

This document captures proposed changes to the key LSST Database tables (Object, Source, FaintSource, DiaSource) to make these tables aligned with the Level 1 and Level 2 data products outlined in the document written for Science Collaboration after the FRS meeting at UC Davis.

## Sizes

This schema is still "under construction", but the following can already give a useful perspective:

If we assume 250 bytes for Object table (rough estimate based on the proposed schema below), 200 bytes for Source (rough estimate) and 24 bytes for FaintSource, and run these numbers through the storage estimates spreadsheet, we are getting:

- FaintSource table = 61%
- Source = 33%, and
- Object = 0.5%

of the total database size.

So:

- optimizing size of the Object table is not making much difference (it still may for the disk IO depending on access patterns)
- we are paying huge price for introducing the FaintSource, every byte added here adds over a petabyte to the final size of the database.
- understanding access patterns for FaintSource and Source is becoming extremely important. Avoiding joins with FaintSource and Source at all cost by pushing summary information into the lightweight Object table is the way to go.

## Object

### Identifiers

```
objectId BIGINT NOT NULL      -- primary key
procHistoryId INTEGER NOT NULL -- for provenance tracking
iauId CHAR(34)
```

iauId is a IAU compliant name for the object. Example: "LSST-DR11 J001234.65-123456.18 GAL". The last 3 characters identify classification. Note that it will not accommodate multiple classifications.

### Best fitting model type for the object

Several types of object model will be fit to the measurements. This column specifies which model type gives the best fit. The number of all models will be small (few). We will be adding new models very infrequently.

Proposed convention: a model will be identified by two characters. We currently have two models: slowly moving point source (?PS?), and small galaxies (?SG?)

```
bestModel char[2]
```

## Multifit model parameters and covariance matrices

The model parameters include the most important object parameters - mean position, fluxes, shape parameters.

```
-- ra/dec and their errors are present in each model
-- expressed in degrees
raPS FLOAT
raSigmaPS FLOAT
decPS FLOAT
decSigmaPS FLOAT
raSG FLOAT
raSigmaSG FLOAT
decSG FLOAT
decSigmaSG FLOAT

-- only PS model has these
muRaPS DOUBLE, muRaSigmaPS FLOAT -- expressed in miliarcsec per year
muDecPS DOUBLE, muDecSigmaPS FLOAT -- expressed in miliarcsec per year
parallaxPS DOUBLE, parallaxSigmaPS FLOAT -- expressed in miliarcsec

-- Small dimension of the position error ellipse, assuming gaussian scatter
-- These are per model and per filter.
-- Name is a bit awkward, but we will eventually rename them
-- (they will become part of covariance matrix)
-- posErr expressed in arcsec. Theta expressed in degrees
uPosErrAPS FLOAT
uPosErrBPS FLOAT
uPosErrThetaPS FLOAT

uPosErrASG FLOAT
uPosErrBSG FLOAT
uPosErrThetaSG FLOAT

-- One flux + error per model per filter
uFluxPS FLOAT, uFluxSigmaPS FLOAT
uFluxSG FLOAT, uFluxSigmaSG FLOAT

-- First and second moments. These are per model per filter, all FLOATs
uIxPS, uIyPS, uIxyPS, uIxxPS, uIyyPS
uIxSG, uIySG, uIxySG, uIxxSG, uIyySG
```

Notice that we are proposing to use `?dec?` instead of `?decl?`. `?Dec?` is strongly preferred by astronomers. We will always have model type attached to the `?dec/decl?`, so the issue with `?dec?` being a reserved word in some RDBMSes goes away.

The moments are defined as:

- $I_x = \text{integral\_over\_footprint}[w(x-x_0,y-y_0)*f(x,y)*(x-x_0)]$
- $I_y = \text{integral\_over\_footprint}[w(x-x_0,y-y_0)*f(x,y)*(y-y_0)]$
- $I_{xy} = \text{integral\_over\_footprint}[w(x-x_0,y-y_0)*f(x,y)*(x-x_0)*(y-y_0)]$
- $I_{xx} = \text{integral\_over\_footprint}[w(x-x_0,y-y_0)*f(x,y)*(x-x_0)*(x-x_0)]$
- $I_{yy} = \text{integral\_over\_footprint}[w(x-x_0,y-y_0)*f(x,y)*(y-y_0)*(y-y_0)]$

Where:

- $x_0, y_0$  is the object center

- $w(x,y)$  is a weighting function
- $f(x,y)$  is the flux

## Elliptical equivalent to object footprint

The ellipse which gives the same weighted moments as the object

```
raFootprint FLOAT
decFootprint FLOAT
aFootprint FLOAT
bFootprint FLOAT
thetaFootprint FLOAT
```

## Bounding box for object

The smallest box on the sky that fully encloses the object footprint. This is what you need for a postage stamp. These ranges are for the center given by (ra, decl). *Need to decide for which model. Best?*

```
raRange FLOAT
decRange FLOAT
```

## Summary statistics for light curve variability

Still TBD. At least give variance around best fitting constant. Likely include Welch-Stetson statistics, or similar.

```
-- this is per filter, number of measurements in light curve
uNumObs INTEGER

-- Probability of variability in % (100% = variable object). Per filter
uVarProb TINYINT

-- Characteristic timescale of flux variations (measured in days). Per filter.
-- Tim: have we decided to keep it or to remove it???
uTimescale FLOAT
```

## Summary statistics for image model residuals

Still TBD. Goal is to provide useful parameters that can be selected on to find interesting objects such as lensing arcs

[TBD]

## Extendedness parameter

The probability that the object is a point source.

[TBD]

## Photometric redshift and associated PDF

The reported photo-Z will be the peak of the PDF. The full PDF will be included.

There will be a single Level 2 photo\_z information

This section will include info about redshift and info about probability distribution function

Redshift: ~20 columns. Same readshift for many objects, so normalize into separate Redshift table.

Probability distribution function: Not sure yet how to encode it, also not clear how many columns will be needed. Need to go back to the old email thread [Tim will try to find it].

Dave Wittman suggested we'd need an array of ~500 numbers to describe the probability distribution of a galaxy's photometric redshift.

What is currently in placeholder\_photoz is not very useful

## Segmentation info

Relationship to other Objects which are part of the same segmentation (deblending) tree

```
-- same as SDSS
parentId
nChild -- number of children
```

## Time related summary statistics

```
earliestObsTime DOUBLE
latestObsTime FLOAT
```

## Flags ==

TBD how many flags and how to call them.

```
flag BIGINT
```

## Spatial indexes

```
zoneId SMALLINT
chunkId SMALLINT
subZoneId SMALLINT
subChunkId SMALLINT
```

## Columns to be removed

The following columns will be removed from the current (DC3b) schema:

- raFlux, declFlux + errors (The flux weighted position for (ra, decl), calculated from the first moment of the pixel values within the footprint.)

- apMag, petroMag (replaced with 1 magnitude per model)
- refrRaAstrom, refrDeclAstrom, errors (astrometric refraction)
- primaryPeriod, errors (summary period for all filters)
- all scalegrams 25 x 6 filters (they are currently in placeholder\_Object table)
- xFlux, yFlux, errors (The flux weighted position for (x,y), calculated from the first moment of the pixel values within the \* footprint)
- xPeak, yPeak ((x,y) position with the peak value in the footprint.)
- raPeak, declPeak
- raAstrom, declAstrom, errors
- xAstrom, yAstrom, errors
- cx, cy, cz, errors (this was introduced to speed up near neighbor queries)
- uAmplitude
- uPeriod

Issues to be discussed:

- coord-system (no ra/dec, no x/y, just per-model ra/dec + "bestModel" ok?).
- Redshift/probability distribution function.
- Segmentation info (parentId + number of children). Kinds of queries?
- Really no scalegrams?
- Max variation between ra/dec from different models?
- Extra columns to describe object variability?
- Queries asking for best model will be messy (case when then else end).
- Bounding box: (ra/dec range big near poles)

## FaintSource

We need to store ellipses for faint sources generated for moving objects, which argues we should split the FaintSource table into separate tables as shown below.

Actually, it is possible splitting it further into per-filter tables would be useful for two reasons:

1. access patterns (looking a a light curve in a given filter)
2. compression (it will compress better because of different ranges of sky background values)

FaintDIASources will be saved for a fairly small subset of all objects, eg these that pass alerts.

```
CREATE TABLE FaintSourceForObject (
  objectId BIGINT,      -- pointer to Object table
  exposureId INT,      -- pointer to FPA_Exposure
  flux INT,            -- from Exposure
  skyBackground INT,  -- from Exposure
  PRIMARY KEY(objectId, exposureId)
)
```

```
CREATE TABLE FaintDiaSourceForObject (
  objectId BIGINT,      -- pointer to Object table
  exposureId INT,      -- pointer to FPA_DIA_Exposure
  fluxDia INT,         -- from DiaExposure
  skyBackgroundDia INT -- from DiaExposure
  PRIMARY KEY(objectId, exposureId)
)
```

```

CREATE TABLE FaintSourceForMovingObject (
  movingObjectId BIGINT, -- pointer to MovingObject
  exposureId INT,       -- pointer to FPA_Exposure
  flux INT,             -- from Exposure
  skyBackground INT,    -- from Exposure
  -- 20 bytes for model predicted ellipse for Exposure, columns TBD
  predictedEllipseDummy1 FLOAT,
  predictedEllipseDummy2 FLOAT,
  predictedEllipseDummy3 FLOAT,
  predictedEllipseDummy4 FLOAT,
  predictedEllipseDummy5 FLOAT,
  PRIMARY KEY(movingObjectId, exposureId)
)

```

```

CREATE TABLE FaintDiaSourceForMovingObject (
  movingObjectId BIGINT, -- pointer to MovingObject
  exposureId INT,       -- pointer to FPA_DIA_Exposure
  fluxDia INT,         -- from DiaExposure
  skyBackgroundDia INT -- from DiaExposure

  -- 20 bytes for model predicted ellipse for DiaExposure, columns TBD
  predictedEllipseDummy1Dia FLOAT,
  predictedEllipseDummy2Dia FLOAT,
  predictedEllipseDummy3Dia FLOAT,
  predictedEllipseDummy4Dia FLOAT,
  predictedEllipseDummy5Dia FLOAT,

  PRIMARY KEY(movingObjectId, exposureId)
)

```

It might be advantageous to split each of the above tables **per filter** for two reasons:

- time series (ALWAYS DONE PER FILTER, right???) will be more efficient due to better clustering.
- it'll compress a bit better

If we decide to split it, we need to come up with appropriate naming convention.

Issues to be discussed:

- only low SNR or low and high SNR?
- Flux/background want INT not FLOAT.
- Ellipses only for MovingObjects. Ok?

## Source

### Identifiers and foreign keys

```

sourceId BIGINT           -- primary key
ccdExposureId BIGINT     -- corresponding CcdExposure from which this measurement was made.
                          -- Note that Source might span multiple amplifiers
filterId TINYINT         -- filter in which this measurement was made. We could get it
                          -- from exposure, kept here for efficiency reasons.
objectId BIGINT          -- pointer to Object or MovingObject
procHistoryId INTEGER    -- provenance tracking

```

## Multi-fit model parameters and covariance matrices for each model

```
-- ra/dec and their errors are present in each model
-- expressed in degrees
raPS FLOAT
raSigmaPS FLOAT
decPS FLOAT
decSigmaPS FLOAT

raSG FLOAT
raSigmaSG FLOAT
decSG FLOAT
decSigmaSG FLOAT

-- Model ellipse convolved with PSF for this Exposure, per model.
-- a and b expressed in arcsec. Theta expressed in degrees
aPS FLOAT
bPS FLOAT
thetaPS FLOAT

aSG FLOAT
bSG FLOAT
thetaSG FLOAT

-- One flux + error per model
fluxPS FLOAT, fluxSigmaPS FLOAT
fluxSG FLOAT, fluxSigmaSG FLOAT

-- Sky background, per model
skyPS FLOAT
skySG FLOAT
```

## Measured ellipse independent of Multifit model

```
a FLOAT
b FLOAT
theta FLOAT

flux FLOAT
```

## Raw measured flux independent of Multifit model

```
apFlux INTEGER
apFluxSigma INTEGER
instFlux INTEGER
instFluxSigma INTEGER
atmCorrFlux INTEGER
atmCorrFluxSigma INTEGER
```

## Photometric calibration extinction

Specific to source location and SED, but not to shape

```
extinction FLOAT
extinctionSigma FLOAT
```

## Residual characterization

```
resCharPlaceholder1 FLOAT  
resCharPlaceholder2 FLOAT  
resCharPlaceholder3 FLOAT  
resCharPlaceholder4 FLOAT  
resCharPlaceholder5 FLOAT
```

## Time related statistics

```
timeMidPoint DOUBLE  
timeRange FLOAT
```

## Flags

TBD how many flags and how to call them.

```
flag BIGINT  
isSynthetic CHAR NULL -- efficiency marker; indicates detection is synthetic
```

## Spatial indexes

```
zoneId SMALLINT  
chunkId SMALLINT  
subZoneId SMALLINT  
subChunkId SMALLINT
```

## Columns that go away

- xFlux, yFlux + errors
- raFlux, declFlux + errors
- xPeak, yPeak
- raPeak, declPeak
- xAstrom, yAstrom + errors
- raAstrom, declAstrom + errors
- psfFlux, modelFlux, petroFlux, nonGrayCorrFlux + errors
- apDia
- second moments
- snr
- chi2

Issues to be discussed:

- schema for synthetic sources of data

## DiaSource

## Identifiers and foreign keys

```
diaSourceId BIGINT NOT NULL
ampDiaExposureId BIGINT NOT NULL
diaSourceToId BIGINT NULL
filterId TINYINT NOT NULL
objectId BIGINT NULL
procHistoryId INTEGER NOT NULL
scId INTEGER NOT NULL
ssmId BIGINT
```

Where:

- diaSourceToId: Id of the corresponding diaSourceId measured in the other exposure from the same visit.
- ampDiaExposureId: point to the difference exposure from which this measurement was made
- objectId: pointer to Object or MovingObject
- scId: pointer to SourceClassif table
- ssmId: Pointer to mops\_SSM table. Might be NULL. Yields the originating SSM object for synthetic detections injected by daytime MOPS.

## Position

```
ra DOUBLE
raSigma FLOAT
decl DOUBLE
decSigma FLOAT
```

Should this be per model? If not: issue with naming this column 'dec'

## SS model predicted ellipse

Only for moving objects, otherwise null

```
-- a and b expressed in arcsec. Theta expressed in degrees
aSS FLOAT
bSS FLOAT
thetaSS FLOAT
```

## Measured ellipse

```
a FLOAT
b FLOAT
theta FLOAT
```

## Raw measured flux

```
flux DOUBLE
fluxSigma FLOAT
```

## Photometric calibration extinction

Specific to source location and SED, but not to shape

```
extinction FLOAT
extinctionSigma FLOAT
```

## Time related statistics

```
timeMidPoint DOUBLE
timeRange FLOAT
```

## Flags

TBD how many flags and how to call them.

```
flag BIGINT
isSynthetic CHAR NULL -- efficiency marker; indicates detection is synthetic
```

## Spatial indexes

```
zoneId SMALLINT
chunkId SMALLINT
subZoneId SMALLINT
subChunkId SMALLINT
```

## Columns to be removed

```
raFlux DOUBLE NULL,
raFluxSigma FLOAT NULL,
declFlux DOUBLE NULL,
declFluxSigma FLOAT NULL,
raPeak DOUBLE NULL,
declPeak DOUBLE NULL,
xAstrom DOUBLE NULL,
xAstromSigma FLOAT NULL,
yAstrom DOUBLE NULL,
yAstromSigma FLOAT NULL,
raAstrom DOUBLE NULL,
raAstromSigma FLOAT NULL,
declAstrom DOUBLE NULL,
declAstromSigma FLOAT NULL,
taiMidPoint DOUBLE NOT NULL,
taiRange FLOAT NOT NULL,
lengthDeg DOUBLE NOT NULL,
psfFlux DOUBLE NOT NULL,
psfFluxSigma FLOAT NOT NULL,
apFlux DOUBLE NOT NULL,
apFluxSigma FLOAT NOT NULL,
modelFlux DOUBLE NOT NULL,
modelFluxSigma FLOAT NULL,
instFlux DOUBLE NOT NULL,
instFluxSigma FLOAT NOT NULL,
nonGrayCorrFlux DOUBLE NULL,
nonGrayCorrFluxSigma FLOAT NULL,
```

```
atmCorrFlux DOUBLE NULL,  
atmCorrFluxSigma FLOAT NULL,  
apDia FLOAT NULL,  
refMag FLOAT NULL,  
xFlux, yFlux, xPeak, yPeak (replaced by flux)  
timeMidPoint DOUBLE,  
timeRange FLOAT,  
Ixx FLOAT NULL,  
IxxSigma FLOAT NULL,  
Iyy FLOAT NULL,  
IyySigma FLOAT NULL,  
Ixy FLOAT NULL,  
IxySigma FLOAT NULL,  
snr FLOAT NOT NULL,  
chi2 FLOAT NOT NULL,  
valx1 DOUBLE NOT NULL,  
valx2 DOUBLE NOT NULL,  
valy1 DOUBLE NOT NULL,  
valy2 DOUBLE NOT NULL,  
valxy DOUBLE NOT NULL,  
obsCode CHAR(3) NULL,  
mopsStatus CHAR(1) NULL,  
flagForAssociation SMALLINT NULL,  
flagForDetection SMALLINT NULL,  
flagForWcs SMALLINT NULL,  
flagClassification BIGINT NULL,
```

Issues to be discussed:

- pairing diasources from the same visit

## WcsSource?

See [?WcsSource in DC3a](#)