`mirrorstats` Group

John D. Baker

November 29, 2022

Contents

mirrorstats Overview

mirrorstats is a J script that queries the SQLite mirror.db database file built by MirrorXref.

mirror.db cross references local image files with online https://conceptcontrol.smugmug.com/ versions. Local images are catalogued in ThumbsPlus SQLite databases. mirror.db matches local ThumbsPlus image and path keys with online keys.

mirrorstats Interface

```
NotDivisible        [7]  albums with image counts that are not divisible by 3 and 5
albdist             [8]  all mean album distances km from position (x)
fsd                 [14] fetch sqlite dictionary array
fst                 [14] fetch sqlite reads table
gpsextremesgallery  [15] list images with gps extremes
meanalbdist         [17] mean km distance of geotagged album images from (x)
```

Using mirrorstats

mirrorstats is typically used from the Jupyter notebook Mirror SmugMug Statistics.ipynb.
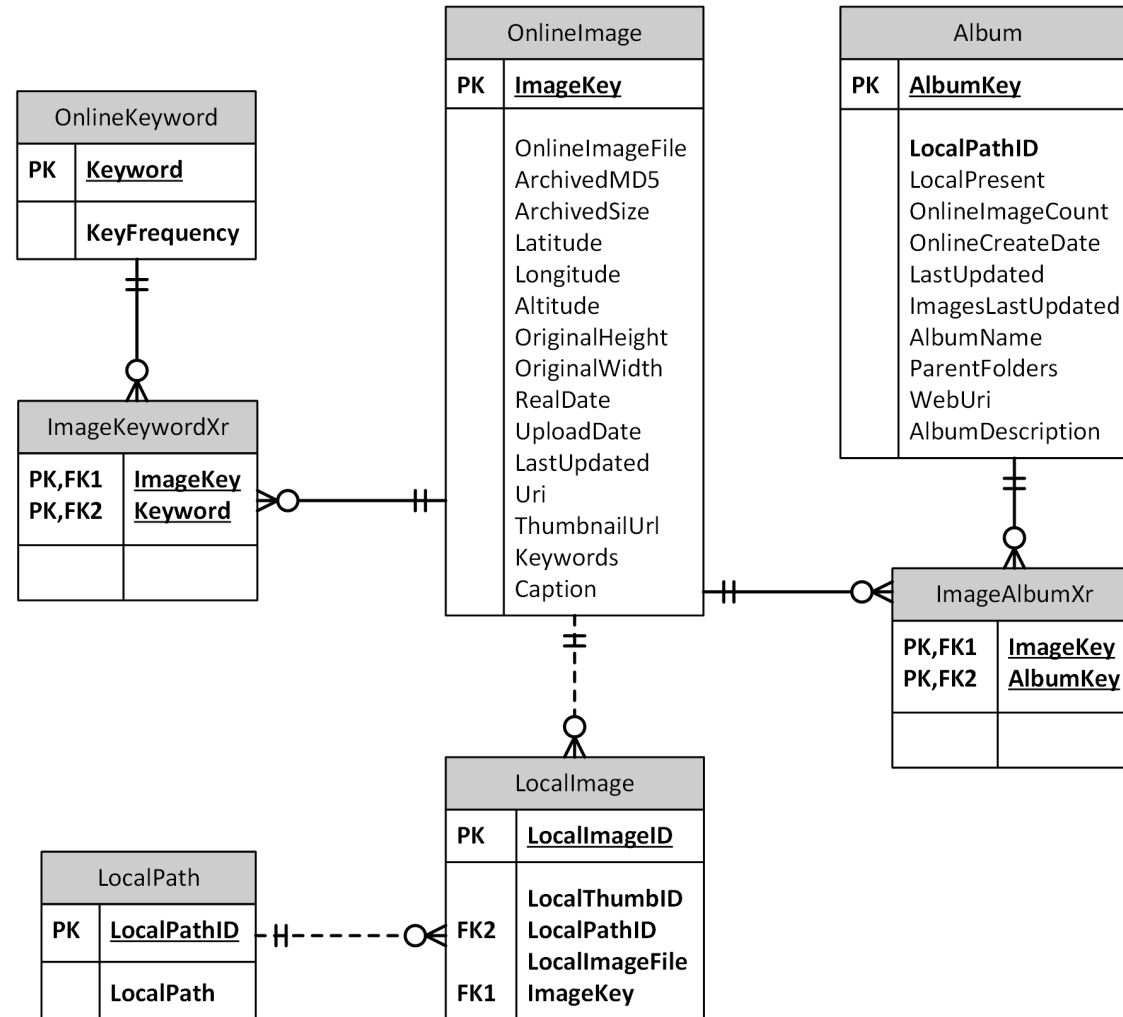
mirror.db Schema

Figure 1: `mirror.db` schema.

mirrorstats Source Code

```
NB.*mirrorstats s-- utils for querying local SmugMug mirror_db metadata.
NB.
NB. verbatim:
NB.
NB. interface word(s):
NB. ---------------------------------------------------------------------
NB.  NotDivisible       - albums with image counts that are not divisible by 3 and 5
NB.  albdist            - all mean album distances km from position (x)
NB.  fsd                - fetch sqlite dictionary array
NB.  fst                - fetch sqlite reads table
NB.  gpsextremesgallery - list images with gps extremes
NB.  meanalbdist        - mean km distance of geotagged album images from (x)
NB.
NB. created: 2022jul07
NB. ---------------------------------------------------------------------
NB. 2022nov29 (gpsextremesgallery) added

require 'data/sqlite'

coclass 'mirrorstats'

NB.*dependents
NB. (*)=: AlbumImageCount_sql GeotaggedAlbumImages_sql
NB.*enddependents
```

```
NB. NOTE: the j sqlite addon is fussy about how sql is formatted.
NB. Running standard sql pretty printers or indenting sql in your
NB. favorite style is likely to produce code that doesn't work.

AlbumImageCount_sql=: (0 : 0)
select count(1) as ImageCnt, a.AlbumKey, b.AlbumName from ImageAlbumXr a
inner join Album b on a.AlbumKey=b.AlbumKey group by a.AlbumKey
)

GeotaggedAlbumImages_sql=: (0 : 0)
select AlbumName, OnlineImageFile, Latitude, Longitude from OnlineImage a
inner join ImageAlbumXr b on a.ImageKey = b.ImageKey
inner join Album c on b.AlbumKey = c.AlbumKey
where (not (a.Latitude = 0 and a.Longitude = 0))
and c.AlbumName like
)


NB.*end-header

ALTMIRRORDBPATH=: 'c:/SmugMirror/Documents/XrefDb/'

NB. interface words (IFACEWORDSmirrorstats) group
IFACEWORDSmirrorstats=: <;._1 ' NotDivisible albdist fsd fst gpsextremesgallery meanalbdist'

NB. sqlite mirror database file name
MIRRORDB=: 'mirror.db'
```

```
NB. mirror database path
MIRRORDBPATH=: 'c:/smugmirror/documents/xrefdb/'


NB. home longitude and latitude, west longitudes +, north latitudes +
MeeusHomeLonLat=: 0 0


NB. root words (ROOTWORDSmirrorstats) group
ROOTWORDSmirrorstats=: <;._1 ' IFACEWORDSmirrorstats NotDivisible ROOTWORDSmirrorstats VMDmirrorstats albdi
>..>st albextent dstat freq fsd fst histogram2 itYMDhms ofreq portchars read'



NB. version, make count and date
VMDmirrorstats=: '0.5.0';15;'29 Nov 2022 17:21:08'


AlbumImageCount=: 3 : 0

NB.*AlbumImageCount v-- execute (AlbumImageCount_sql) query.
NB.
NB. monad:  bt =. AlbumImageCount clMirrorDb
NB.
NB.   AlbumImageCount ALTMIRRORDBPATH,MIRRORDB

NB. get album image counts and names
AlbumImageCount_sql fsd y
)
```

```
NotDivisible=: 3 : 0
```

```
NB.*NotDivisible  v--  albums with  image  counts  that  are  not
NB. divisible by 3 and 5.
NB.
NB. This  verb  finds  albums  with  image  counts that  are  not
NB. divisibe by  3 and 5. This weird requirement was motivated by
NB. how the SmugMug iPhone App displays galleries. It breaks  the
NB. images into rows  of three or five. I  don't  like incomplete
NB. terminal rows.
NB.
NB. monad:  bt =. NotDivisible clMirrorDb
NB.
NB.   trg=. 'c:/smugmirror/documents/xrefdb/mirror.db'
NB.   NotDivisible trg
NB.
NB. dyad:   bt =. ia NotDivisible clMirrorDb
NB.
NB.   4 NotDivisible trg
```

```
3 5 NotDivisible y
:
NB. !(*)=. ImageCnt AlbumName
(0 {"1 d)=. 1 {"1 d=. AlbumImageCount y
```

```
NB. works for integer (x) without common divisors
'common factor(s)' assert -.pwcf x
```

```
a=. x , */x

b=. *./ 0 < r=. a |"0 1 ImageCnt
c=. (a *"1 >. ImageCnt %"0 1 a) - ImageCnt
c=. (/: |."1 b # c) { (<"0 b # ImageCnt,.c) ,. b # AlbumName
c ,~ '[Count]';(<"0 a),<'[Album Name]'
)

albdist=: 3 : 0

NB.*albdist v-- all mean album distances km from position (x).
NB.
NB. monad:  bt =. albdist uuIgnore
NB.
NB.    albdist 0
NB.
NB. dyad:  bt =. flLonLat albdist uuIgnore
NB.
NB.    0 0 albdist 0  NB. distances km from lb origin

MeeusHomeLonLat albdist 0
:
a=. nonemptyalbums 0
d=. x&meanalbdist&> 1 {"1 a
((<"0 b#d) ,. b # 1 {"1 a) [ b=. 0 < d
)

NB. mean km distance of geotagged album images from album "centroid"
```

```
albextent=: meanalblonlat meanalbdist ]


antimode=: 3 : 0


NB.*antimode v-- finds the least frequently occurring  item(s) in
NB. a list.
NB.
NB. monad:  ul =. antimode ul
NB.
NB.   antimode ?.500#100
NB.   antimode ;:'blah blah blah yada yada wisdom'



if. 0 < # y =. ,y do.     NB. no antimodes for null lists
  f =. #/.~ y              NB. nub frequency
  (~. y) #~ f e. <./ f   NB. lowest frequency items
else. y
end.
)


NB. arc tangent
arctan=: _3&o.


NB. signal with optional message
assert=: 0 0"_ $ 13!:8^:((0: e. ])`(12"_))


NB. retains string before first occurrence of (x)
beforestr=: ] {.~ 1&(i.~)@([ E. ])
```

```
charsub=: 4 : 0
```

```
NB.*charsub v-- single character pair replacements.
NB.
NB. dyad:  clPairs charsub cu
NB.
NB.   '-_$ ' charsub '$123 -456 -789'
```

```
'f t'=. ((#x)$0 1)<@,&a./.x
t {~ f i. y
)
```

```
NB. cosine radians
cos=: 2&o.
```

```
NB. double quotes - doubles internal " quotes like (quote)
dbquote=: '"'&,@(,&'"')@(#~ >:@(=&'"'))
```

```
NB. deviation about mean
dev=: -"_1 _ mean
```

```
dstat=: 3 : 0
```

```
NB.*dstat v-- descriptive statistics
NB.
NB. monad: ct =. dstal nl
NB.
```

```
NB.    dstat   ?.1000#100
NB.
NB. dyad:   ct =.   faRound dstat nl
NB.
NB.    0.1 dstat   ?.1000#100


0.0001 dstat y
:
t=. '/sample size/minimum/maximum/1st quartile/2nd quartile/3rd quartile/first mode'
t=. t , '/first antimode/mean/std devn/skewness/kurtosis'
min=. <./
max=. >./
t=. ,&': ' ;._1 t
v=. $,min,max,q1,median,q3,({.@mode2),({.@antimode),mean,stddev,skewness,kurtosis
t,. ": x round ,. v , y
)


earthdist=: 4 : 0


NB.*earthdist v-- distance in km between n points on the Earth's surface.
NB.
NB. dyad:   (fl | ft) earthdist (fl | ft)
NB.
NB.    NB. Paris longitude, latitude
NB.    NB. ddfrdms computes decimal degrees from degree, minutes, seconds
NB.    l1     =. ddfrdms _2 _20 _14    NB.  2d 20m 14s (East)
NB.    theta1 =. ddfrdms 48 50 11      NB. 48d 40m 11s (North)
NB.
```

```
NB.    NB. Washington
NB.    l2      =. ddfrdms 77 3 56       NB. 77d  3m 56s (West)
NB.    theta2 =. ddfrdms 38 55 17       NB. 38d 55m 17s (North)
NB.
NB.    NB. rounded to 2 decimals matches Meeus
NB.    6181.63 = ". '0.2' 8!:2 (l1,theta1) earthdist l2,theta2
NB.
NB.    NB. table arguments
NB.    (|: 5 # ,: l1,theta1) earthdist |: 5 # ,: l2,theta2


a=.  6378.14      NB. Earth's mean radius (km)
fl=. % 298.257    NB. Earth's flattening (a * 1 - fl) is polar radius


NB. zero distances mask
b=.  *./ x = y


NB. longitudes and latitudes in decimal degrees
NB. western longitudes +, northern latitudes +
NB. (*)=. l1 l2 theta1 theta2
'l1 theta1'=.  x [ 'l2 theta2'=. y


f=.      rfd -: theta1 + theta2
g=.      rfd -: theta1 - theta2
lambda=. rfd -: l1 - l2


sqrsin=. *: @ sin
sqrcos=. *: @ cos
```

```
sinlam=.  sqrsin lambda [ coslam=. sqrcos lambda
sqrcosg=. sqrcos g [ sqrsing=. sqrsin g
sqrsinf=. sqrsin f [ sqrcosf=. sqrcos f

s=. (coslam * sqrsing) + sinlam * sqrcosf
c=. (coslam * sqrcosg) + sinlam * sqrsinf

omega=. arctan %: s % c
r3=. 3 * (%: s * c) % omega
d=.  +: omega * a
h1=. (<: r3) % +: c
h2=. (>: r3) % +: s

NB. required distance
d=. d * (>: fl*h1*sqrsinf*sqrcosg) - fl*h2*sqrcosf*sqrsing

NB. handle any zero distances
if. +./ b do.
  NB. cannot do b*d as d is undefined _. for zero distances
  if. #$ d do. 0 (I. b)} d elseif. b do. 0 elseif. 1 do. d end.
else.
  d
end.
)

NB. frequency distribution
freq=: ~. ; #/.~
```

```
fsd=: 4 : 0

NB.*fsd v-- fetch sqlite dictionary array.
NB.
NB. dyad:  clSql fsd clDb
NB.
NB.   trg=. 'c:/smugmirror/documents/xrefdb/mirror.db'
NB.   sql=. 'select ImageKey, OriginalWidth, OriginalHeight, OnlineImageFile, Keywords from OnlineImage'
NB.   sql fsd trg

NB. require 'data/sqlite' !(*)=. sqlclose__db sqldict__db sqlopen_psqlite_
d [ sqlclose__db '' [ d=. sqldict__db x [ db=. sqlopen_psqlite_ y
)


fst=: 4 : 0

NB.*fst v-- fetch sqlite reads table.
NB.
NB. dyad:  bt =. clSql fst clDb
NB.
NB.   trg=. 'c:/smugmirror/documents/xrefdb/mirror.db'
NB.   sql=. 'select ImageKey, OriginalWidth, OriginalHeight, OnlineImageFile, Keywords from OnlineImage'
NB.   sql fst trg

NB. require 'data/sqlite' !(*)=. sqlclose__db sqlreads__db sqlopen_psqlite_
d [ sqlclose__db '' [ d=. sqlreads__db x [ db=. sqlopen_psqlite_ y
)
```

```
gpsextremesgallery=: 3 : 0

NB.*gpsextremesgallery v-- list images with gps extremes.
NB.
NB. monad:  gpsextremesgallery clDb
NB.
NB. trg=. 'c:/smugmirror/documents/xrefdb/mirror.db'
NB. gpsextremesgallery trg
NB.
NB. dyad:  iaN gpsextremesgallery clDb

4 gpsextremesgallery y
:
NB. overide mixed assignments (<:)=:
sql=. 'select ImageKey, OnlineImageFile, Latitude, Longitude, Altitude from OnlineImage'
({."1 r)= {:"1 r=. sql fsd y

NB. geotagged images !(*)=. Altitude ImageKey Latitude Longitude OnlineImageFile
bgt=. (0 ~: Longitude) +. 0 ~: Latitude
lba=. bgt #  Latitude ,. Longitude ,. Altitude
gti=. (bgt # ImageKey ,. OnlineImageFile) ,. <"1 lba

NB. distance from lb origin 0 0
dst=. 0 0 earthdist |: 1 _1  (*"1) 0 1 {"1 lba
gti=. gti ,. <"0 dst

NB. images near and far from origin
```

```
nf=. ,/ (x,-x) {.&> < gti {~ /: dst
```

```
NB. highest elevations - altitudes in db do not indicate
NB. above or below sea level - not collected in metadata
NB. rare below sea level images manually managed for now
he=. x {. gti {~ \: {:"1 lba
```

```
NB. images near and far equator
ord=. /: | 0 {"1 lba
ei=. (x {. ord{gti) , (-x) {. ord{gti
```

```
NB. images near and far prime meridian
ord=. /: | 1 {"1 lba
pm=. (x {. ord{gti) , (-x) {. ord{gti
```

```
NB. unique images by origin distance
gti=. ~. nf,he,ei,pm
gti {~ /: 3 {"1 gti
)
```

```
NB. variation on (histogram) uses left open intervals (xi, xi+1]
histogram2=: <:@(#/.~)@(i.@>:@#@[ , |.@[ (#@[ - I.) ])
```

```
NB. sqlite iso character timestamps to Y M D h m s table - ignores timezones
itYMDhms=: [: _1&".&> '- T : '&charsub@('+'&beforestr)&.>
```

```
NB. kurtosis
kurtosis=: # * +/@(^&4)@dev % *:@ssdev
```

```
NB. mean value of a list
mean=: +/ % #


meanalbdist=: 3 : 0


NB.*meanalbdist v-- mean km  distance  of geotagged album  images
NB. from (x).
NB.
NB. monad:  fa =. meanalbdist clAlbumName
NB.
NB.   meanalbdist 'Weekenders'    NB. has geotagged images
NB.   meanalbdist 'Alpha Layered' NB. no geotagged images - 0 result
NB.
NB. dyad: fa =. flLonLat meanalbdist clAlbumName
NB.
NB.   NB. mean distance from lb origin off west Africa
NB.   0 0 meanalbdist 'Ghana 1970''s'


MeeusHomeLonLat meanalbdist y
:
db=. ALTMIRRORDBPATH mirrorcn 0
(;0{r)=. ;1{r=. sqlread__db GeotaggedAlbumImages_sql,' ' ,dbquote y
NB. !(*)=. Longitude Latitude
if. #Longitude do. mean x earthdist (-Longitude) ,: Latitude else. 0 end.
)


meanalblonlat=: 3 : 0
```

```
NB.*meanalblonlat v-- mean  longitude and  latitude of  geotagged
NB. album images.
NB.
NB. The  point computed  is roughly  the "centroid" of  geotagged
NB. album images. Uses Meeus  conventions:  western longitudes +,
NB. northern latitudes +.
NB.
NB. monad:  meanalblonlat clAlbumName

db=. ALTMIRRORDBPATH mirrorcn 0
(;0{r)=. ;1{r=. sqlread__db GeotaggedAlbumImages_sql,' ' ,dbquote y
NB. !(*)=. Longitude Latitude
if. #Longitude do. mean"1 (-Longitude) ,: Latitude else. 0 0 end.
)


NB. median value of a list
median=: -:@(+/)@((<. , >.)@midpt {  /:~) ::_:


NB. mid-point
midpt=: -:@<:@#


mirrorcn=: 3 : 0


NB.*mirrorcn v-- connect to mirror database.
NB.
NB. monad:  ba =. mirrorcn uuIgnore
```

```
MIRRORDBPATH mirrorcn 0
:
NB. require 'data/sqlite' !(*)=. sqlopen_psqlite_
sqlopen_psqlite_ x,MIRRORDB
)


mode2=: 3 : 0

NB.*mode2 v-- finds   the  most frequently occurring item(s) in  a
NB. list.
NB.
NB. monad:  ul =. mode2 ul
NB.
NB.   mode2 ?.500#100
NB.   mode2 ;:'I do what I do because I am what I am'

if. 0 < # y =. ,y do.     NB. null lists have no modes
  f =. #/.~ y              NB. nub frequency
  (~. y) #~ f e. >./ f     NB. highest frequency items
else. y
end.
)


nonemptyalbums=: 3 : 0

NB.*nonemptyalbums v-- nonempty albums.
NB.
```

```
NB. monad:  bt =. nonemptyalbums uuIgnore

db=. ALTMIRRORDBPATH mirrorcn 0
(;0{r)=. ;1{r=. sqlread__db AlbumImageCount_sql
NB. !(*)=. ImageCnt AlbumName
(<"0 b#ImageCnt) ,. b#AlbumName [ b=. 0 < ImageCnt
)


NB. like (freq) but results in descending frequency
ofreq=: [: ((([: < [: \: [: ; 1 {  ]) { &.> ]) ~. ; #/.~


NB. portable box drawing characters
portchars=: [: 9!:7 '+++++++++|-'"_ [ ]


NB. 1 if (x) has at least one pair with common factor(s) - see long document
pwcf=: 1 < [: >./ [: +/ [: +./@e.&>/~ 0 -.&.>~ [: <"1 [: ~."1 q:


NB. first quartile
q1=: median@((median > ]) # ]) ::_:


NB. third quartile
q3=: median@((median < ]) # ]) ::_:


NB. reads a file as a list of bytes
read=: 1!:1&(]`<@.(32&>@(3!:0)))
```

```
NB. radians from degrees
rfd=: *&0.0174532925199432955


NB. round (y) to nearest (x) (e.g. 1000 round 12345)
round=: [ * [: (<.) 0.5 + %~


NB. sine radians
sin=: 1&o.


NB. skewness
skewness=: %:@# * +/@(^&3)@dev % ^&1.5@ssdev


NB. sum of square deviations (2)
ssdev=: +/@:*:@dev


NB. standard deviation (alternate spelling)
stddev=: %:@:var


NB. var
var=: ssdev % <:@#


NB.POST_mirrorstats post processor.

smoutput IFACE=:  (0 : 0)
NB. (mirrorstats) interface word(s): 20221129j172108
NB. ----------------------------
```

```
NB. NotDivisible       NB. albums with image counts that are not divisible by 3 and 5
NB. albdist            NB. all mean album distances km from position (x)
NB. fsd                NB. fetch sqlite dictionary array
NB. fst                NB. fetch sqlite reads table
NB. gpsextremesgallery  NB. list images with gps extremes
NB. meanalbdist        NB. mean km distance of geotagged album images from (x)
)

cocurrent 'base'
coinsert  'mirrorstats'
```

Index