Intro to VRE

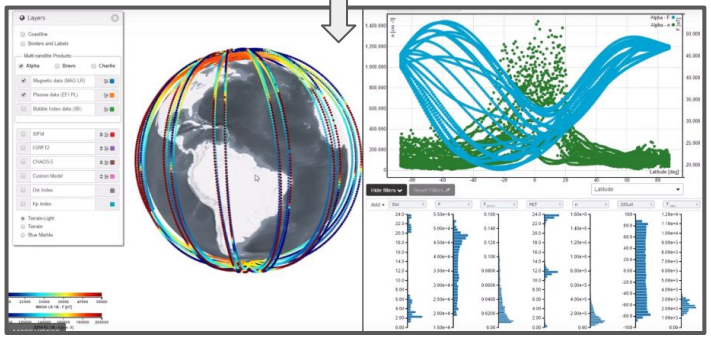
Ashley Smith (PDRA, Uni of Edinburgh, UK)
ashley.smith@ed.ac.uk

Virtual Research Environment (VRE) for Swarm

What is it?

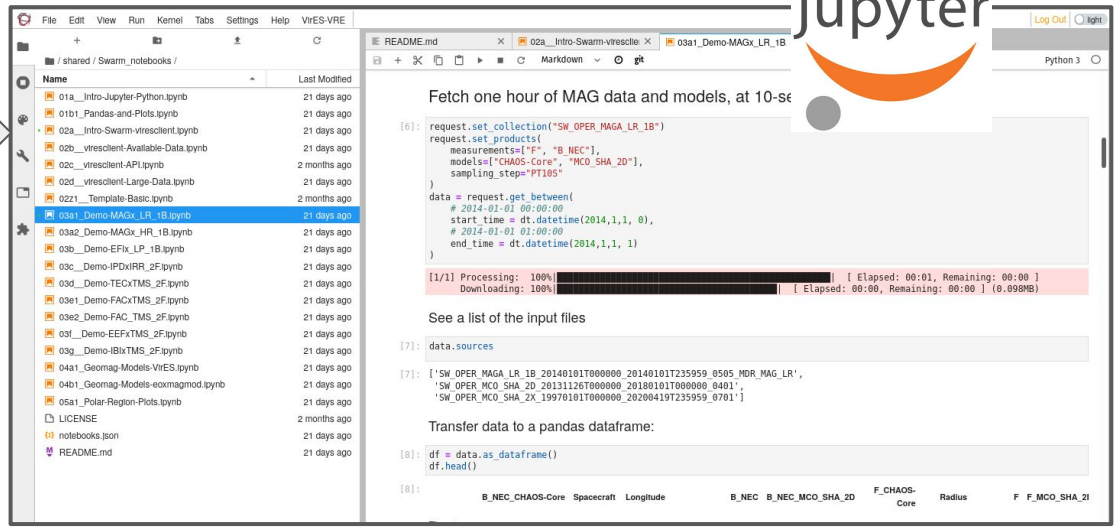
Free-to-use Jupyter environment in the cloud to easily access Swarm products

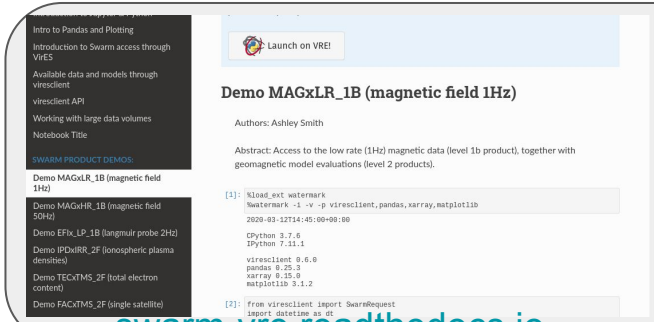
Extension of VirES (GUI)



vires.services

vre.vires.services



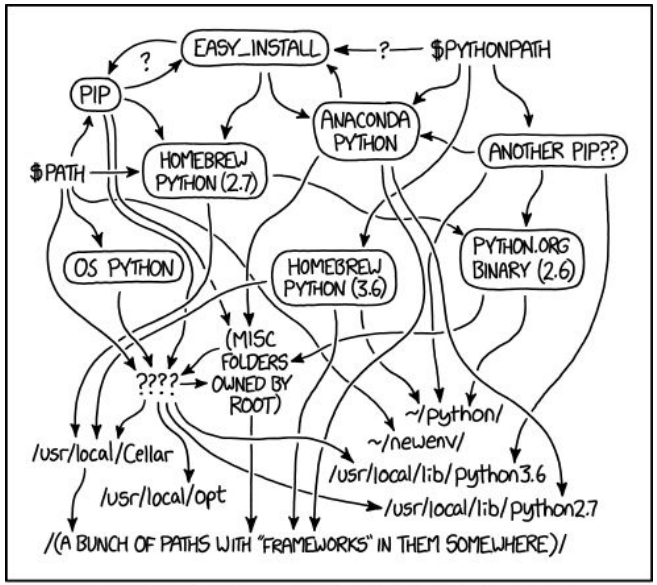


swarm-vre.readthedocs.io

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

Aims of VRE: *improve accessibility of Swarm*

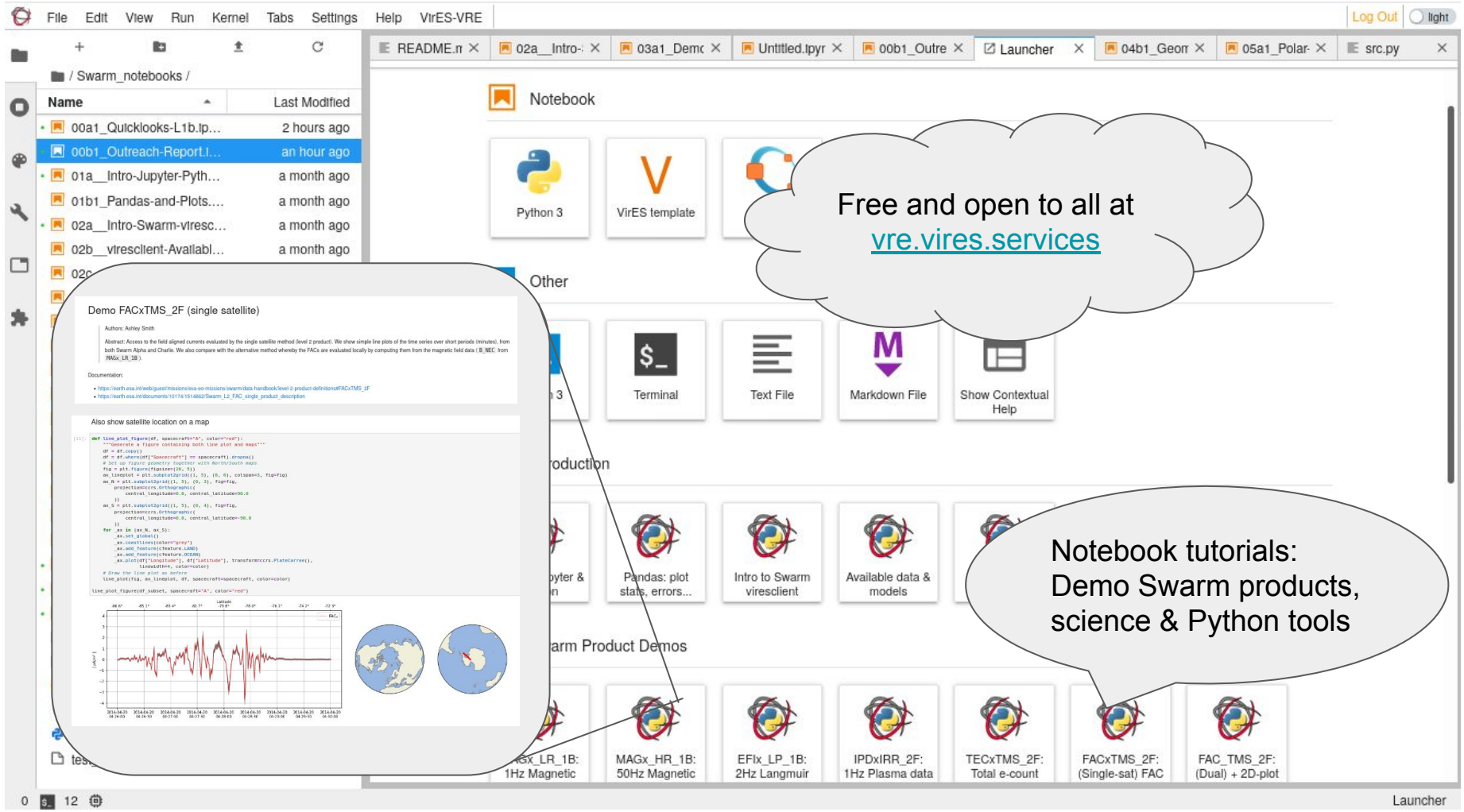
- Reduce IT skills necessary to just get started
- Easier navigation of & access to Swarm products



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

xkcd.com/1987

- Level 2 Products
  - Core Field
    - MCO_SHA_2C MCO_SHA_2D MCO_VAL_2C MCO_VAL_2D MCO_VAL_2
    - 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_2
  - Mantle Conductivity
    - MIN_1DM_2 MCR_1DM_2 MI1_VAL_2 MC1_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_2 IBIxTMS_2F TECxTMS_2F
    - FAC_TMS_2F FACxTMS_2F EEFxTMS_2F IPDxIRR_2F MIO_SHA_2E
    - IBP_CLI_2


File Edit View Run Kernel Tabs Settings Help VirES-VRE
Log Out
light
README.m
02a__Intro-
03a1_Demc
Untitled.ipyr
00b1_Outre
Launcher
04b1_Geom
05a1_Polar
src.py
/ Swarm_notebooks /
Name
Last Modified
00a1_Quicklooks-L1b.Ip… 2 hours ago
00b1_Outreach-Report.I… an hour ago
01a__Intro-Jupyter-Pyth… a month ago
01b1_Pandas-and-Plots.… a month ago
02a__Intro-Swarm-vIresc… a month ago
02b__vIresclient-Availabl… a month ago
Notebook
Python 3
VirES template
Other
Terminal
Text File
Markdown File
Show Contextual Help
Pandas: plot stats, errors...
Intro to Swarm viresclient
Available data & models
Product Demos
MAGx_LR_1B: 1Hz Magnetic
MAGx_HR_1B: 50Hz Magnetic
EFIx_LP_1B: 2Hz Langmuir
IPDxIRR_2F: 1Hz Plasma data
TECxTMS_2F: Total e-count
FACxTMS_2F: (Single-sat) FAC
FAC_TMS_2F: (Dual) + 2D-plot
Demo FACxTMS_2F (single satellite)
Also show satellite location on a map
Free and open to all at
vre.vires.services
Notebook tutorials:
Demo Swarm products,
science & Python tools
Launcher

VirES (**Vir**tual workspaces for **E**arth observation **s**cientists)

Data & model parameters

Various formats etc.

VirES server (for Swarm)
Time series access
Forward evaluation of models

Consistent interface to all products

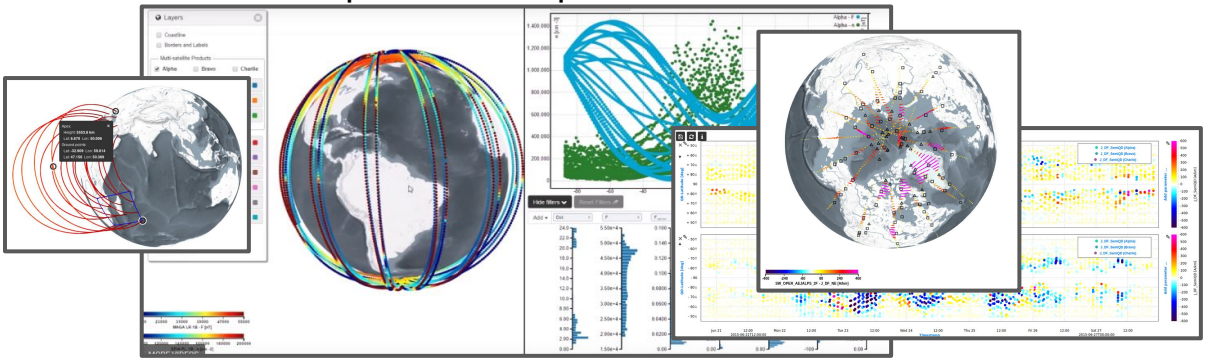

```
from viresclient import SwarmRequest

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)
```

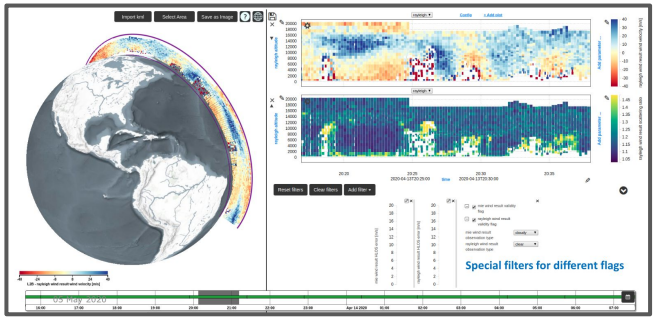
`viresclient` plugs into Python ecosystem

VRE service
Run your custom code

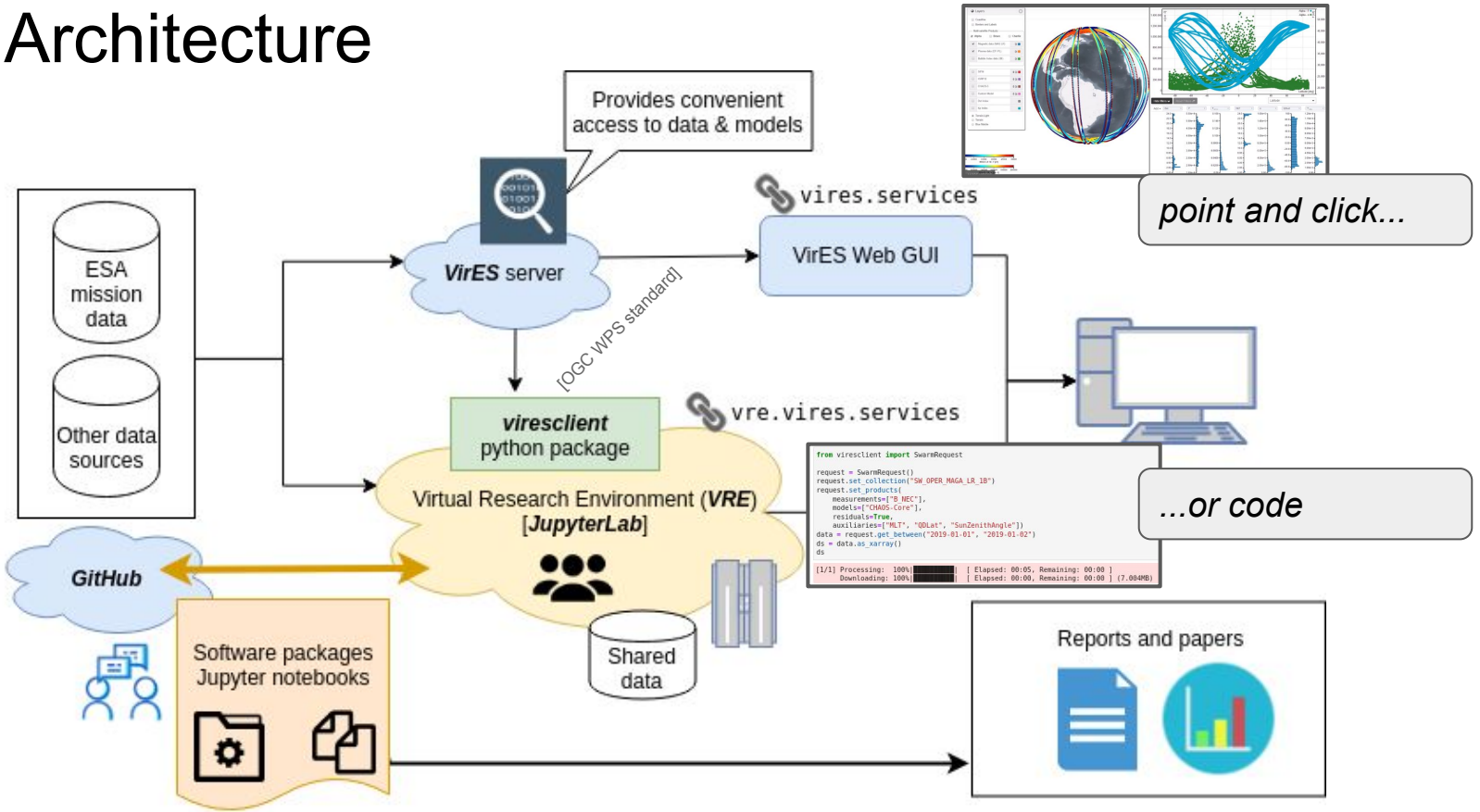
Web client provides bespoke visualisations



(Same system applied to Aeolus)



VRE Architecture



viresclient.readthedocs.io

viresclient

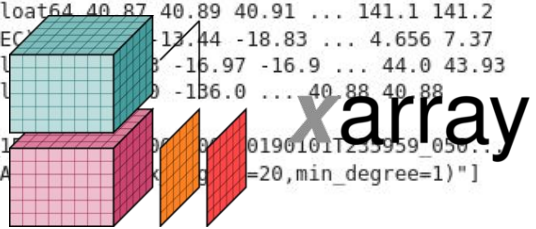
Access to **time series** of (most of) the Swarm products as pandas/xarray

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

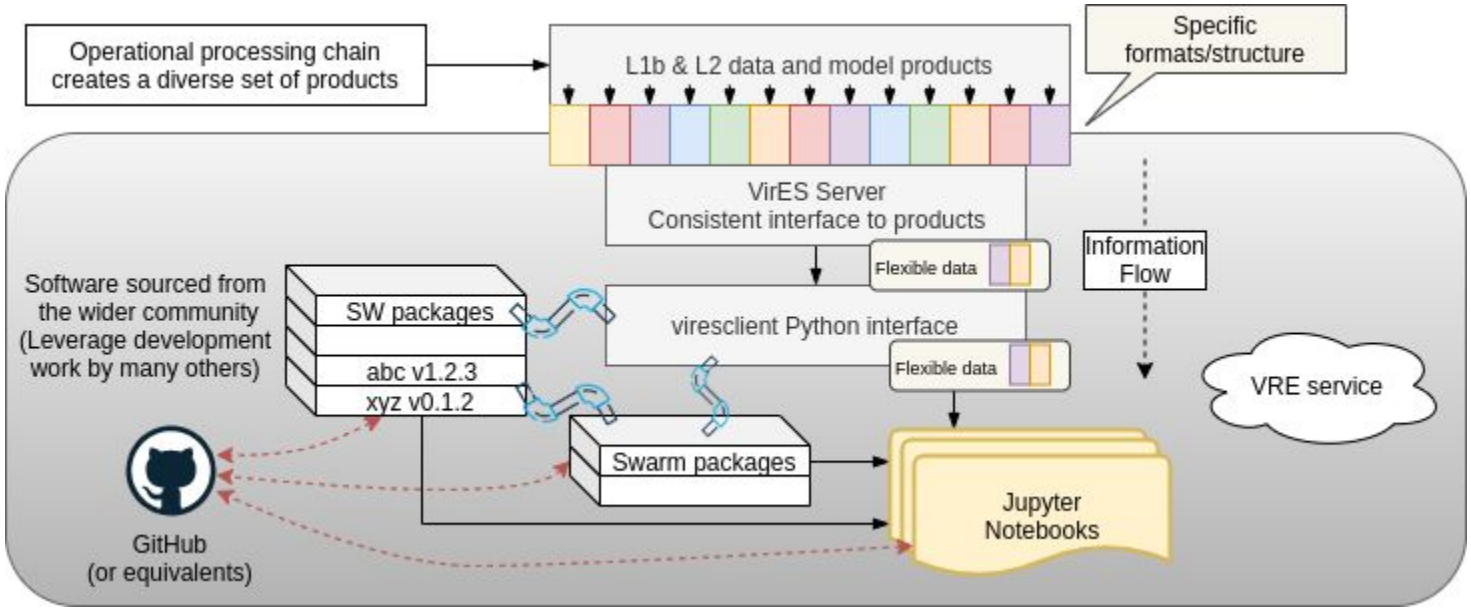
Geomagnetic models and other parameters **evaluated on-the-fly** on the server

```python
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) -13.44 -18.83 ... 4.656 7.37
    Latitude               (Timestamp) fl  -16.97 -16.9 ... 44.0 43.93
    Longitude              (Timestamp) fl  -136.0 ... 40.88 40.88
Attributes:
    Sources:          ['SW_OPER_MAGA_LR_1  ...  20190101T235959_050...']
    MagneticModels:   ["CHAOS-Core = 'CHA  ...  =20,min_degree=1)"]
    RangeFilters:     []
```



Information Architecture



Challenges & opportunities for software-based scientific research and services

Sophistication

Complexity

Tools

Data- & compute-heavy workloads

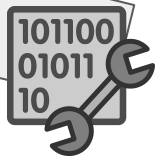
Bring the code to the data
Shared elastic resources

Sustainability

Onboarding difficulty

Development & maintenance workload

Reliability



Cloud services

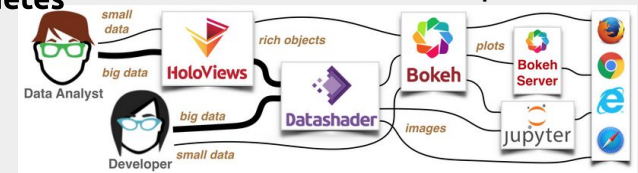
aws

openstack

kubernetes

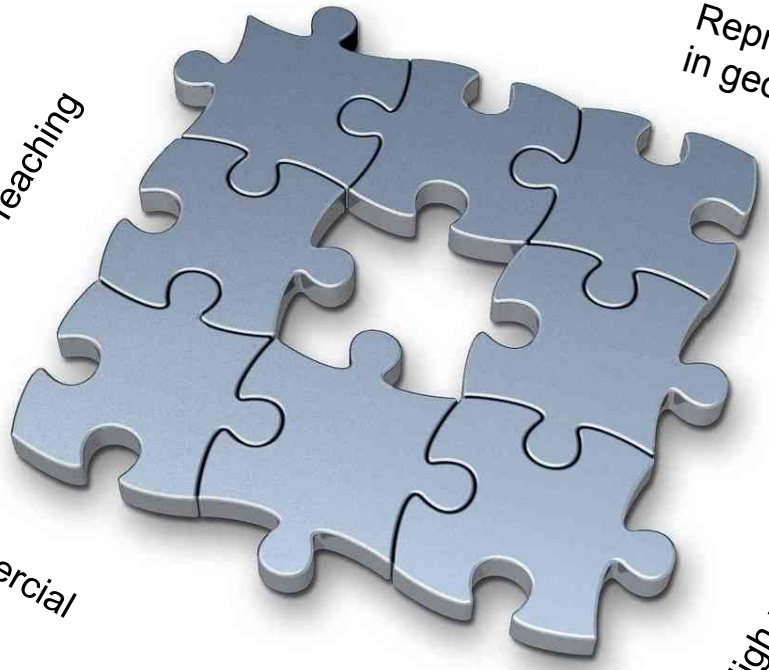
docker

Vast software landscapes



Accessibility and ease of use

Other Jupyter-y examples



data8.org

syzygy.ca

EDINA Noteable

Teaching

Reproducability in geospace

Resen (InGeO)

cocalc.com

Google Colab

Commercial

High performance
Big data geoscience

pangeo.io

\+ Lots more!

Plans

- Stimulate collaboration
  - Development of packages
  - Sharing of notebooks
- Encourage best practices

- 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

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

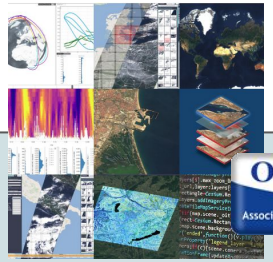
- Main GitHub repos [responsibility]:
  - [EOX] VirES Server github.com/ESA-VirES/VirES-Server
  - [EOX] VirES Web GUI github.com/ESA-VirES/WebClient-Framework
  - [EOX] eoxmagmod github.com/ESA-VirES/MagneticModel
  - [AS / EOX] viresclient github.com/ESA-VirES/VirES-Python-Client
  - [EOX] (VRE config 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.

eox.at/



EOX IT Services
Expertise in geospatial software and services

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

esa

Wider research communities

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

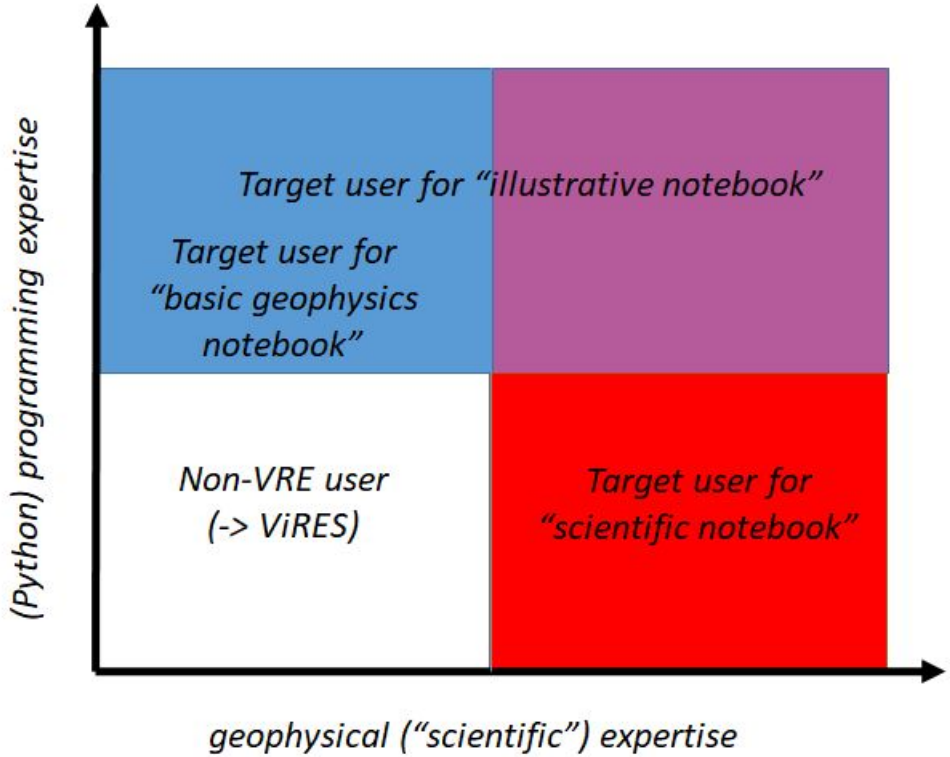
Other Earth & space obs

Notebook development & community connections

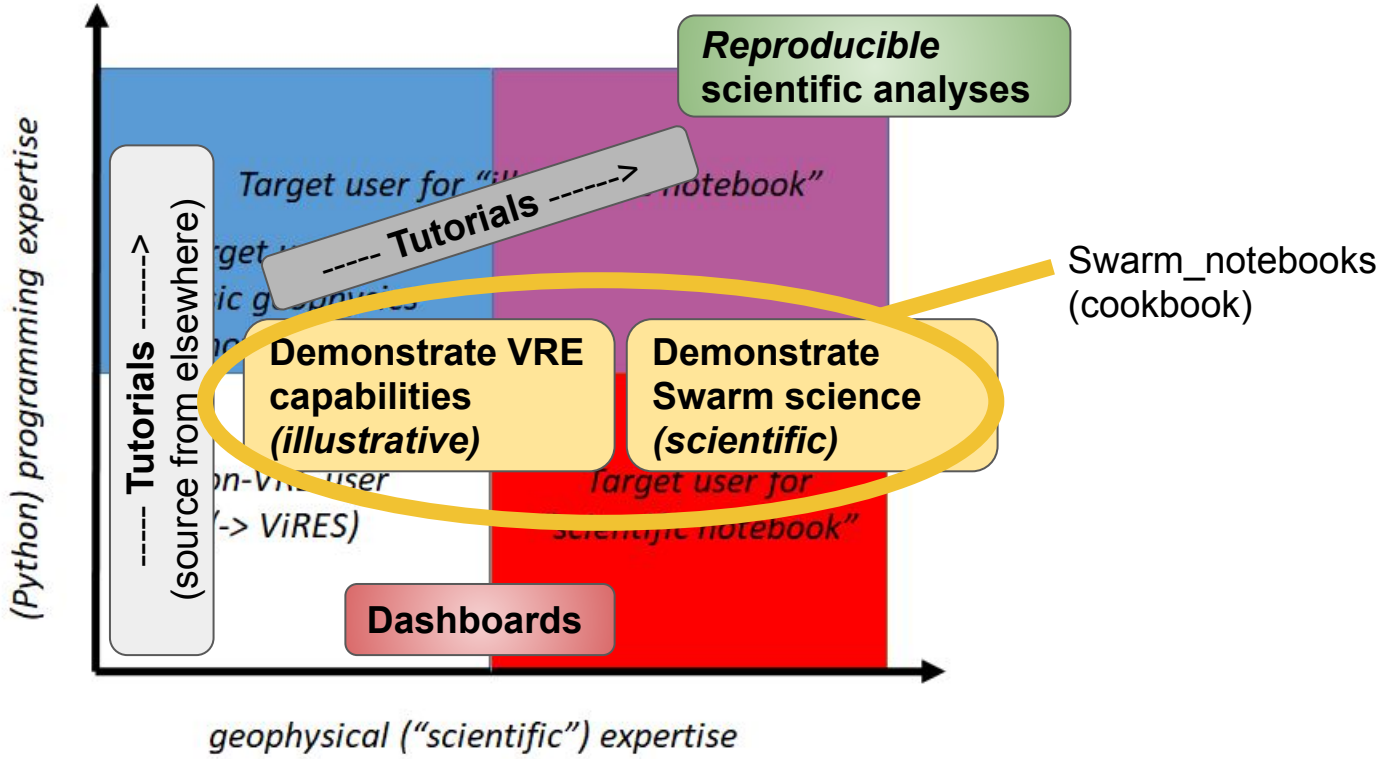
Current notebooks

- 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

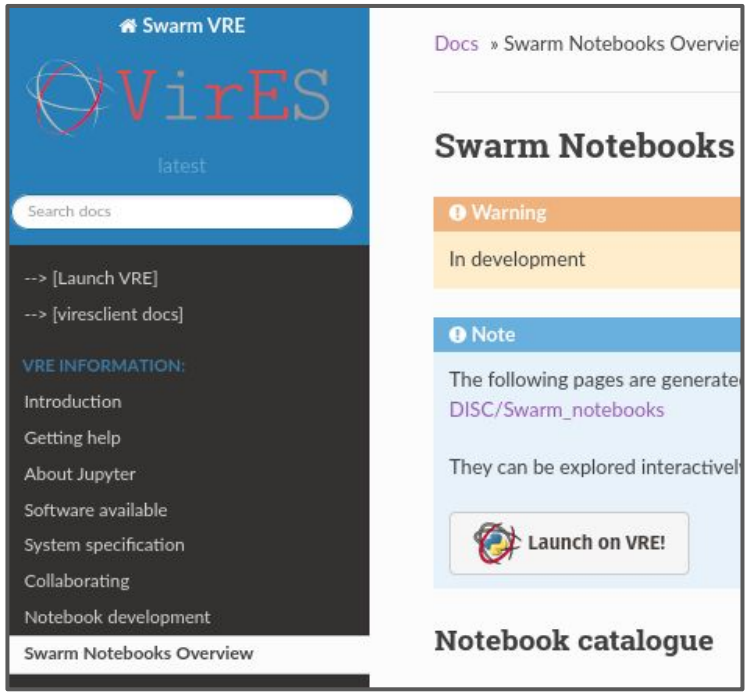


Themed collections in [GitHub/Swarm-DISC/Swarm_notebooks]

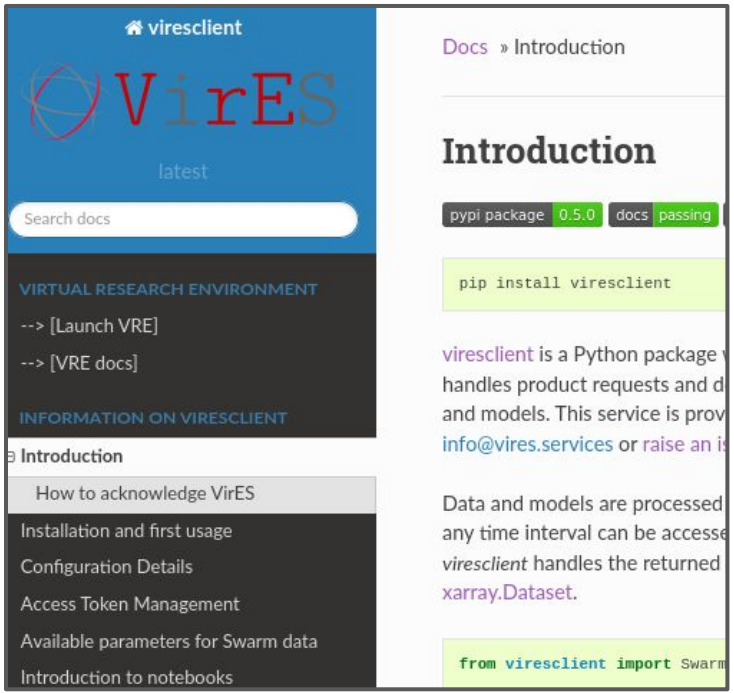
	First notebook name of each theme	***Description***
Illustrative	01a__Intro-Jupyter-Python ...	Generic Python guide
Illustrative	02a__Intro-Swarm-viresclient ...	Guide to VirES concepts & related libs
Illustrative	03a__Product-Demo-MAG ...	Demo access to each Swarm product
Scientific	04a__Magnetic-Models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

Documentation sites

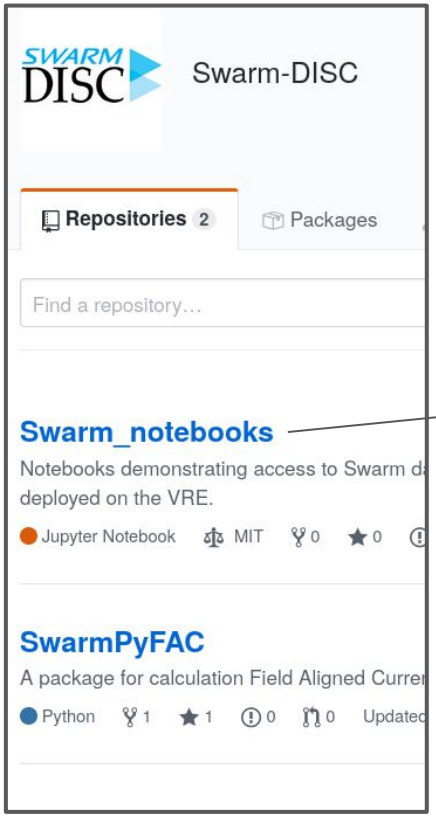
swarm-vre.readthedocs.io

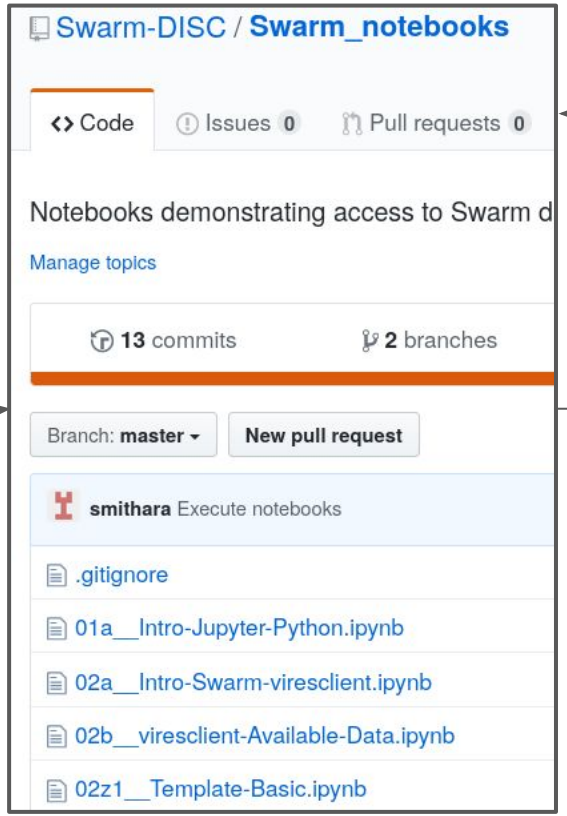


viresclient.readthedocs.io



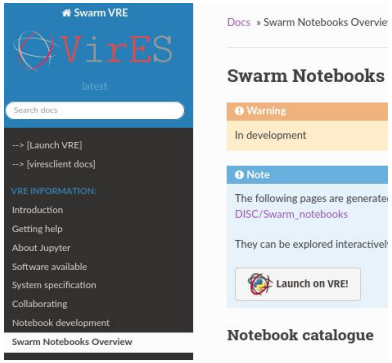
Coordination through GitHub





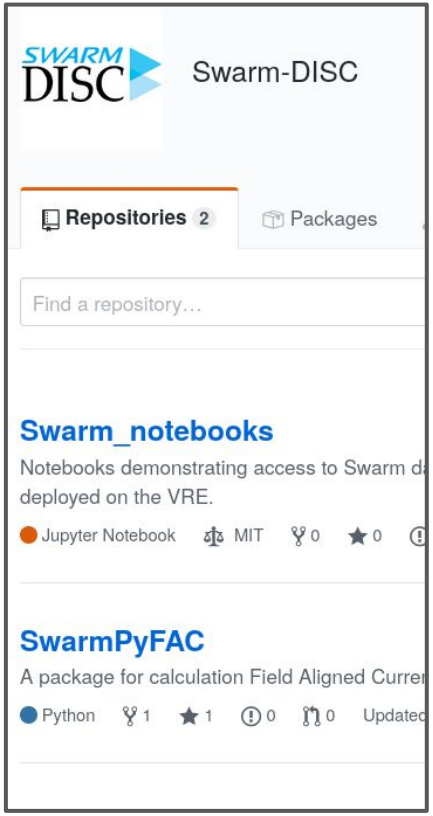
Run NBs

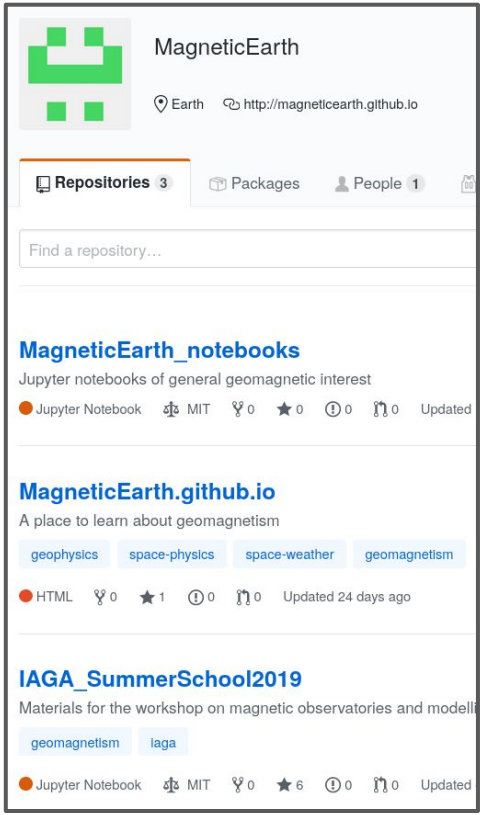
View NBs and guidance

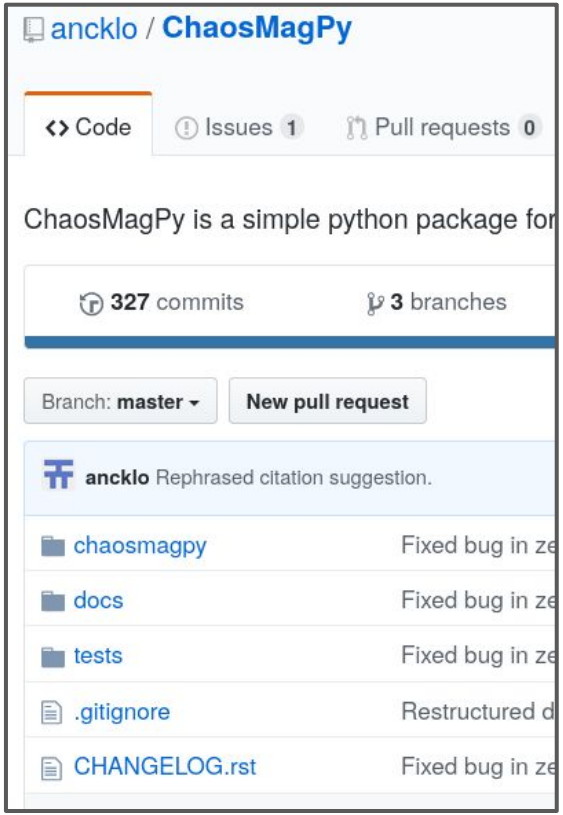


Expose Swarm products and usage

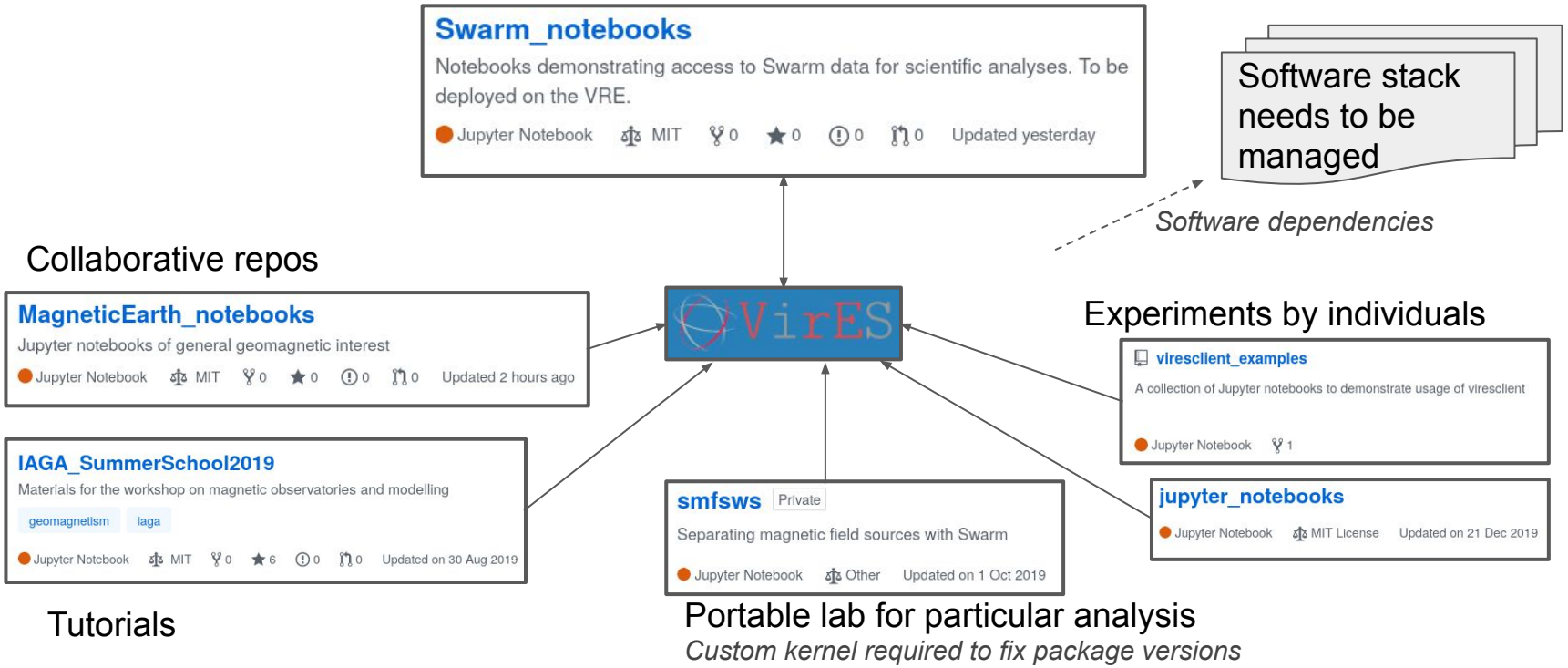
Coordination through GitHub - Orgs & Packages



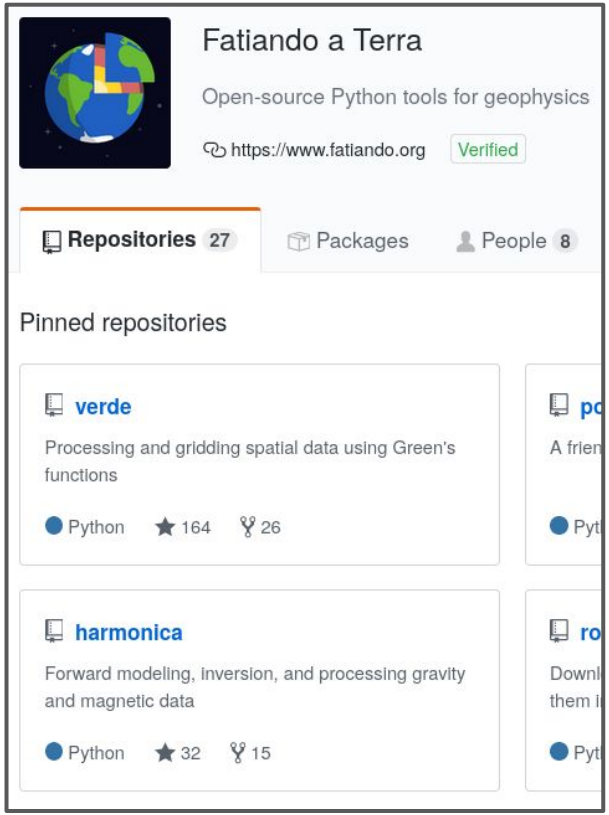


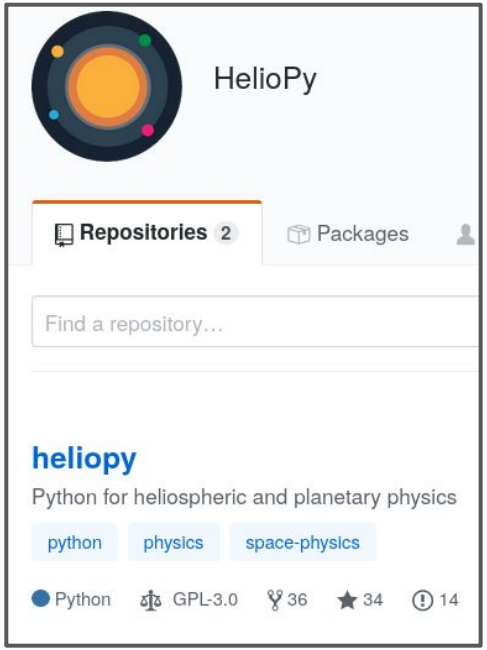


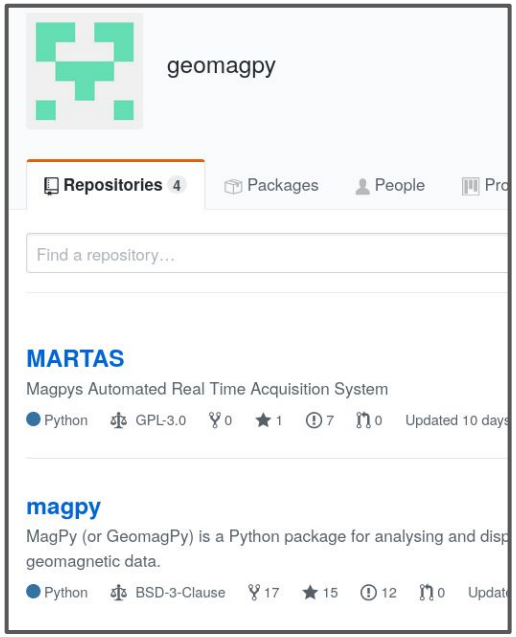
Coordination through GitHub - Deploy NBs



Coordination through GitHub - Other Projects?







....

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

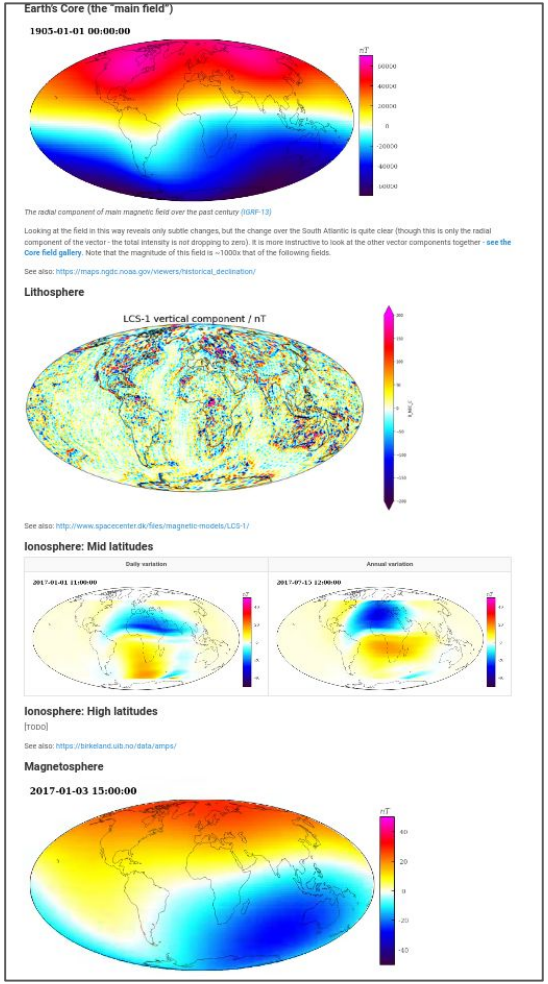
Challenges for science with Swarm

- 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



magneticearth.org/pages/magsources.html

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*

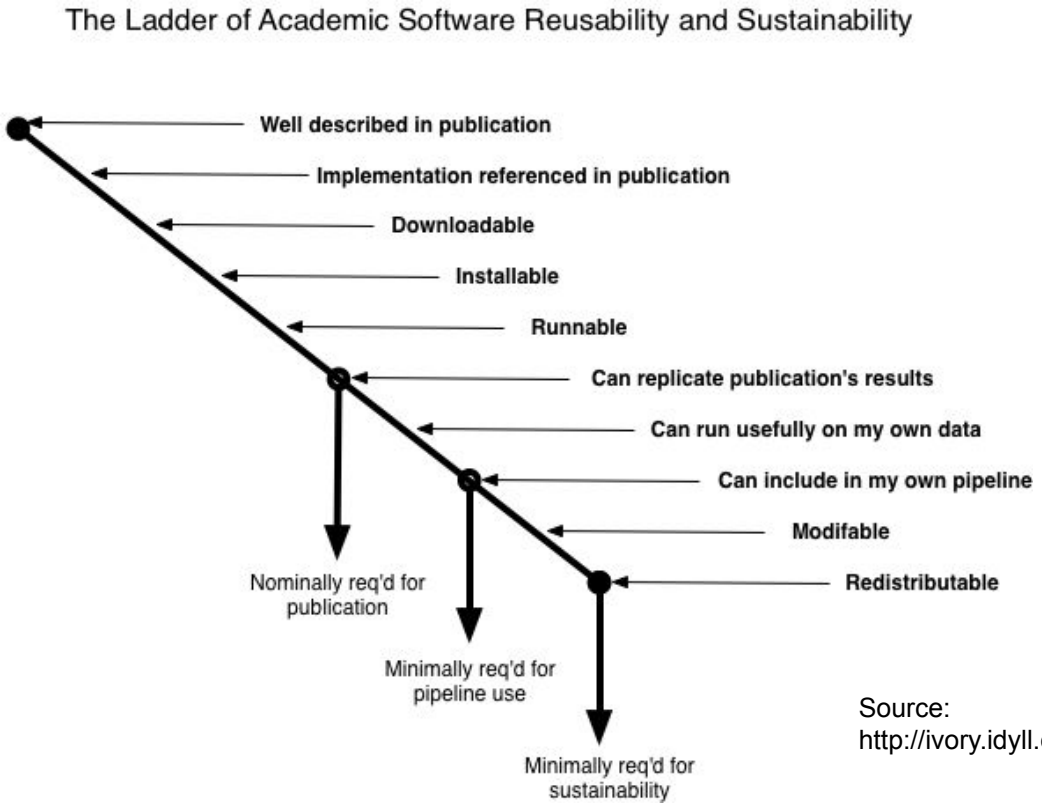
VRE limitations (& solutions?)

- 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

https://software.ac.uk/resources/get-speed

State of software in geomagnetism?



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?”

+

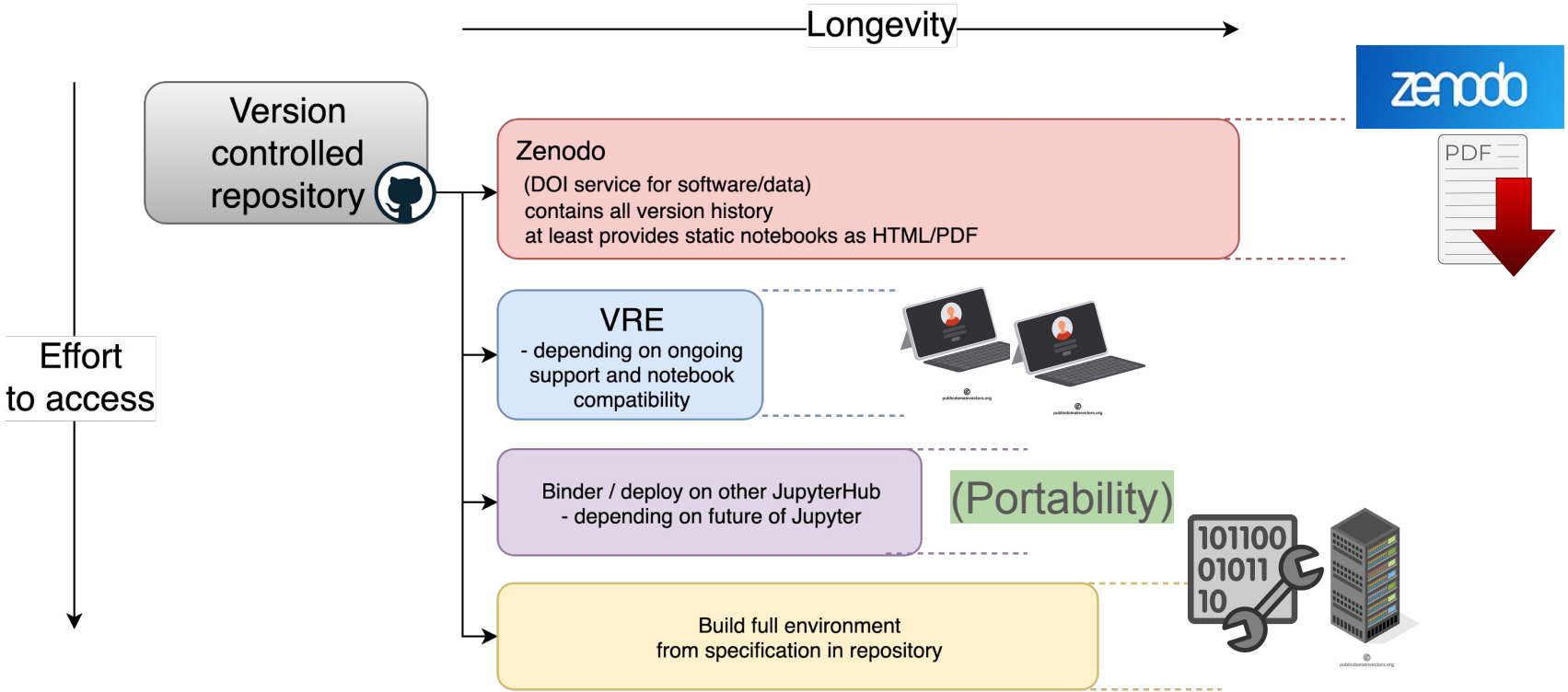
Portability “... on a different system?”

+

Extensibility “... in a different way?”

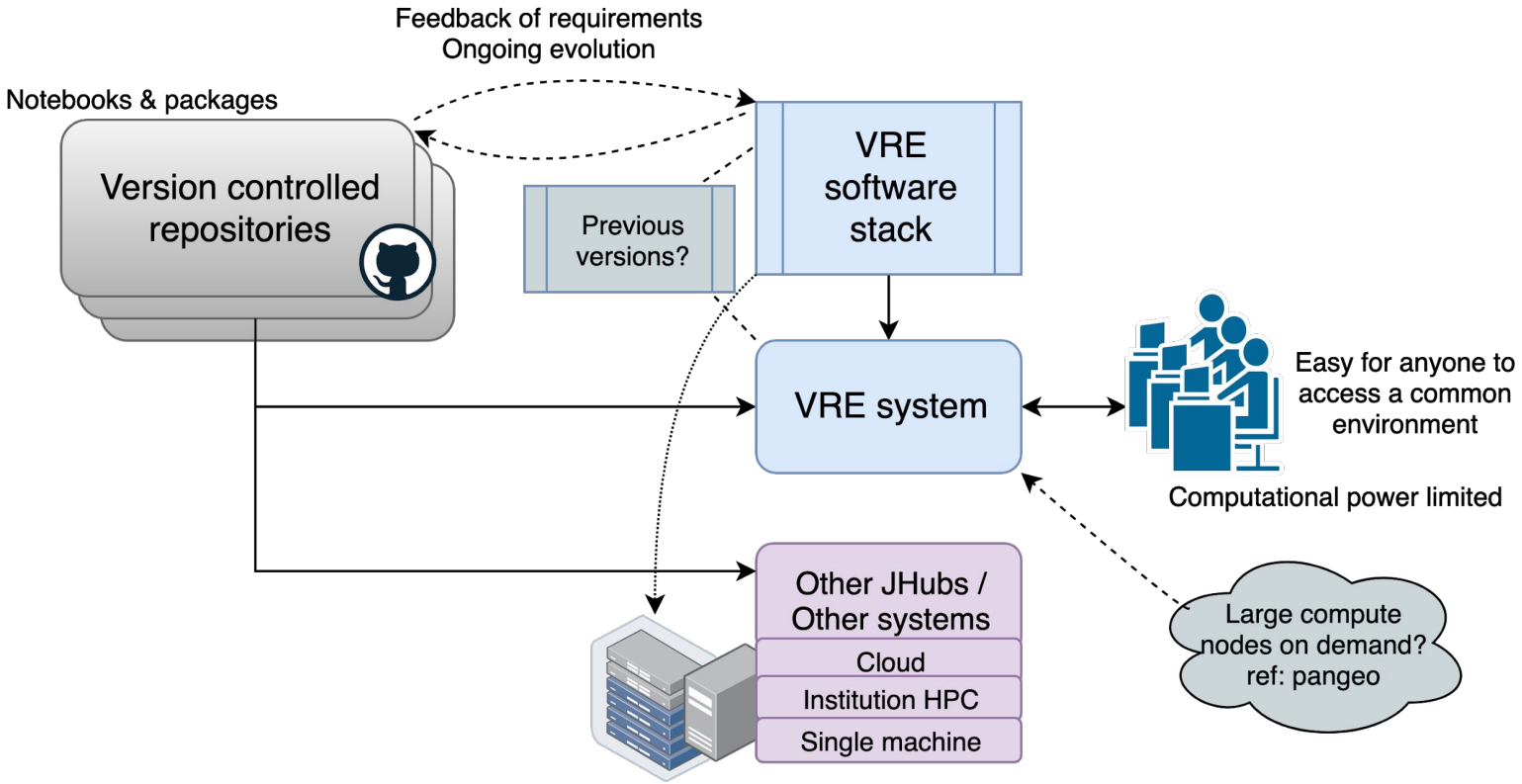
Reproducibility of notebooks

Notebooks as reproducible scientific analyses / reports



Notebooks as tools to share and collaborate on science

Ease of access and portability



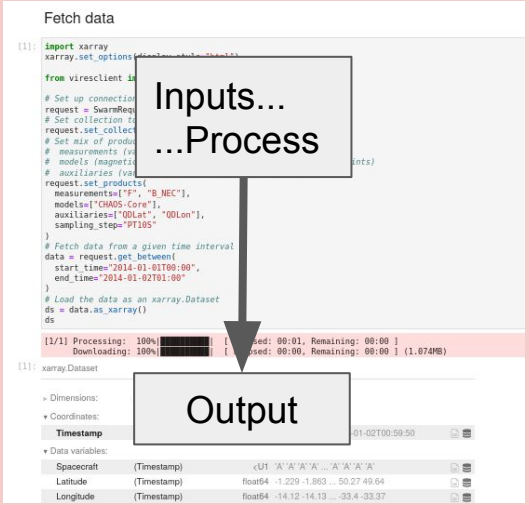
Extensibility

From notebooks to packages

Notebooks as testing grounds

Packages for long term extensibility

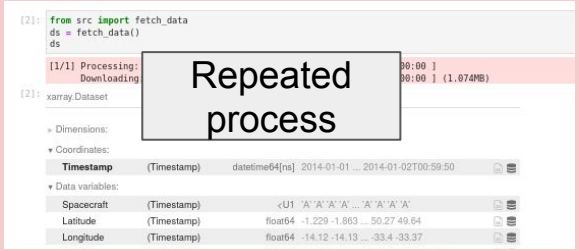
notebook.ipynb



Usable in *this* notebook

notebook.ipynb
module.py

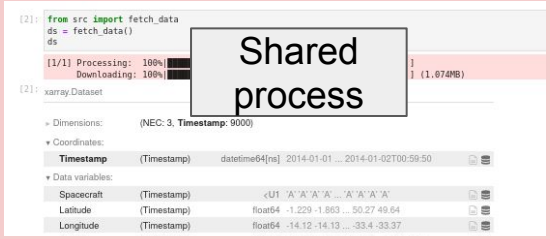
```
from module import x
```



Usable by *all* notebooks in this repository

notebook.ipynb
[installed ***package***]

```
from package import x
```



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

- 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