

# Data Release Production WIP F16 Release Notes



Do not edit

This page is frozen for the v13 release. Please do not edit!

These *draft* notes cover the major updates made by Data Release Production to the LSST stack since release 12 (Winter/Extra 2016). Please add material here which should ultimately be incorporated into the notes accompanying the Fall 2016 release.

## Major Functionality and Interface Changes

### Add a measurement extension which performs Kron photometry

The KronFluxAlgorithm measures the flux in an elliptical aperture as defined by [Kron \(1980\)](#). The plugin is not enabled by default, but may be switched on by requesting `ext_photometryKron_KronFlux` in measurement configuration.

 [DM-3440](#) - add `meas_extensions_photometryKron` to `lsstsw`, `lsst_distrib` DONE

 [DM-6564](#) - Fix order of flags in Kron photometry DONE

### Package version tracking and checking ("stack provenance")

A package version tracking and checking capability has been included in the stack and integrated with the `CmdLineTask` system. This keeps track of what versions of packages are being used in order to prevent the user from unwittingly processing data with multiple, subtly different, versions of the code. This provides a provisional implementation of a provenance tracking system: it is essential for short-term use, but will be rendered obsolete before we move to operations. A [more detailed description](#) of this system is available on the LSST Community Forum.

 [DM-3372](#) - Port, replace, or defer HSC-side provenance of EUPS products DONE

### ngmix-based measurement

Erin Sheldon's `ngmix` has been integrated with the LSST codebase: it can be installed as package `ngmix`, and `meas_extensions_ngmix` package provides an interface between it and the LSST measurement framework. At time of writing, these packages are purely optional, and have not been included in any of the standard Science Pipelines top-level packages (`lsst_apps`, `lsst_distrib`). Currently, we provide an `ngmix`-based plugins for approximating the PSF and for fitting sources using a mixture of Gaussians in both single frame and forced measurement.

 [DM-5429](#) - Add plugins for approximating the PSF as Gaussians with `ngmix` DONE

 [DM-6123](#) - Build SFM housing for PSF approximation using `ngmix` code DONE

 [DM-6124](#) - Testing `ngmix` Psf plugin with `CModel` DONE

 [DM-6125](#) - Do robustness tests of `ngmix` PSF approx plugin DONE

 [DM-2253](#) - add third-party package builds for `ngmix` dependencies DONE

 [DM-5867](#) - `ngmix` is a TaP package DONE

 [DM-5432](#) - Add SFM plugin for `ngmix` fitting DONE

 [DM-6749](#) - `meas_extensions_ngmix` has no `SConstruct` DONE

 [DM-6724](#) - Ship `ngmix` with license DONE

 [DM-6300](#) - Extend galaxy shear fitting results to cover `ngmix` DONE

 [DM-6788](#) - Document `meas_extensions_ngmix` DONE

### Multiband coadd processing now saves metadata

Metadata describing the operations of algorithms within the [multiband coadd processing framework](#) is now correctly persisted along with their outputs.

 [DM-4994](#) - Save algorithm metadata in `multiband.py` DONE

### "Mosaic" calibration

The `meas_mosaic` package for performing simultaneous astrometric calibration across multiple overlapping exposures has been ported from Hyper Suprime-Cam. While this is an essential capability for LSST, we do not expect `meas_mosaic` itself to survive to operations: it is slated for replacement by [Jointcal](#). However, `meas_mosaic` provides an essential point of comparison for `Jointcal`, and is a key capability until `Jointcal` is fully available.

 [DM-2674](#) - Get `meas_mosaic` working on HSC data with LSST stack DONE

## Disabled oversampling in PSFEx

Experiments with Hyper Suprime-Cam have established that oversampling does not work well in PSFEx. It has therefore been disabled in the default configuration used for LSST.

 [DM-6982](#) - Fix oversampling settings in psfex DONE

## Add support for the HSC I2 filter

This filter is now defined in the obs\_subaru camera package.

 [DM-7045](#) - Support reduction of i2 observations DONE

## Multiband coadd processing does not select flagged bands as reference

The algorithm used to select the reference band in multi-band coadd processing has been updated to avoid bands with certain user-specified flags set. By default, these flags include the source having an interpolated centre, or errors detected with PSF, Kron or CModel fluxes.

 [DM-7044](#) - Additional constraints on reference band selection for multiband DONE

## Renamed the "afterburner" system to CatalogCalculation

The "afterburner" system the stack in the previous (v12) release; it provides a capability to run algorithms akin to our regular measurements, but which only take inputs from catalogs (i.e. not pixel data). To avoid ambiguity, this system has been renamed `CatalogCalculation`.

 [DM-6949](#) - Please rename "afterburners" DONE

 [DM-7421](#) - Update processFile to work in modern mid-2016 stack: fix SIP and ->characterizImageTask DONE

## Centroid position checker

Centroid finding measurement algorithms were updated to ensure that the centroids they find fall within the `Footprint` being centroided. If not, the returned value is forced to correspond to the position of the most significant peak within the `Footprint`, and a flag is set to indicate the error.

 [DM-4926](#) - Centroids fall outside Footprints DONE

## Added ability to run ctrl\_pool processes without a batch system

The option `--batch-type none` (or `None`) runs the command in your process as normal. This means that *e.g.* the pipe\_drivers calibration scripts need not rely on MPI.

 [DM-7900](#) - Add --batch-type None to possibilities, disabling any MPI DONE

 [DM-7943](#) - Even with --batch-type none ctrl\_pool can call `mpi.COMM\_WORLD.Abort()` DONE

## HSC R2 filter

The obs\_subaru package now includes support for the R2 filter on Hyper Suprime-Cam.

 [DM-7510](#) - Add support for HSC-R2 filter DONE

## Added DirectMatchTask

The new `DirectMatchTask` matches sources against a reference catalog without performing any rotation, offset or WCS fitting. This provides a simpler and less error-prone option for matching sources for quality assessment. The `MeasureMergedCoaddSourcesTask` has been modified to make use of this new `DirectMatchTask`.

 [DM-7447](#) - measureCoaddSources fails with "RuntimeError: Unable to match sources" DONE

## Added --show history=config.foo.bar

You can now specify a command line argument `-show history=config.foo.bar` to see where configuration parameter `foo.bar` is set and overridden (the leading `config.` is optional). The output may not be very intuitive, but it is what `pex_config` provides and is sometimes just what you need. N.B. pipe to `cat` to turn off colourization (or investigate `lsst.pex.config.history.Color`).

DM-5424 - Add --show history option to cmdLineTask **DONE**

DM-7949 - Disable colourisation when not writing to a terminal **DONE**

## frame defaults to None when creating a new Display

When using the `afw.display` system to create a new `Display` object, the `frame` argument is now optional: it will default to `None` if not specified.

RFC-244 - Minor changes to `afw.display.Device` interface **IMPLEMENTED**

DM-7848 - Implement minor `afw.display` changes as per RFC-244 **DONE**

## Attribute access to Display objects is passed through to the underlying implementation

Some `afw.display` device types may provide implementation-specific functionality. This may now be accessed directly through the `Display` object, which will delegate to implementation-specific methods, if they exist, when a non-standard method is called on the `Display`.

RFC-244 - Minor changes to `afw.display.Device` interface **IMPLEMENTED**

DM-7848 - Implement minor `afw.display` changes as per RFC-244 **DONE**

## New fitter for astrometric distortions.

A new fitter task, `meas.astrom.FitSipDistortionTask` has been added that will eventually replace the current fitter, `meas.astrom.FitTanSipWcsTask`. The new fitter should be more robust and certainly uses a better-motivated algorithm, but it has not yet received enough testing to be used as the default and hence must be enabled explicitly. When running `processCcd.py`, this can be done with the following config file:

```
from lsst.meas.astrom import FitSipDistortionTask
config.charImage.astrometry.wcsFitter.retarget(FitSipDistortionTask)
config.calibrate.astrometry.wcsFitter.retarget(FitSipDistortionTask)
```

The new fitter is not compatible with `meas.astrom.ANetAstrometryTask`.

DM-3549 - Further improve the TAN-SIP WCS fitter **DONE**

## Unified exposure metadata

Exposures now contain a new `VisitInfo` object that contains useful information about the overall visit (none of it specific to the individual CCD), including the exposure time, date and ID, boresight position and airmass. See [Unified exposure metadata](#) for more information.

RFC-499 - Add new metadata to `ExposureInfo` **ADOPTED**

DM-5593 - Implement single interface to sanitized exposure metadata **DONE**

## Convenient access to catalog metadata

It is now possible to read the metadata, length and schema of a catalog through a `Butler` interface without loading the whole catalog. For example, given a particular data ID (`dataId`) and catalog dataset type (`someCatalog`), one can run:

```
butler.get("someCatalog_schema", dataId)
butler.get("someCatalog_len", dataId)
butler.get("someCatalog_md", dataId)
```

DM-8211 - Add support for reading a catalog schema **DONE**

DM-8212 - Add support for reading metadata, length and schema of a catalog **DONE**

## Return a PropertyList when reading FITS headers

`lsst.afw.image.readMetadata` now returns a `PropertyList`, which maintains the intrinsic ordering of the header keywords in the FITS file it is reading. Previous versions used a `PropertySet`, which is unordered and hence lost this information.

DM-8243 - `afw.image.readMetadata` returns a `PropertySet` **DONE**

## Measurement without deblending

The deblender sometimes runs into trouble with cluster galaxies, leading to inaccurate fluxes. To mitigate, we can now provide measurements performed without any deblending having taken place. To activate this functionality, specify the measurement plugins to be run without deblending in the `undeblender` registry attached to the measurement class (this is analogous to the `plugins` registry used when performing measurement on deblended sources). By default, no undeblended measurement is performed. This functionality is available in both forced and single-frame measurement.

 [DM-6795](#) - Port parent/child measurement from HSC DONE

 [DM-7889](#) - Activate HSC afterburner functionality DONE

## Warp then match when building PSF-matched coadds

WarpAndPsfMatchTask has been updated so that, by default, it will first warp inputs to the coadd projection and then perform PSF matching; this is the reverse of its previous behaviour. The new behaviour is preferable, but a configuration option makes it possible to revert to the old method on request.

 [DM-8088](#) - When making PSF-matched coadds, warp first then PSF-match DONE

## Reduce galaxy “shredding” when deblending

If galaxies have substructure, such as face-on spirals, the process of identifying peaks can “shred” the galaxy into many pieces. The templates of shredded galaxies are typically quite similar because they represent the same galaxy. We try to identify these “degenerate” peaks by looking at the inner product (in pixel space) of pairs of templates. If they are nearly parallel, we only keep one of the peaks and reject the other. This behaviour is enabled by the `removeDegenerateTemplates` option to `SourceDeblendConfig`; by default, it is disabled.

 [DM-4550](#) - Add SDSS anti-shredding algorithm to deblender DONE

## Bug Fixes

### Corrected configuration dictionary comparisons

Resolved a bug whereby comparison of (intrinsically unordered) configuration dictionaries incorrectly compared their ordering rather than just their contents. This led to unexpected failure of re-runs due to incorrectly identified “mismatched configurations”.

 [DM-6664](#) - ConfigDictField says “Inequality in keys for...” even if I give 2 same configurations DONE

### Correct CModel table schema units

Units for the time field changed from seconds to second to comply with the Astropy convention.

 [DM-6282](#) - Fix wrong units in CModel schema DONE

### Restore capability of calculating temporary local backgrounds in source detection

Our `SourceDetectionTask` provides the optional capability of fitting and temporarily removing a local background before detecting sources. This capability was accidentally disabled in the X/W16 release (v12); it has now been restored. It is disabled by default.

 [DM-5323](#) - `estimateBackground` should not make a deep copy of the exposure DONE

### Resolve a memory error in convolution

A memory handling error could cause our convolution code to generate incorrect results on some platforms (problems observed on OS X when the code is built without optimization; no errors seen on Linux, or when building with `-O2` or `-O3` on OS X). This appears to originate in the [Boost Generic Image Library](#), an external dependency of the LSST stack. We have been unable to isolate the problem in Boost, but have worked around it for our particular use case.

 [DM-5822](#) - AFW fails unit test for convolve depending on compiler optimisation level DONE

### Patched GalSim to raise an exception in some error conditions

On certain error conditions, [GalSim](#), a third party package used by the LSST stack, generates an assertion failure which is hard for our code to handle appropriately. LSST’s version of GalSim has been patched to raise a more appropriate exception instead. This change has been [submitted](#) for inclusion in GalSim itself.

 [DM-7114](#) - Guard against assertion failure DONE

### Ensure star selector results are reproducible

Resolved an issue whereby the star selection could change depending on the number of CPU cores in use.

 [DM-7040](#) - Stars selected by starSelector change when number of cores varies DONE

## Single frame measurement exploits multiple CPU cores

Resolved an issue whereby the `SingleFrameDriverTask` failed to use more than one CPU core regardless of configuration.

 [DM-7134](#) - singleFrameDriver is only running with a single process DONE

## Fix typo in CoaddSrcTransformTask

A typo which rendered `CoaddSrcTransformTask` inoperable was resolved. In the process, the tests for this task were significantly upgraded to avoid a repeat of this, or other, problems.

 [DM-6635](#) - Typo in CoaddSrcTransformTask DONE

## Do not write duplicate AR\_NAME headers when persisting data

When persisting objects to FITS, we were incorrectly writing duplicate `AR_NAME` headers. This has been resolved.

 [DM-7224](#) - TUNIT header is not uniformly applied to columns in FITS table output from afw.table DONE

## Be more rigorous about units when persisting angles

We now correctly write `TUNIT` headers describing the units used when storing angles, and are more careful about checking units when reading them.

 [DM-7224](#) - TUNIT header is not uniformly applied to columns in FITS table output from afw.table DONE

## Infinite recursion in afw table

A bug has been discovered in the `afw` table library which could lead to infinite recursion. While the ultimate cause of the bug is still under investigation, a temporary workaround has been put in place to mitigate its impact.

 [DM-7377](#) - meas\_base testMonteCarlo randomly triggers recursion on Python 3 DONE

 [DM-7385](#) - Make temporary workaround for DM-7377 DONE

## Fixed a division by zero error in measurement of aperture corrections

Attempting to calculate an aperture correction for a measurement of zero flux would previously result in an unhandled division by zero.

 [DM-7190](#) - Port HSC aperture correction fix DONE

## Fix dataset lookup in multibandDriver.py

The multiband driver script was previously checking for the existence of the pre-calexp version of the coadd. This is not guaranteed to exist if the coadds were built using the driver scripts (where there is no need to persist both the pre-calexp and calexp versions) and it is only the calexp coadds that are used in the multiband tasks, so it is the calexp version whose existence should be checked for in `multibandDriver.py`.

 [DM-7746](#) - Fix dataset lookup in multibandDriver.py DONE

## Fix incorrect definition of mask when building fringe frames

The code in `pipe_drivers` to construct fringe frames, `FringeTask`, was incorrectly using a mask *plane* when in fact a mask *bit* was required. This has now been corrected.

 [DM-8033](#) - Port HSC fix to fringe construction DONE

## Fix failures in MPI-based processing

Resolved an issue whereby the `ctrl_pool` distributed processing middleware was unable to transmit large volumes of data over MPI due to a limitation of the Python 2.x "pickle" system.

 [DM-8024](#) - Deal with large pickles DONE

## Fix failure to retrieve optimizer state when fitting `DoubleShapeletPsf`

Resolved an issue whereby the optimizer state was being read incorrectly, which could lead to fitting failures.

 [DM-6562](#) - `DoubleShapeletApproximation` fails to fit Gaussian PSF DONE

## Build & Code Improvements

### Migration from Boost to standard library equivalents

Continuing work started in the previous release, we have audited the stack for use of the `Boost` libraries. Where possible, Boost has been replaced with C++11 standard library equivalents.

 [DM-5880](#) - Audit use of Boost in the stack and remove it where possible DONE

 [DM-6089](#) - Use fixed width integer types from `std` instead of boost DONE

 [DM-6090](#) - Replace `boost::lexical_cast` with `std::to_string` where possible DONE

 [DM-6091](#) - Replace `boost::lambda` with C++11 `lambda` DONE  [DM-6092](#) - Remove use of `boost::noncopyable` DONE

 [DM-6093](#) - Replace `boost::random` with `std::random` where possible DONE

 [DM-6094](#) - Use type traits from `type_traits` over boost equivalents where possible DONE  [DM-6095](#) - Replace `boost::ref` DONE

 [DM-6096](#) - Replace `boost::scoped_array` with `std::unique_ptr` specialisation DONE  [DM-6097](#) - Replace `boost::bind` with `std::bind` DONE

 [DM-6325](#) - Replace `BOOST_STATIC_ASSERT` with `static_assert` DONE

### Properly log warnings in `ObjectSizeStarSelector`

Previously, non-fatal NumPy warnings of the form `RuntimeWarning: invalid value encountered` in less could appear on the console, but not be caught and logged anywhere. It was therefore easy to miss problems arising when running the algorithm. These warnings are now ingested into the LSST logging system, and are hence recorded properly.

 [DM-5428](#) - `ObjectSizeStarSelector` can produce numpy warnings DONE

### Unit tests for colorterm framework

A set of unit tests have been written which properly exercise the colorterm framework which was adapted from `Hyper Suprime-Cam` in the S15 cycle.

 [DM-2837](#) - Add unit tests for the new colorterms code DONE

### FlagHandler accessible from Python

The `FlagHandler` system for efficiently and safely setting flags corresponding to measurement algorithm failures on source records is now accessible from Python. Previously, it was only available for use in C++ code.

 [DM-4099](#) - Allow `FlagHandler` to be used from Python DONE

### Exact image comparison utility code

The `assertImagesEqual` and `assertMaskedImagesEqual` methods have been added to added to LSST's test cases. These provide explicit mechanisms for checking for exact equality of image data.

 [DM-6644](#) - Add exact variants of image comparison test utilities DONE

### Add integration test for stellaricity of sources used to measure PSF

Added a test to the `ci_hsc` package to monitor the fraction of sources used to model the PSF in when processing a CCD image and ensure that the vast majority of them are ultimately classified as stars. This provides an automated check that we are not inadvertently estimating the PSF based on extended objects.

 [DM-6054](#) - Add extendedness vs. star selector test to single-visit validation in `ci_hsc` DONE

## Use CatalogCalculation to handle aperture corrections

The `CatalogCalculation` facility was added to the stack in the previous (v12) release (where it was referred to as "afterburners"); it provides a capability to run algorithms akin to our regular measurements, but which only take inputs from catalogs (i.e. not pixel data). This has now been used to clean up the logic used to apply aperture corrections within our regular data processing flow. This old method for handling aperture corrections was a persistent source of bugs and confusion: these should be avoided by the new approach.

 [DM-5877](#) - Use Afterburners to clean up aperture correction logic DONE

## Builds are now optimized by default

Following an RFC, `sconsUtils` was updated to enable optimization by default.

 [RFC-202](#) - Builds are optimised by default IMPLEMENTED  [DM-7040](#) - Builds should be optimised by default DONE

## Fixed a build failure of mpi4py on some versions of Mac OS X

Added extra logic to ensure we target a version of the OS X SDK which is compatible both with the version of Mac OS installed and the Python interpreter in use.

 [DM-6133](#) - `mpi4py` does not compile under Yosemite due to hardcoded `MACOSX_DEPLOYMENT_TARGET` DONE

## Centralised Butler datasets

Butler dataset definitions which are common to multiple cameras have been consolidated in a central location in `daf_butlerUtils` rather than being repeated in each camera package.

 [DM-6858](#) - Mapper tests require modification when new datasets are added DONE

 [RFC-204](#) - Move all config and tract/patch dataset definitions to base `CameraMapper` IMPLEMENTED

 [DM-7049](#) - Move patch/tract and config mapping definitions to `daf_butlerUtils` DONE

 [DM-7677](#) - Create RFC documents for changes to `Mapper.paf` files DONE

## Order in which SConscript files are read is configurable

The `sconsUtils` build system now reads `SConscript` files, which describe the build, in a user configurable order. This makes it possible to specify that some components of a package must be built before others.

 [DM-7179](#) - `sconsUtils` reads `SConscript` files in alphabetical order DONE

## Ndarray is now an external package

The `ndarray` package in the LSST stack now installs vanilla upstream version 1.3.0.

 [DM-2005](#) - switch `ndarray` to external package DONE  [DM-7519](#) - Rebase `pybind11` branch onto unit test and Python 3 changes DONE

## SpherePoint now available

The `SpherePoint` class allows locations on the celestial (or any other) sphere to be represented and manipulated without the coordinate-system complications associated with `Coord`.

 [DM-5529](#) - Implement `SphPoint` DONE