SCIENCE DATA PROCESSING WITH HIGH-END COMPUTING

JON M. JENKINS

SCIENCE PIPELINES GROUP LEAD

NASA ADVANCED SUPERCOMPUTING DIVISION

AMES RESEARCH CENTER

Ames Partnership Days

29 April 2025

Problem Statement

- Science data processing software is typically custom-built for every science mission
  - This incurs technical, cost and schedule risk that is often realized
- Science data processing software is frequently assigned lower priority than other mission elements, which introduces risk to the mission science
  - For cost-constrained missions this typically falls on a postdoc who shows an aptitude and/or interest in software
  - This often results in a brittle workflow, not a robust software project
- This is an identified weakness in the Earth Venture-class missions

Science Pipelines at NASA's Advanced Supercomputing Division

We've built and operated several major science pipelines for NASA astrophysics missions.

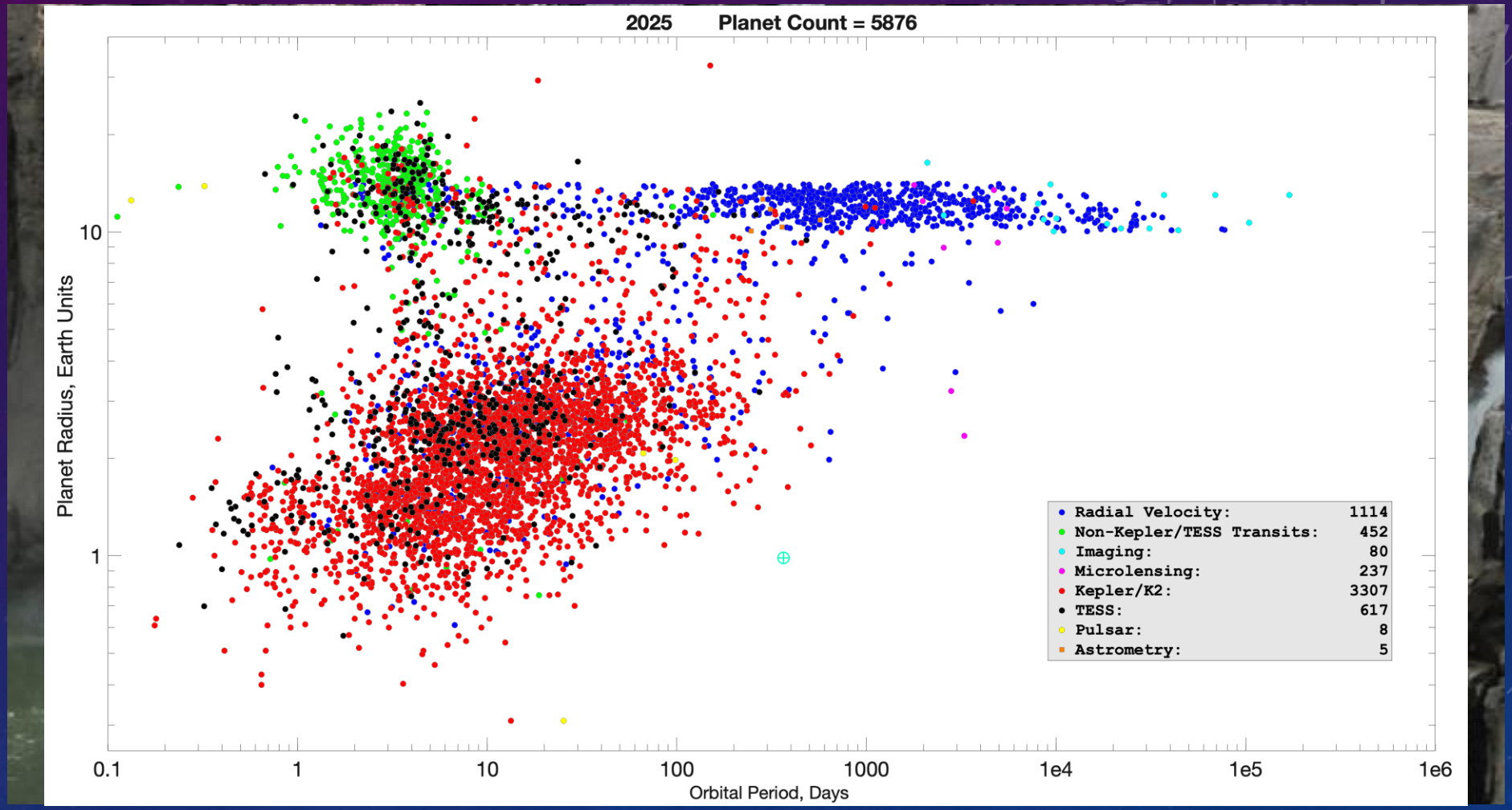


*As per NExScI data base as of 11/6/24
https://exoplanetarchive.ipac.caltech.edu/

Science Pipelines at NASA's Advanced Supercomputing Division

We've built and operated several major science pipelines for NASA astrophysics missions.

Kepler Mission:
Discovered 3307 of the 5876 known exoplanets to date



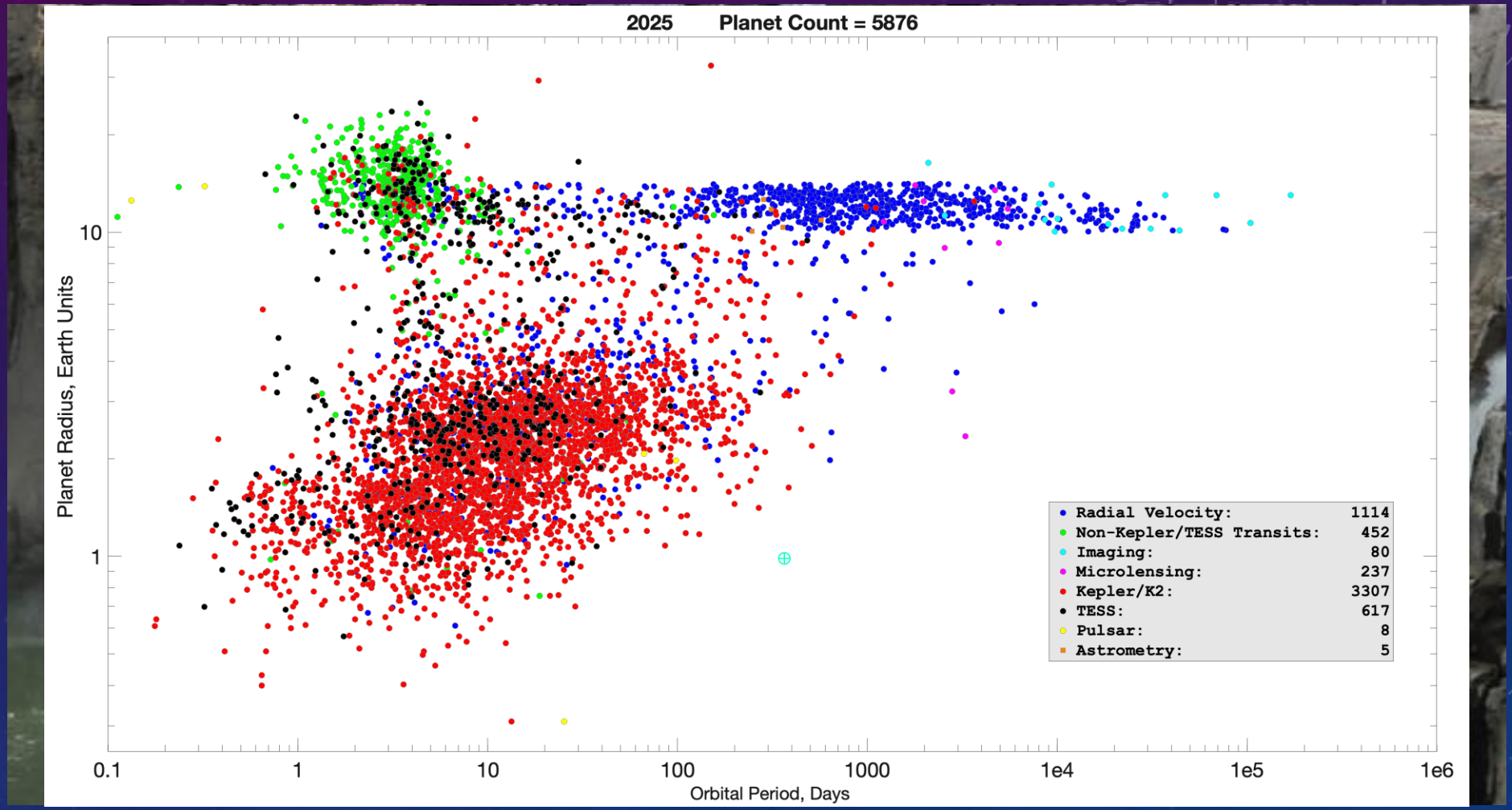
*As per NExScI data base as of 4/23/25
https://exoplanetarchive.ipac.caltech.edu/

Science Pipelines at NASA's Advanced Supercomputing Division

We've built and operated several major science pipelines for NASA astrophysics missions.

Kepler Mission:
Discovered 3307 of the 5876 known exoplanets to date

Transiting Exoplanet Survey Satellite:
617 Exoplanets
122 < 2.5 R_{earth} with measured masses



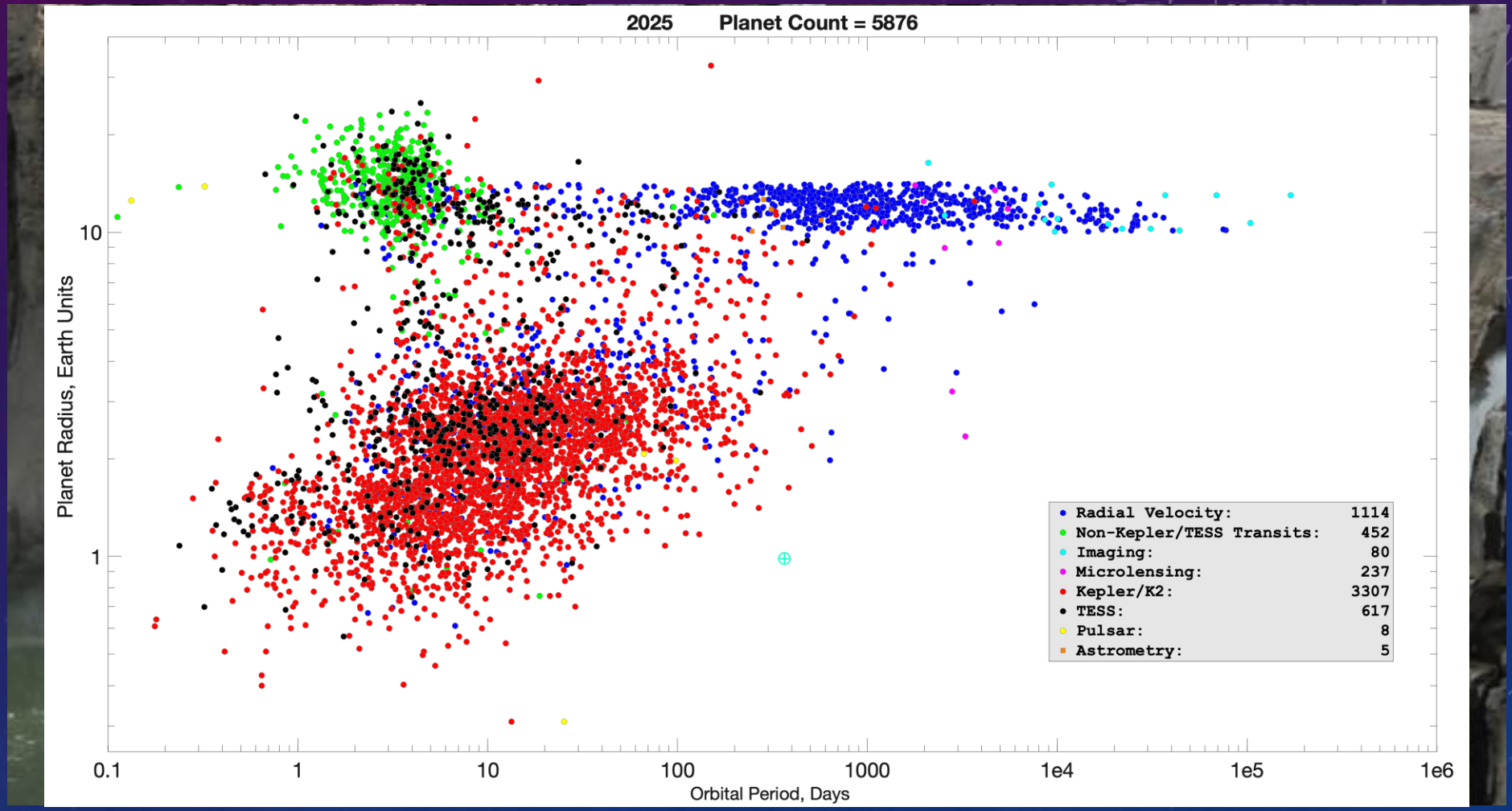
*As per NExScI data base as of 4/23/25
https://exoplanetarchive.ipac.caltech.edu/

Science Pipelines at NASA's Advanced Supercomputing Division

We've built and operated several major science pipelines for NASA astrophysics missions.

Kepler Mission:
Discovered 3307 of the 5876 known exoplanets to date

Transiting Exoplanet Survey Satellite:
617 Exoplanets
129 < 2.5 R_{earth} with measured masses

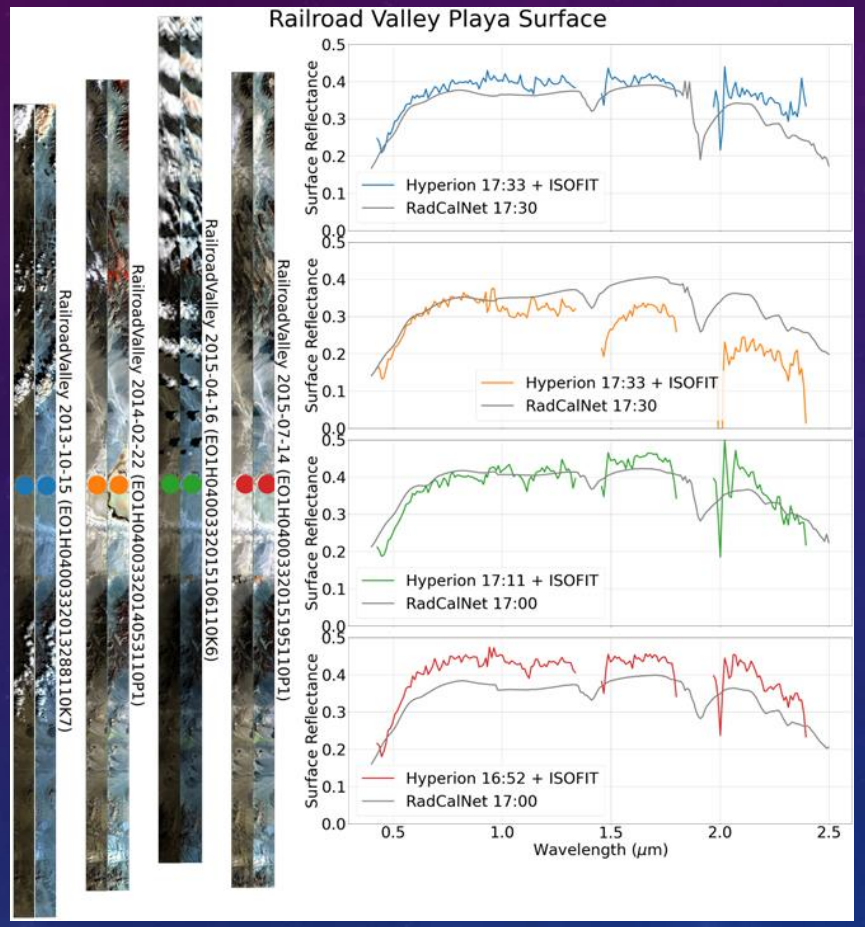


*As per NExScI data base as of 4/23/25
https://exoplanetarchive.ipac.caltech.edu/

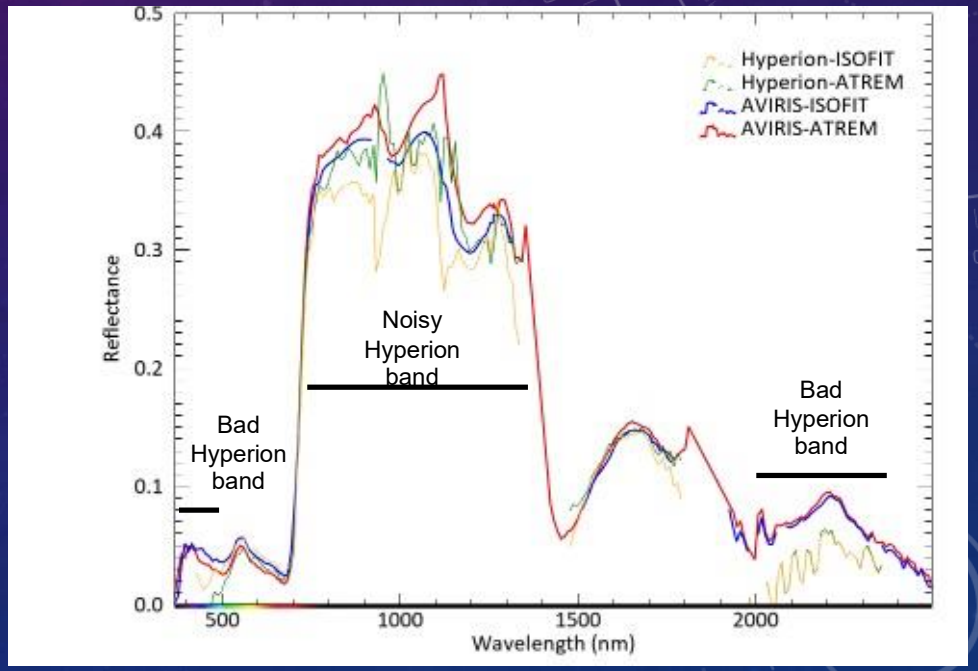
The NAS has enabled discovery of 90% of the exoplanets smaller than Neptune!

Surface Biology & Geology Pathfinder Studies

- We've resurrected the 55-TB Hyperion data set as proxy for SBG data
- SBG will collect 2.4 TB day-1 of hyperspectral image data and will produce ~40 TB day-1 of data products
- This study demonstrated the NAS could easily keep up with the processing demands for SBG



Comparison of RadCalNet measurements with Hyperion surface reflectance retrievals for scenes observed in Railroad Valley.



Comparison of surface reflectance spectra retrieved with the Hyperion and AVIRIS sensors, each with ISOFIT and ATREM retrieval algorithms. The results are shown for a vegetation site near Half Moon Bay, CA, observed on April 30, 2015.

Designing, Developing and Deploying New Science Data Systems is Expensive



Kepler Mission: SPIfY 1.0

Time to develop SDS:
78 person years, or ~$25M

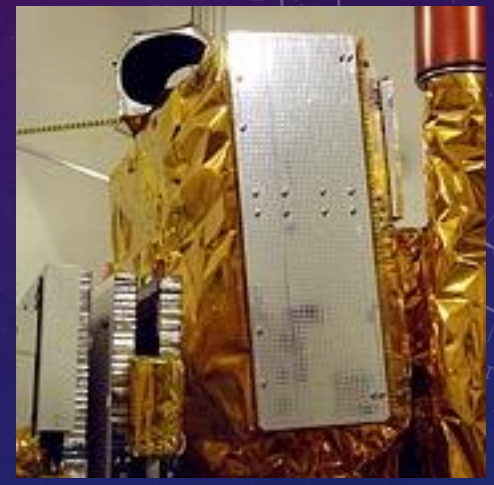
Data Rate: 1.1 GB day^{-1}
Data Volume: 1.6 TB



TESS Mission SPIfY 2.0

Time to develop SDS:
~31 person years or ~$10M

Data Rate: 13-130 GB day^{-1}
Data Volume to Date: 156 TB



Hyperion/EO-1: Ziggy 1.0

Time to develop SDS:
~3 person years or ~$1M

Data Rate: ~8.8 GB day^{-1}
Data Volume: 56 TB

Designing, Developing and Deploying New Science Data Systems is Expensive



Kepler Mission: SPIfY 1.0

Time to develop SDS:
78 person years, or ~$25M

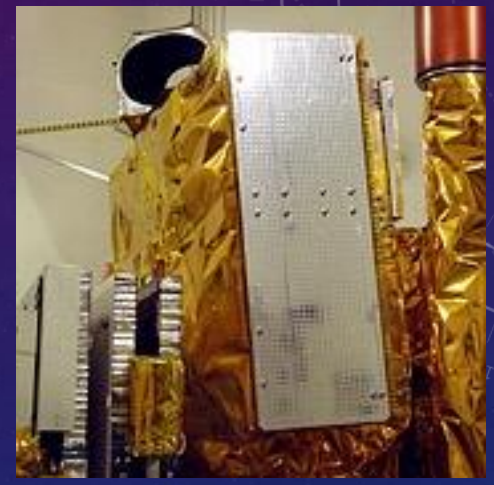
Data Rate: 1.1 GB day^{-1}
Data Volume: 1.6 TB



TESS Mission SPIfY 2.0

Time to develop SDS:
~31 person years or ~$10M

Data Rate: 13-130 GB day^{-1}
Data Volume to Date: 156 TB



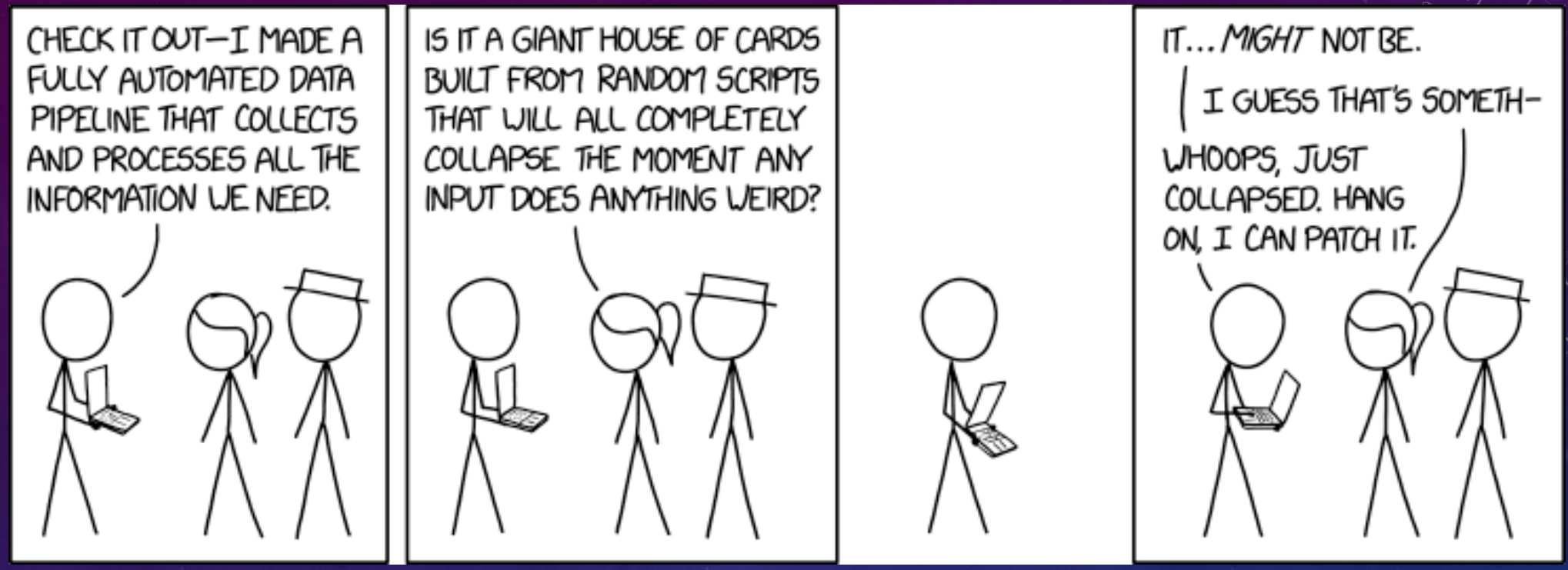
Hyperion/EO-1: Ziggy 1.0

Time to develop SDS:
~3 person years or ~$1M

Data Rate: ~8.8 GB day^{-1}
Data Volume: 56 TB

Ziggy scales easily to data rates of 5+ TB day^{-1}

xkcd.com. Used under terms of Creative Commons License 2.5.

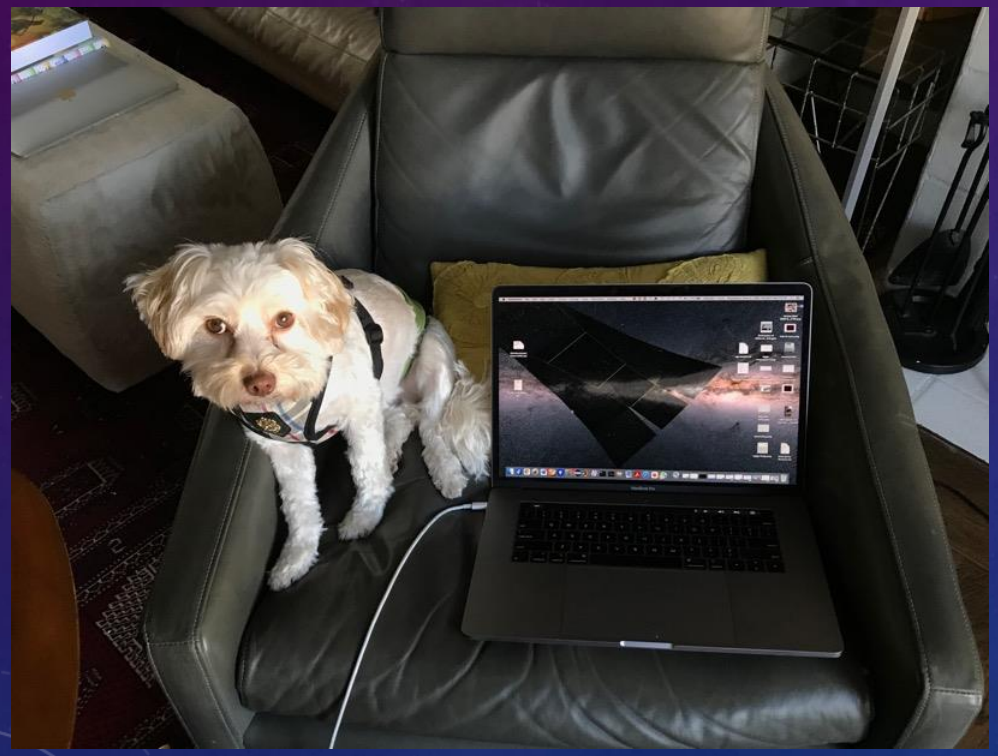


Science data pipelines need to do a lot more than science:

- Logging
- Execution flow
- Execution monitoring
- Data accountability
- Configuration management
- Error handling
- And much more!

ziggy handles all the not-science and lets the scientists get on with the science!

Develop here* …



… deploy here!



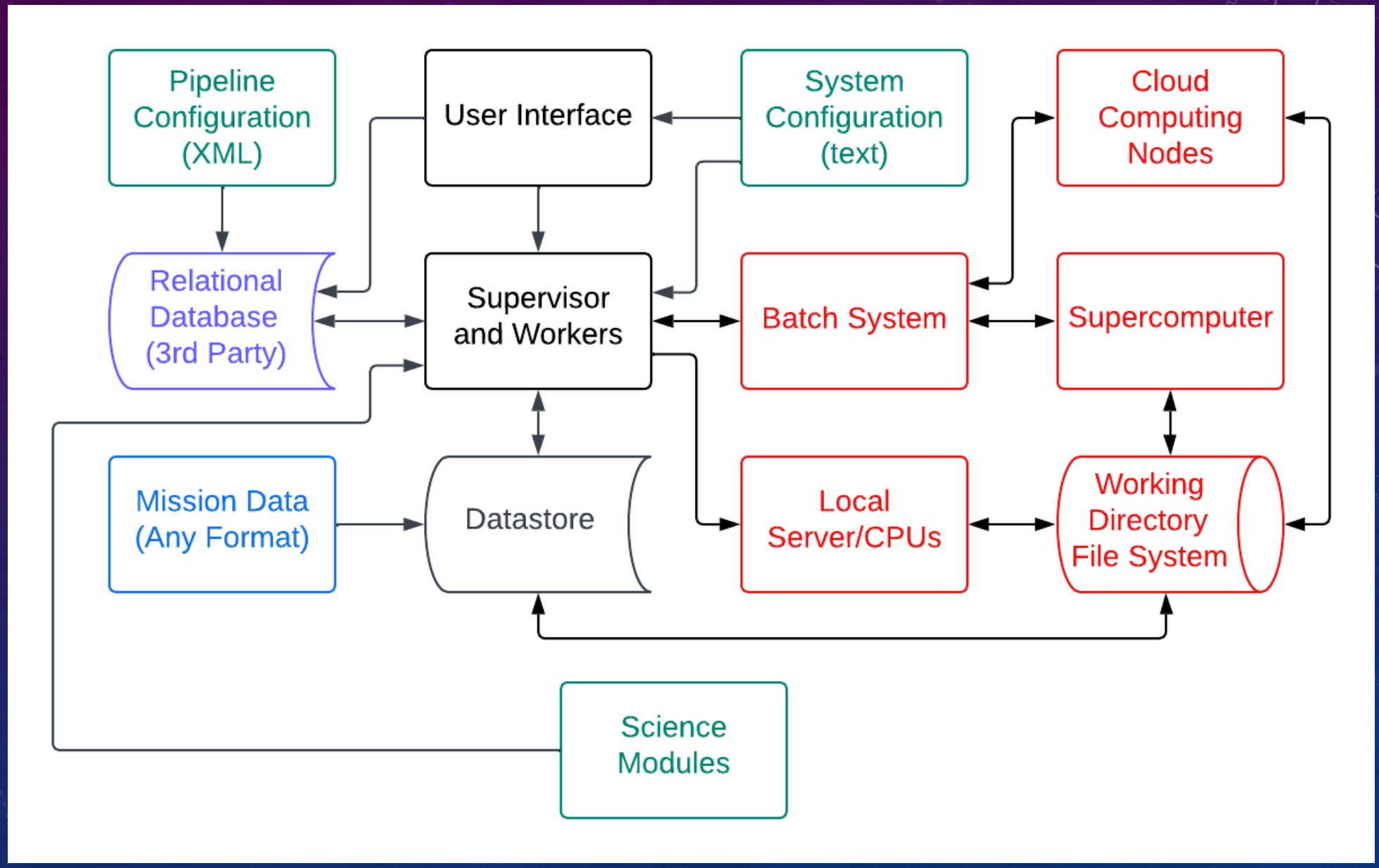
*Dog not included.

Ziggy is Lightweight

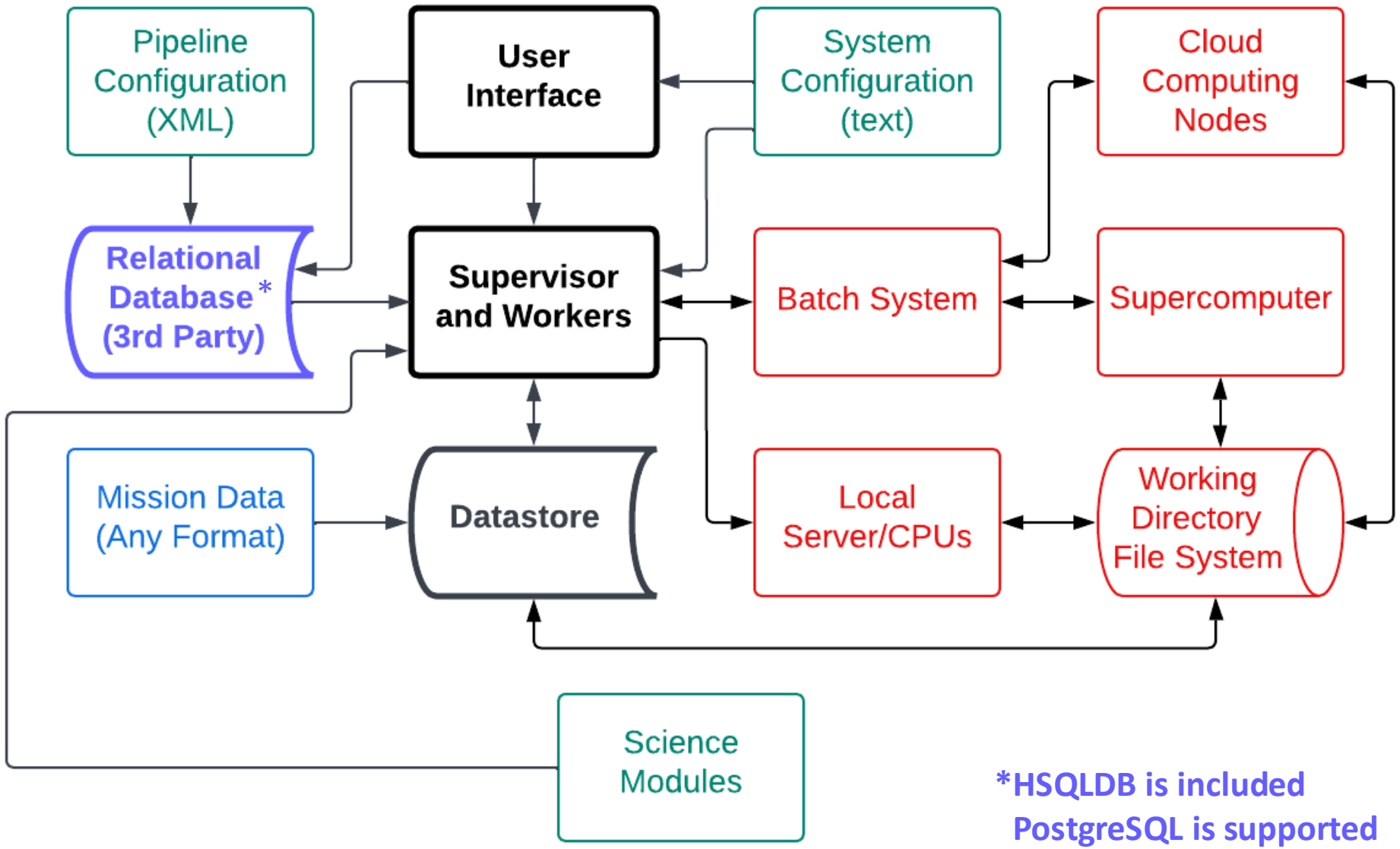
Conclusions

- Developing and deploying production pipelines for NASA flight missions can be expensive and risky
- The NAS has extensive experience assisting project teams build and run science pipelines enabling phenomenal science return
- The Science Pipelines Group can help projects develop and operate new complex science pipelines at far less risk and cost than traditional approaches
- Ziggy
  - Supports Open-Source Science & SPD-41a: released by NASA to github.com/nasa/Ziggy
  - Easy to use and ready to use: new pipelines can be set up in days
  - Ensures replicability and repeatability of science results: stores pipeline and algorithm configurations in a relational database
  - Allows for highly-automated, event-driven operations
  - TRL-8, NPR-7150-compliant as a class C software project

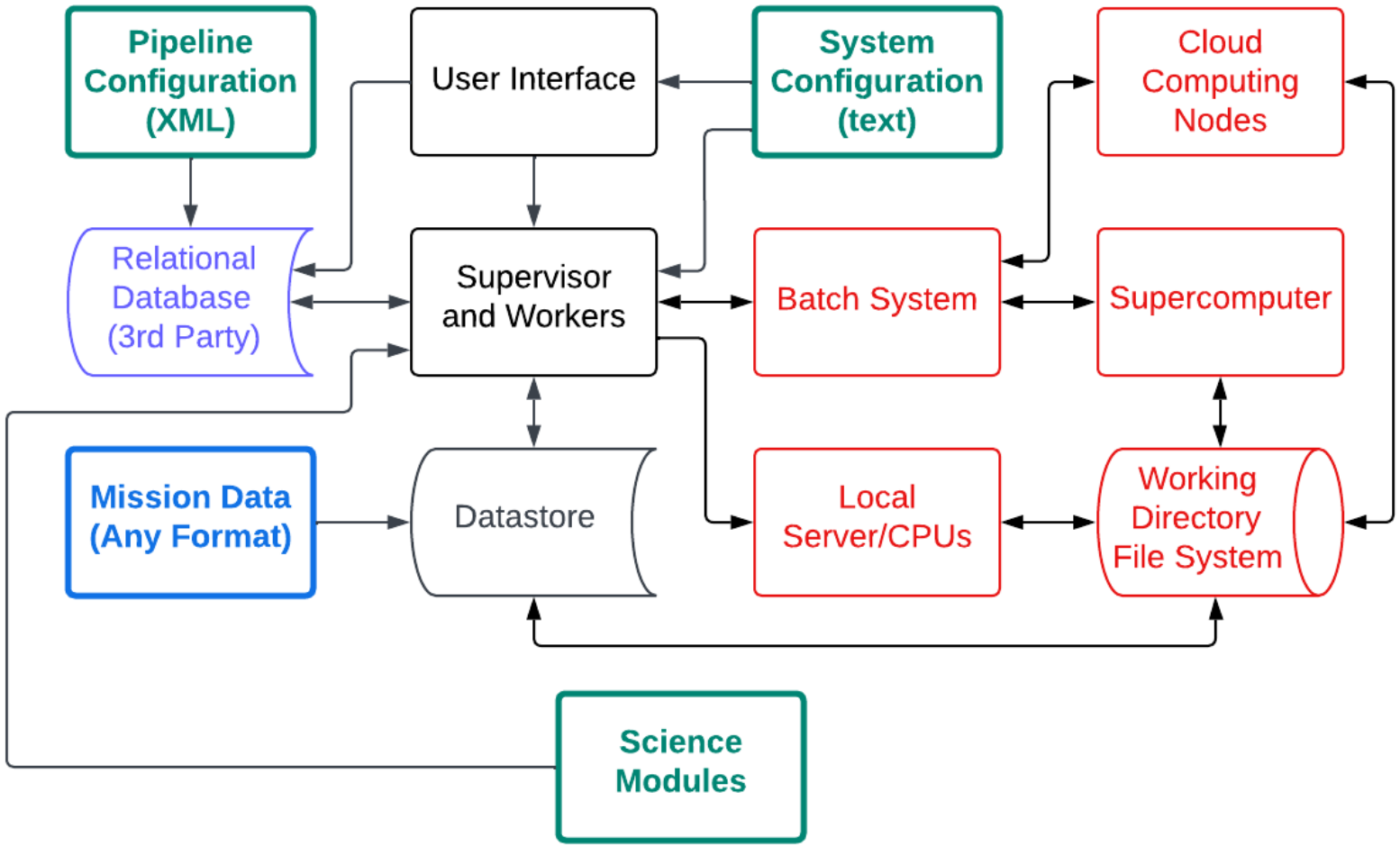
Deployment Architecture



Ziggy-Supplied Components



User-Supplied Components



Subject Matter Experts supply

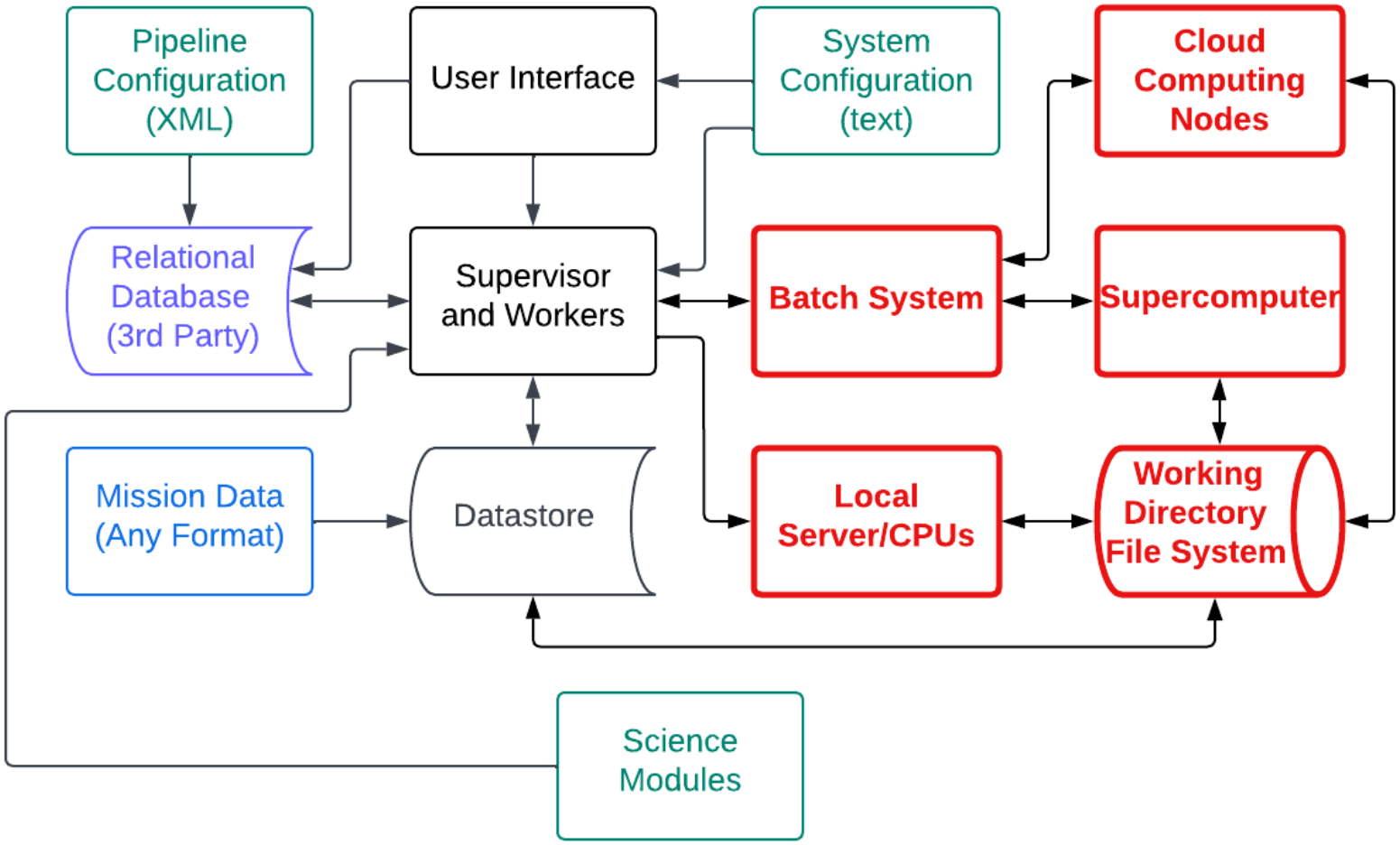
- **Algorithms**
- **Pipeline and system configuration and**
- **Mission data**

Algorithms "As-is" – no design rules that must be followed

Polyglot – "mix and match" any languages

So far we've used MATLAB, C++, Python, Shell Script – it all just works

Execution Options



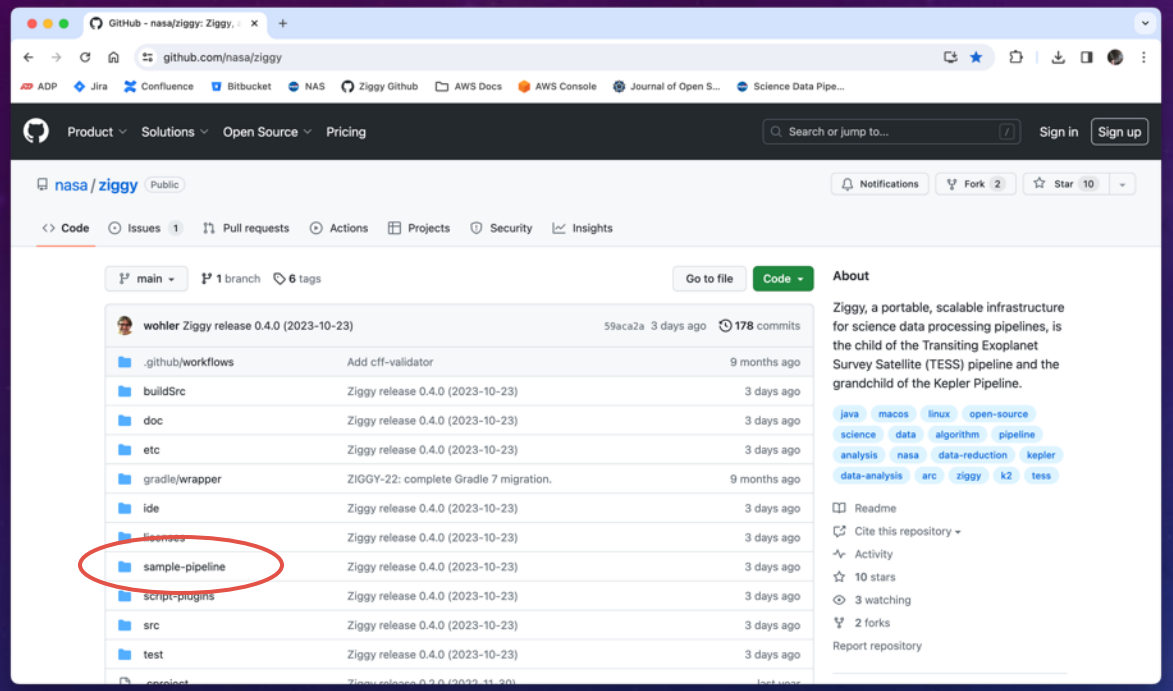
Select execution in server or in HPC / Cloud on a per-algorithm basis

Current support for NAS (PBS Batch)

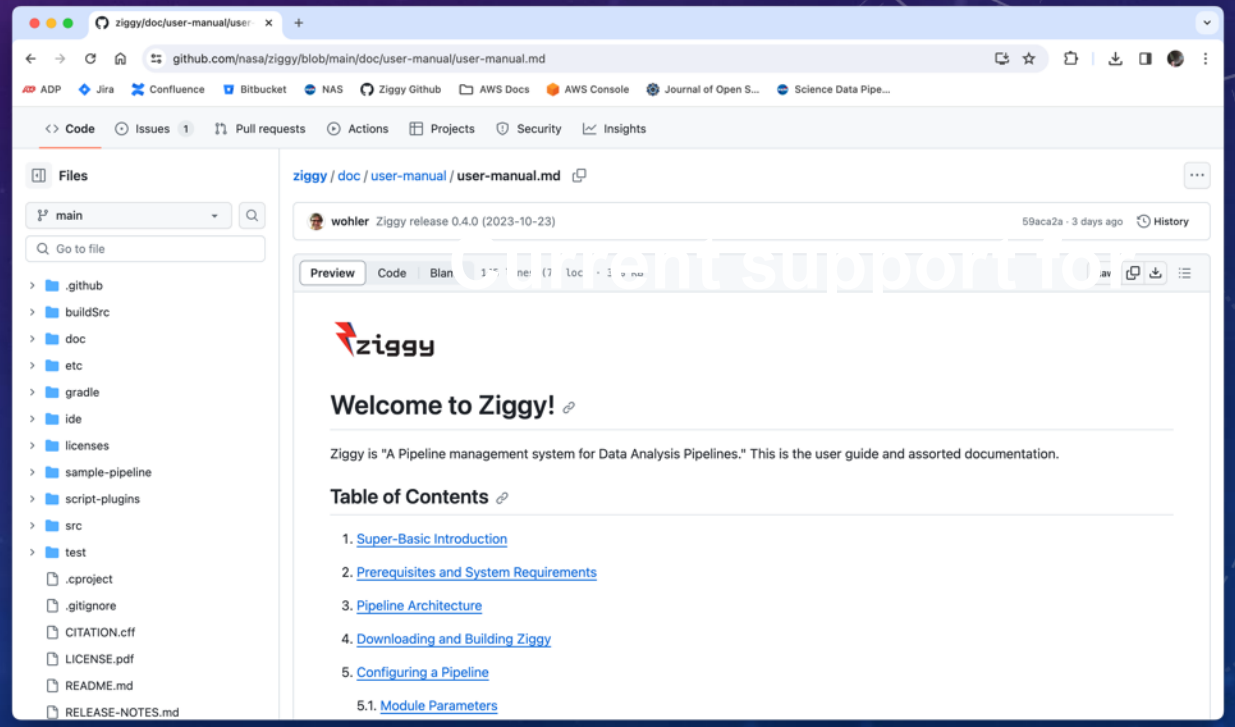
Coming soon: support for AWS / Slurm batch

Ziggy is Open Source!

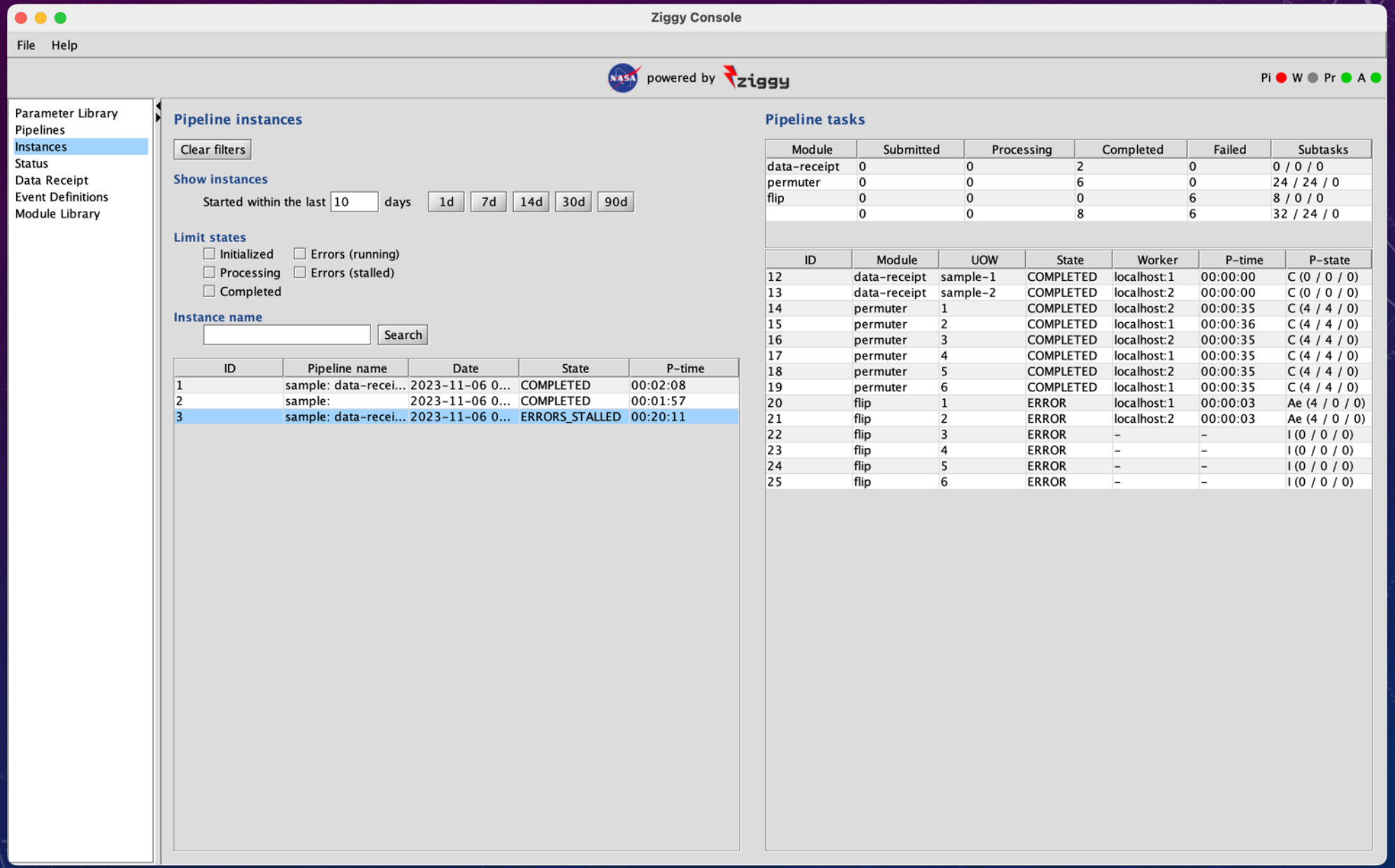
github.com/nasa/ziggy



Extensive easy-to-read documentation



Super-basic sample pipeline that shows you how to do it all


Ziggy Console
File Help
powered by ziggy
Pi W Pr A
Parameter Library
Pipelines
Instances
Status
Data Receipt
Event Definitions
Module Library
Pipeline instances
Clear filters
Show instances
Started within the last 10 days
1d 7d 14d 30d 90d
Limit states
Initialized
Errors (running)
Processing
Errors (stalled)
Completed
Instance name
Search
ID | Pipeline name | Date | State | P-time
1 | sample: data-recei... | 2023-11-06 0... | COMPLETED | 00:02:08
2 | sample: | 2023-11-06 0... | COMPLETED | 00:01:57
3 | sample: data-recei... | 2023-11-06 0... | ERRORS_STALLED | 00:20:11
Pipeline tasks
Module | Submitted | Processing | Completed | Failed | Subtasks
data-receipt | 0 | 0 | 2 | 0 | 0 / 0 / 0
permuter | 0 | 0 | 6 | 0 | 24 / 24 / 0
flip | 0 | 0 | 0 | 6 | 8 / 0 / 0
| 0 | 0 | 8 | 6 | 32 / 24 / 0
ID | Module | UOW | State | Worker | P-time | P-state
12 | data-receipt | sample-1 | COMPLETED | localhost:1 | 00:00:00 | C (0 / 0 / 0)
13 | data-receipt | sample-2 | COMPLETED | localhost:2 | 00:00:00 | C (0 / 0 / 0)
14 | permuter | 1 | COMPLETED | localhost:2 | 00:00:35 | C (4 / 4 / 0)
15 | permuter | 2 | COMPLETED | localhost:1 | 00:00:36 | C (4 / 4 / 0)
16 | permuter | 3 | COMPLETED | localhost:2 | 00:00:35 | C (4 / 4 / 0)
17 | permuter | 4 | COMPLETED | localhost:1 | 00:00:35 | C (4 / 4 / 0)
18 | permuter | 5 | COMPLETED | localhost:2 | 00:00:35 | C (4 / 4 / 0)
19 | permuter | 6 | COMPLETED | localhost:1 | 00:00:35 | C (4 / 4 / 0)
20 | flip | 1 | ERROR | localhost:1 | 00:00:03 | Ae (4 / 0 / 0)
21 | flip | 2 | ERROR | localhost:2 | 00:00:03 | Ae (4 / 0 / 0)
22 | flip | 3 | ERROR | – | – | I (0 / 0 / 0)
23 | flip | 4 | ERROR | – | – | I (0 / 0 / 0)
24 | flip | 5 | ERROR | – | – | I (0 / 0 / 0)
25 | flip | 6 | ERROR | – | – | I (0 / 0 / 0)