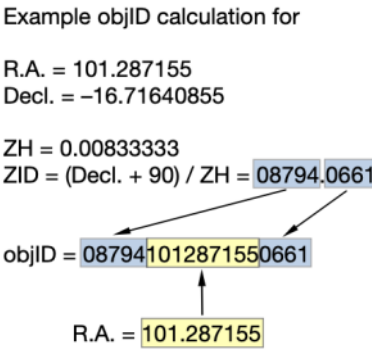
PS1 Object Identifiers

The PS1 object identifiers (`objID` in the database tables) are computed from the object positions. We do not recommend trying to extract the positions directly from the `objID` because subsequent calibrations have significantly improved the positional accuracy of the RA and Dec columns. However, the `objID` is the primary key in most PS1 tables, and it is useful to know how the `objID` is related to position. Partitioning the table by `objID` approximately divides the table into declination strips.

Contents



How the object identifier is computed from RA and Dec

The `objID` index is derived from right ascension and declination. While it is possible to calculate the RA and Dec from the `objID`, it is not recommended to do this, because the `objID` is based on the astrometric solutions from individual exposures and stacks as they are ingested during an early phase of the PS1 data processing. Consequently the positions derived from the `objID` have not been calibrated against 2MASS or Gaia. It is recommended to use `raMean`, `decMean` from ObjectThin. Included below is the C code for the translation between R.A. and Dec., for users interested in the relationship.

C code to compute objID

```
uint64_t CreatePSPSObjectID(double ra, double dec)
{
double zh = 0.0083333;
double zid = (dec + 90.) / zh; // 0 - 180*60*2 = 21600 < 15 bits
int izone = (int) floor(zid);
double zresid = zid - ((float) izone); // 0.0 - 1.0
uint64 t part1, part2, part3;
part1 = (uint64 t)( izone * 10000000000000LL);
part2 = ((uint64 t)(ra * 1000000.)) * 10000 ; // 0 - 360*1e6 = 3.6e8 (< 29
bits)
part3 = (int) (zresid * 10000.0) ; // 0 - 10000 (1 bit == 30/10000 arcsec)
(< 14 bits)
return part1 + part2 + part3;
}
```

The figure below (adapted from a similar figure from Heather Flewelling) shows an example of computing the `objID` from specific RA and Dec values.