Rubin Observatory

The Software Behind the Science

NSF AURA U.S. DEPARTMENT OF ENERGY SLAC CHARLES AND LISA SIMONYI FUND FOR ARTS AND SCIENCES LSST CORPORATION

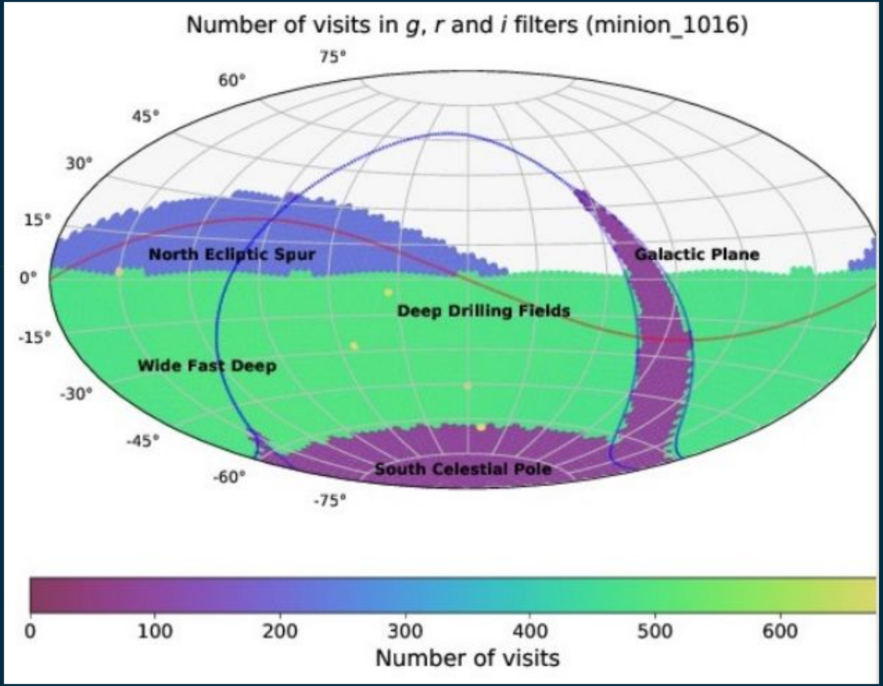
Rubin Observatory

What is Rubin Obs?

Rubin Observatory

A facility that will conduct an optical/near-IR survey of half the sky over a 10 year period. This survey is called the legacy survey of space and time (LSST).

- 90% of the time spent on uniform survey, with 800-1000 observations a night with most observations repeated 20-30 min apart
- ~10 million transient events a night
- 3.2 GigaPixel Camera
- Will generate 100PB of data over the course of the survey, with catalogs for over 40 billion objects



U.S. DEPARTMENT OF ENERGY Office of Science

Rubin Observatory

Who are we?

The data management team began in 2008 with a handful of members and has grown to a group of ~100 developers split over 5 institutions today

PRINCETON UNIVERSITY

NCSA

UNIVERSITY *of* WASHINGTON

SLAC

NOAO

Data processing

Rubin Observatory

- Nightly processing of transients, within 60s of every image taken on the telescope
- Yearly processing of all data to date

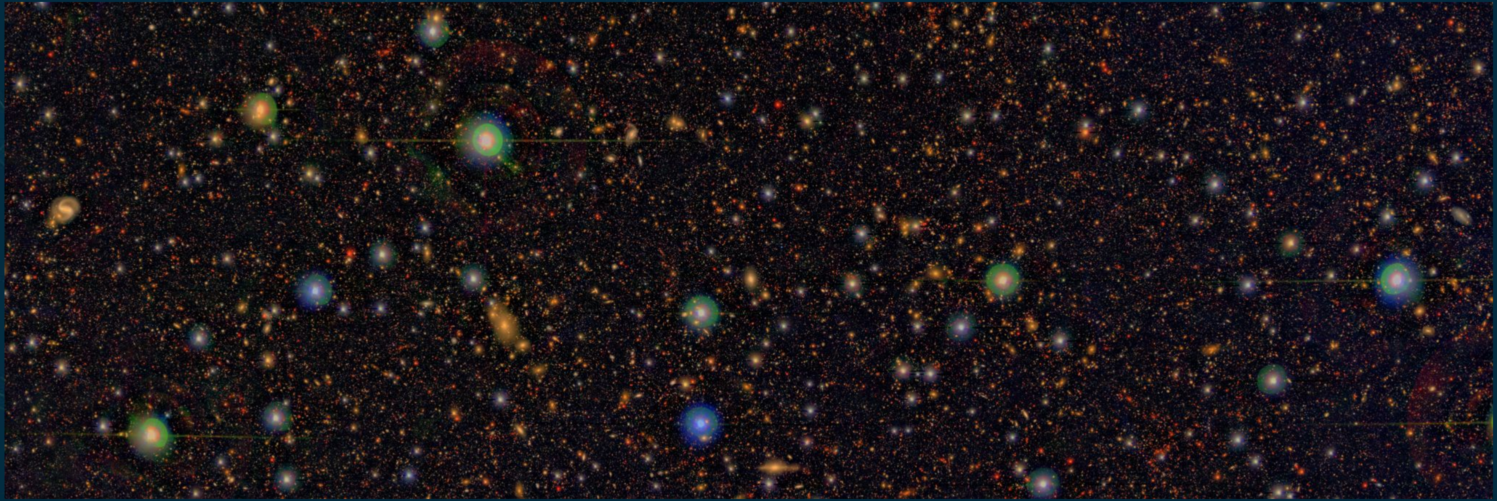
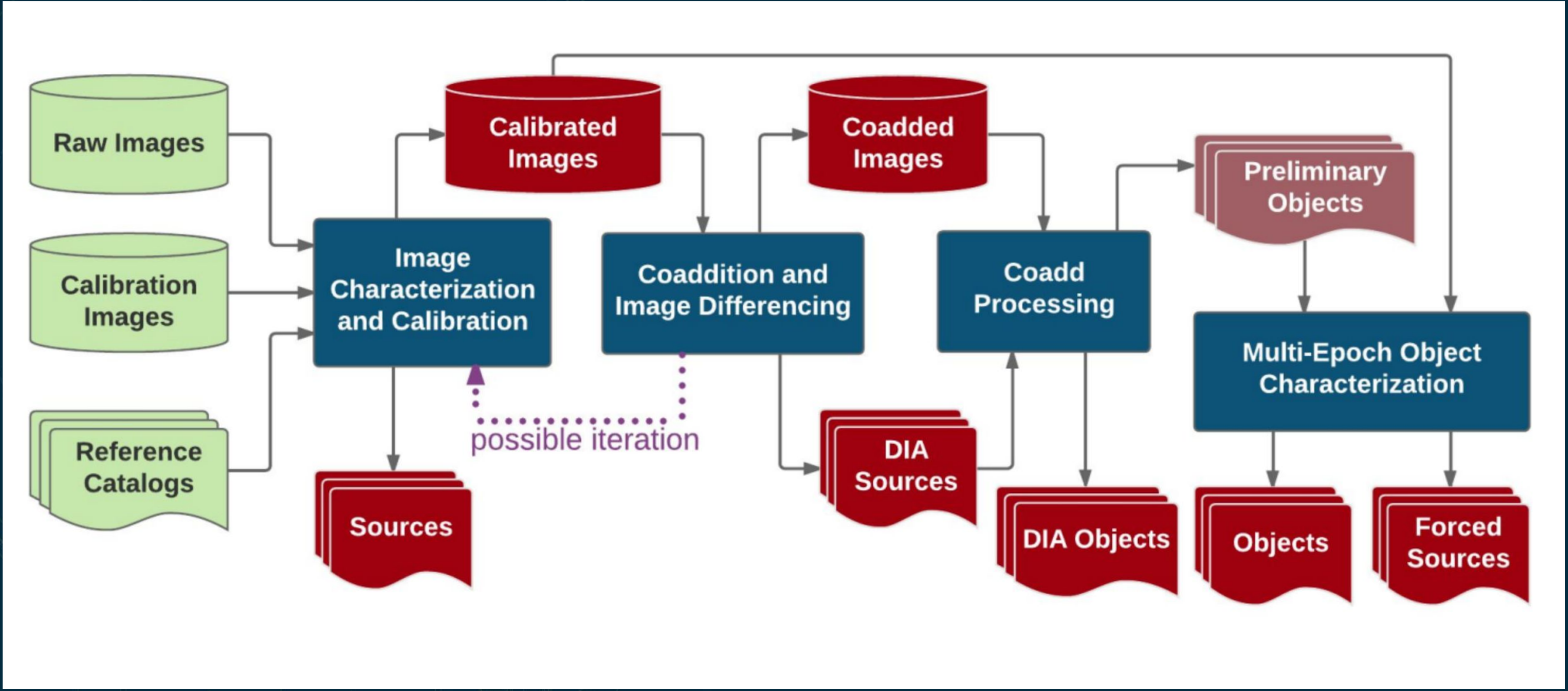


Image processed with LSST software using data from HSC

From pixels to catalogs

Rubin
Observatory



Science Platform

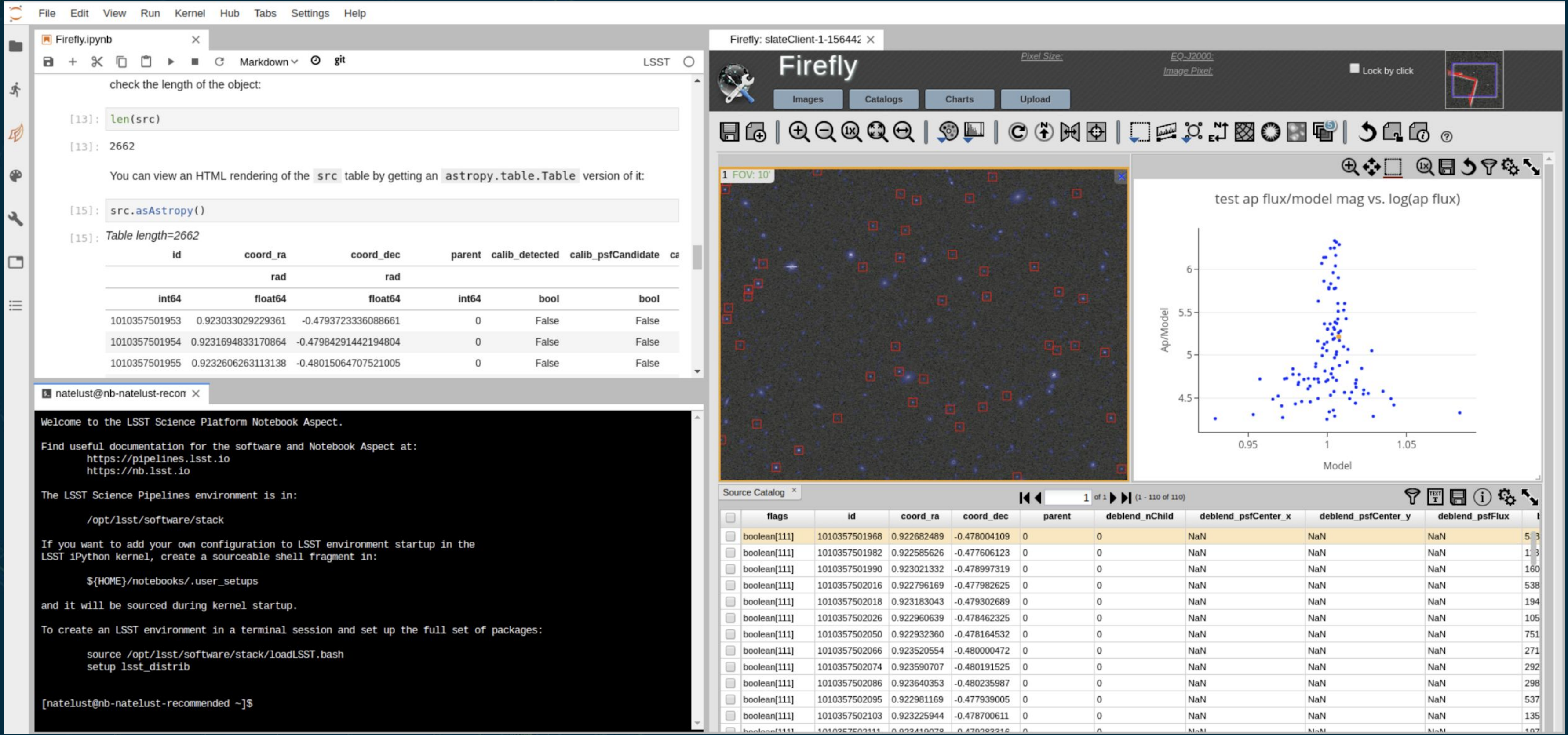
Rubin Observatory

- Too much data to take to scientists, so bring scientists to data
- Allows analysis with same tools used to product the data
- Uses Jupyter lab
- Cloud based
- Per user containerized environments with persistent storage
- Choose from multiple versions of the software stack pre-setup in the environment
- Data visualization tools built in

U.S. DEPARTMENT OF ENERGY | Office of Science

Science Platform

Rubin Observatory



check the length of the object:

```
[13]: len(src)
[13]: 2662
```

You can view an HTML rendering of the `src` table by getting an `astropy.table.Table` version of it:

```
[15]: src.asAstropy()
```

[15]: *Table length=2662*

id	coord_ra	coord_dec	parent	calib_detected	calib_psfCandidate	ca
	rad	rad				
int64	float64	float64	int64	bool	bool	
1010357501953	0.923033029229361	-0.4793723336088661	0	False	False	
1010357501954	0.9231694833170864	-0.47984291442194804	0	False	False	
1010357501955	0.9232606263113138	-0.48015064707521005	0	False	False	

```
Welcome to the LSST Science Platform Notebook Aspect.

Find useful documentation for the software and Notebook Aspect at:
        https://pipelines.lsst.io
        https://nb.lsst.io

The LSST Science Pipelines environment is in:

        /opt/lsst/software/stack

If you want to add your own configuration to LSST environment startup in the
LSST iPython kernel, create a sourceable shell fragment in:

        ${HOME}/notebooks/.user_setups

and it will be sourced during kernel startup.

To create an LSST environment in a terminal session and set up the full set of packages:

        source /opt/lsst/software/stack/loadLSST.bash
        setup lsst_distrib


[natelust@nb-natelust-recommended ~]$
```

flags	id	coord_ra	coord_dec	parent	deblend_nChild	deblend_psfCenter_x	deblend_psfCenter_y	deblend_psfFlux	
boolean[111]	1010357501968	0.922682489	-0.478004109	0	0	NaN	NaN	NaN	[illegible]
boolean[111]	1010357501982	0.922585626	-0.477606123	0	0	NaN	NaN	NaN	[illegible]
boolean[111]	1010357501990	0.923021332	-0.478997319	0	0	NaN	NaN	NaN	160
boolean[111]	1010357502016	0.922796169	-0.477982625	0	0	NaN	NaN	NaN	538
boolean[111]	1010357502018	0.923183043	-0.479302689	0	0	NaN	NaN	NaN	194
boolean[111]	1010357502026	0.922960639	-0.478462325	0	0	NaN	NaN	NaN	105
boolean[111]	1010357502050	0.922932360	-0.478164532	0	0	NaN	NaN	NaN	751
boolean[111]	1010357502066	0.923520554	-0.480000472	0	0	NaN	NaN	NaN	271
boolean[111]	1010357502074	0.923590707	-0.480191525	0	0	NaN	NaN	NaN	292
boolean[111]	1010357502086	0.923640353	-0.480235987	0	0	NaN	NaN	NaN	298
boolean[111]	1010357502095	0.922981169	-0.477939005	0	0	NaN	NaN	NaN	537
boolean[111]	1010357502103	0.923225944	-0.478700611	0	0	NaN	NaN	NaN	135

U.S. DEPARTMENT OF ENERGY | Office of Science

Python as a language

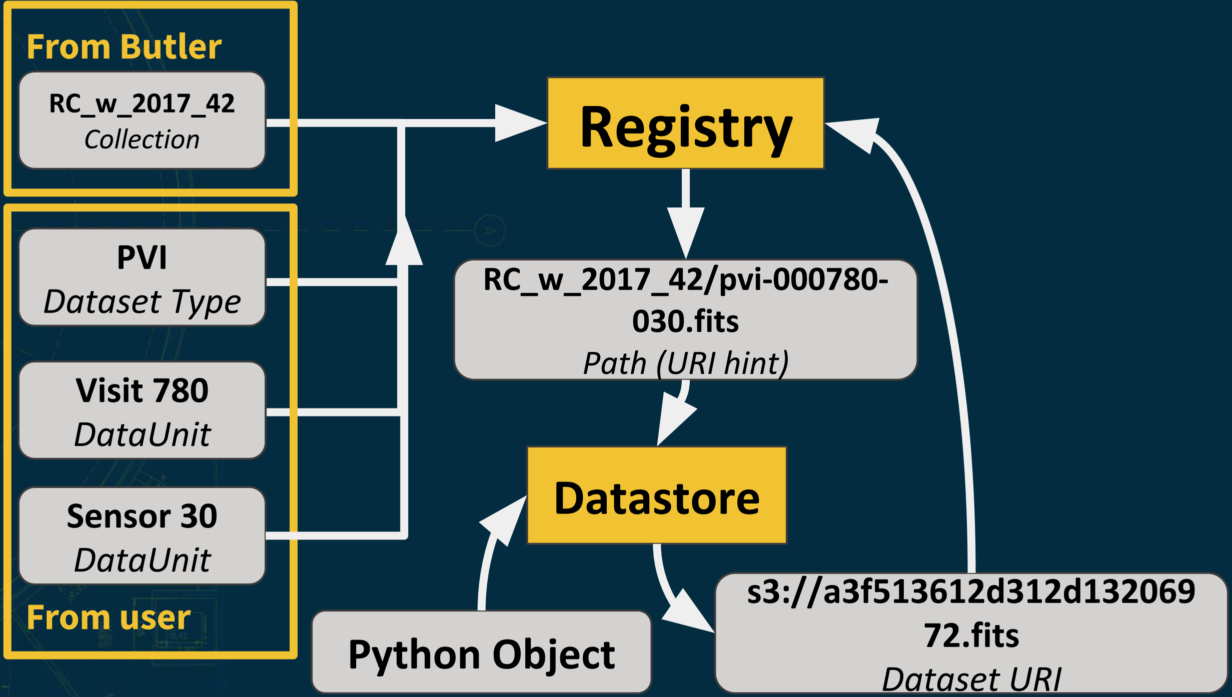
Rubin Observatory

- Easy to write code such that beginners to advanced users all have what they need
- Great for getting things going quickly, but this can have software engineering costs when things scale up
- Python has started addressing some of these issues with things like abstract base classes, type annotations, python 3 being new-style class only, etc.
- Great for bringing together code written in lower level languages

U.S. DEPARTMENT OF ENERGY | Office of Science

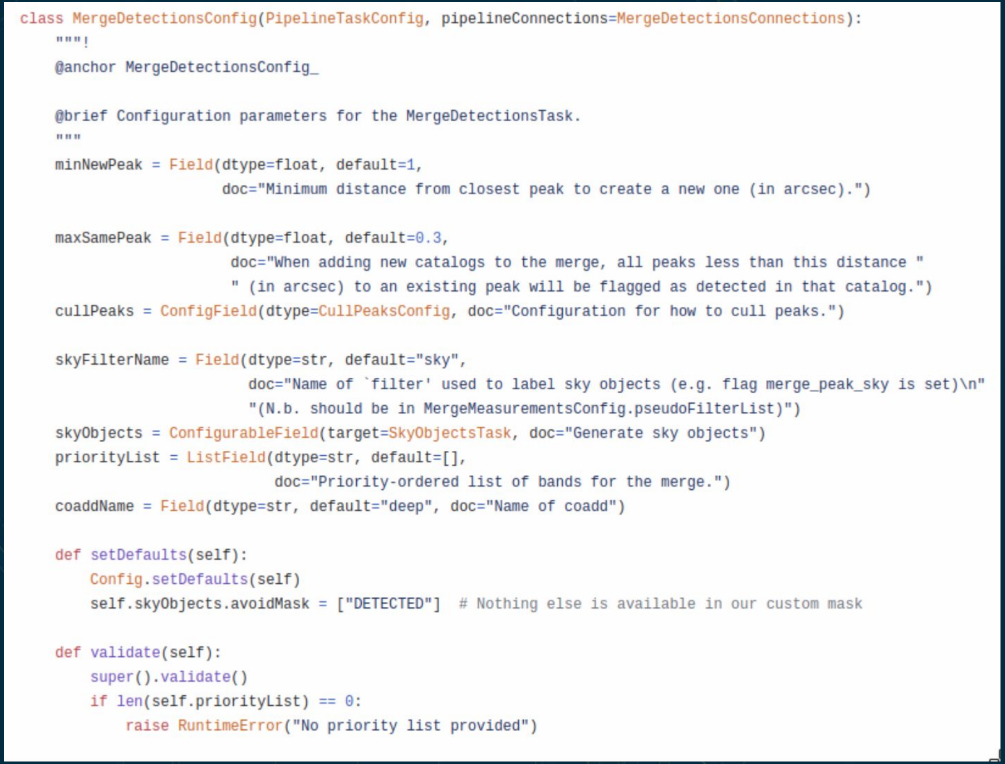
Python in the workflow

A middleware system for storage indexing and retrieval (get just what you need, right when you need it)



Python in the workflow

Rubin Observatory



```
class MergeDetectionsConfig(PipelineTaskConfig, pipelineConnections=MergeDetectionsConnections):
    """!
    @anchor MergeDetectionsConfig_

    @brief Configuration parameters for the MergeDetectionsTask.
    """
    minNewPeak = Field(dtype=float, default=1,
                       doc="Minimum distance from closest peak to create a new one (in arcsec).")

    maxSamePeak = Field(dtype=float, default=0.3,
                        doc="When adding new catalogs to the merge, all peaks less than this distance "
                        " (in arcsec) to an existing peak will be flagged as detected in that catalog.")
    cullPeaks = ConfigField(dtype=CullPeaksConfig, doc="Configuration for how to cull peaks.")

    skyFilterName = Field(dtype=str, default="sky",
                          doc="Name of `filter' used to label sky objects (e.g. flag merge_peak_sky is set)\n"
                          "(N.b. should be in MergeMeasurementsConfig.pseudoFilterList)")
    skyObjects = ConfigurableField(target=SkyObjectsTask, doc="Generate sky objects")
    priorityList = ListField(dtype=str, default=[],
                             doc="Priority-ordered list of bands for the merge.")
    coaddName = Field(dtype=str, default="deep", doc="Name of coadd")

    def setDefaults(self):
        Config.setDefaults(self)
        self.skyObjects.avoidMask = ["DETECTED"]  # Nothing else is available in our custom mask

    def validate(self):
        super().validate()
        if len(self.priorityList) == 0:
            raise RuntimeError("No priority list provided")
```

Declarative (nestable) schema based configuration system that tracks what changed a config option, and when it was changed.

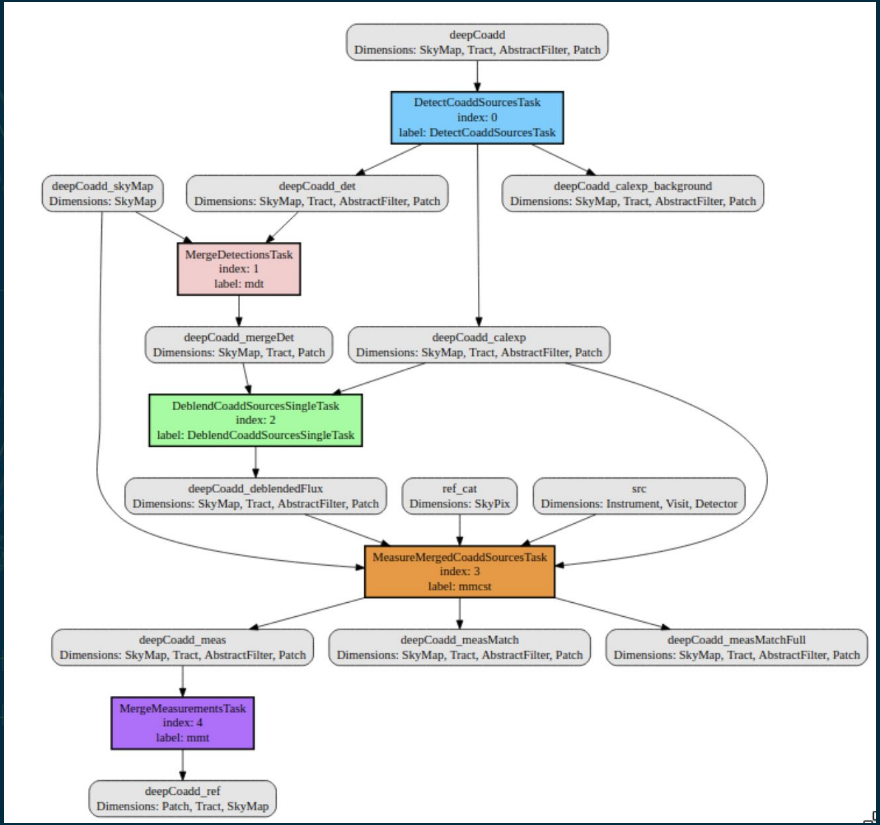
An example for this class may be:

```
config = MergeDetectionsConfig()
config.priorityList = \
    [“I-Band”, “R-Band”, “Z-Band”]
```

Python in the workflow

Rubin Observatory

Task based (nestable) operations that can be arranged into various “pipelines”, with automatic execution ordering based on data-set dependency relations

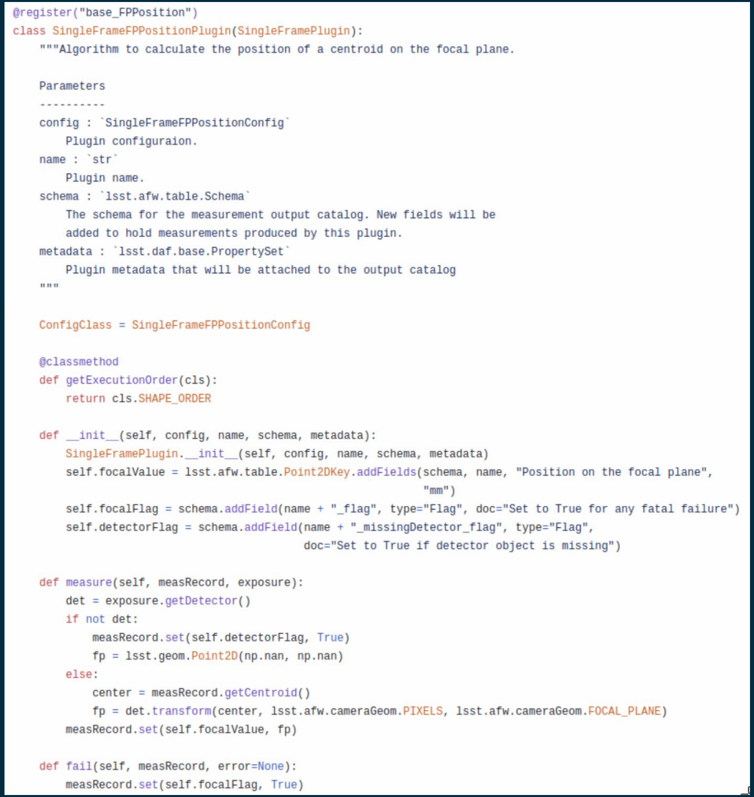

deepCoadd
Dimensions: SkyMap, Tract, AbstractFilter, Patch
DetectCoaddSourcesTask
index: 0
label: DetectCoaddSourcesTask
deepCoadd_skyMap
Dimensions: SkyMap
deepCoadd_det
Dimensions: SkyMap, Tract, AbstractFilter, Patch
deepCoadd_calexp_background
Dimensions: SkyMap, Tract, AbstractFilter, Patch
MergeDetectionsTask
index: 1
label: mdt
deepCoadd_mergeDet
Dimensions: SkyMap, Tract, Patch
deepCoadd_calexp
Dimensions: SkyMap, Tract, AbstractFilter, Patch
DeblendCoaddSourcesSingleTask
index: 2
label: DeblendCoaddSourcesSingleTask
deepCoadd_deblendedFlux
Dimensions: SkyMap, Tract, AbstractFilter, Patch
ref_cat
Dimensions: SkyPix
src
Dimensions: Instrument, Visit, Detector
MeasureMergedCoaddSourcesTask
index: 3
label: mmcst
deepCoadd_meas
Dimensions: SkyMap, Tract, AbstractFilter, Patch
deepCoadd_measMatch
Dimensions: SkyMap, Tract, AbstractFilter, Patch
deepCoadd_measMatchFull
Dimensions: SkyMap, Tract, AbstractFilter, Patch
MergeMeasurementsTask
index: 4
label: mmt
deepCoadd_ref
Dimensions: Patch, Tract, SkyMap

NSF U.S. DEPARTMENT OF ENERGY Office of Science

Python in the workflow

Rubin
Observatory

Pluggable scientific algorithms with automatic runtime ordering



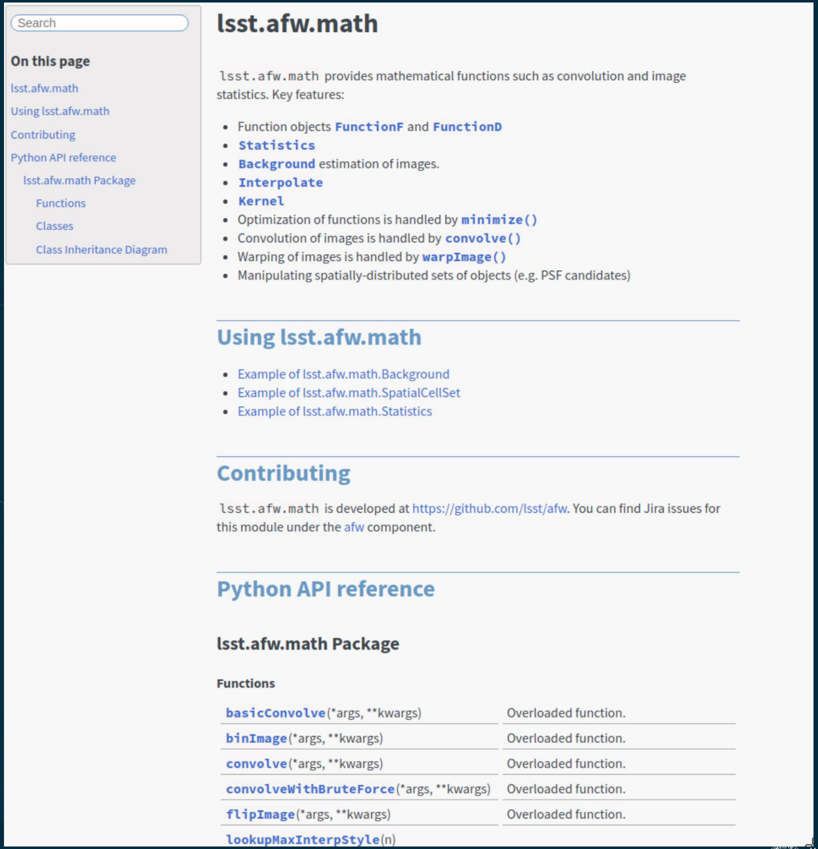
```
@register("base_FPPosition")
class SingleFrameFPPositionPlugin(SingleFramePlugin):
    """Algorithm to calculate the position of a centroid on the focal plane.

    Parameters
    ----------
    config : `SingleFrameFPPositionConfig`
        Plugin configuraion.
    name : `str`
        Plugin name.
    schema : `lsst.afw.table.Schema`
        The schema for the measurement output catalog. New fields will be
        added to hold measurements produced by this plugin.
    metadata : `lsst.daf.base.PropertySet`
        Plugin metadata that will be attached to the output catalog
    """

    ConfigClass = SingleFrameFPPositionConfig

    @classmethod
    def getExecutionOrder(cls):
        return cls.SHAPE_ORDER

    def __init__(self, config, name, schema, metadata):
        SingleFramePlugin.__init__(self, config, name, schema, metadata)
        self.focalValue = lsst.afw.table.Point2DKey.addFields(schema, name, "Position on the focal plane",
                                                              "mm")
        self.focalFlag = schema.addField(name + "_flag", type="Flag", doc="Set to True for any fatal failure")
        self.detectorFlag = schema.addField(name + "_missingDetector_flag", type="Flag",
                                            doc="Set to True if detector object is missing")

    def measure(self, measRecord, exposure):
        det = exposure.getDetector()
        if not det:
            measRecord.set(self.detectorFlag, True)
            fp = lsst.geom.Point2D(np.nan, np.nan)
        else:
            center = measRecord.getCentroid()
            fp = det.transform(center, lsst.afw.cameraGeom.PIXELS, lsst.afw.cameraGeom.FOCAL_PLANE)
        measRecord.set(self.focalValue, fp)

    def fail(self, measRecord, error=None):
        measRecord.set(self.focalFlag, True)
```

U.S. DEPARTMENT OF ENERGY | Office of Science

Python in the workflow

Rubin Observatory

Extensive library of primitives



Search

On this page

- lsst.afw.math
- Using lsst.afw.math
- Contributing
- Python API reference
  - lsst.afw.math Package
    - Functions
    - Classes
    - Class Inheritance Diagram

lsst.afw.math

`lsst.afw.math` provides mathematical functions such as convolution and image statistics. Key features:

- Function objects `FunctionF` and `FunctionD`
- `Statistics`
- `Background` estimation of images.
- `Interpolate`
- `Kernel`
- Optimization of functions is handled by `minimize()`
- Convolution of images is handled by `convolve()`
- Warping of images is handled by `warpImage()`
- Manipulating spatially-distributed sets of objects (e.g. PSF candidates)

Using lsst.afw.math

- Example of lsst.afw.math.Background
- Example of lsst.afw.math.SpatialCellSet
- Example of lsst.afw.math.Statistics

Contributing

`lsst.afw.math` is developed at https://github.com/lsst/afw. You can find Jira issues for this module under the afw component.

Python API reference

lsst.afw.math Package

Functions

`basicConvolve`(*args, **kwargs)	Overloaded function.
`binImage`(*args, **kwargs)	Overloaded function.
`convolve`(*args, **kwargs)	Overloaded function.
`convolveWithBruteForce`(*args, **kwargs)	Overloaded function.
`flipImage`(*args, **kwargs)	Overloaded function.
`lookupMaxInterpStyle`(n)	

NSF

U.S. DEPARTMENT OF ENERGY Office of Science

Data Analysis code base

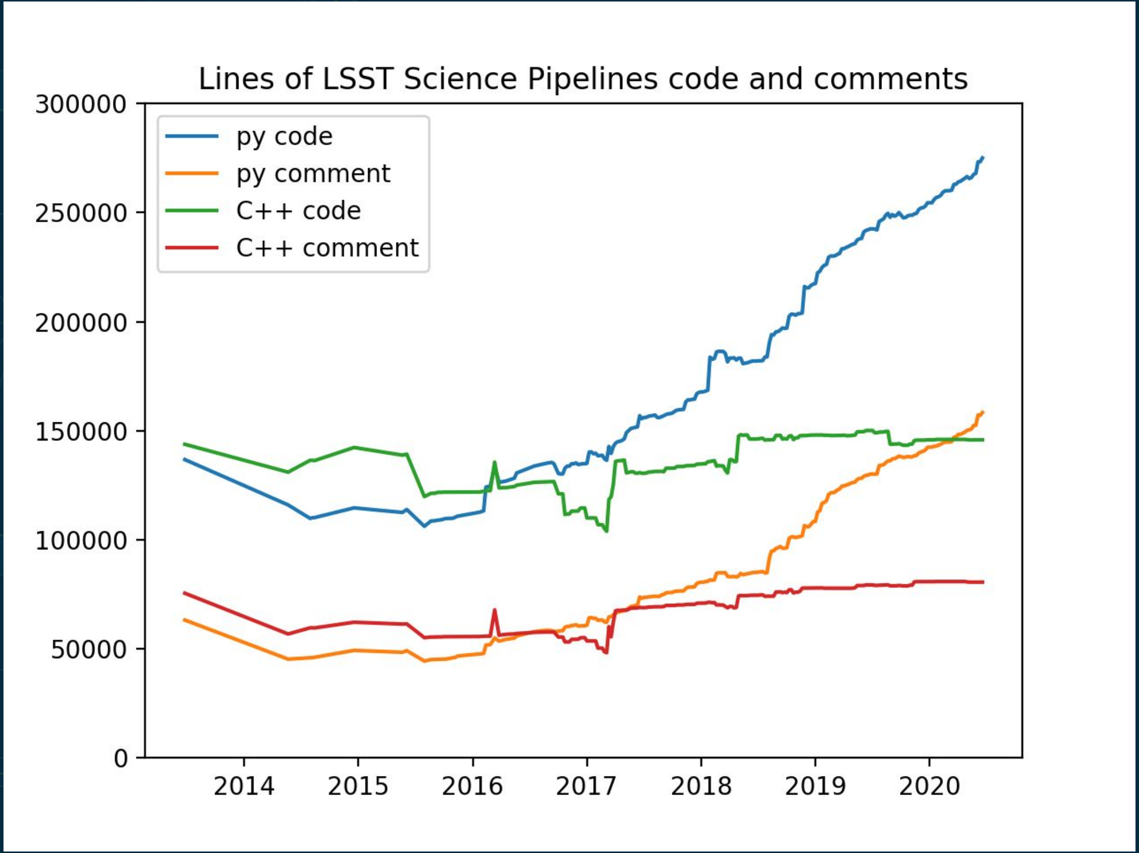
Rubin Observatory

	C++	Python	Total
# of flles	1,353	2,438	3,791
# lines of comments	80,613	158,342	238,955
# lines of code	145,830	275,045	420,875

Split over ~80 packages

The codebase through time

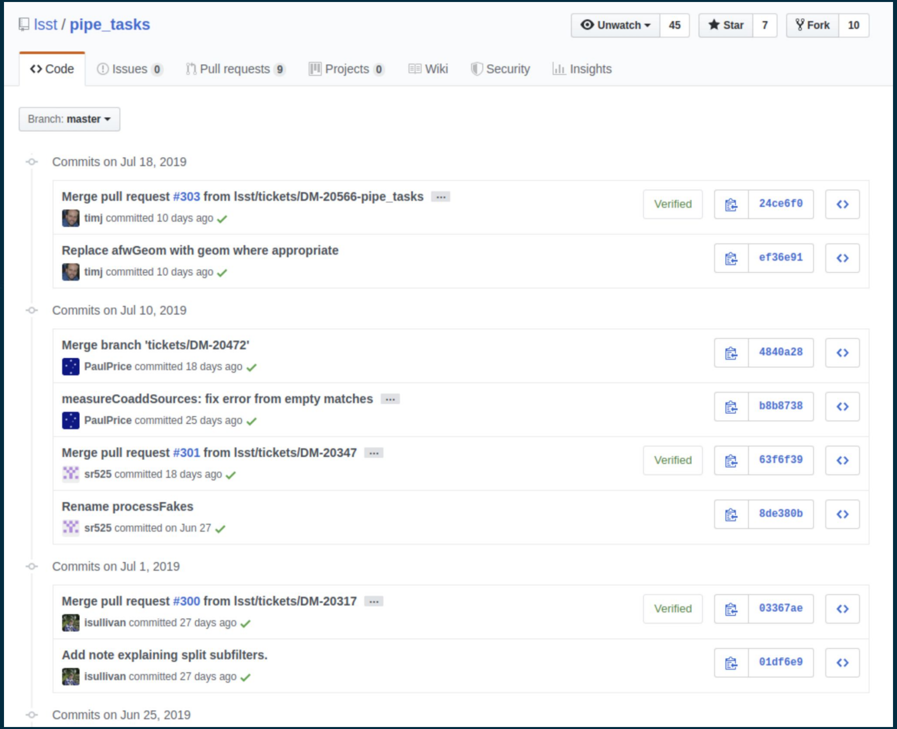
Rubin Observatory



Development Process

Rubin Observatory

- All work managed through tickets
- Any developer can create tickets but, major changes subject to request for comment (RFC) discussions with whole project
- Code tracked with git, changes happen on “ticket” branches
- Code reviews done with Github
- Merges are done with no fast forward, to help associate with changes with tickets
- Commits are squashed to functioning logical units of work



Commits on Jul 18, 2019

Merge pull request #303 from lsst/tickets/DM-20566-pipe_tasks
timj committed 10 days ago ✓ — Verified — 24ce6f0

Replace afwGeom with geom where appropriate
timj committed 10 days ago ✓ — ef36e91

Commits on Jul 10, 2019

Merge branch 'tickets/DM-20472'
PaulPrice committed 18 days ago ✓ — 4840a28

measureCoaddSources: fix error from empty matches
PaulPrice committed 25 days ago ✓ — b8b8738

Merge pull request #301 from lsst/tickets/DM-20347
sr525 committed 18 days ago ✓ — Verified — 63f6f39

Rename processFakes
sr525 committed on Jun 27 ✓ — 8de380b

Commits on Jul 1, 2019

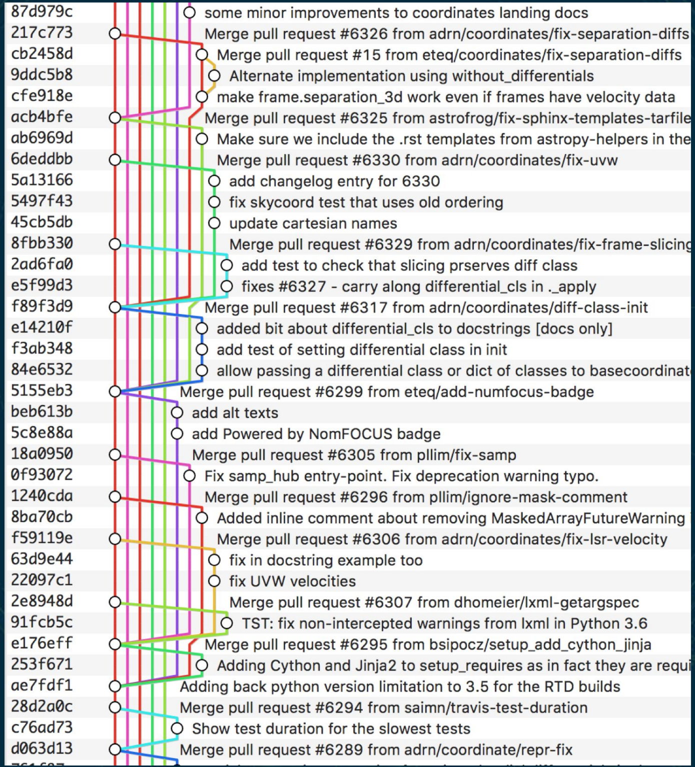
Merge pull request #300 from lsst/tickets/DM-20317
isullivan committed 27 days ago ✓ — Verified — 03367ae

Add note explaining split subfilters.
isullivan committed 27 days ago ✓ — 01df6e9

Commits on Jun 25, 2019

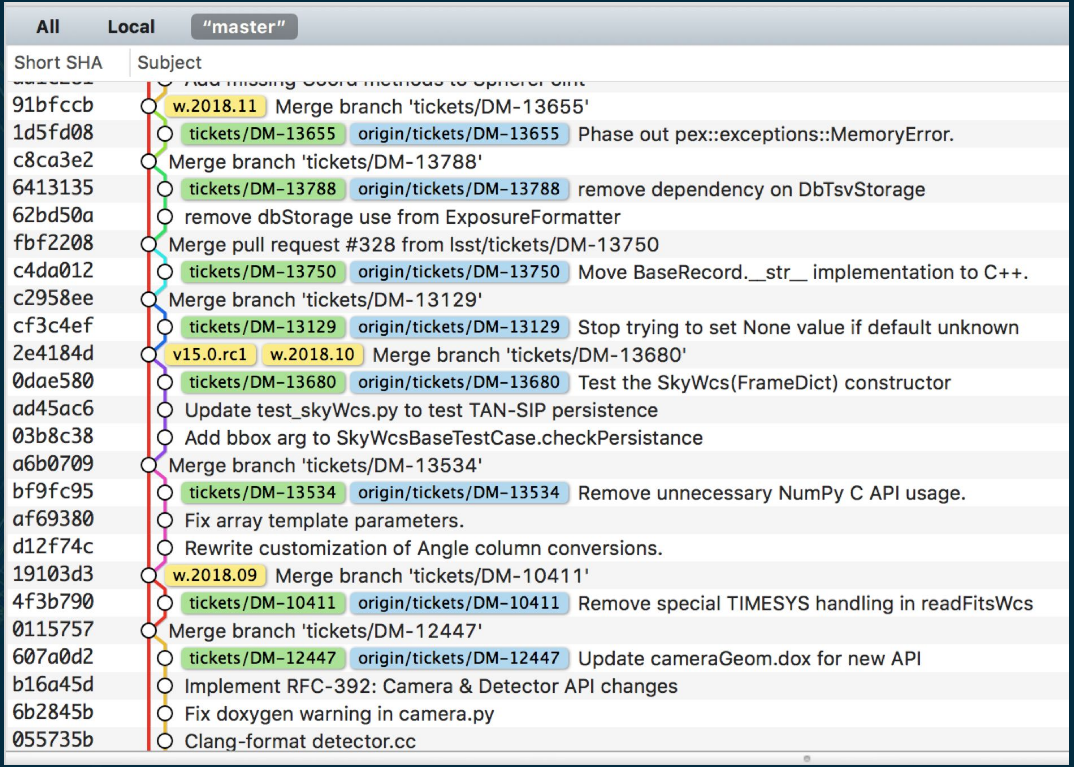
Development Process

Rubin Observatory



87d979c	some minor improvements to coordinates landing docs
217c773	Merge pull request #6326 from adrn/coordinates/fix-separation-diffs
cb2458d	Merge pull request #15 from eteq/coordinates/fix-separation-diffs
9ddc5b8	Alternate implementation using without_differentials
cfe918e	make frame.separation_3d work even if frames have velocity data
acb4bfe	Merge pull request #6325 from astrofrog/fix-sphinx-templates-tarfile
ab6969d	Make sure we include the .rst templates from astropy-helpers in the
6deddbb	Merge pull request #6330 from adrn/coordinates/fix-uvw
5a13166	add changelog entry for 6330
5497f43	fix skycoord test that uses old ordering
45cb5db	update cartesian names
8fbb330	Merge pull request #6329 from adrn/coordinates/fix-frame-slicing
2ad6fa0	add test to check that slicing prserves diff class
e5f99d3	fixes #6327 - carry along differential_cls in ._apply
f89f3d9	Merge pull request #6317 from adrn/coordinates/diff-class-init
e14210f	added bit about differential_cls to docstrings [docs only]
f3ab348	add test of setting differential class in init
84e6532	allow passing a differential class or dict of classes to basecoordinat
5155eb3	Merge pull request #6299 from eteq/add-numfocus-badge
beb613b	add alt texts
5c8e88a	add Powered by NomFOCUS badge
18a0950	Merge pull request #6305 from pllim/fix-samp
0f93072	Fix samp_hub entry-point. Fix deprecation warning typo.
1240cda	Merge pull request #6296 from pllim/ignore-mask-comment
8ba70cb	Added inline comment about removing MaskedArrayFutureWarning
f59119e	Merge pull request #6306 from adrn/coordinates/fix-lsr-velocity
63d9e44	fix in docstring example too
22097c1	fix UVW velocities
2e8948d	Merge pull request #6307 from dhomeier/lxml-getargspec
91fcb5c	TST: fix non-intercepted warnings from lxml in Python 3.6
e176eff	Merge pull request #6295 from bsipocz/setup_add_cython_jinja
253f671	Adding Cython and Jinja2 to setup_requires as in fact they are requi
ae7fdf1	Adding back python version limitation to 3.5 for the RTD builds
28d2a0c	Merge pull request #6294 from saimn/travis-test-duration
c76ad73	Show test duration for the slowest tests
d063d13	Merge pull request #6289 from adrn/coordinate/repr-fix

Another open source python package



All | Local | "master"

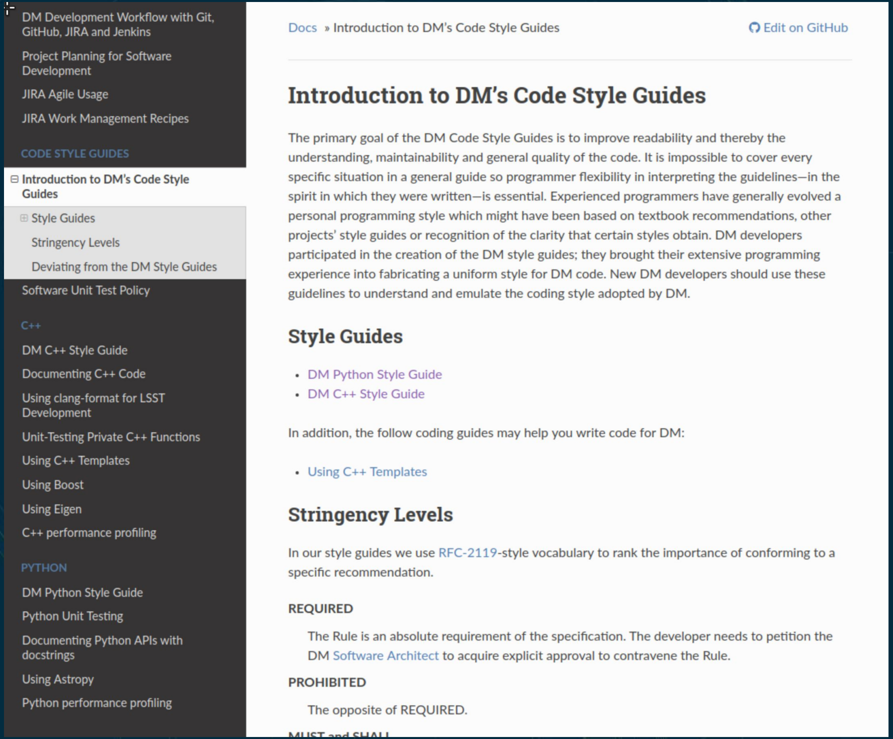
Short SHA	Subject
91bfccb	w.2018.11 Merge branch 'tickets/DM-13655'
1d5fd08	tickets/DM-13655 origin/tickets/DM-13655 Phase out pex::exceptions::MemoryError.
c8ca3e2	Merge branch 'tickets/DM-13788'
6413135	tickets/DM-13788 origin/tickets/DM-13788 remove dependency on DbTsvStorage
62bd50a	remove dbStorage use from ExposureFormatter
fbf2208	Merge pull request #328 from lsst/tickets/DM-13750
c4da012	tickets/DM-13750 origin/tickets/DM-13750 Move BaseRecord.__str__ implementation to C++.
c2958ee	Merge branch 'tickets/DM-13129'
cf3c4ef	tickets/DM-13129 origin/tickets/DM-13129 Stop trying to set None value if default unknown
2e4184d	v15.0.rc1 w.2018.10 Merge branch 'tickets/DM-13680'
0dae580	tickets/DM-13680 origin/tickets/DM-13680 Test the SkyWcs(FrameDict) constructor
ad45ac6	Update test_skyWcs.py to test TAN-SIP persistence
03b8c38	Add bbox arg to SkyWcsBaseTestCase.checkPersistance
a6b0709	Merge branch 'tickets/DM-13534'
bf9fc95	tickets/DM-13534 origin/tickets/DM-13534 Remove unnecessary NumPy C API usage.
af69380	Fix array template parameters.
d12f74c	Rewrite customization of Angle column conversions.
19103d3	w.2018.09 Merge branch 'tickets/DM-10411'
4f3b790	tickets/DM-10411 origin/tickets/DM-10411 Remove special TIMESYS handling in readFitsWcs
0115757	Merge branch 'tickets/DM-12447'
607a0d2	tickets/DM-12447 origin/tickets/DM-12447 Update cameraGeom.dox for new API
b16a45d	Implement RFC-392: Camera & Detector API changes
6b2845b	Fix doxygen warning in camera.py
055735b	Clang-format detector.cc

An LSST repository

U.S. DEPARTMENT OF ENERGY | Office of Science

Development Process

Rubin Observatory



DM Development Workflow with Git, GitHub, JIRA and Jenkins
Project Planning for Software Development
JIRA Agile Usage
JIRA Work Management Recipes

CODE STYLE GUIDES

⊟ Introduction to DM's Code Style Guides
⊞ Style Guides
Stringency Levels
Deviating from the DM Style Guides
Software Unit Test Policy

C++

DM C++ Style Guide
Documenting C++ Code
Using clang-format for LSST Development
Unit-Testing Private C++ Functions
Using C++ Templates
Using Boost
Using Eigen
C++ performance profiling

PYTHON

DM Python Style Guide
Python Unit Testing
Documenting Python APIs with docstrings
Using Astropy
Python performance profiling

Docs » Introduction to DM's Code Style Guides   Edit on GitHub

Introduction to DM's Code Style Guides

The primary goal of the DM Code Style Guides is to improve readability and thereby the understanding, maintainability and general quality of the code. It is impossible to cover every specific situation in a general guide so programmer flexibility in interpreting the guidelines—in the spirit in which they were written—is essential. Experienced programmers have generally evolved a personal programming style which might have been based on textbook recommendations, other projects' style guides or recognition of the clarity that certain styles obtain. DM developers participated in the creation of the DM style guides; they brought their extensive programming experience into fabricating a uniform style for DM code. New DM developers should use these guidelines to understand and emulate the coding style adopted by DM.

Style Guides

- DM Python Style Guide
- DM C++ Style Guide

In addition, the follow coding guides may help you write code for DM:

- Using C++ Templates

Stringency Levels

In our style guides we use RFC-2119-style vocabulary to rank the importance of conforming to a specific recommendation.

REQUIRED

The Rule is an absolute requirement of the specification. The developer needs to petition the DM Software Architect to acquire explicit approval to contravene the Rule.

PROHIBITED

The opposite of REQUIRED.

- Use of style guide for uniformity for both code and git practices
- Style guide itself is a repository subject to same change process as all code repos
- Travis + Jenkins for unit/integration tests
- developer.lsst.io

Lessons and musings

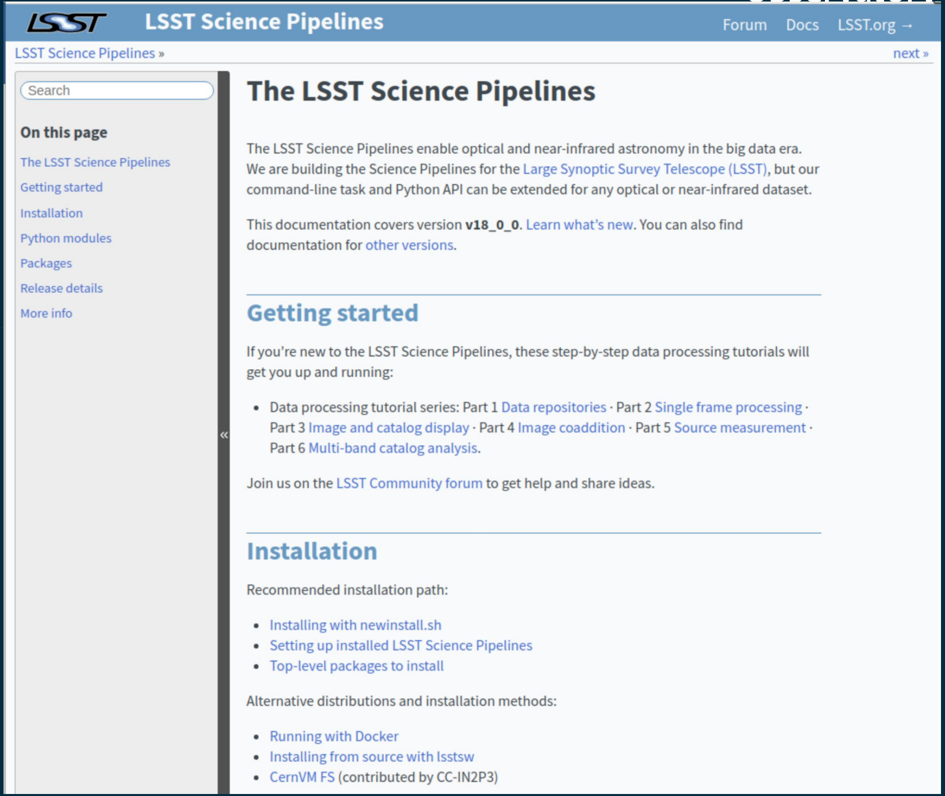
Rubin Observatory

- Make objects (conceptually) immutable whenever you can, python shares everywhere, in big complex systems, it is likely a variable is “owned” elsewhere and hard to trace down side effects happen
- Use interfaces that are separate from implementations (ABCs for example). With the ease of Python programming, unintentional lock-in to a design happens, making refactoring and managing dependencies much harder
- Organize your code according to dependencies, not related ideas. I.E. not necessarily attaching functionality as methods or putting a function in a low level package to be shared just because it might be useful to someone

Lessons and musings

Rubin Observatory

- Packages/ci/build systems always take more time than you expect
- Documentation is important, a community culture around docs is key to making sure they are written
- Documentation bit-rots faster than code.
- Test are important, unit tests are not always the best way to test code (note we do love our unit tests), integration tests are often more revealing of relevant behavior
- Linters and the like have really helped saved time, catch bit-rot, and enforce standards



The LSST Science Pipelines

The LSST Science Pipelines enable optical and near-infrared astronomy in the big data era. We are building the Science Pipelines for the Large Synoptic Survey Telescope (LSST), but our command-line task and Python API can be extended for any optical or near-infrared dataset.

This documentation covers version **v18_0_0**. Learn what's new. You can also find documentation for other versions.

Getting started

If you're new to the LSST Science Pipelines, these step-by-step data processing tutorials will get you up and running:

- Data processing tutorial series: Part 1 Data repositories · Part 2 Single frame processing · Part 3 Image and catalog display · Part 4 Image coaddition · Part 5 Source measurement · Part 6 Multi-band catalog analysis.

Join us on the LSST Community forum to get help and share ideas.

Installation

Recommended installation path:

- Installing with newinstall.sh
- Setting up installed LSST Science Pipelines
- Top-level packages to install

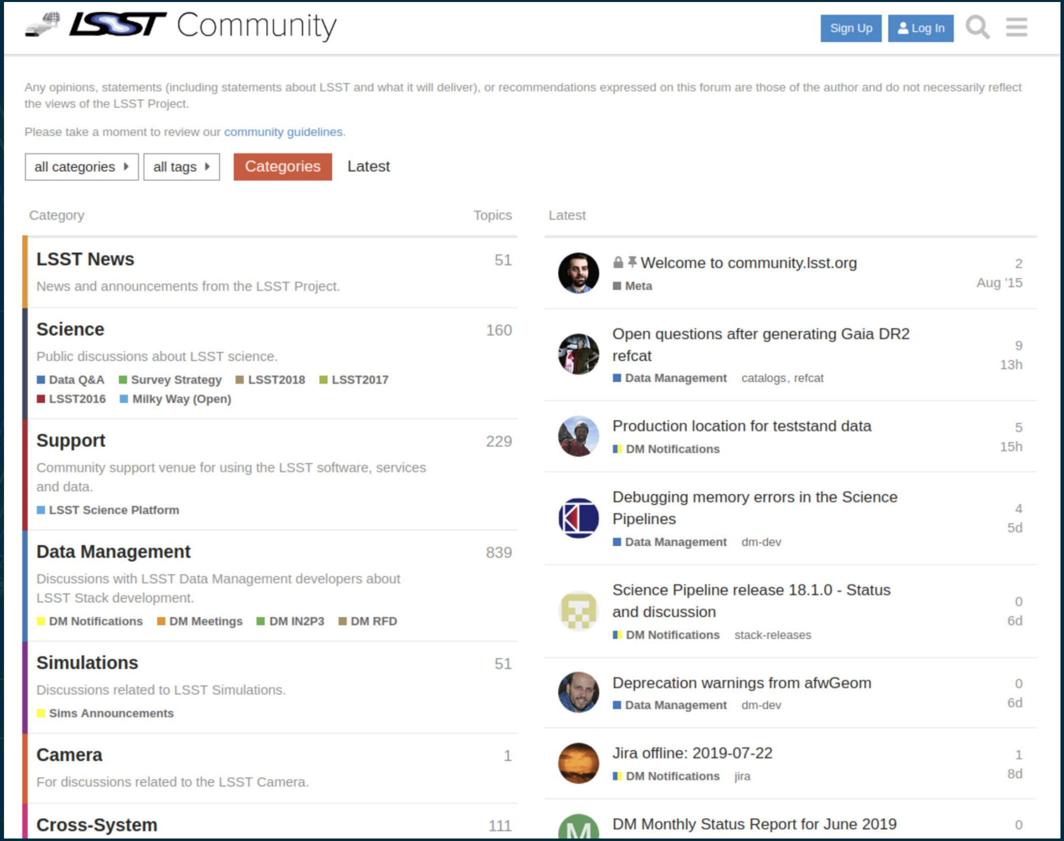
Alternative distributions and installation methods:

- Running with Docker
- Installing from source with lsstsw
- CernVM FS (contributed by CC-IN2P3)

Wrapping up

Rubin Observatory

- We continue to grow alongside the tools we use
- Our goals are to create a welcoming community environment
- All of our software is open source under GPL v3 and developed in the open
- This includes our infrastructure
- We welcome feedback and collaboration
- Best way to ask questions or get in contact is at community.lsst.org under Data Management, you can mention me with @natelust



Any opinions, statements (including statements about LSST and what it will deliver), or recommendations expressed on this forum are those of the author and do not necessarily reflect the views of the LSST Project.

Please take a moment to review our community guidelines.

Category	Topics
LSST News News and announcements from the LSST Project.	51
Science Public discussions about LSST science. Data Q&A · Survey Strategy · LSST2018 · LSST2017 · LSST2016 · Milky Way (Open)	160
Support Community support venue for using the LSST software, services and data. LSST Science Platform	229
Data Management Discussions with LSST Data Management developers about LSST Stack development. DM Notifications · DM Meetings · DM IN2P3 · DM RFD	839
Simulations Discussions related to LSST Simulations. Sims Announcements	51
Camera For discussions related to the LSST Camera.	1
Cross-System	111

Latest		
Welcome to community.lsst.org — Meta	2	Aug '15
Open questions after generating Gaia DR2 refcat — Data Management — catalogs, refcat	9	13h
Production location for teststand data — DM Notifications	5	15h
Debugging memory errors in the Science Pipelines — Data Management — dm-dev	4	5d
Science Pipeline release 18.1.0 - Status and discussion — DM Notifications — stack-releases	0	6d
Deprecation warnings from afwGeom — Data Management — dm-dev	0	6d
Jira offline: 2019-07-22 — DM Notifications — jira	1	8d
DM Monthly Status Report for June 2019	0	

U.S. DEPARTMENT OF ENERGY Office of Science