

Running main.py as it is currently written, passing @ for event in self.events
 It starts by processing MER & log files associated w/ P-09.
 $(\sim L41S)$

First P-09 MER file w/ events: 09-SB6F01F6.MER

dives.py, ~L41S

self.mind-name = '09-SB6F01F6.MER' using merlog2date &
 self.log-name = '09-SB69BE96.LOG' (2018-08-07T15:45:26Z)

FINDING: log ^{name} ~4 days earlier than .MER

L230 of log: 37853 bytes written to 09/SB6F01F6.MER @
 epoch = 183441755 \Rightarrow (2018-08-11T15:35:55Z)

1st line epoch of log file = 2018-08-07T15:4526Z

last line epoch of log file = 2018-08-11T15:36:00Z

so 1: log file name = hex epoch time of its first line,
 (verified w/ real log: 09-SB6F0260.LOG)

2: log file starts before first GPS in MER file

3: Every dive grants 1 log & 1 MER (w/ overflow ext. off, etc.)
 and contains GPS before & after dive.
 \Rightarrow log name = before dive; MER name = after dive

4: "GPSACK" in LOG \rightarrow GPSINFO DATE in MER \rightarrow
 nearly same location printed in ~~MER~~ log file

i.e., the same GPS fixes are printed in the log and
 MER; a few before the dive & @ last one after

Three GPS times in LOG

① 2018-08-07T15:46:00Z

② 2018-08-07T15:51:38Z

③ 2018-08-11T15:33:41Z

↑

epoch times of "GPSACK"

Three GPS times in MER

2018-08-07T15:46:01Z

2018-08-07T15:51:39Z

↓

2018-08-11T15:33:48Z

↑ "GPSINFO DATE" all
 when seconds of "GPSACK"
 in corresponding log file

log name = start of dive
 MER name = end of dive

automated v1.0.0

... back^{to} drives.py -> compute_events_station_locator(self, next_dive)
current dive (b06)

concerning self, the current dive (b06)

self.gps_list.date = these times in MER file

next_dive = the Dive instance associated w/ 09-5B6F0260.L06;
what is the next log in "switching to..."

next_dive.gps_list.date = the times in the corresponding 09-5B77394A.MER
next_dive.mond_name

3 minutes after last GPS
of previous dive → = 2018-08-11T15:36:38Z
gps-before-dive = 2018-08-11T15:46:55Z
gps-after-dive = 2018-08-11T15:50:13Z } next-dive.gps-list.date
= 2018-08-11T21:08:00Z

FINDING: The first GPS of a MER file is ~ a few minutes
after the last GPS of the previous MER file (they
are not the same)

∴ gps-before-dive = all GPS in MER file except final

gps-after-dive = the final GPS fix in a MER file

* by default it uses the GPS in the MER, but
they also exist @ nearly same time in log

Q We know the "GPSACK" in b06 and "GPSINFO DATE" in MER
~~one~~ one run in time; are they equally the same location?

A The GPS locations in MER & b06 differ slightly both in time
& location; one is written to the b06 first & then another (different)
one is written to the MER.

FINDING: the first GPS in a MER file is after the dive
(all others before)

automated VI.0.0

... still @ for event in self.event: in dws.py (~L415)

There are two events associated w/ this file instance →

- the MER file has two (2) <EVENT>...INFO DATE...<EVENT>
data blocks.

Finding: in Dws.py, self compiles the LOB & MER
in Events.py, self compiles the data in the MER file

self.event[0].date = 2013-08-11T09:48:52Z

self.event[1].date = 2013-08-10T05:59:38Z

↑ these times have been adjusted from their
trigger times (INFO DATE), and they have
had a linear dark-drift corrector applied

In the parlance of automated → • LOB = dws

• MER = event

The next page details the gps interpolation to generate
the first 500 files for 109 using: 09-SB69BE96.LOB +
09⁵SB6F01F6.MER

The drift is broken into three separate regions:

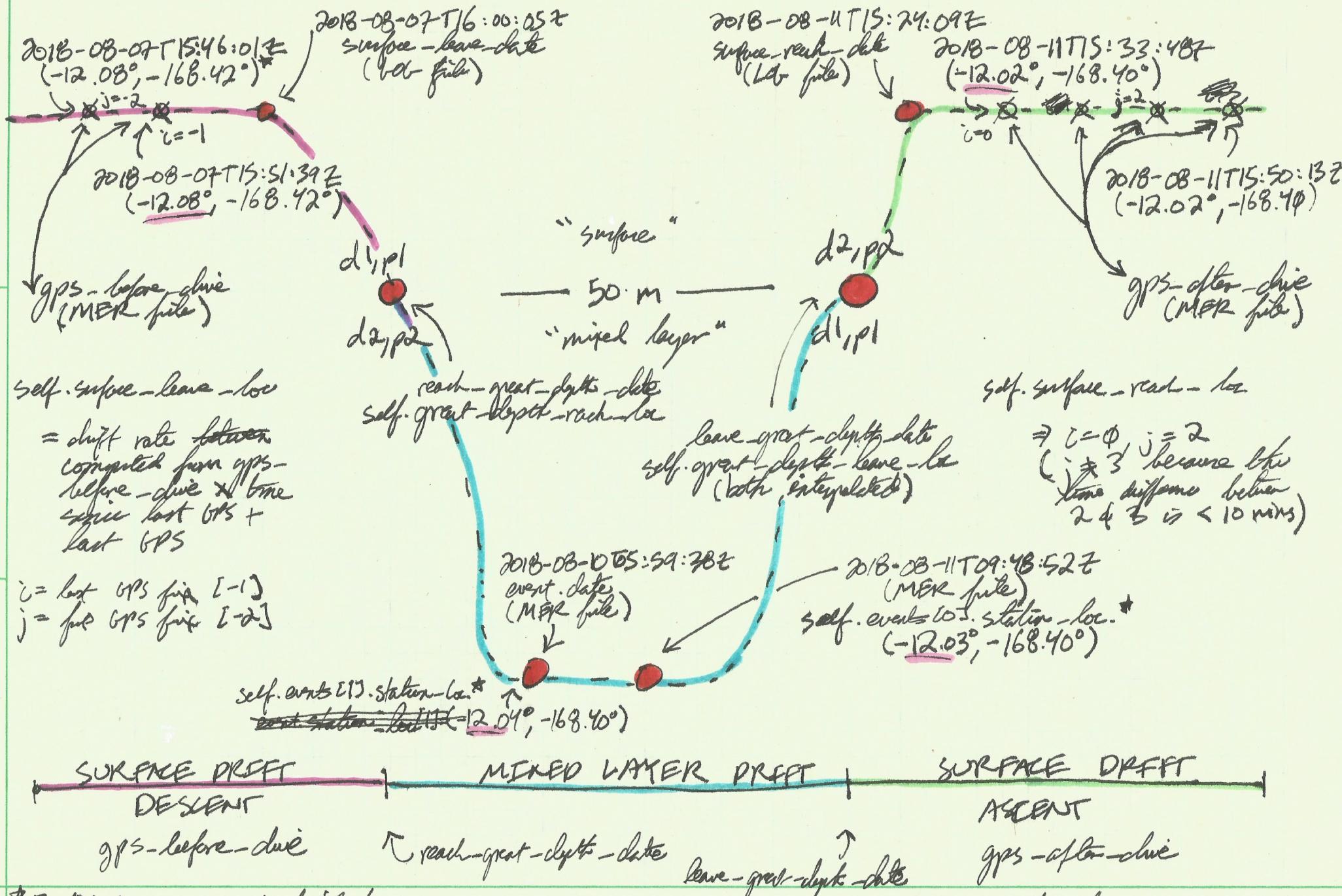
① surface descent

② mixed layer (deep) drift

③ surface ascent

w/ locations interpolated w/in each layer.

automated v1.0.0: Dir. compute - events - station - location \Rightarrow 09-5B6F01F6.MER



* I didn't print enough digits to see a difference, but they are different
(when the #'s are rounded, not truncated)

⊗ = GPS fix from LOB or MER
● = interpolated location and/or date

addendum

21-Sep-2020

[1] Both the GPS times & latitudes are different in MER & LOB:

LOB:

1533656760 & GPSACK ← NB: In automated < v1.0.0 (and maybe
Unix epoch time
= 2018-08-07T15:46:00Z

+
... a few hrs later...

later versions, too) the GPS fix
that is printed a few hrs later
in the LOB has this "gps acquisition"
time as its corresponding time

S12 deg 04.807 min, W168 deg 25.245 min

⇒

MER:

2018-08-07T15:46:01Z LAT = -120 4.8090 lon = -168 25.2490

NB: In automated < v1.0.0 (and maybe later versions, too) LOB
parses the lats to be in the form of degrees & minutes,
where the first two integers are degrees & the rest are ~~degrees~~
degrees

In the same format as by: S12deg04.8090min, W168deg25.2450min

So considering the first GPS point in both:

LOB:

diff → 2018-08-07T15:46:00Z
diff → S12deg 04.807 min,
W168 deg 25.245 min

MER:

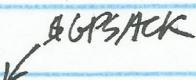
2018-08-07T15:46:01Z
S12deg 04.809 min,
W168deg 25.245 min

at/a2

[2] The locations in MER files are degrees-minutes, ~~not~~
not decimal degrees, where first two integers are degrees

$$\text{LAT} = -1204.809 \Rightarrow 51^{\circ}24'04.809\text{ mn}$$

[3] The locations in the ~~.log~~ files are the same as .vbt, but
the time in the VBT ~ minutes later

LOG:  GPSACK

2018-08-07T15:46:00Z
(same location)

VBT

2018-08-07T15:47:02 (+1 min)
(same location)

Summary, as best as I can reason

- (1) GPS ACK in LOG
- + SECS (2) GPS ACK in MER (different from (1))
- + TMINS (3) The GPS location in .log ~~writes to .vbt~~, @ a later time
NB, the later time does not equal the epoch time of the
"51^o24'04.807, W168^o25.245mn" in .log; still later
- (4) Not all the GPS fixes are written to ~~.log~~ .vbt

∴ vbt+bl.m only records some of the GPS fixes, and further,
the times recorded there are delayed w.r.t. those in LOG

By default, automated L2V1.0.0 uses the GPS fixes in .MER

[4] → there are more, and more accurate, GPS fixes in log/MER than .vbt.