

Intro to VRE

Ashley Smith (PDRA, Uni of Edinburgh, UK)
ashley.smith@ed.ac.uk



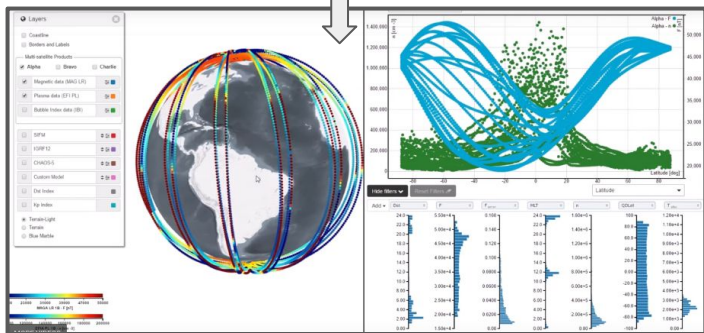
Virtual Research Environment (VRE) for Swarm



What is it?

Free-to-use Jupyter environment in the cloud to easily access Swarm products

Extension of VirES (GUI)



vires.services

vire.services

The image shows a Jupyter Notebook interface. The top part of the notebook contains code to fetch one hour of MAG data and models at 10-second intervals. The code uses the `request` library to get data between specific start and end times. Below the code, there is a progress bar indicating that the data is being processed and downloaded. The bottom part of the notebook shows a list of input files, including data sources and a pandas dataframe.

The image shows the Swarm-VRE ReadTheDocs page. It features a 'Launch on VRE!' button, a 'Demo MAGxLR_1B (magnetic field 1Hz)' section, and a list of authors and abstracts. The page also includes a 'SWARM PRODUCT DEMOS' section with links to various demo notebooks.

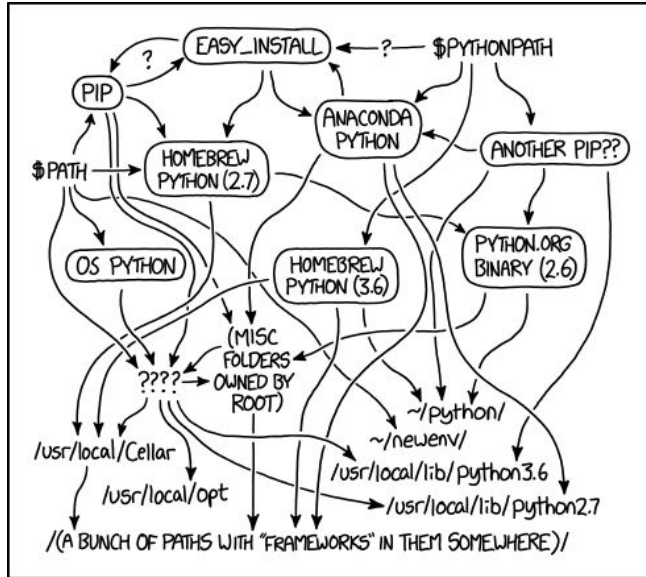
swarm-vre.readthedocs.io



Integrated notebooks to help new users (intro to Swarm products and Python tools)

Aims of VRE: *improve accessibility of Swarm*

- Reduce IT skills necessary to just get started
- Easier navigation of & access to Swarm products



MY PYTHON ENVIRONMENT HAS BECOME SO DEGRADED THAT MY LAPTOP HAS BEEN DECLARED A SUPERFUND SITE.

- Level 2 Products

- Core Field

- MCO_SHA_2C ◦ MCO_SHA_2D ◦ MCO_VAL_2C ◦ MCO_VAL_2D ◦ MCO_VAL_2E
 - MCO_SHA_2F ◦ MCO_SHA_2X

- Lithospheric Field

- MLI_SHA_2C ◦ MLI_SHA_2D ◦ MLI_SHA_2E ◦ MLI_VAL_2C ◦ MLI_VAL_2D
 - MLI_VAL_2E

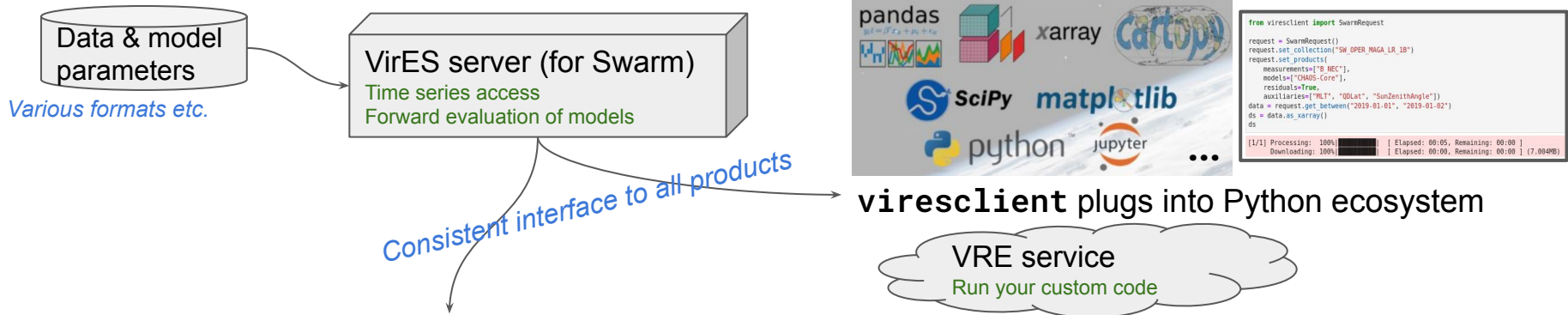
- Mantle Conductivity

- MIN_IDM_2 ◦ MCR_IDM_2 ◦ MI1_VAL_2 ◦ MCI_VAL_2

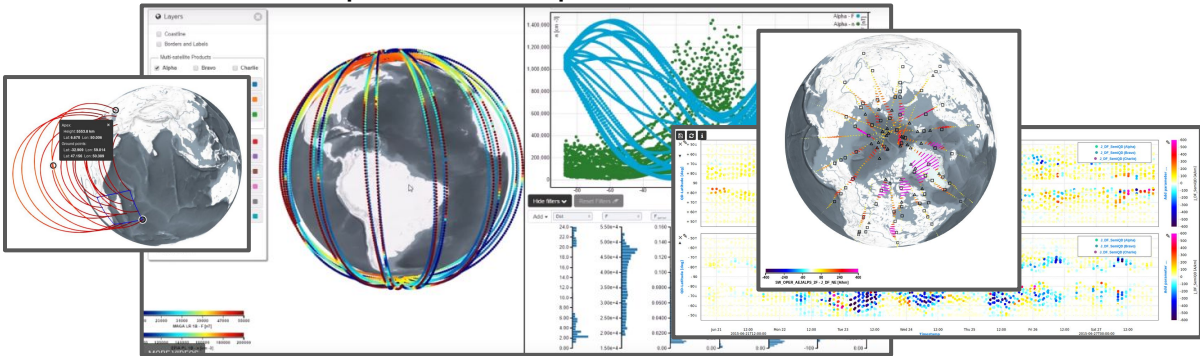
- External Current Systems

- MMA_SHA_2C ◦ MMA_VAL_2C ◦ MMA_SHA_2F ◦ MIO_SHA_2C ◦ MIO_SHA_2D
 - MIO_VAL_2C ◦ MIO_VAL_2D ◦ MIO_VAL_2E ◦ IBIxTMS_2F ◦ TECxTMS_2F
 - FAC_TMS_2F ◦ FACxTMS_2F ◦ EEFxTMS_2F ◦ IPDxIRR_2F ◦ MIO_SHA_2E
 - IBP_CLI_2

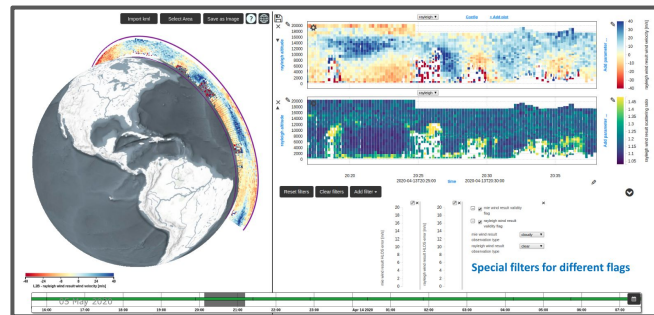
VirES (Virtual workspaces for Earth observation scientists)



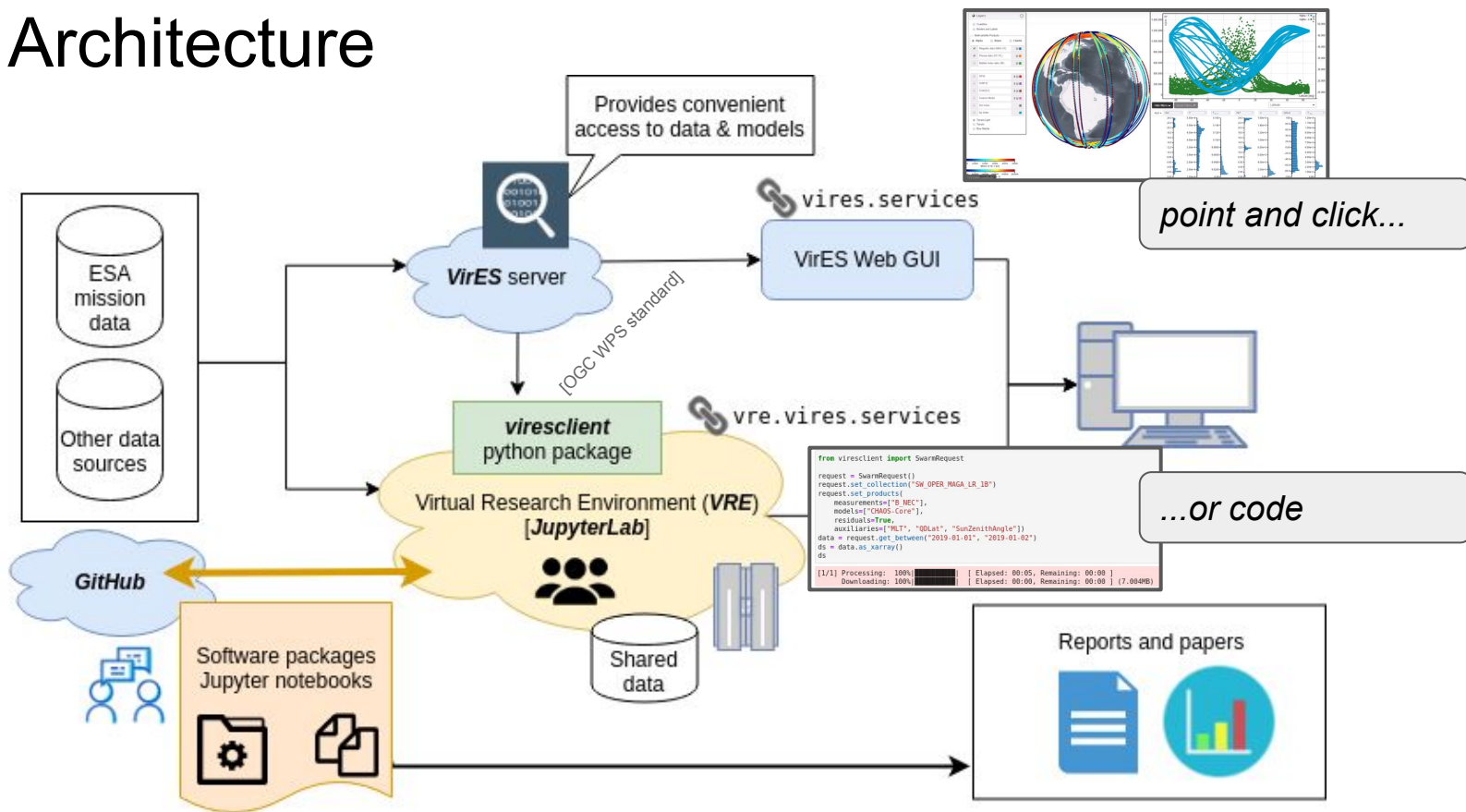
Web client provides bespoke visualisations



(Same system applied to [Aeolus](#))



VRE Architecture



viresclient

Access to **time series** of (most of) the Swarm products as pandas/xarray

Choose any time interval, add custom **subsampling** and **filtering** (subselection)

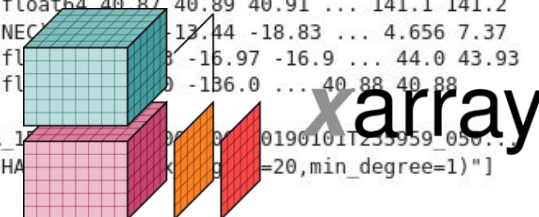
Geomagnetic models and other parameters **evaluated on-the-fly** on the server

```
from viresclient import SwarmRequest

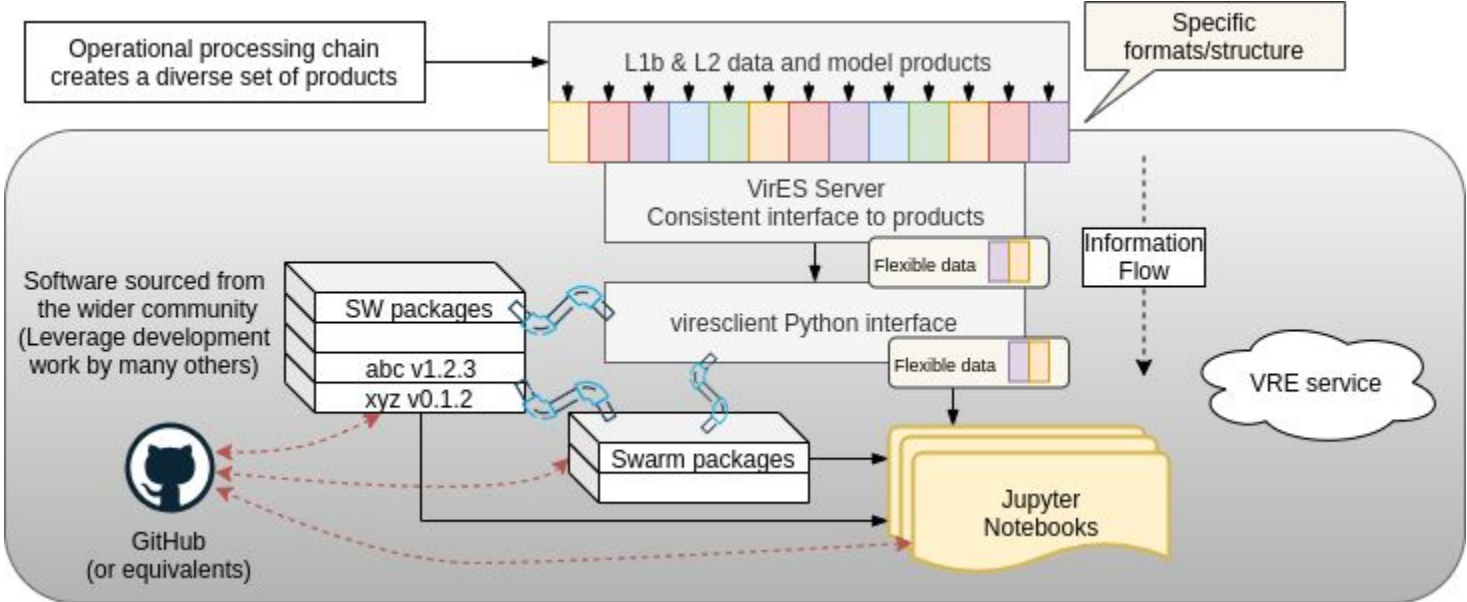
request = SwarmRequest()
request.set_collection("SW_OPER_MAGA_LR_1B")
request.set_products(
    measurements=["B_NEC"],
    models=["CHAOS-Core"],
    residuals=True,
    auxiliaries=["MLT", "QDLat", "SunZenithAngle"])
data = request.get_between("2019-01-01", "2019-01-02")
ds = data.as_xarray()
ds
```

```
[1/1] Processing: 100% ██████████ [ Elapsed: 00:05, Remaining: 00:00 ]
      Downloading: 100% ██████████ [ Elapsed: 00:00, Remaining: 00:00 ] (7.004MB)
```

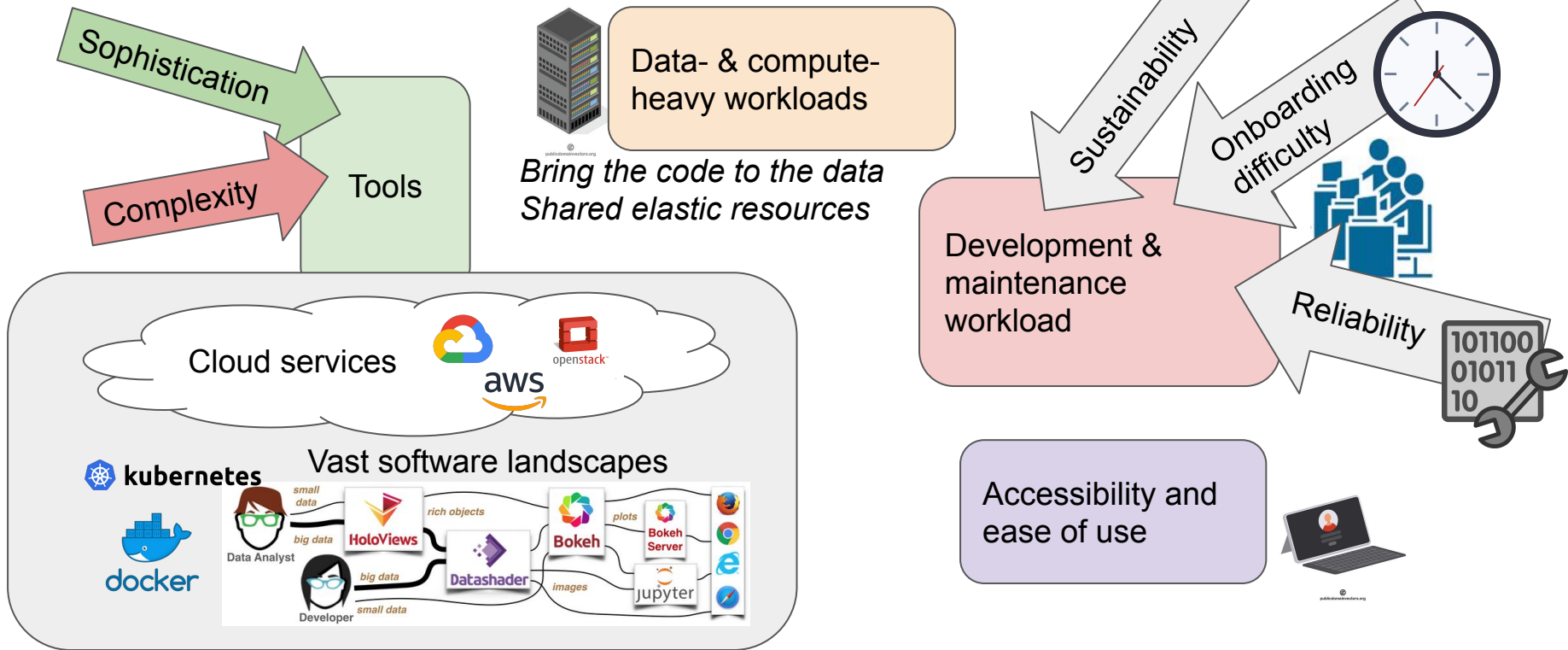
```
<xarray.Dataset>
Dimensions:                (NEC: 3, Timestamp: 86400)
Coordinates:
  * Timestamp                (Timestamp) datetime64[ns] 2019-01-01 ... 2019-01-01T23:59:59
  * NEC                      (NEC) <U1 'N' 'E' 'C'
Data variables:
  Spacecraft                (Timestamp) object 'A' 'A' 'A' 'A' ... 'A' 'A' 'A' 'A'
  MLT                      (Timestamp) float64 14.82 14.82 14.82 ... 2.445 2.445
  Radius                    (Timestamp) float64 6.819e+06 6.819e+06 ... 6.809e+06
  QDLat                    (Timestamp) float64 -13.94 -13.88 ... 39.74 39.67
  SunZenithAngle            (Timestamp) float64 40.87 40.89 40.91 ... 141.1 141.2
  B_NEC_res_CHAOS-Core     (Timestamp, NEC) float64 -13.44 -18.83 ... 4.656 7.37
  Latitude                  (Timestamp) float64 13.88 -16.97 -16.9 ... 44.0 43.93
  Longitude                 (Timestamp) float64 136.0 -136.0 ... 40.88 40.88
Attributes:
  Sources:                  ['SW_OPER_MAGA_LR_1B']
  MagneticModels:          ["CHAOS-Core = 'CHAOS-Core', k=20, min_degree=1)"]
  RangeFilters:            []
```



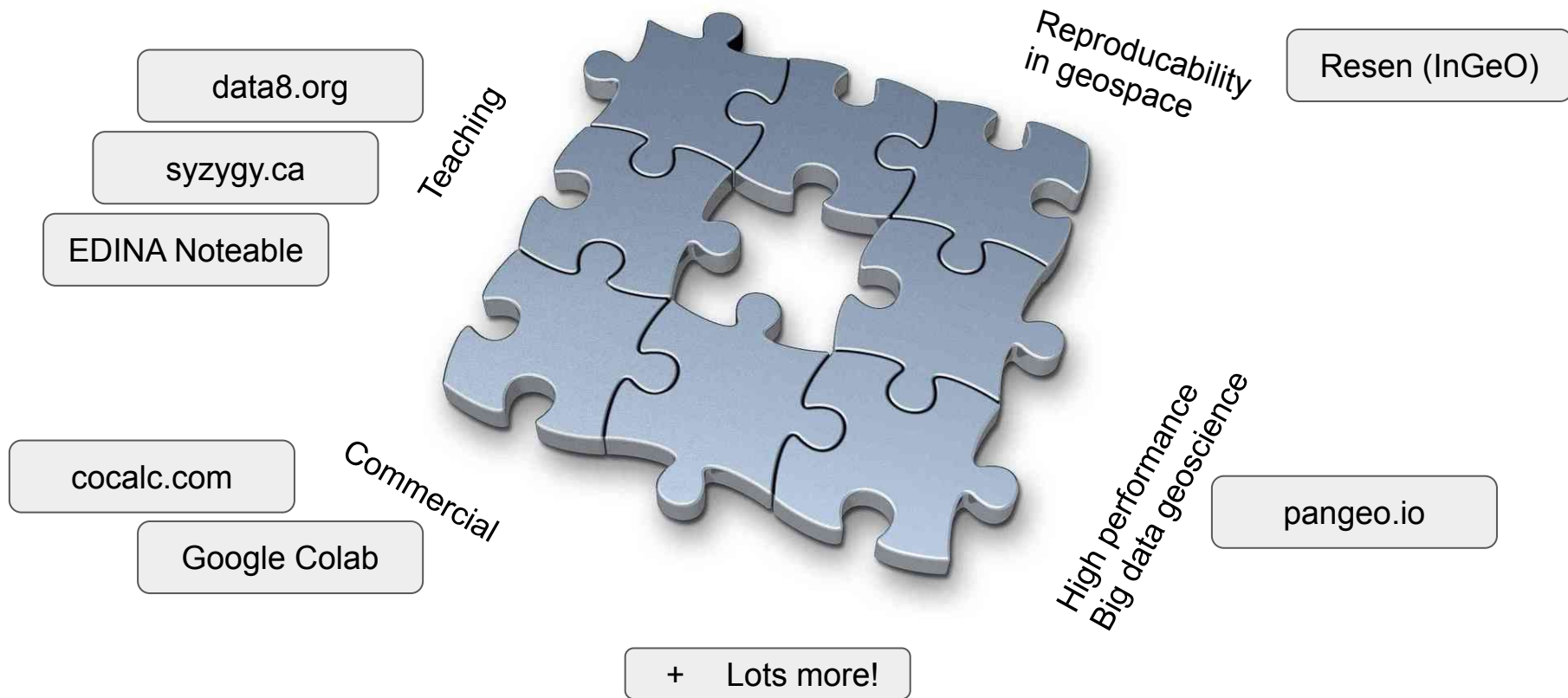
Information Architecture



Challenges & opportunities for software-based scientific research and services



Other Jupyter-y examples



Plans

- viresclient:
 - Integration with PyHC and other communities
 - Pre-configured quicklook visualizations
 - Make it easy to cite/acknowledge software and data correctly
- Notebooks (as a cookbook):
 - Need to improve content; want to stimulate more community involvement
- Access to new products:
 - Auroral electrojet and oval boundaries (AEJ_LPL, AEJ_LPS, AEJ_PBL, AEJ_PBS, AOB_FAC)
 - Ground Observatories (INTERMAGNET) & Virtual Observatories
- VRE software stack:
 - Improve portability (instructions for recreating similar environment)
 - Add more packages!
- New services:
 - E.g. “live” plots at magneticearth.org/pages/quicklooks
- Stimulate collaboration
 - Development of packages
 - Sharing of notebooks
- Encourage best practices

Some talking points

- Looking for an easier way to share notebooks
 - Email: bad (awkward to comment, update etc.)
 - Github: hard (need to learn git)
 - Hub? E.g. <https://exploratory.openhumans.org/>
- What do you want to do with Swarm?
- Swarm-DISC is looking for new ideas for tools/services

Service: vires.services
Info & guide: swarm-vre.readthedocs.io
Python package: viresclient.readthedocs.io

Come talk to me!
ashley.smith@ed.ac.uk

Weekly online open
office hours, check:
smithara.github.io

Links

Service: vires.services
Info & guide: swarm-vre.readthedocs.io
Python package: viresclient.readthedocs.io

- Main GitHub repos [[responsibility](#)]:
 - [[EOX](#)] VirES Server github.com/ESA-VirES/VirES-Server
 - [[EOX](#)] VirES Web GUI github.com/ESA-VirES/WebClient-Framework
 - [[EOX](#)] eoismagmod github.com/ESA-VirES/MagneticModel
 - [[AS / EOX](#)] viresclient github.com/ESA-VirES/VirES-Python-Client
 - [[EOX](#)] (VRE config currently hidden - aim to make useful docker file public?)
 - [[AS / DISC](#)] Swarm notebooks github.com/Swarm-DISC/Swarm_notebooks
 - [[AS](#)] VRE user documentation github.com/ESA-VirES/Swarm-VRE
- Friendly introduction to geomagnetism: magneticearth.org
- CI for notebooks: treebeard.io (beta)

AS: ashley.smith@ed.ac.uk

Background organisations

Swarm DISC (Data, Innovation, & Science Cluster)
Comprises groups at various Institutions
Production of Swarm data/model products etc.



**ESA funds DISC to deliver Swarm mission goals;
EOX for IT innovation**

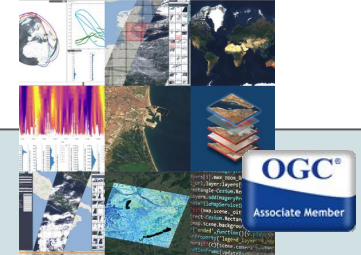


eox.at/



EOX IT Services

Expertise in geospatial software and services



Wider research communities

Geo & space physics
Core, mantle, crust,
Ionosphere, magnetosphere,
Space weather ...

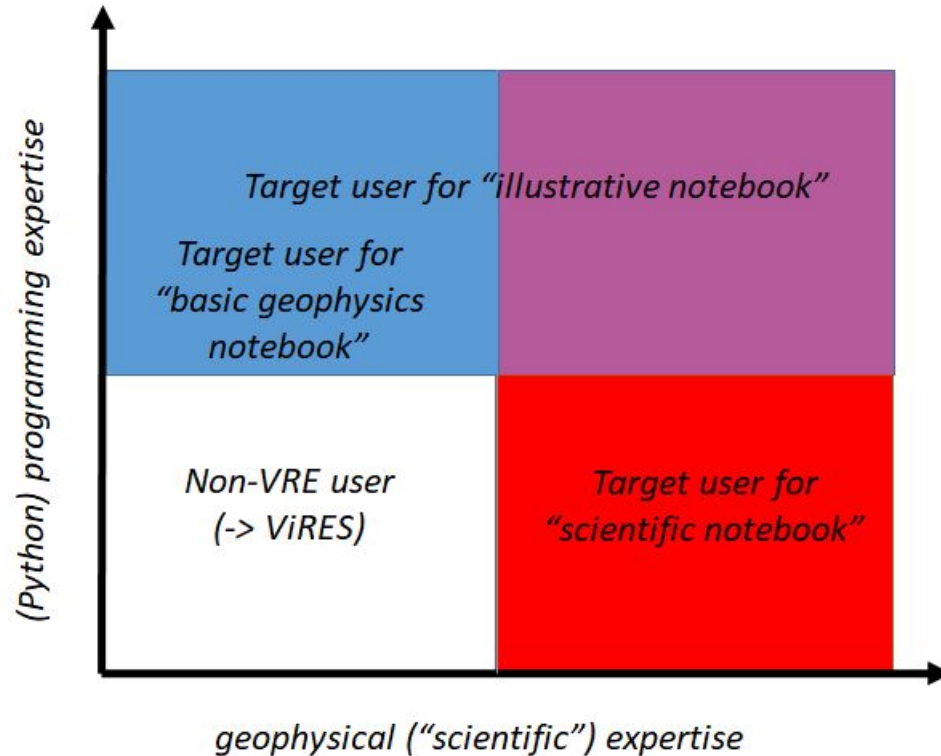
Other Earth & space obs

Notebook development & community connections

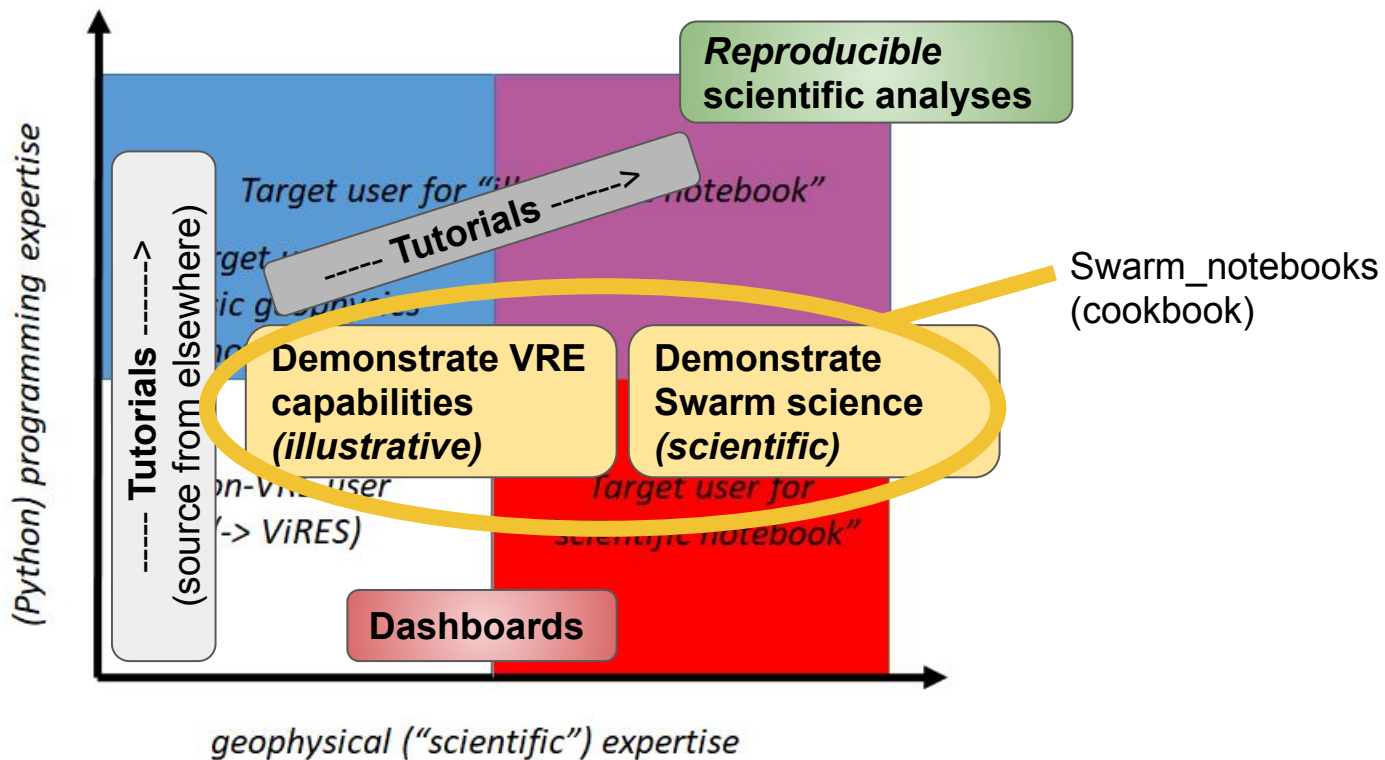
Current notebooks

- Browse them at <https://swarm-vre.readthedocs.io/>
 - Easy one-click to run on VRE
- Main collection “Swarm_notebooks”
 - To be updated and maintained as a cookbook
 - Talk to me to augment it with your own examples
- Publish your own collections on GitHub
- [Demo one of the NBs and JLab, then 10 mins for people to have a go?]

Target audiences



Targeted notebook themes

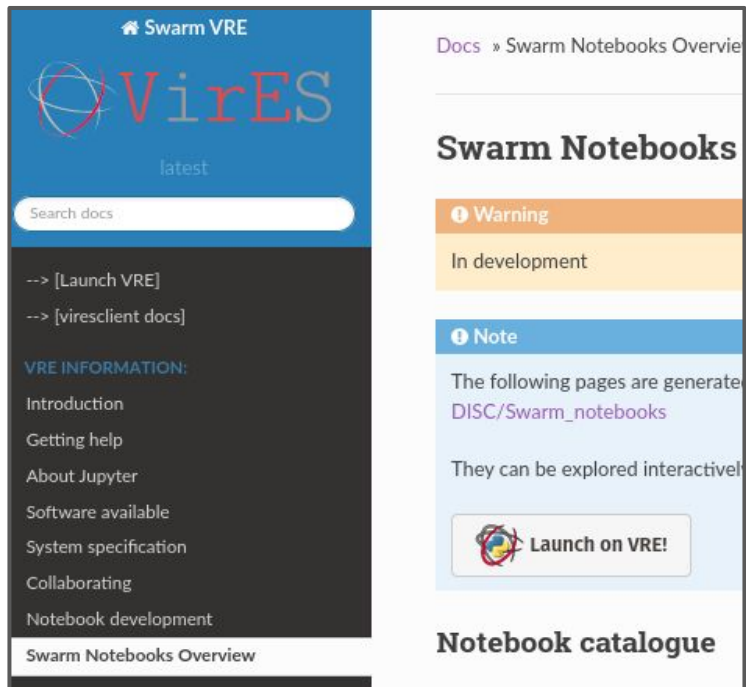


Themed collections in [GitHub/Swarm-DISC/Swarm_notebooks](https://github.com/Swarm-DISC/Swarm_notebooks)

	<i>First notebook name of each theme</i>	<i>Description</i>
Illustrative	01a__Intro-Jupyter-Python ...	Generic Python guide
	02a__Intro-Swarm-viresclient ...	Guide to VirES concepts & related libs
	03a__Product-Demo-MAG ...	Demo access to each Swarm product
Scientific	04a__Magnetic-Models ...	Geomagnetic Models
	05a__...	Ionosphere
	...	

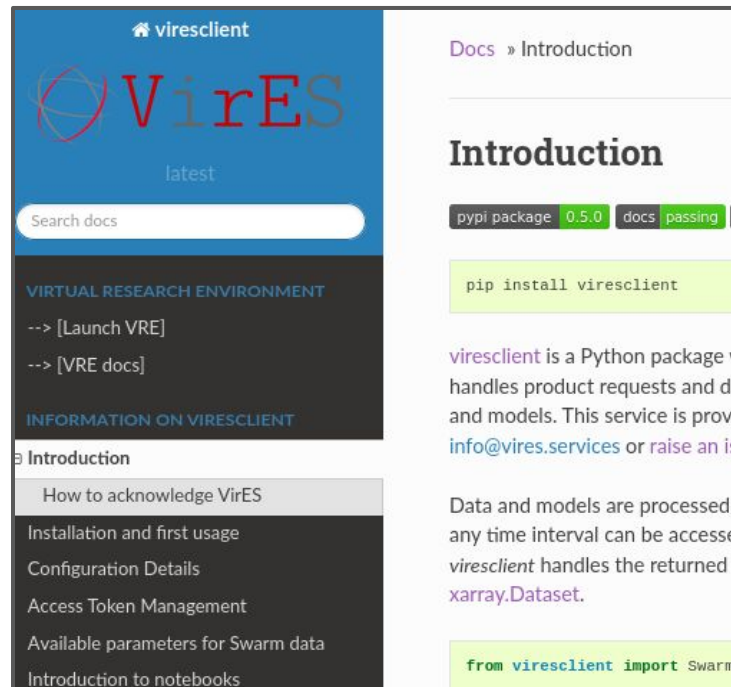
Documentation sites

swarm-vre.readthedocs.io



The screenshot shows the Swarm VRE documentation page. The header includes the 'Swarm VRE' logo and the 'VirES' logo with 'latest' version information. A search bar is present. The left sidebar contains navigation links: '[Launch VRE]', '[viresclient docs]', 'VRE INFORMATION:' (Introduction, Getting help, About Jupyter, Software available, System specification, Collaborating, Notebook development), and 'Swarm Notebooks Overview'. The main content area is titled 'Swarm Notebooks Overview' and features a 'Warning' box stating 'In development' and a 'Note' box stating 'The following pages are generated by DISC/Swarm_notebooks. They can be explored interactively.' Below this is a 'Launch on VRE!' button. At the bottom, there is a 'Notebook catalogue' section.

viresclient.readthedocs.io



The screenshot shows the viresclient documentation page. The header includes the 'viresclient' logo and the 'VirES' logo with 'latest' version information. A search bar is present. The left sidebar contains navigation links: '[Launch VRE]', '[VRE docs]', 'VIRTUAL RESEARCH ENVIRONMENT' (Launch VRE, VRE docs), and 'INFORMATION ON VIRESCIENT' (Introduction, How to acknowledge VirES, Installation and first usage, Configuration Details, Access Token Management, Available parameters for Swarm data, Introduction to notebooks). The main content area is titled 'Introduction' and features a 'Warning' box stating 'pypi package 0.5.0 docs passing'. Below this is a code block: `pip install viresclient`. The text describes viresclient as a Python package that handles product requests and data, and provides contact information: `info@vires.services` or `raise an issue`. It also mentions that data and models are processed every time interval and can be accessed via `viresclient` which handles the returned `xarray.Dataset`. At the bottom, there is a code block: `from viresclient import Swarm`.

Coordination through GitHub



SWARM DISC Swarm-DISC

Repositories 2 Packages

Find a repository...

Swarm_notebooks

Notebooks demonstrating access to Swarm d
deployed on the VRE.

Jupyter Notebook MIT 0 0

SwarmPyFAC

A package for calculation Field Aligned Curren

Python 1 1 0 0 Updated

Swarm-DISC / Swarm_notebooks

Code Issues 0 Pull requests 0

Notebooks demonstrating access to Swarm d

Manage topics

13 commits 2 branches

Branch: master New pull request

smithara Execute notebooks

.gitignore

01a__Intro-Jupyter-Python.ipynb

02a__Intro-Swarm-viresclient.ipynb

02b__viresclient-Available-Data.ipynb

02z1__Template-Basic.ipynb



Run NBs

View NBs and guidance

Swarm VRE

VirES

latest

Search docs

--> [Launch VRE]

--> [viresclient docs]

VRE INFORMATION:

Introduction

Getting help

About Jupyter

Software available

System specification

Collaborating

Notebook development

Swarm Notebooks Overview

Docs > Swarm Notebooks Overview

Swarm Notebooks

Warning

In development

Note

The following pages are generate
DISC/Swarm_notebooks

They can be explored interactiv

Launch on VRE!

Notebook catalogue

Expose Swarm products and usage

Coordination through GitHub - Orgs & Packages



SWARM DISC Swarm-DISC

Repositories 2 Packages

Find a repository...

Swarm_notebooks
Notebooks demonstrating access to Swarm d
deployed on the VRE.
Jupyter Notebook MIT 0 0 0 Updated

SwarmPyFAC
A package for calculation Field Aligned Curre
Python 1 1 0 0 Updated

MagneticEarth
Earth http://magneticearth.github.io

Repositories 3 Packages People 1

Find a repository...

MagneticEarth_notebooks
Jupyter notebooks of general geomagnetic interest
Jupyter Notebook MIT 0 0 0 Updated

MagneticEarth.github.io
A place to learn about geomagnetism
geophysics space-physics space-weather geomagnetism
HTML 0 1 0 0 Updated 24 days ago

IAGA_SummerSchool2019
Materials for the workshop on magnetic observatories and modell
geomagnetism laga
Jupyter Notebook MIT 0 6 0 0 Updated

ancklo / **ChaosMagPy**

Code Issues 1 Pull requests 0

ChaosMagPy is a simple python package for

327 commits 3 branches

Branch: master New pull request

ancklo Rephrased citation suggestion.

chaosmagpy	Fixed bug in ze
docs	Fixed bug in ze
tests	Fixed bug in ze
.gitignore	Restructured d
CHANGELOG.rst	Fixed bug in ze

Coordination through GitHub - Deploy NBs



Swarm_notebooks

Notebooks demonstrating access to Swarm data for scientific analyses. To be deployed on the VRE.

Jupyter Notebook MIT 0 0 0 0 Updated yesterday

Software stack needs to be managed

Software dependencies

Collaborative repos

MagneticEarth_notebooks

Jupyter notebooks of general geomagnetic interest

Jupyter Notebook MIT 0 0 0 0 Updated 2 hours ago

IAGA_SummerSchool2019

Materials for the workshop on magnetic observatories and modelling

geomagnetism iaga

Jupyter Notebook MIT 0 6 0 0 Updated on 30 Aug 2019

Tutorials



Experiments by individuals

virescient_examples

A collection of Jupyter notebooks to demonstrate usage of virescient

Jupyter Notebook 1

smfsws Private

Separating magnetic field sources with Swarm

Jupyter Notebook Other Updated on 1 Oct 2019

jupyter_notebooks


Jupyter Notebook MIT License Updated on 21 Dec 2019

Portable lab for particular analysis

Custom kernel required to fix package versions

Coordination through GitHub - Other Projects?






Fatiando a Terra

Open-source Python tools for geophysics

<https://www.fatiando.org> Verified

Repositories 27 **Packages** **People** 8


Pinned repositories



verde

Processing and gridding spatial data using Green's functions


Python ★ 164 🔗 26



harmonica

Forward modeling, inversion, and processing gravity and magnetic data

Python ★ 32 🔗 15



Heliopy

Repositories 2 **Packages**

Find a repository...


heliopy

Python for heliospheric and planetary physics

[python](#) [physics](#) [space-physics](#)

Python 🔗 GPL-3.0 🔗 36 ★ 34 🔔 14

....



geomagpy

Repositories 4 **Packages** **People**

Find a repository...

MARTAS

Magpys Automated Real Time Acquisition System

Python 🔗 GPL-3.0 🔗 0 ★ 1 🔔 7 🔗 0 Updated 10 days

magpy

MagPy (or GeomagPy) is a Python package for analysing and displaying geomagnetic data.

Python 🔗 BSD-3-Clause 🔗 17 ★ 15 🔔 12 🔗 0 Updated

pandas
 $y_i = \beta^T x_i + \mu_i + \epsilon_i$

xarray

Cartopy

SciPy

matplotlib

python™

jupyter

...

Virtual Research Environment For Swarm

Moving to cloud systems

Developing your own notebooks

Developing a notebook

Intro template & written guidelines at

https://swarm-vre.readthedocs.io/en/latest/notebook_development.html

Live demo. Demo some pitfalls and how to make NB cleaner

Sharing notebooks

Review options

VRE as a research tool

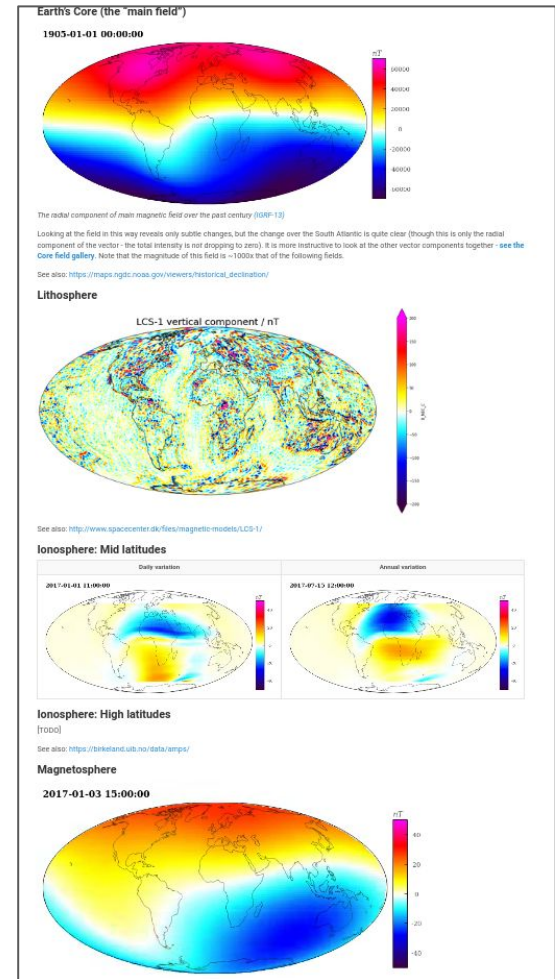
Challenges for science with Swarm

- Constantly improving and evolving portfolio of data and models (collectively “products”):
 - Bewildering array of products
 - Products with changing version numbers
 - Different “choices” of products possible (geomagnetic field models; FAC algorithms etc.) and newer ones developed
- Complexity of the science:
 - Intersection of geo- and space- physics
 - Necessity to support different user groups and help communication between them

Swarm products and communities

Geomagnetic field modelling [Geophysics]

Models of the near-Earth magnetic field



Why VRE?

- Centralise IT effort and reduce skills necessary for scientists
 - Managed Python environment with aim to cover “most of” Swarm science
- Provide access to all, lowering barrier for entry
- Provide an identical environment to all to help coordination
- Benefits of cloud service
 - Elastic provision of resources to a large user base
 - Proximity to other cloud services for performance
- Difficulties
 - Free service means available compute power is limited
 - Tied to Jupyter ecosystem and concept
 - Evolving software stack breaks reproducibility

VRE aspects

- Infrastructure & support
 - A place to run code; Anyone can log in and use it
- Software stack
 - Provides integration with VirES for Swarm data access (`viresclient`)
 - Everyone is running the same software environment
 - Solves the “it worked on *my* machine...” problem
 - Stimulate a community of Swarm-related software that is inter-operable
 - The software is all **portable** and can be redeployed elsewhere
 - E.g. on your individual computer or on your institution’s systems
 - But requires some knowledge + effort on your part
- Notebooks
 - Resource to learn/teach/share *science*

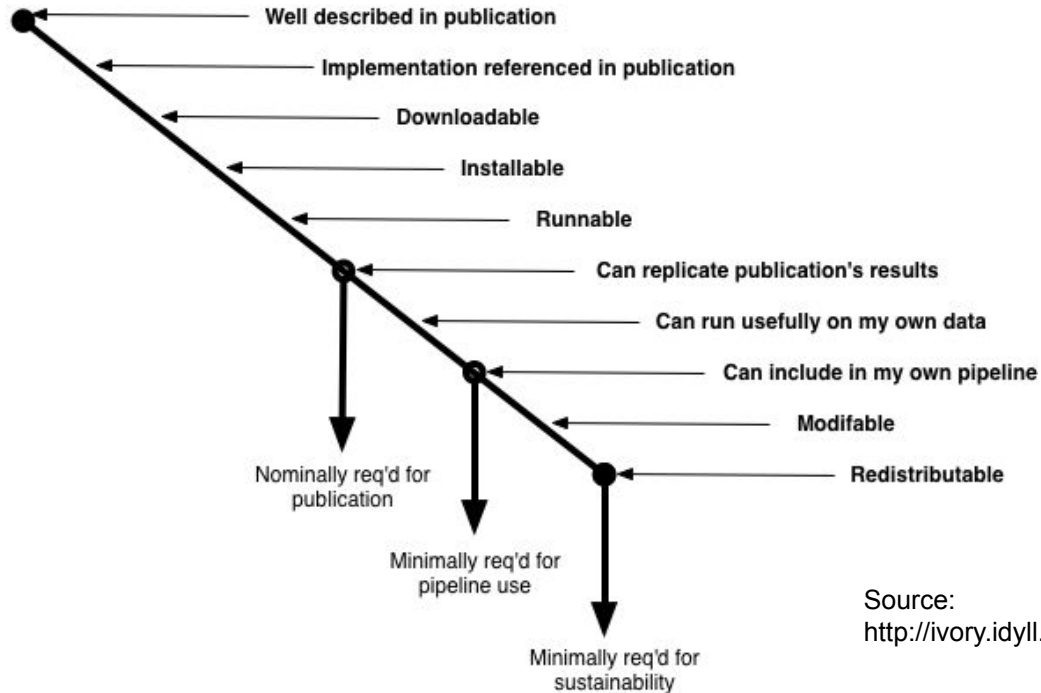
VRE limitations (& solutions?)

- Limited computational power
 - Potential to extend through dask+kubernetes (see Pangeo concept)
 - Costs a lot more; problems of managing usage
 - Needs more community s/w development & education to be useful
- Limited storage space
 - Raise it on per-user requests
 - Support connectivity with other services
- Less freedom to install/run other software
 - Can be expanded over time according to user demand (a clear pathway for this is needed)
 - Inevitably limited to what fits in the Jupyter paradigm
- Other difficulties related to the cloud nature and in-browser interaction

Sustainable software + notebooks

State of software in geomagnetism?

The Ladder of Academic Software Reusability and Sustainability



Source:

<http://ivory.idyll.org/blog/ladder-of-academic-software-notsuck.html>

Making code more useful and shareable

“ Can I run it ... ”

Sustainability ≡

Repeatability

“ ... again later? ”



Portability

“ ... on a different system? ”



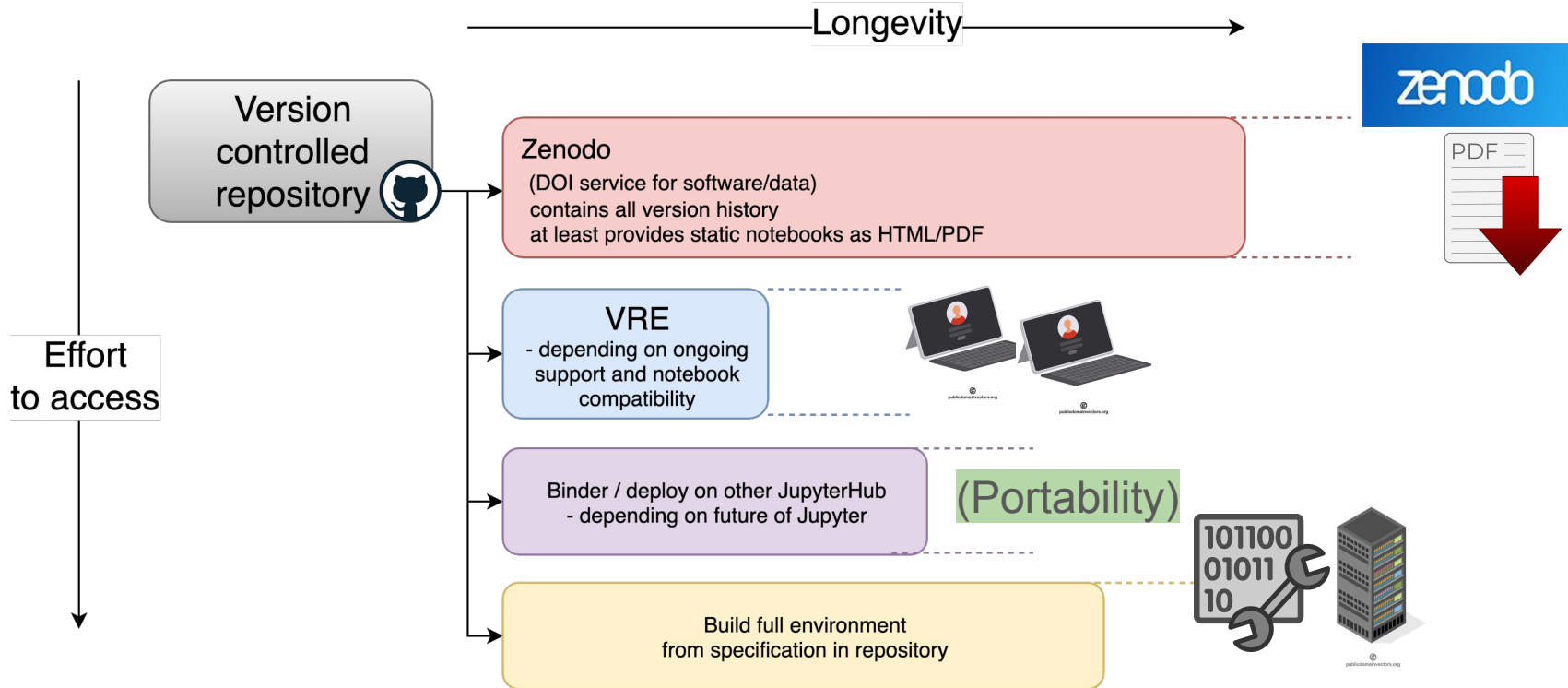
Extensibility

“ ... in a different way? ”



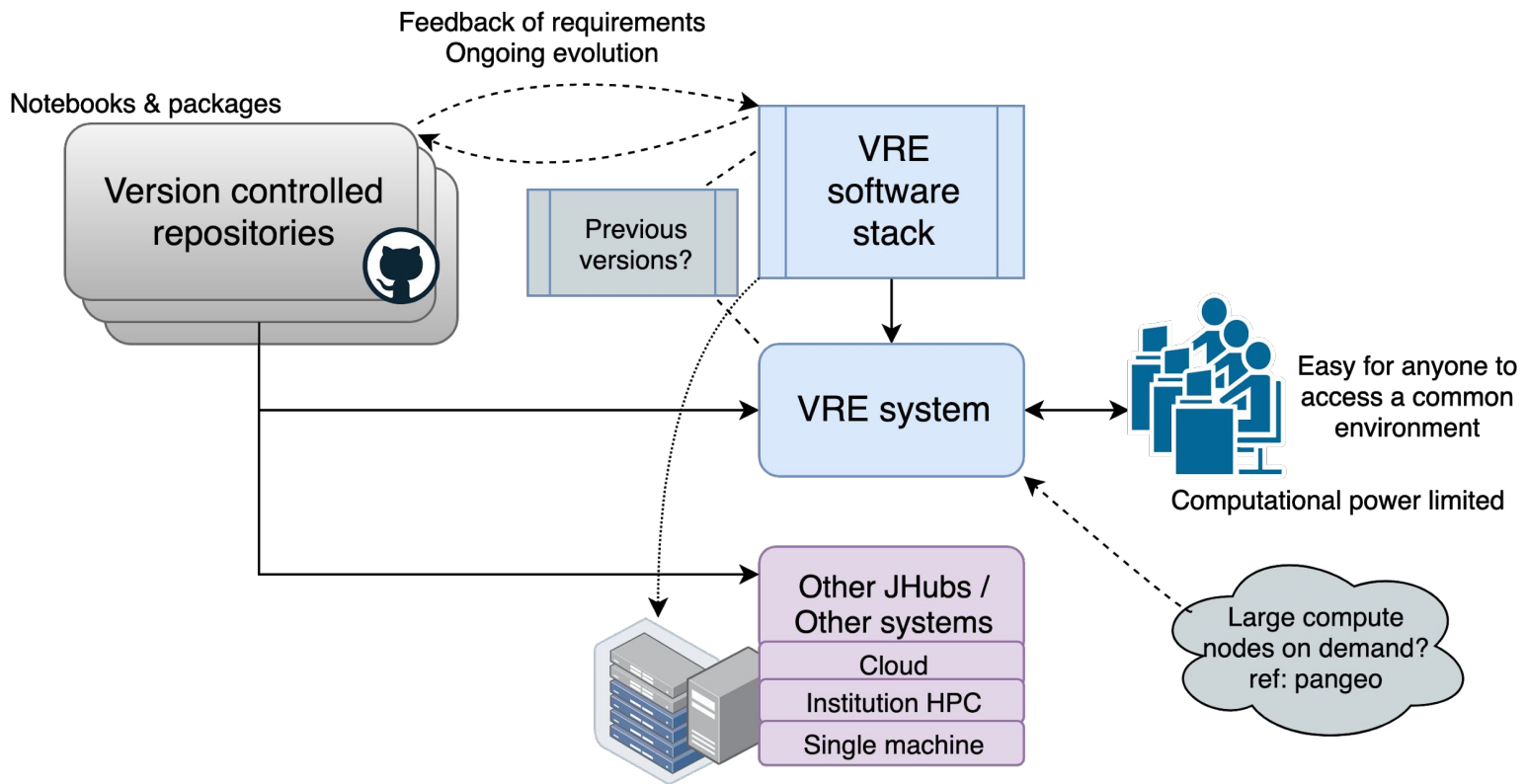
Reproducibility of notebooks

Notebooks as reproducible scientific analyses / reports



Ease of access and portability

Notebooks as tools to share and collaborate on science



From notebooks to packages

Notebooks as testing grounds

Packages for long term extensibility

notebook.ipynb

```
Fetch data

[1]: import xarray
xarray.set_options(keep_attrs=True)

from viresclient import ViresClient

# Set up connection
request = SaurReq
# Set collection id
request.set_collection_id('...')
# Set size of products
# Set size of products
# measurements (vs)
# models (magneti)
# auxiliaries (vol)
request.set_products(
    measurements=['F', 'B_NEC'],
    models=['CHADS-Core'],
    auxiliaries=['00Lat', '00Lon'],
    sampling_steps='PT10S'
)

# Fetch data from a given time interval
data = request.get_between(
    start_time='2014-01-02T00:00',
    end_time='2014-01-02T01:00'
)

# Load the data as an xarray.Dataset
ds = data.as_xarray()

[1/1] Processing: 100% [██████████] [Elapsed: 00:01, Remaining: 00:00 ]
[1/1] Downloading: 100% [██████████] [Elapsed: 00:00, Remaining: 00:00 ] (1.074MB)

[1]: xarray.Dataset

+ Dimensions:
+ Coordinates:
+ Timestamp (Timestamp)  datetime64[ns]  2014-01-01 ... 2014-01-02T00:59:50
+ Data variables:
Spacecraft (Timestamp)  <U1 'A'A'A'A...A'A'A'A'
Latitude (Timestamp)    float64  -1.229 -1.863 ... 50.27 49.64
Longitude (Timestamp)   float64  -14.12 -14.13 ... -33.4 -33.37
```

Inputs...
...Process

Output

Usable in *this* notebook

notebook.ipynb
module.py

```
from module import x

[2]: from src import fetch_data
ds = fetch_data()

[1/1] Processing: 100% [██████████] [Elapsed: 00:00, Remaining: 00:00 ]
[1/1] Downloading: 100% [██████████] [Elapsed: 00:00, Remaining: 00:00 ] (1.074MB)

[2]: xarray.Dataset

+ Dimensions:
+ Coordinates:
+ Timestamp (Timestamp)  datetime64[ns]  2014-01-01 ... 2014-01-02T00:59:50
+ Data variables:
Spacecraft (Timestamp)  <U1 'A'A'A'A...A'A'A'A'
Latitude (Timestamp)    float64  -1.229 -1.863 ... 50.27 49.64
Longitude (Timestamp)   float64  -14.12 -14.13 ... -33.4 -33.37
```

Repeated process

Usable by *all* notebooks in this repository

notebook.ipynb
[installed *package*]

```
from package import x

[2]: from src import fetch_data
ds = fetch_data()

[1/1] Processing: 100% [██████████] [Elapsed: 00:00, Remaining: 00:00 ]
[1/1] Downloading: 100% [██████████] [Elapsed: 00:00, Remaining: 00:00 ] (1.074MB)

[2]: xarray.Dataset

+ Dimensions:
+ Coordinates:
+ Timestamp (Timestamp)  datetime64[ns]  2014-01-01 ... 2014-01-02T00:59:50
+ Data variables:
Spacecraft (Timestamp)  <U1 'A'A'A'A...A'A'A'A'
Latitude (Timestamp)    float64  -1.229 -1.863 ... 50.27 49.64
Longitude (Timestamp)   float64  -14.12 -14.13 ... -33.4 -33.37
```

Shared process

Usable by *anyone*

- Package must be maintained
- Should be *extensible*
- Needs input from others

Maturity & usability



Reproducible analysis

Barriers

- Workflow issues
 - Unscripted steps (analysis as a DAG)
 - Data extracted through a GUI
 - Data passed from someone else
 - Difficult to follow (poor documentation)
- Data dependencies
 - External data (e.g. from another analysis)
 - Large data requiring long processing
 - Non-free data
- Software dependencies
 - Libraries difficult to install / configure
 - Non-free software
- Heavy computation needed



Solutions

- Complex learning process...
 - Find scriptable solutions
 - Date notebooks showing progress
 - Explain steps within notebooks (besides other forms of documentation)
- Different solutions for different data
 - Small data packaged with the analysis
 - Larger data stored elsewhere, with scripts to pull it in; store intermediate steps in processing?
- Push for better packaging & distribution of software
- Avoid non-free software & data
- Software like Dask help make computation portable

Visualisation tools

Menagerie of viz tools

HoloViz-maintained libraries



- Main resources:
 - **PyViz**: <https://pyviz.org/> (general tools across Python)
 - **HoloViz**: <https://holoviz.org/> (coordinated ecosystem)
- Interactive plots within notebooks
 - Planned to support through **HoloViz**
 - Can develop bespoke visualisations for specific tasks
- Dashboarding
 - Possibility to create live Python-backed applications
 - Options: **Panel** (part of HoloViz), Dash, Voila (<https://pyviz.org/dashboarding/index.html>)
 - Dashboards present a clean view (hiding the code) of tools that are actually part of a (optionally hidden) notebook - this makes them more easily modified by users themselves
 - See examples at <https://panel.holoviz.org/index.html>

