LINCC Frameworks

Scalable analysis across large datasets

Neven Caplar - Rubin Observatory/University Of Washington

LINCC

VERA C. RUBIN OBSERVATORY

LSST Discovery Alliance

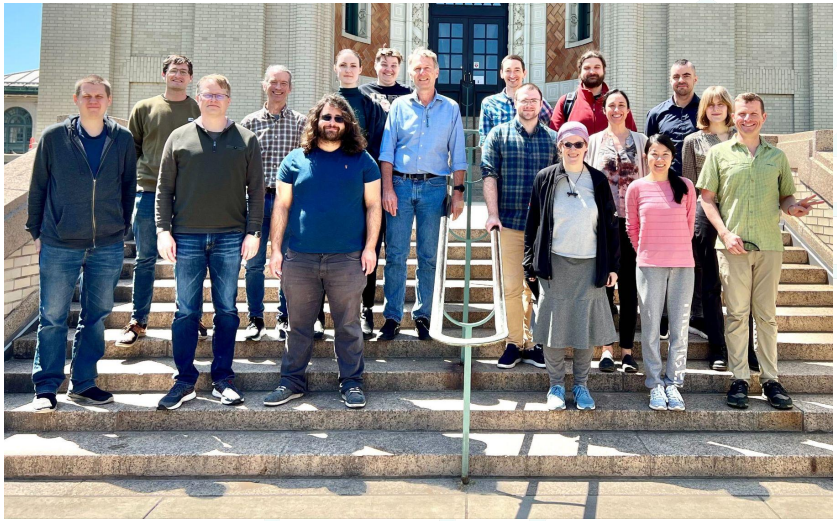
Schmidt Sciences

The LINCC Frameworks Project

LSST Interdisciplinary Network For Collaboration And Computing

- A collaboration between University of Washington, Carnegie-Mellon University, LSST Discovery Alliance, University of Pittsburgh, and NOIRLab [NSF's National Optical-Infrared Astronomy Research Laboratory] to build software systems for key LSST[Large Survey of Space and Time] science

- **Science** software infrastructure: combining user algorithms & code, astro packages, and industry tools to build scalable science analysis packages



Incubators

- Incubators provide support for researchers to work directly with LINCC Frameworks team to apply their new tools to research problems.

- Goal: Establish long-term software development collaborations that serve both the selected teams and LINCC Frameworks.

- Next proposal deadline is June 17, 2024
  - Selection to be announced mid July
  - 2 stage proposal process
  - Duration of the project - 2024 September through November (flexible)

Incubator example: Solar system simulation

Sorcha

PI: Meg Schwamb (Queen's University Belfast)

Goal of the incubator: **Make it fast and efficient!** Generate DP0.3 [Data Preview 0.3] simulated dataset in less than a day rather than the weeks it took

Original implementation:

- Too slow for widespread use (~3 weeks to run one one full scale Solar System simulation - 16 million simulated objects run on ~500-1000 cores)
- Multi-step read in and out of files
- Ephemeris generator (orbit to sky positions) generating terabytes of temporary files
- Clunky two step process with two software packages

Incubator example: Solar system simulation

PI: Meg Schwamb (Queen's University Belfast)

Goal of the incubator: **Make it fast and efficient!** Generate DP0.3 [Data Preview 0.3] simulated dataset in less than a day rather than the weeks it took

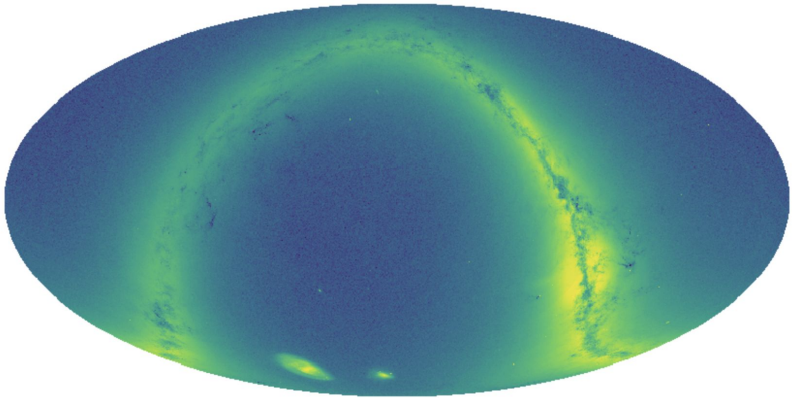
Results of the incubator:

- Continuous integration, automatic documentation generation, and PyPI distribution.
- Expanded the code to support pluggable light curve and cometary activity magnitude modifications contributed by users.
- Eliminated a file IO bottleneck and integrated new super fast ephemeris generation algorithm into the code base.
- Integrated a more efficient detection linking algorithm.
- Reduced runtime for DP0.3 with 100 cores from ~3 weeks to ~5 hours **(100-fold speed up)**

Scalable cross matching and analytics - LSDB

- Data is stored in a hierarchical data storage scheme, where the sky is split into HEALPix tiles until each tile has roughly a similar number of objects (rows).
- These tiles are stored as Parquet files within a directory tree that encodes their location on the sky.
- https://github.com/astronomy-commons/lsdb

Enables: fast spatial lookup, distributed analytics, distributed joining and cross-matching. Based on 10+ yrs of thinking/experience/experimentation.

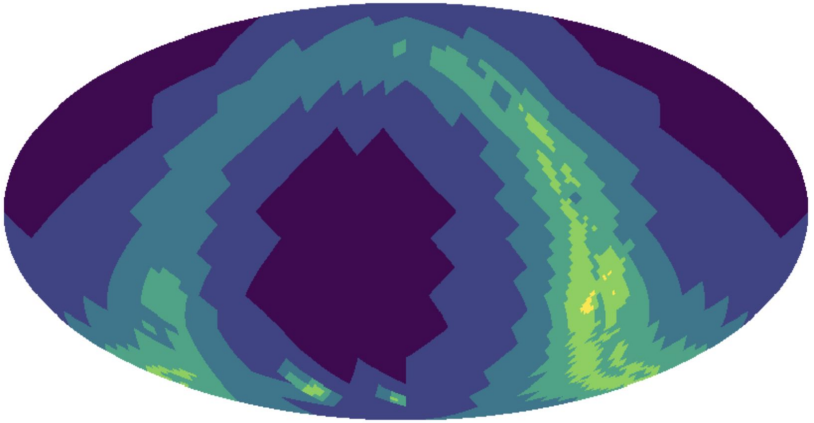


```
Norder=0/Dir=0/Npix=0/catalog.parquet
…
Norder=1/Dir=0/Npix=28/catalog.parquet
Norder=1/Dir=0/Npix=29/catalog.parquet
Norder=1/Dir=0/Npix=30/catalog.parquet
Norder=2/Dir=0/Npix=112/catalog.parquet
Norder=2/Dir=0/Npix=113/catalog.parquet
Norder=2/Dir=0/Npix=114/catalog.parquet
Norder=2/Dir=0/Npix=115/catalog.parquet
…
Norder=0/Dir=0/Npix=11/catalog.parquet
```

Scalable cross matching and analytics - LSDB

- Data is stored in a hierarchical data storage scheme, where the sky is split into HEALPix tiles until each tile has roughly a similar number of objects (rows).
- These tiles are stored as Parquet files within a directory tree that encodes their location on the sky.
- https://github.com/astronomy-commons/lsdb

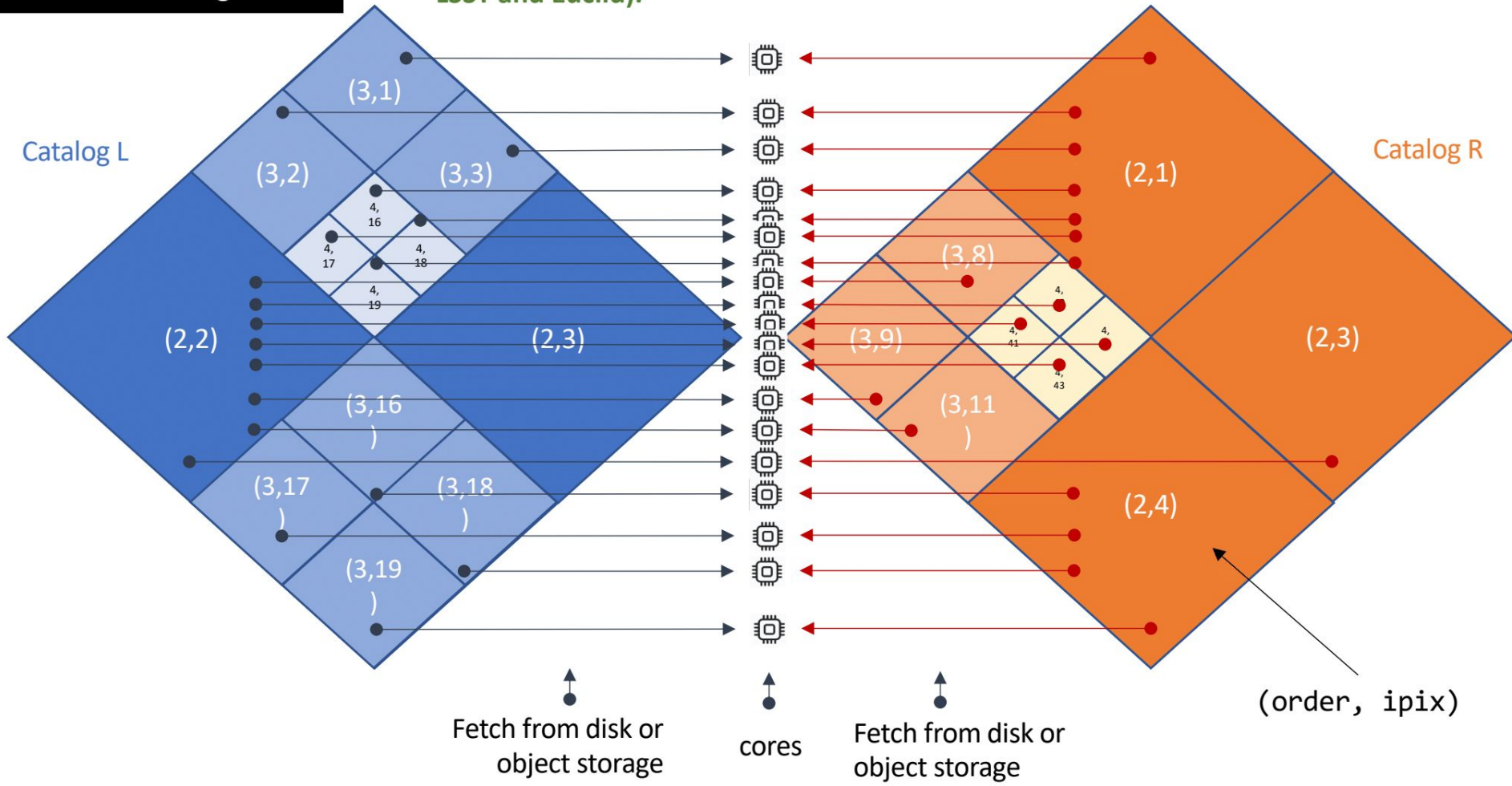
Enables: fast spatial lookup, distributed analytics, distributed joining and cross-matching. Based on 10+ yrs of thinking/experience/experimentation.



```
Norder=0/Dir=0/Npix=0/catalog.parquet
…
Norder=1/Dir=0/Npix=28/catalog.parquet
Norder=1/Dir=0/Npix=29/catalog.parquet
Norder=1/Dir=0/Npix=30/catalog.parquet
Norder=2/Dir=0/Npix=112/catalog.parquet
Norder=2/Dir=0/Npix=113/catalog.parquet
Norder=2/Dir=0/Npix=114/catalog.parquet
Norder=2/Dir=0/Npix=115/catalog.parquet
…
Norder=0/Dir=0/Npix=11/catalog.parquet
```

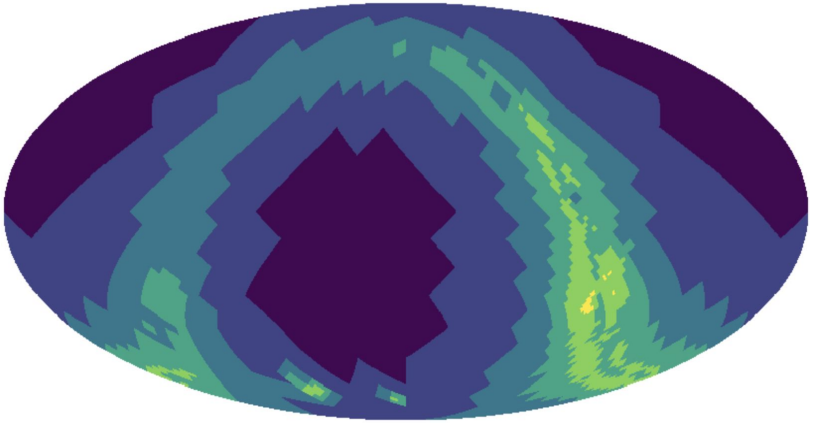
Efficient, parallel, joins and crossmatching

Use case #4: distributed analysis on data from two catalogs (example: LSST and Euclid).



Scalable cross matching and analytics - LSDB

- Collaborating with NASA to provide datasets in this spatial sharded format
- Currently implementing on Fornax - NASA science platform in development to enable cloud-computing resources to community
- Tested with various datasets (DELVE, S-Plus survey, TRILEGAL simulations, Zubercal, ZTF, PanSTARRS, GAIA, Neowise) at multiple clusters
- Currently at IVOA meeting being proposed as standard.

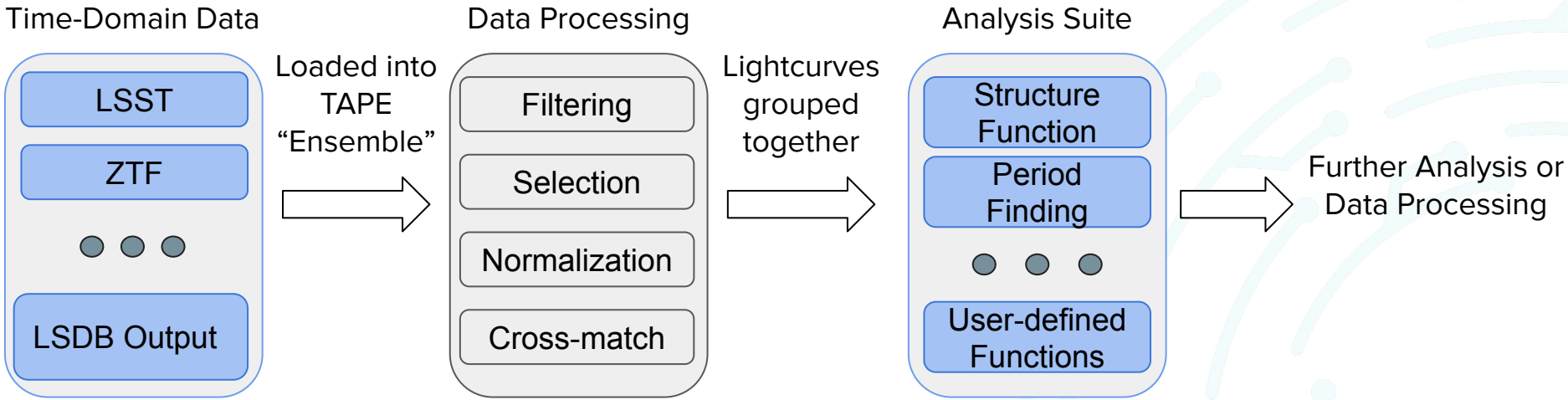


```
Norder=0/Dir=0/Npix=0/catalog.parquet
…
Norder=1/Dir=0/Npix=28/catalog.parquet
Norder=1/Dir=0/Npix=29/catalog.parquet
Norder=1/Dir=0/Npix=30/catalog.parquet
Norder=2/Dir=0/Npix=112/catalog.parquet
Norder=2/Dir=0/Npix=113/catalog.parquet
Norder=2/Dir=0/Npix=114/catalog.parquet
Norder=2/Dir=0/Npix=115/catalog.parquet
…
Norder=0/Dir=0/Npix=11/catalog.parquet
```

Time domain - TAPE

Details: https://tape.readthedocs.io/en/latest/

- **TAPE**: Package for performing scalable analysis of time-domain astronomy data
- Built on top of **Dask**, which provides a parallelized dataframe object and enables larger-than-memory computation on single or multiple machines.



Nested pandas/dask

Nested-Pandas is motivated by time-domain use cases, where we see two levels of information, about astronomical objects and then an associated set of N measurements of those objects.

Core advantages being:

- hierarchical column access
- efficient packing of nested informatio into inputs to custom user functions
- avoiding costly groupby operations

Load in Parquet Data

In [8]:
```
%%time
#Read in parquet data
nf = read_parquet(
data="objects.parquet",
to_pack={"ztf_sources": "ztf_sources.parquet", "ps1_sources": "ps1_sources.parquet"},
)
nf
```

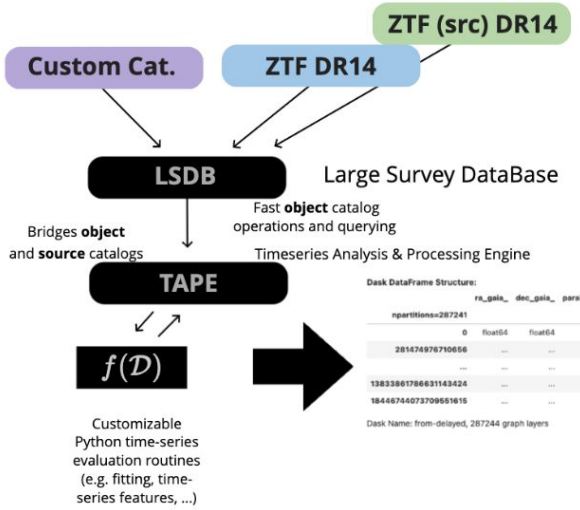
```
CPU times: user 144 ms, sys: 21.4 ms, total: 166 ms
Wall time: 153 ms
```

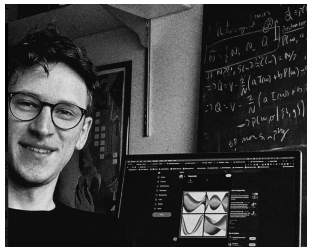
Out[8]:

	ra	dec	ztf_sources	ps1_sources
0	17.447868	35.547046	mjd flux band 0 8.420511...	mjd flux band 0 0.091356...
1	1.020437	4.353613	mjd flux band 0 14.143429...	mjd flux band 0 12.475696...
2	3.695975	31.130105	mjd flux band 0 7.190259...	mjd flux band 0 13.717712...
3	13.242558	6.099142	mjd flux band 0 1.708140...	mjd flux band 0 16.759764...
4	2.744142	48.444456	mjd flux band 0 18.837824...	mjd flux band 0 18.139101...
...	...	...	...	...
995	6.547263	40.249140	mjd flux band 0 4.055585...	mjd flux band 0 5.474614...
996	18.391919	17.643616	mjd flux band 0 10.358167...	mjd flux band 0 11.889307...
997	18.587638	46.568135	mjd flux band 0 3.871603...	mjd flux band 0 16.421570...
998	10.871655	6.719466	mjd flux band 0 0.886458...	mjd flux band 0 14.044775...
999	15.466982	13.620714	mjd flux band 0 15.703350...	mjd flux band 0 3.585283...

1000 rows × 4 columns

Example - Finding dipping stars

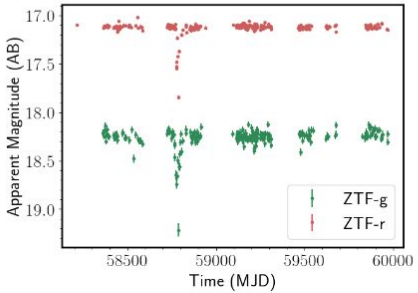


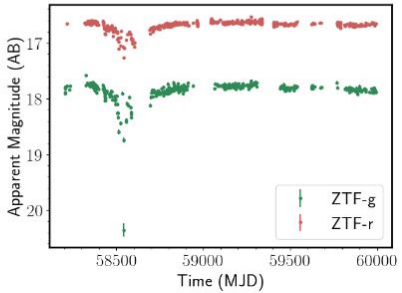


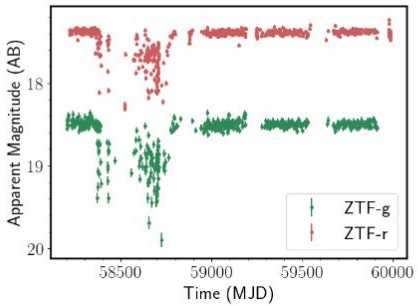
Slides by Andy Tzanidakis, to find dipping Main Sequence stars

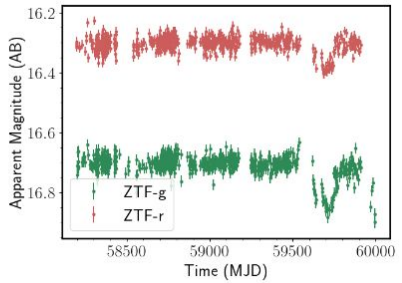
Preliminary Results

Tzanidakis et al. (2024; in-prep)



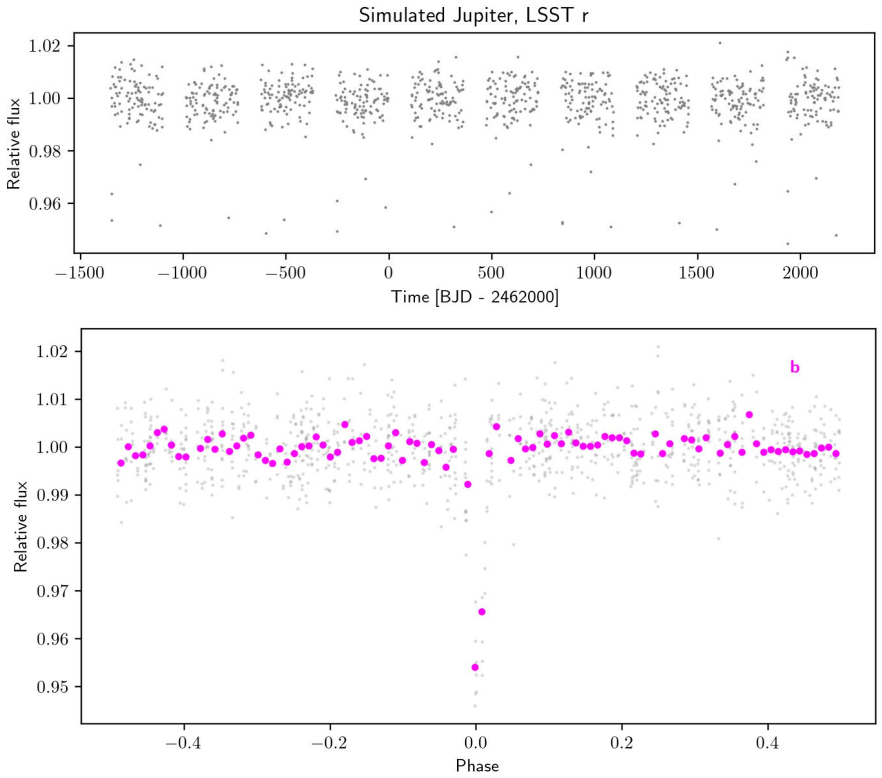




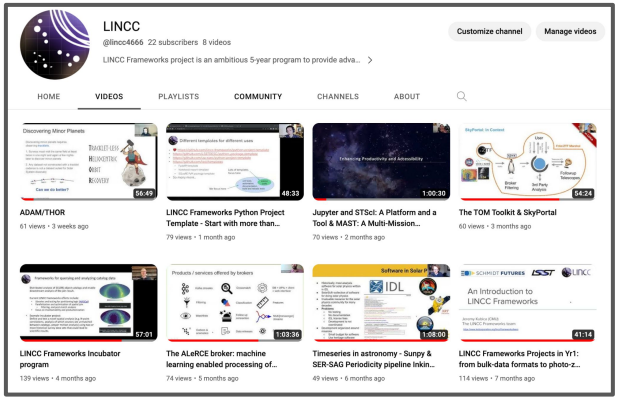


Example - Finding stars with dips

- High-cadence photometric data (e.g., from Kepler, TESS) are typically searched for periodic transits using the computationally-demanding Box Least Squares (BLS).
- The number of searchable light curves in Rubin will 1,000 times larger than for TESS.
- Led by Tansu Daylan (Wash. U.)



LINCC Frameworks Tech Talks



- Talks that showcase the work done by the broad Rubin software and archives community.
- So far: Brokers, in-kind contributors, data centers, LINCC Frameworks software engineers…
- Future: Roman software group & LSST data management
- June 13 - Even more about LSDB/TAPE

2nd Thursday of the month, 10 am Pacific at https://ls.st/lincc-talks

Demo Tomorrow

- **4pm, Wednesday, KIVA room**
  - At the start of TDA/MMA breakout session
  - If you wish to run the demo at the same time as us we have two options
    - On your local machine
    - On our cloud [Recommended]
  - **Before the demo** - follow the Getting Started Instructions:
    - github.com/lincc-frameworks/Rare_Gems_Demo
    - Especially if running on our cloud; we need you to fill a form and to be approved and you should do that by the end of the day today!
  - Join **#lsdb_tape_tutorial** in conference slack channel OR **#lincc-lsdb** in LSSTC slack channel for questions/comments/help

Summary

- **LINCC**
  - We are working on enabling scientist to use big data
  - Incubators - we can help you scale up your code
  - LSDB/TAPE; new way to shard data, to crossmatch and to do large-scale analytics
  - Join of for the demo tomorrow!
  - Follow Getting Started instructions at:
    - https://github.com/lincc-frameworks/Rare_Gems_Demo