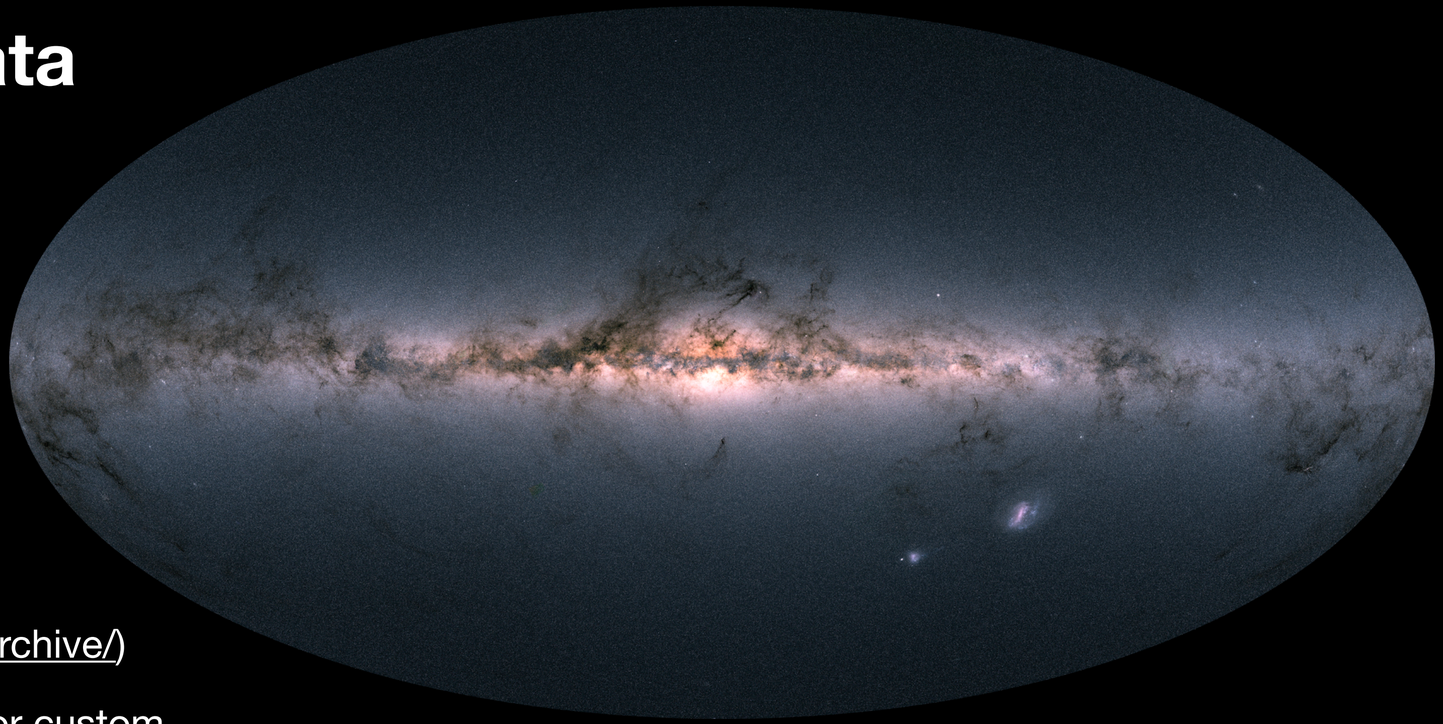
Remote DataFrames and interactive exploration of Big astronomical datasets with Vaex

Building big data dashboards with Voila and Vaex

Maarten Breddels (vaex.io), Joshua Peek (STScI), Sergey Koposov (CMU)

Main problem
Big, or uncomfortable big data



- Example
  - Gaia DR2: ~1.7 billion stars/rows, 95 columns
  - 1 TB of data (1+ day @10mb/s)
  - Barely fits on most laptops
  - How do you process it?
  - SQL/ADQL interface (e.g. https://gea.esac.esa.int/archive/)
    - Limited in scope, not a ‘PyData’ API, too slow for custom filtering and aggregation
- Others
  - Pan-STARSS
  - WISE
  - 2MASS
  - LSST

Main problem

Data at Your Fingertips: A Real-Time Discovery Engine for Gaia

- Proposed
  - A system to query N-d histograms
  - Custom filtering
  - Create 1d and 2d histograms
  - Create and share notebooks/dashboards
- Solution almost existed
  - Vaex

Sergey Koposov (CMU)

Joshua Peek (STScI)

Vaex

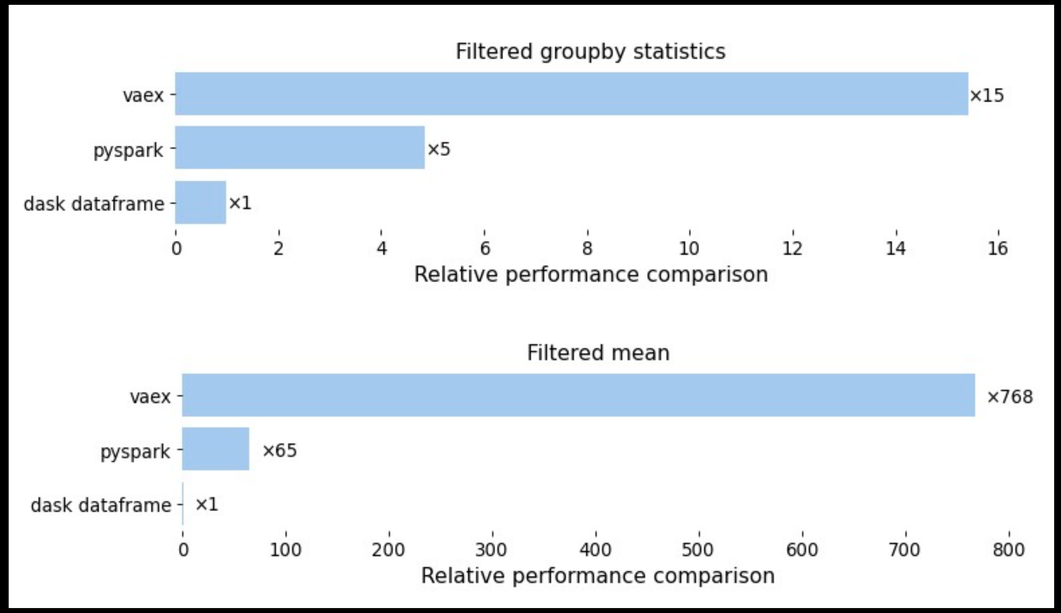
Out-Of-Core DataFrame library for Python

- Process ~1 billion rows / second (no cluster)
- Out-of-Core means larger than memory
- Pip/conda installable
  - pip install vaex
  - conda install -c conda-forge vaex
- DataFrame API - pandas like
  - Remote DataFrame

Vaex is fast (and memory efficient)

https://towardsdatascience.com/beyond-pandas-spark-dask-vaex-and-other-big-data-technologies-battling-head-to-head-a453a1f8cc13

- Comparison between
  - (Py)Spark
  - Koalas
  - Dask.DataFrame
  - Vaex
  - (py)datatable
  - Turicreate



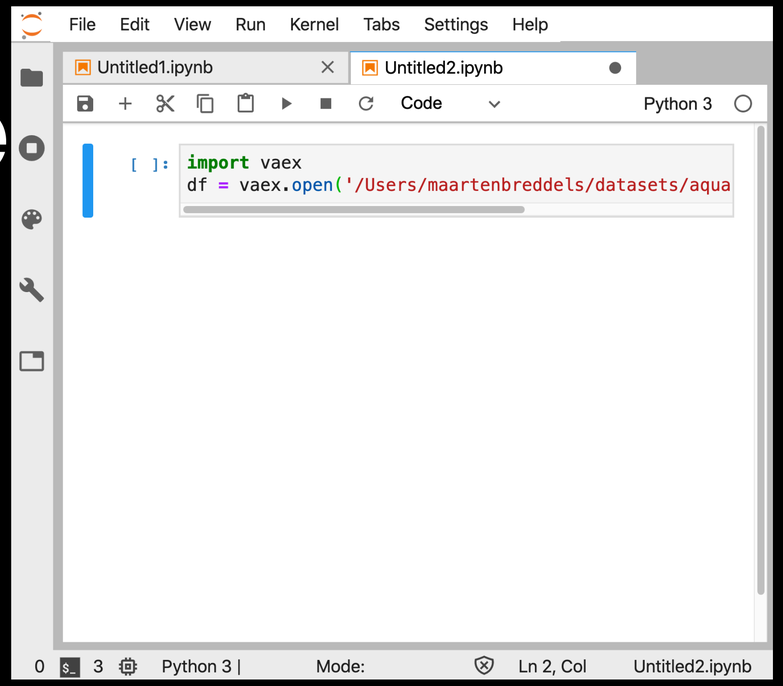
Notebook demo

- Work with the Gaia DR2 dataset
- Key features
- Visualization
- Voila intro

Remote DataFrame

- ‘Bring the compute to the data’
  - Jupyter notebook/lab/hub
  - Dask
- Arbitrary code execution
- DataFrame server
  - May not want to execute the ‘application’ code on the ‘data’ server
  - Use local jupyter environment
  - Use your local Python editor (vscode/vim/emacs)
  - Use from a machine in the cloud: e.g. mybinder running Voila
- dataframe.gaia.vaex.io
- dataframe.vaex.io

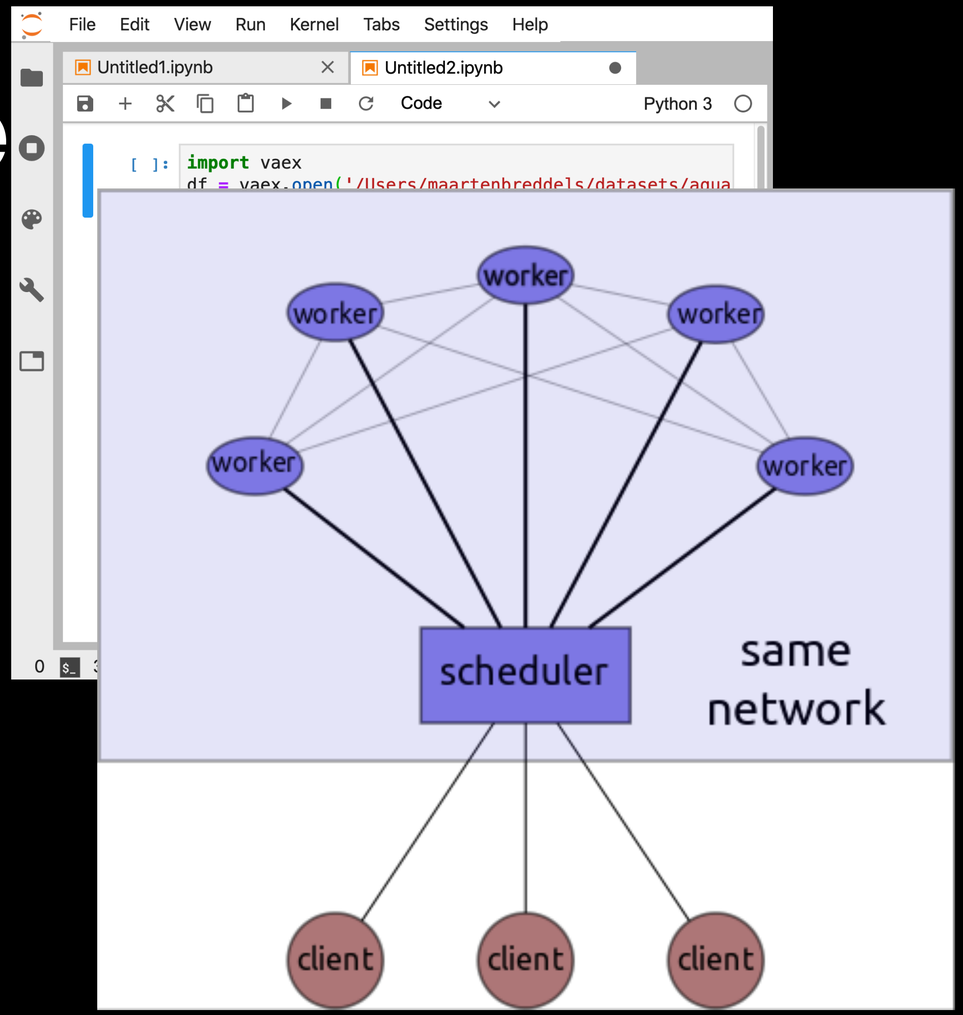
Remote DataFrame



- ‘Bring the compute to the data’
  - Jupyter notebook/lab/hub
  - Dask
- Arbitrary code execution
- DataFrame server
  - May not want to execute the ‘application’ code on the ‘data’ server
  - Use local jupyter environment
  - Use your local Python editor (vscode/vim/emacs)
  - Use from a machine in the cloud: e.g. mybinder running Voila
- dataframe.gaia.vaex.io
- dataframe.vaex.io

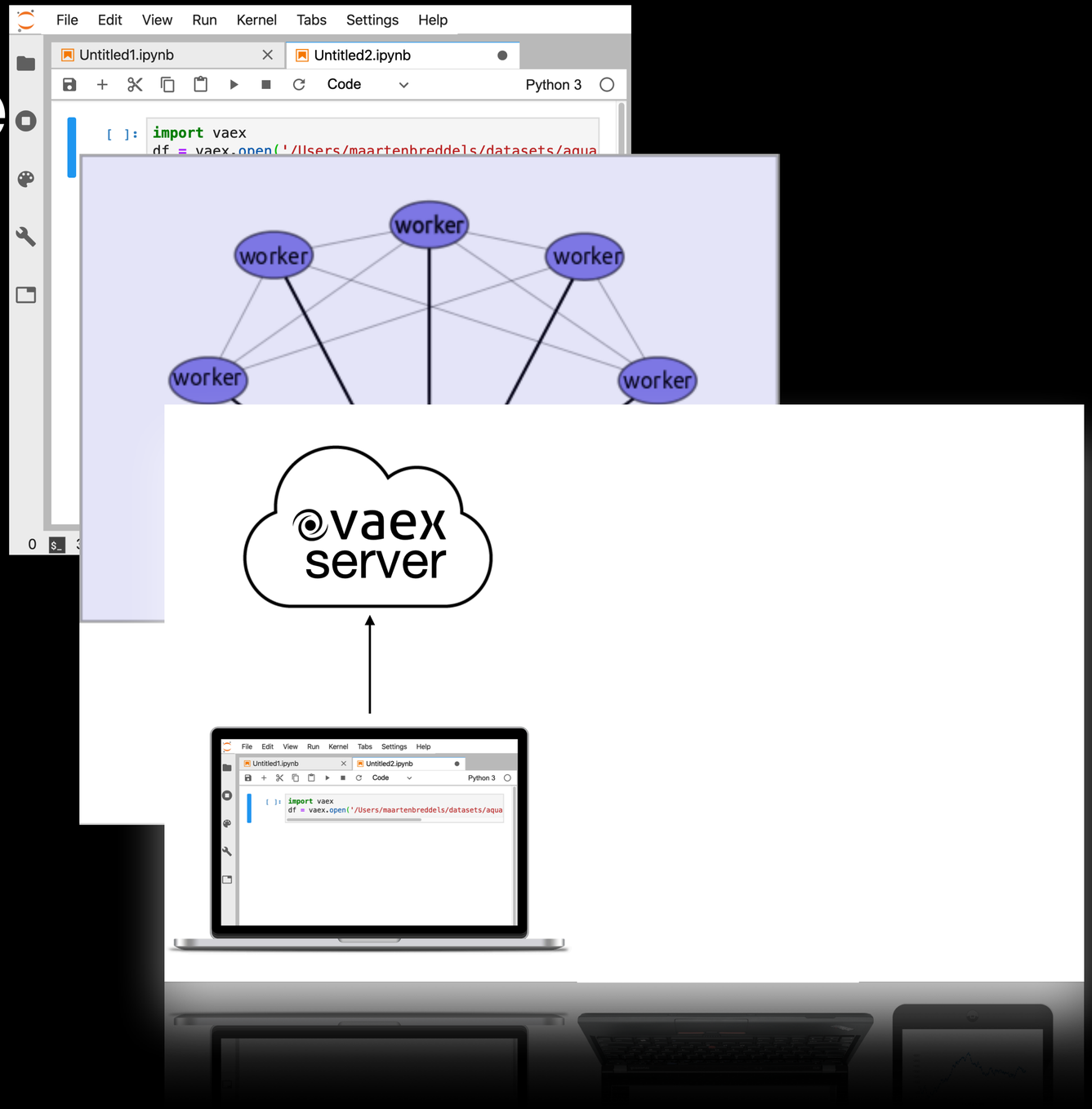
Remote DataFrame

- ‘Bring the compute to the data’
  - Jupyter notebook/lab/hub
  - Dask
- Arbitrary code execution
- DataFrame server
  - May not want to execute the ‘application’ code on the ‘data’ server
  - Use local jupyter environment
  - Use your local Python editor (vscode/vim/emacs)
  - Use from a machine in the cloud: e.g. mybinder running Voila
- dataframe.gaia.vaex.io
- dataframe.vaex.io



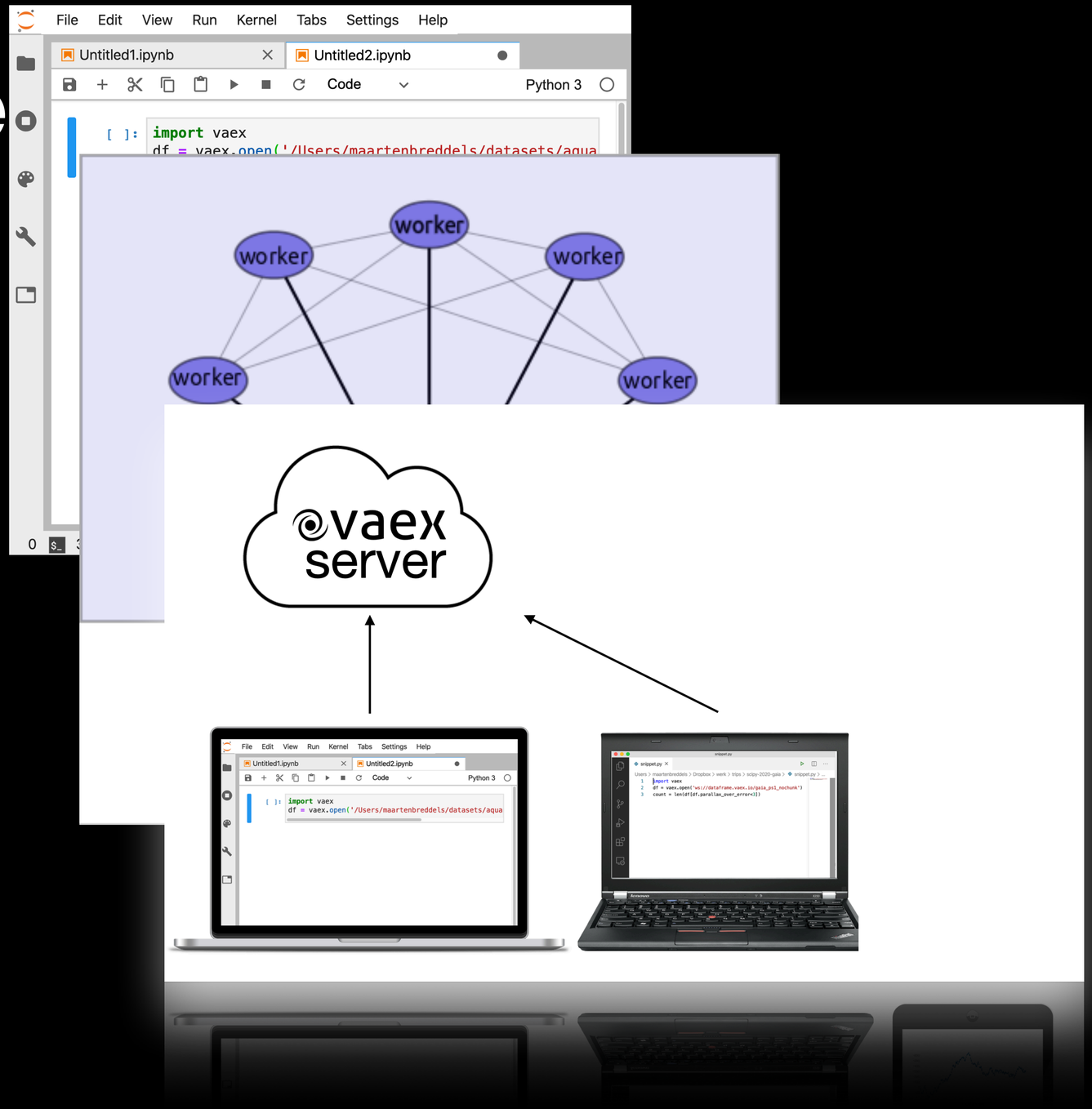
Remote DataFrame

- ‘Bring the compute to the data’
  - Jupyter notebook/lab/hub
  - Dask
- Arbitrary code execution
- DataFrame server
  - May not want to execute the ‘application’ code on the ‘data’ server
  - Use local jupyter environment
  - Use your local Python editor (vscode/vim/emacs)
  - Use from a machine in the cloud: e.g. mybinder running Voila
- dataframe.gaia.vaex.io
- dataframe.vaex.io



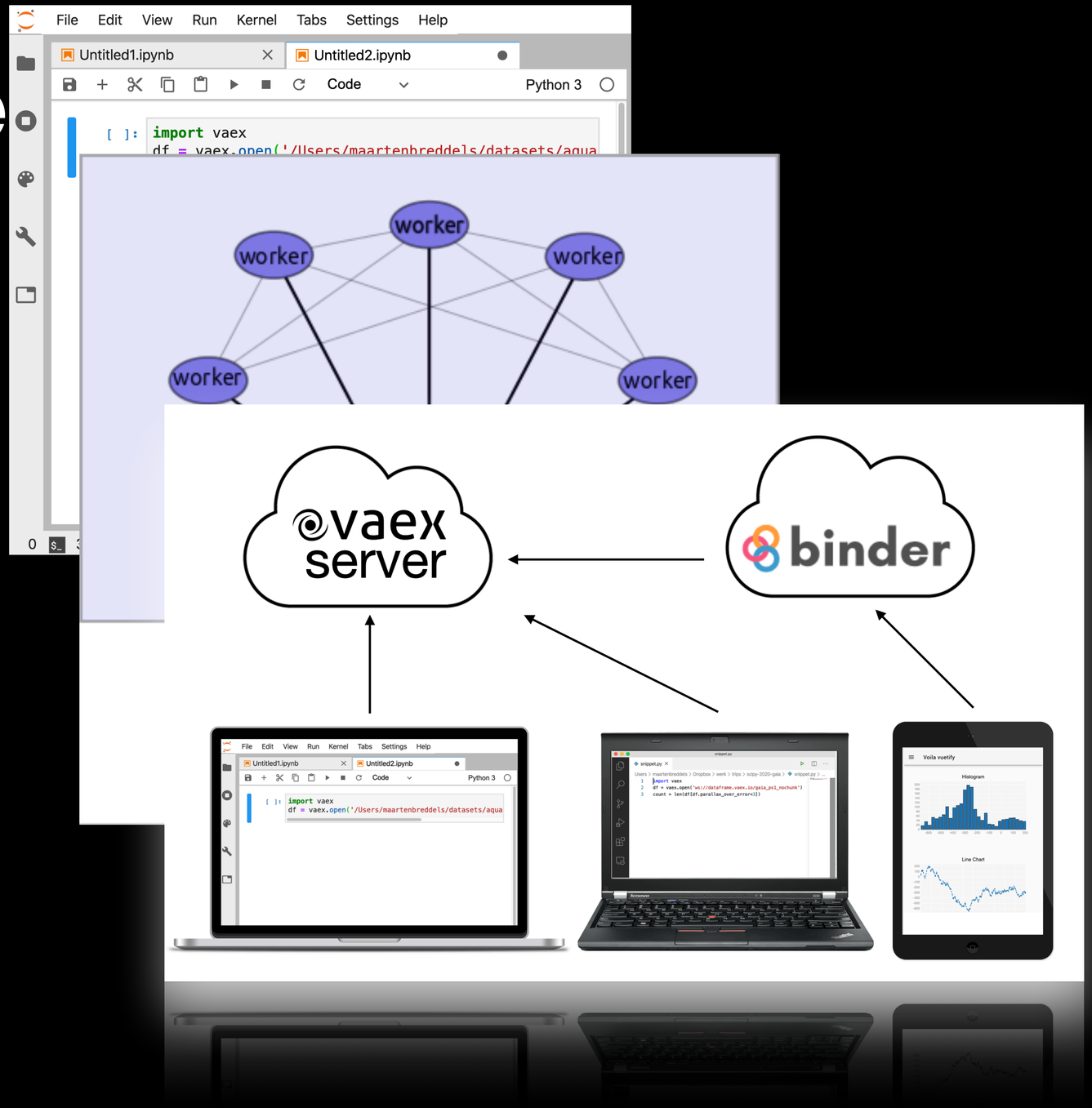
Remote DataFrame

- ‘Bring the compute to the data’
  - Jupyter notebook/lab/hub
  - Dask
- Arbitrary code execution
- DataFrame server
  - May not want to execute the ‘application’ code on the ‘data’ server
  - Use local jupyter environment
  - Use your local Python editor (vscode/vim/emacs)
  - Use from a machine in the cloud: e.g. mybinder running Voila
- dataframe.gaia.vaex.io
- dataframe.vaex.io



Remote DataFrame

- ‘Bring the compute to the data’
  - Jupyter notebook/lab/hub
  - Dask
- Arbitrary code execution
- DataFrame server
  - May not want to execute the ‘application’ code on the ‘data’ server
  - Use local jupyter environment
  - Use your local Python editor (vscode/vim/emacs)
  - Use from a machine in the cloud: e.g. mybinder running Voila
- dataframe.gaia.vaex.io
- dataframe.vaex.io



Summary

- Vaex
  - vaex-core: fast data processing (1 billion / second)
  - vaex-jupyter: Interactive viz for Jupyter
  - vaex-server: Remote aggregation
- Voila
  - voila: renders notebooks as dashboards
  - Ipyvuetify: Rich widgets library
  - voila-vuetify: Responsive web apps
- Vaex+Voila
  - Create interactive dashboard for big datasets without a cluster

Future Plans

- Depending on funding
  - Examples/widgets
  - Allow remote joins/groupby
  - Distributed DataFrame (Dask, Ray, …)
    - $10^{10\text{-}12}$ (trillion) rows
  - Glue/Glue-Jupyter integration
    - https://glueviz.org/

Resources

- https://github.com/maartenbreddels/gde-examples (more Gaia notebook examples)
- Packages
  - https://github.com/vaexio/vaex/
  - https://github.com/voila-dashboards/voila
  - https://github.com/voila-dashboards/voila-vuetify
  - https://github.com/mariobuikhuizen/ipyvuetify
- Documentation
  - https://docs.vaex.io/en/latest/tutorial.html
  - https://docs.vaex.io/en/latest/tutorial_jupyter.html
- Contact
  - Maarten Breddels <maartenbreddels@gmail.com>
  - Joshua Peek <jegpeek@stsci.edu>
  - Sergey Koposov <skoposov@cmu.edu>