules content and relationships in a formal way (UML, etc.)



UML

Example documentation schema for a medium-size web-app

EMBAL dashboard: classes and modules diagram



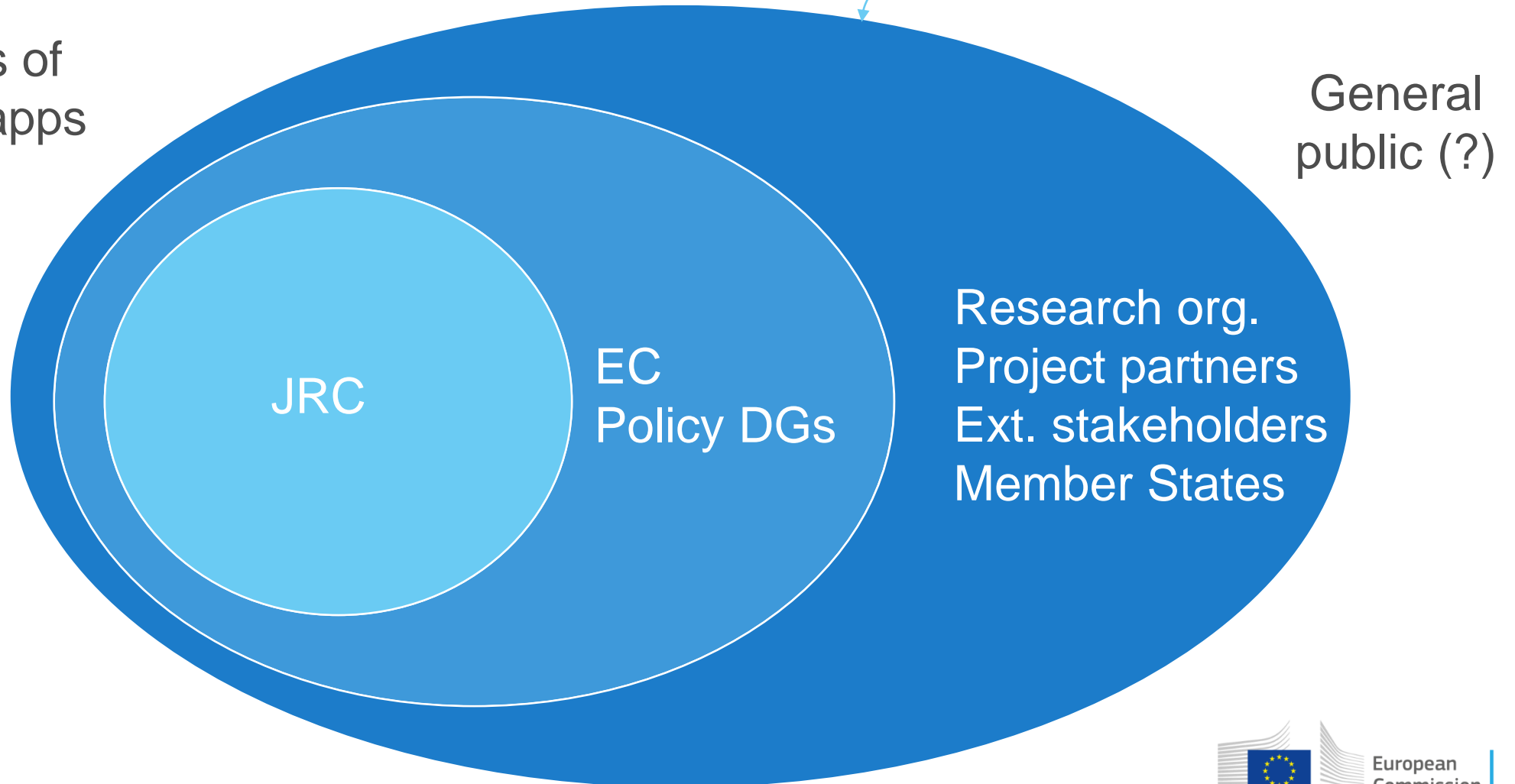
Application types

- All are solutions based on data (data-centric apps)
- Many use geospatial datasets or data that is linked to a location
- Tools to ease the discovery and usage of complex datasets or to link/merge multi-media datasets
- Communication of project and research results to (non-expert) hierarchy
- Dashboards to let readers of a scientific paper experiment with the data, replicate the results, test the derived products
- Demo applications for promoting projects, ideas, engage stakeholders
- Open data, open access → Open applications

Application targets



The users of our web-apps are:



Live demos

- JRC:
 - ❖ Pubsy mining (smart AI search on all JRC publications)
 - ❖ Collections Explorer (browse the STAC collections catalogue and GIS overlay of datasets)
- EC Policy DGs
 - ❖ LUCAS (Eurostat: Land use and land cover across the European Union)
 - ❖ EMBAL (DG-Agri - European Monitoring of Biodiversity in Agricultural Landscapes)
- External partners
 - ❖ SHERPA (Screening for high emission reduction potential on air)
 - ❖ FAIRMODE (Forum for Air quality Modeling)

Fully public Voilà web apps: Huggingface Spaces

Tested <https://huggingface.co/spaces> for free Voilà dashboard deploy

The screenshot shows the Hugging Face Spaces interface for a repository named 'voila' by user 'DavideDeMarchi'. The repository is currently 'Running'. The file list includes:

File/Folder	Size	Action	Time
Upload TestRasterAPI.ipynb			5 months ago
.github		Duplicate from giswqs/voila-geospatial	9 months ago
notebooks		Upload TestRasterAPI.ipynb	5 months ago
.gitattributes	1.56 kB	Upload 17 files	9 months ago
.gitignore	1.81 kB	Duplicate from giswqs/voila-geospatial	9 months ago
Dockerfile	685 Bytes	Update Dockerfile	6 months ago
README.md	477 Bytes	Updates	9 months ago
requirements.txt	56 Bytes	Update requirements.txt	5 months ago
run.sh	217 Bytes	Update run.sh	5 months ago

Fully public Voilà web apps: Ploomber Cloud

Applications / Create










▼ Overview
Choose the project where the application will be deployed

Select project

Set project name ⓘ

Set labels

▼ Framework
The framework used by your application

 Voilà	 Docker	 Panel
 Solara	 Streamlit	 Shiny (R)
 Dash		

Other (free) deployments

Voici is a tool for generating static dashboards from Jupyter Notebooks. It can be used as a drop-in replacement for [Voilà](#) and it has the same commands and supports most of Voila's configuration options.

Voici uses [WebAssembly](#) (Wasm) kernels to render notebooks in the browser, making the resulting dashboard entirely self-contained and distributable.

<https://github.com/voila-dashboards/voici>

A static dashboard created with Voici can be deployed, for instance, in [Gitlab Pages](#)

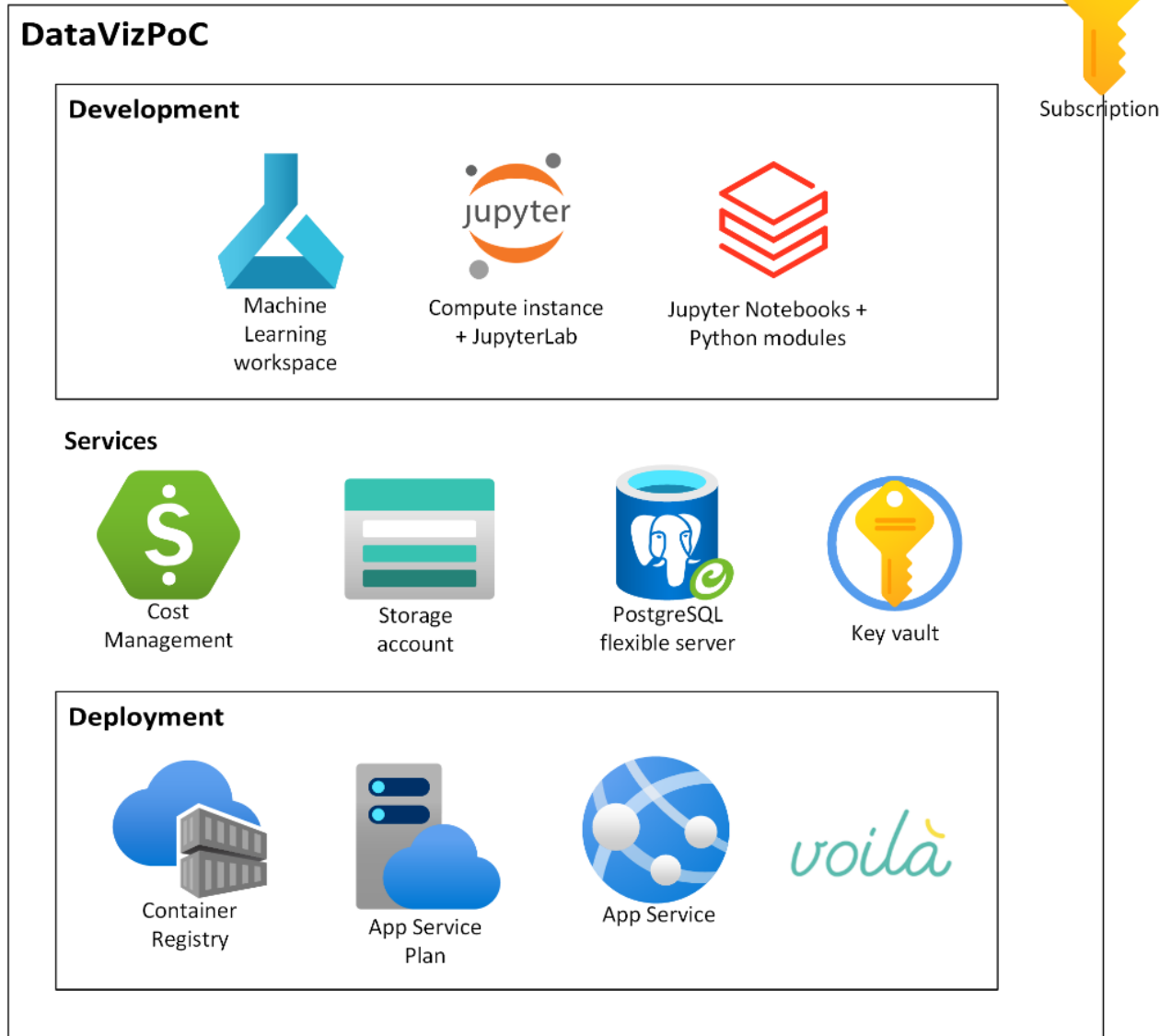
MS Azure DataViz PoC

JRC started a Proof of Concept (feasibility study) with [Avanade](#) to test the creation of Voilà web applications in the Azure Cloud platform:

- Access to data
- Open to the public

An excellent tutorial:

<https://medium.com/microsoftazure/from-jupyter-notebook-to-azure-web-app-in-5-easy-steps-2783f8fd847d>



Takeaway

- Voilà demonstrates its versatility in both allowing for quick&dirty notebook conversion and in the creation of fully fledged data-centric web applications
- It is a powerful visualization and communication tool that is deeply nested into the Jupyter data science environment (needs programming skills!)
- VOIS library is used inside the JRC BDAP Cloud Platform to communicate scientific results to a wider audience and to allow for easy interaction with complex datasets (open source from <https://code.europa.eu/jrc-bdap/vois>)
- Voilà and vois library can be used in many commercial cloud environments even with free accounts
- We see a near future where an Hybrid Cloud infrastructure can provide Voilà development, deploy and data access services through APIs

EU Solidarity with Ukraine

The EU stands united with Ukraine



Thank you!



© European Union 2024

Unless otherwise noted the reuse of this presentation is authorised under the [CC BY 4.0](https://creativecommons.org/licenses/by/4.0/) license. For any use or reproduction of elements that are not owned by the EU, permission may need to be sought directly from the respective right holders.