DAC Argo Trajectory Workshop

Hamburg, Germany
December 1 -2, 2017

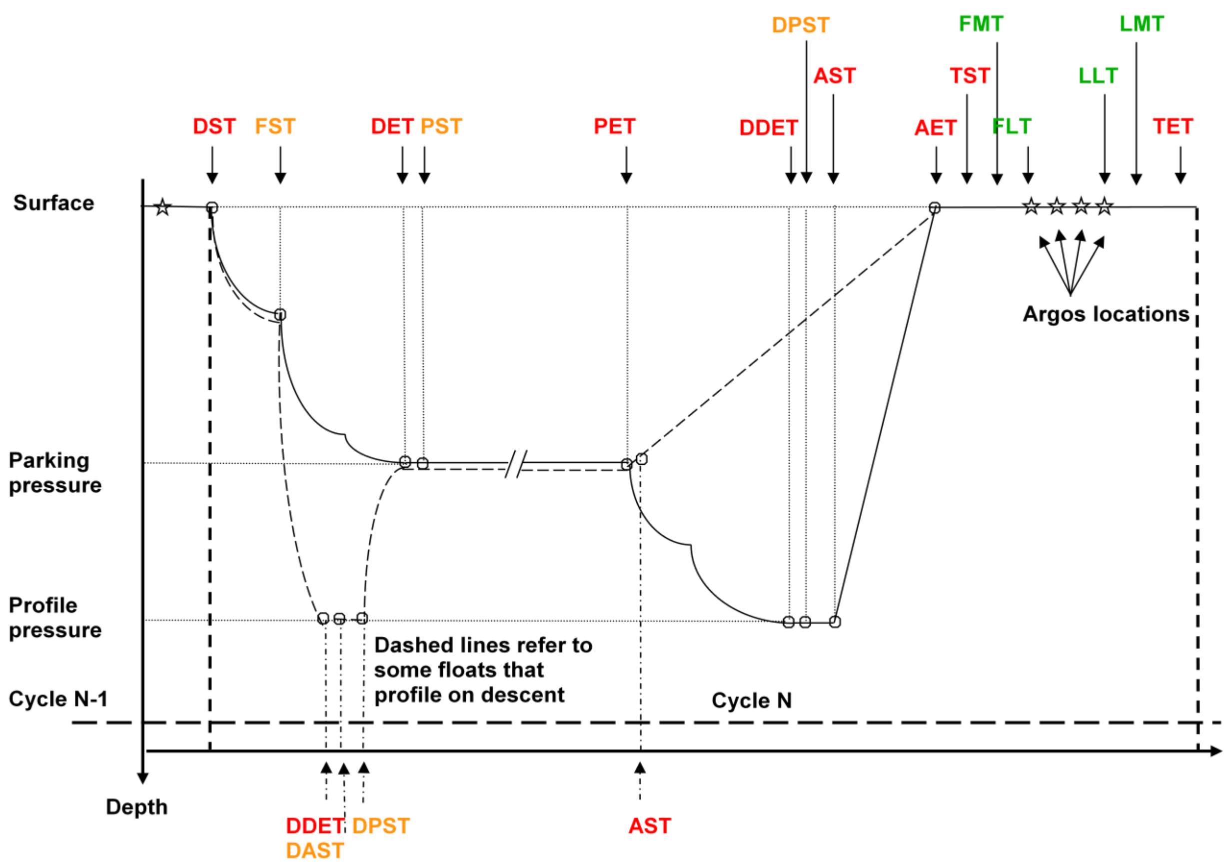
Contents

1. Welcome and Overview of Workshop goals and review of trajectory file

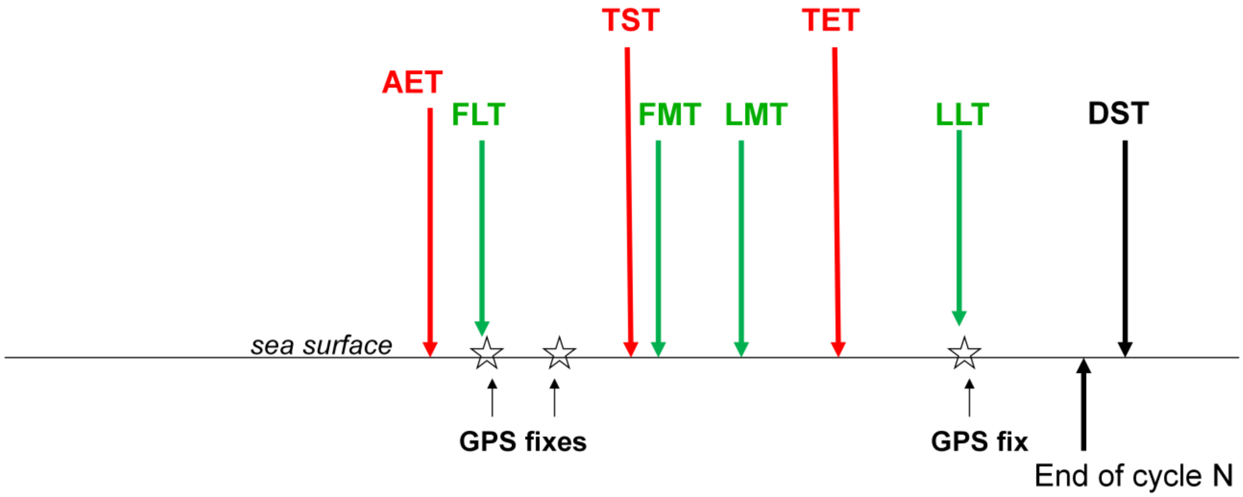
M. Scanderbeg gave a presentation with some background on the way the trajectory v3.1 file format was created, some instructions on how to populate the files and goals for the workshop. The v3.1 file format was created in response to work done by M. Ollitrault and J-P Rannou to estimate velocities based on the trajectory files. Their work helped Argo understand what information might be missing in the trajectory files. Therefore, the need to add more cycle timing information and indicate where, and how, in the cycle measurements were taken was also a motivator for creating v3.1 trajectory files. Finally, it was hoped to create a format that fit into the Argo 'standard' practices with respect to retaining raw data and allowing for adjustments in both real time and delayed mode.

File format v3.1 included the identification, and sometimes addition, of key cycle timing events that Argo would like measured to enable the estimation of drift velocities. Measurement codes were added to indicate what type of measurement was taken and when during the cycle. Additional variables were added to help indicate how JULD timing information was obtained and to allow for estimated times, including clock drift adjustments. Positioning system errors and satellite name variables were added, but these are very Argos centric.

M. Scanderbeg showed a diagram of the cycle timing variables with primary and secondary measurement codes highlighted in red and orange accordingly. Green times indicate surface times.



She noted that this diagram is accurate for Argos floats, but not for Iridium/GPS floats. Below is a more accurate diagram of Iridium floats on the surface.



Next, it was stated that events are to be arranged in the order they occurred, not in ascending measurement code (MC) order or ascending JULD. For JULD, this is partly because there are two sources of timing in JULD: the floats themselves and the satellites. Therefore, depending on the float's clock drift, JULD may not be in chronological order, as we assume that the satellite clocks are accurate. If no time is associated with an event, it should be placed between the event prior and the event following it with FillValue in JULD and '9' for JULD_STATUS indicating that it may be possible to estimate the time at a later date.

All Primary and Secondary MC events that are experienced by the float are required to be present in the N_MEASUREMENT array and redundantly in the N_CYCLE variables. All other codes are voluntary. For example, many float types have a shallow cycle immediately after deployment that does not have a drift phase. In this case, it is not necessary to include MCs concerned with the drift phase.

The MC Table was explained with MC codes ranging from 0 – 800. The Primary codes are mandatory and are (100, 200, 300, 400, 500, 600, 700 and 800). The Secondary codes are suggested, float specific and are on the '50' (150, 250, 350, etc). Relative Generic Codes (MC minus 24 to MC minus 1) are designed to be generic definitions that apply to a wide range of floats and data. Examples include a series of measurements made when transitioning to the next MC or a single measurement made when transitioning to the next MC. Finally, there are Relative Specific Codes (MC plus 1 to MC plus 25) and these are designed to apply to only a specific type of float(s). The hope is to minimize the Relative Specific codes since that may require different handling for different float types. In the discussion that followed, it became clear that the wording may need to be changed somewhat to reflect that Primary/Mandatory MCs may not appear in the trajectory file if the float did not experience that event.

M. Scanderberg wrapped up the presentation with a reminder that the v3.1 trajectory files were designed to be flexible and allow events to be added with new MCs if necessary. It is crucial to have float experts and manufacturers to help match raw data from floats to MCs and this information must be captured in the DAC Trajectory Cookbook. In addition, when estimates are made, these procedures need to be agreed upon and documented in the DAC Trajectory Cookbook to be consistent across DACs and dmode operators. Good communication between DACs and PIs is essential to ensure all data is delivered, decoded and understood properly.

2. Working Group Activity

M. Scanderbeg introduced the group activity where raw data from APEX APF9 with Argos, APF9 with Iridium, APEX APF11, NAVIS, ARVOR and BGC APEX data would be examined to match up with measurement codes. Participants were asked to self select into groups based on the APF9 with Argos, the NAVIS/APF9 with Iridium, the APF11, Arvor and BGC floats. After that, the blank MC table was described and groups were set free to do their best to match up the raw data to MC codes, keeping in mind the order in which events were to be recorded, the cycle in which the data are reported and the type of file the data can be found in. Questions for the group were to be indicated somehow. After a period of time, each working group was expected to report back their results and pose questions encountered during the exercise.

This report will not include all the details of the discussions following each presentation. Instead, there is an action item on each group to finalize their MC tables by float type and send them around for feedback. Following that, they will be included in the next version of the DAC Trajectory Cookbook. However, there were some general items that were agreed upon after hearing all the presentations. First off, it was suggested to change the meaning of Ascent End Time (AET) to mean when the float switches into surface mode and not the exact time the float finishes its ascent and is at the surface. Many floats report the first time, but not the second, and clarifying this in the Cookbook and User's Manual is important so that users understand what AET means. It was also suggested that MC 702 and 704 which are the times of first and last transmission should not be required for Iridium floats with RUDICS which do not send this information. These two codes are very important for Argos floats, but not for Iridium floats with RUDICS.

> **Action item 1:** Working groups were formed with the intent to prepare 'finalized' MC tables which will then be passed to the rest of the ADMT for feedback. Further revisions can be made and then included in a new DAC Trajectory Cookbook. Tables should be completed by AST-19.
>
> **Action item 2:** M. Scanderbeg to update other parts of the DAC Trajectory Cookbook reflecting changes discussed at workshop. Pass to ADMT for feedback.
>
> **Action item 3:** M. Scanderbeg to update MC table reflecting changes discussed during meetings held this week. Pass to ADMT for feedback and then included in both DAC Trajectory Cookbook and Argo User's Manual.

3. Trajectory File Checker

M. Ignaszewski presented several examples of rejections from the operational test version of the Trajectory File Checker. The checks themselves are documented in the File Checker Cookbook (http://doi.org/10.13155/46120). The purpose of this presentation was to work through example rejections and confirm them or modify them and to explain the format of the rejection. After working through each example, everything that was presented was confirmed to be kept in the Trajectory File Checker. The issue of whether any particular MC can be repeated in a single cycle came up and a working group was formed to study this issue. Additionally, it was requested that the File Checker ensure that floats that are greylisted have a QC flag of '3' or '4' in <PARAM> variables in the trajectory file. Finally, there was a discussion on when the Trajectory File Checker should become active. Right before the meeting, M. Ignaszewski reported that about 12% of trajectory files were being rejected by the Trajectory File Checker, with 88% making it through. He did note that some trajectory files were rejected for a variety of reasons and that the checker could be modified to allow some things through with a warning vs. rejection. He also noted that when the Trajectory File Checker first went into operational test mode that many more files were being rejected and that DACs have been responding to the warnings and made improvements in their files. However, the number of rejected files seems to have plateaued in the past few months. The Workshop felt that the AST should be asked for guidance on whether to make the Trajectory File Checker active and allow only consistent, well formatted files on at the risk of losing 12% of files for which the reason for rejection may be small and many cycles of usable data would be lost.

> **Action item 4:** Working group to study which MCs can be repeated and which can appear only once. M. Scanderbeg, J. Gilson, M. Ignaszewski

> **Action item 5:** Ask AST for guidance on when to make the Trajectory File Checker active. M. Scanderbeg, M. Ignaszewski

4. Real Time QC tests

R. Cowley gave a presentation on the Real Time QC tests that are currently in the Argo User's Manual for trajectory files. There are 6 tests currently in the manual and she attempted to run all trajectory files on the GDAC through her version of the RT QC tests and compare the results to what was recorded in the trajectory files. Not all files were able to go through this test, but overall the results showed that several DACs were not performing some of the RT QC tests in the manual, especially the Global Range test and for some DACs, the Impossible Date and Impossible Speed test. This was a useful inventory and several DACs acknowledged that they were not yet running these tests and it was noted that the QC flag must be 0 in these cases to indicate that nothing was done. DACs also requested that R. Cowley send the results to them or make them available so they can check them against what they believe is occurring. The plots of the data and a summary of the checks for each DAC can be found at: https://cloudstor.aarnet.edu.au/plus/s/aHUnvxglHELm8Wg

A discussion followed on what can be done with trajectory files in real time. B. Owens raised the issue of transferring known pressure and salinity offsets to trajectory files in real time. This was agreed to be a good idea. M. Scanderbeg wondered if the real time QC tests related to GPS positions could be improved. There was not a real consensus on this issue. In the end, it was requested that DACs perform the RTQC tests that are currently in the User's manual.

> **Action item 6:** Ask DACs to ensure that information from the greylist is being applied to trajectory parameters as well. This will involve looking at the time range in the greylist and converting that to cycle numbers and changing parameter QC flags accordingly in the affected cycles.

> **Action item 7:** If offsets to pressure and salinity are being applied in real time to profile files, ask DACs to apply these same real time offsets in pressure and salinity to trajectory parameters.

> **Action item 8:** Ask DACs to perform RTQC tests on trajectory files

5. Code Sharing and Git Hub

Due to time constraints, some parts of the Agenda were skipped and the final time was spent on a discussion about code sharing. The issue here is that in the current set up, most DACs have to develop decoders for each new float type that arrives. This results in wasted effort and a potential lack of consistency across DACs. It was recognized that for past float types, not much can be done, but with the new APF11 coming out, Argo could work to change how things are done. The idea of float manufacturers having to supply a decoder with their floats was presented at the Profiling Float and Sensor Workshop held earlier in the year in Seattle and the manufacturers seemed somewhat open to this idea. There is also the difficulty that comes when the manufacturer makes a small firmware revision that requires a major re-writing of decoders. M. Donnelly suggested that if Argo moves towards developing one decoder for all DACs to use for certain float types, there are three important things to consider: 1) using a forum for developing code (such as GitHub) that can keep track of changes and 2) deciding who can have editing access to this code, 3) clear idea of what it desired as the final outcome. Some discussion revolved around this and it was thought that a small number of people should have editing access including some within Argo and some from the float manufacturer. The outcome discussion did not really happen, but still needs to be discussed. It could be an Argo netcdf file or plain ASCII. It was suggested that PIs include a request for a decoder in their float tenders. B. King and R. Cowley volunteered to take the lead on working with the TWR manufacturers to develop an APF11 decoder.

> **Action item 9**: AST, B. King and R. Cowley to work with manufacturers on developing a decoder.

6. Meeting Wrap UP

M. Scanderbeg closed the meeting by thanking everyone for their work during the workshop and requested continued work on the action items. The DAC Cookbook will be updated with the results of the improved MC tables for the various float types.