HTML flight plan update

John Bell
thejohnbell@gmail.com
1/16/2019

Contents

Summary

I like the iPad as an EFB. However, there are certain things within iOS that make using the html flight plan more difficult. This update adds some new features, but the biggest change is to adapt the flight plan to make it more useful in the iOS environment.

Persistence: When you make changes such as inputting times and highlighting, this information will remain if you close and then re-open the same flight plan. Opening a different flight and release will reset the saved data. Export to .pdf if you want to archive the flight plan.

Waypoint reporting: You can add fuel, times, and comments at waypoints. Click on the "---" or "DAL RPT" in the right column of the Times section or on the "log" in the HOWGOZIT. You can transfer this to another tablet by using the "Report" section.

Highlighting: iOS can still be tricky to select. Try a long tap to select a word or number. Use the handles to extend the area if necessary. After highlighting, choose the Highlight button to turn the selection to a highlight. You still cannot highlight across different html elements, but you only get the error message once, you can return to your highlighted area from the error message, your highlights will save if you close and re-open the file.

ETP points: The ETP points have been reformatted. Tap on any point in Jepp FD. Paste the whole text into the name. Cut the latitude and longitude from the name and paste it into the latitude box. Cut the longitude and paste it into the longitude. Tap on Add to Route.

Persistence

The biggest problem with using the current html flight plan is that any changes made to the flight plan from an e-mail attachment are lost if the flight plan is closed and re-opened. This makes features such as highlighting of little value since they are so easily lost. This has now been fixed. Any changes are now saved so that you can open the file, make changes, close the attachment, and then have the changes remain.

How persistence is implemented

When you open the html flight plan as an e-mail attachment, the file actually opens in a preview window. Think of this preview window as being like a special version of Safari without all of the features.

When you make an update such as a highlight or update the waypoint information, the dynamic portion of the html file saves the updated page into the permanent section called local storage. When you open the file, the first thing that it does is to see if there is a saved version with the same flight, city pair, date, and release. If so, then it will open the saved version from local storage instead of the version in the attachment. If the flight plan is a new flight or new release, then the fresh version of the html flight plan will open.

There is only room for one flight plan to be cached into local storage at a time – the active flight. The flight plan must be activated before the changes start caching in local storage. Once the flight plan is activated any changes are stored in local storage. Additionally, changes are stored as they are made. The requirement to activate the flight plan is to prevent the local storage containing the active flight from being overwritten if you open the wrong flight plan.

Figure 1 This is what opening a new flight plan looks like.

Figure 2 This is what the flight plan looks like after it is activated

Cautions and or technique for clearing changes

Opening and then activating a different flight plan than is currently cached will replace the current flight plan in local storage. This is a warning if you do not want to erase the changes to the currently active flight plan. You can use this as a technique in that you can open a different flight plan and go back to the previous flight plan to clear any changes to the previous flight plan.

Archiving a flight plan

There have been requests to be able to archive changes. There is no way to export the edited flight plan that is saved in local storage. Perhaps there is a way to do this by developing a dedicated app, but I am not even sure about the feasibility of doing this with html.

If you want to archive the flight plan, the best solution is to export the flight plan as a .pdf file. The problem with .pdf is that it is not possible to create the interactivity of the html version. However, this interactivity is not needed for archiving and one could even argue that freezing the state of the file by making it a pdf is desirable in the context of archiving the flight.
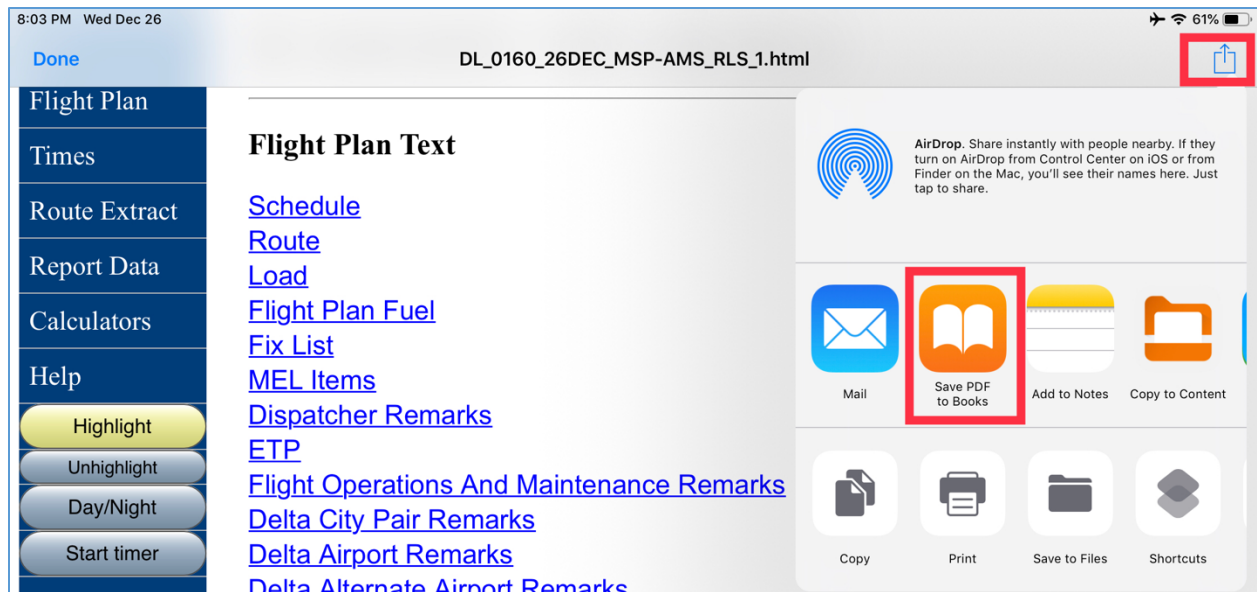
Figure 3

Waypoint reporting

Enter the template to add the values by clicking on the link in the right column of the times report or the word "log" in the HOWGOZIT. The link in the times section will be labeled "---" or "DAL POS." You can make the update in either the Times or HOWGOZIT section and it will be reflected in the other section. The waypoint names are hyperlinks and you can tap on the waypoint name to jump to the same waypoint in the HOWGOZIT from the Times section or vice versa.

There is also a comment section. This is used to add things that you might scribble down on the flight plan such as RDA times, HF frequencies, etc.

I give the time and fuel deltas relative to the originally calculated times based on the original takeoff time. I base the fuel on the on the fuel printed on the flight plan fuel. This is the same process used with the paper flight plan.

```
------------------------------------------------------------------
 DCT            N 4621.5  M070  276/.81  565  :02  7:04    0.3
SAW            W08723.8  T069            13  2210  log    106.8
                                  Actual  2211         107.0
                                          -1            +0.2


  ** DAL POSN RPT **

  ** FIR CZYZ (TORONTO) ** N4707.9 W08526.7 **
------------------------------------------------------------------
 DCT            N 4834.3  M063  278/.81  528  :31  6:33    7.3
YTS            W08122.2  T059  FL350    278  2241  log     99.5

  ** FIR CZUL (MONTREAL)  ** N4918.9 W07905.7 **
------------------------------------------------------------------
 DCT            N 4943.4  M073  278/.81  514  :19  6:14    4.3
NM             W07744.5  T063  FL350    158  2300  log     95.2
------------------------------------------------------------------
 DCT            N 5108.6  M086  278/.81  507  :36  5:38    8.4
ROBBE          W07000.0  T071  FL350    308  2336  log     86.8
                                  Actual  2337          86.9
                                          -1            +0.1
```

Figure 4 HOWGOZIT showing reporting

Wpt		Ttl	UTC	MSP	AMS	Trmg	ALT	PRMG	Sun	Report
1	KMSP	0:00	2134	1534	2234	7:40	CLB	121.7	8	---
2	DWIYT	0:11	2145	1545	2245	7:29	CLB	115.5	5	---
3	WLSTN	0:18	2152	1552	2252	7:22	CLB	112.5	4	---
4	*T-O-C*	0:34	2208	1608	2308	7:06	350	107.1	0	---
5	SAW	0:36	2210	1610	2310	7:04		106.8	0	DAL POS
	Actual		2211	-1				107.0	+0.2	
6	YTS	1:07	2241	1641	2341	6:33	350	99.5	-10	---
7	NM	1:26	2300	1700	0000	6:14	350	95.2	-15	---
8	ROBBE	2:02	2336	1736	0036	5:38	350	86.8	-26	DAL POS
	Actual		2337	-1				86.9	+0.1	
9	ROBBE400	2:51	0025	1825	0125	4:49	350	75.6	-40	---
10	LOMSI	3:03	0037	1837	0137	4:37	370	72.6	-43	DAL POS

ATA [   ]   Actual Fuel [   ]

Comment: [   ]   (ENTER)

Figure 5 Times section showing reporting and input

Discussion on time and fuel delta logic

This section can be skipped unless you are involved in the design of the eCBP.

One of the things that inspired me to finally add the waypoint reporting feature was to see the implementation of waypoint reporting in an app written by a major company. My implementation represents how Delta currently uses the flight plan, but there are other interpretations.

The html flight plan imitates the current method used in the paper flight plan. The time and fuel at each waypoint is based on the time initially calculated based on the off time. The time and fuel at each subsequent waypoint is not updated by entering the time and fuel at a waypoint. If you want to know my reasoning, read on.

On international flights, the relief pilot goes through and writes down the time at each waypoint based on the off time. The html flight plan does this when you enter the off time. Now let's say that you are 1 minute late at a waypoint A. The html flight is pretty clear that you are now 1 minute late. As you travel further in the flight you pass waypoint B. At waypoint B, you are 1 minute behind the time that you are supposed to be there based on the flight plan and initial off time and the html flight plan will show that you are still one minute late based on the original time.

Logic A (the one that is used in the html flight plan): The logic in the html flight plan is that you are 1 minute late at waypoint B as based on the expected time based on the flight plan and off time.

Logic B: When you passed waypoint A one minute late, there is a logic that your new expected time for all subsequent waypoints would reflect this one minute. This is somewhat analogous to the how the FMS works. When you past waypoint B you would be on time based on the fact that the waypoint B and all waypoints after waypoint A were updated based on being one minute late over Waypoint A.

Although the html flight plan uses logic A, it would be easy to re-program it use logic B. One of the biggest reasons that I use logic A is that it reflects current practice in using the flight plan. It would be impractical to take out a pencil and revise each subsequent waypoint on the paper flight plan. However, I think using logic A is valuable in its own right as opposed to simply reflecting the limitations of the logic used with paper. Logging the waypoint fuel and times is not only a way of updating the time and fuel information. It is also a good way of validating the flight plan. Logic A allows a pilot to simply scan the trend to compare the actual time and fuel with the planned time and fuel.

I currently think that logic A for an electronic flight plan makes sense as does logic B for the FMS. I have seen apps that use logic B. While one can argue the trade offs for using logic A vs B, I hope those involved in making the decision understand the issue and understand that using logic A has merit beyond simply reflecting the limitations of the current paper process.

Waypoint verification

Delta ETOPS procedures require verifying some waypoints by circling, drawing a slash, and completing the slash into an X. The html flight plan has a basic way of doing this. Tapping in the longitude immigrates this by cycling through the characters "O", "/", and "X".

```
---------------------------------------------------------------
  DCT             O  N 5600.0   M096   260/.80   483   :43   3:22      9.1
 5640N               W04000.0   T076   FL370      345  0152   log      56.4
   POSITION   ATA       FL       ESTIMATE    ETA    NEXT            FUEL

 56N040W                         57N030W             57N020W

   ** DAL POSN RPT **
---------------------------------------------------------------
```

Figure 6 Circle

```
---------------------------------------------------------------
  DCT             /  N 5600.0   M096   260/.80   483   :43   3:22      9.1
 5640N               W04000.0   T076   FL370      345  0152   log      56.4
   POSITION   ATA       FL       ESTIMATE    ETA    NEXT            FUEL

 56N040W                         57N030W             57N020W

   ** DAL POSN RPT **
---------------------------------------------------------------
```

Figure 7 Draw a line

```
---------------------------------------------------------------
  DCT             X  N 5600.0   M096   260/.80   483   :43   3:22      9.1
 5640N               W04000.0   T076   FL370      345  0152   log      56.4
   POSITION   ATA       FL       ESTIMATE    ETA    NEXT            FUEL

 56N040W                         57N030W             57N020W

   ** DAL POSN RPT **
---------------------------------------------------------------
```

Figure 8 Draw the second line

Exchanging the reporting data with other pilots on the crew

There is a new section of the flight plan called Report Data. As you enter times, fuel, and comments at each waypoint, the html flight plan keeps these in a list. If you know how to use Air Drop, you can paste the text from the report section into a note and Air Drop this list to the other pilot(s). Air Drop is great, but you have to have the settings correct on two iPads. You can search the internet for better instructions than I could give.

A quick way of transferring the data is to use a QR code. Here is the technique:

1. Open the camera app on the receiving iPad.
2. Aim the camera at the QR code. You do not have to take a picture. The app will scan the QR just like it processes the focus before you take the picture.
3. Acknowledge that you would like to open the e-mail.
4. Copy the entire text of the body of the e-mail.
5. Reopen the html flight plan.
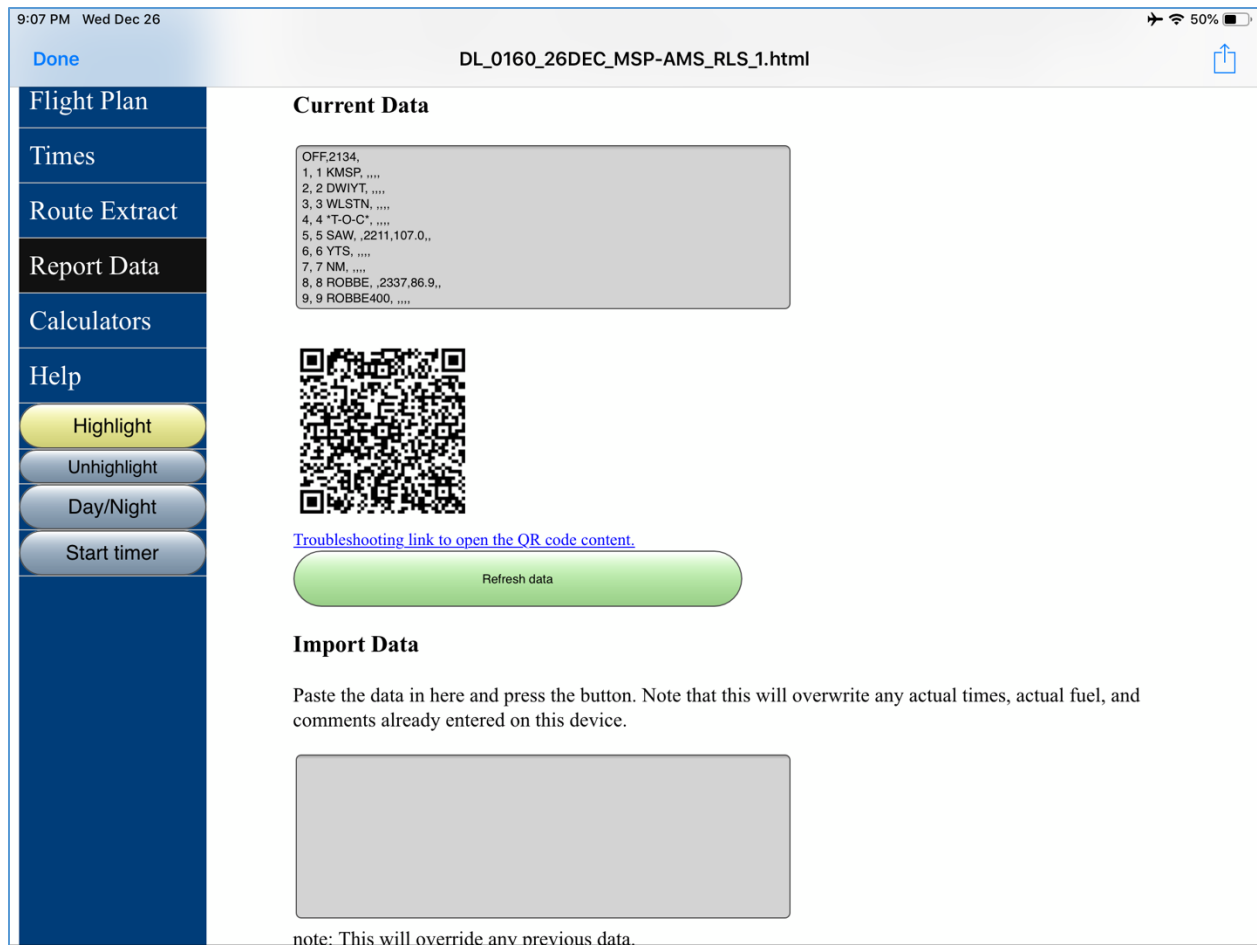6. Paste the text into the input section of the html flight plan and tap the import button.

Figure 9

Importing data will overwrite previous data. It is conceivable to design a system that would be like Google Docs in that when one pilot makes and update, it would be reflected on the tablets of the other pilots. What is easy to conceive is not always feasible to execute. This would take a dedicated app and you would have the issue of connecting the iPads to each other. This would probably best be done by each device logging into an aircraft interface device or using GoGo as a server to coordinate this information. It is probably feasible at some point, but it is beyond the capability of the html flight plan and is certainly beyond my capability.

The way that I see this feature being used is for the pilot keeping the data transferring it to one of the pilots coming on duty. The pilot coming on duty would then become the new master.

Even though I have added this feature, I am still a fan of paper on a clipboard as a communications device between pilots. This is especially true on 4-man crews that involve a complete crew change. I really like the idea of a compact form HOWGOZIT printed off of the ACARS. I think that simply replacing paper will require quite a bit of discussion and should not be taken lightly. This feature in the html flight plan is meant as a first step to experiment as part of this discussion.

Highlighting

Highlighting has been relatively useless in the html flight plan in iOS. The error message is embedded into the bottom of the page because iOS will not open a pop up menu from the html flight plan opened as an attachment. I got around this by sending you to a message at the bottom of the page. In this release, I have added a button to go back to where you were and I only give you the error message once per flight plan. Additionally, the highlighting is now persistent if you close and re-open the file.

It might be worth a review of how highlighting works. After selecting the text, click on the highlight button to turn the selection into a highlight.

The tricky part is to highlight the text. I find that a long tap works well where you hold your finger down for a second or two. This should select the whole word or number which is often all you need. You can drag the sizing handles if you need to select more. I find that after a little practice, highlighting works most of the time. I have had times where the highlight looks correct, but will not highlight and re-selecting what appears to be the same area will work.

There is still a limitation that you cannot select text across an html element. Often these elements are obvious such as a hyperlink. Other times they are not obvious such as embedded paragraph marks. While it has its limitations, the feature allows you to quickly and easily select things such as fuel loads, reporting points, etc.

QR route

This does not work and will not work until Jeppesen enables the feature for Jeppesen FD Pro.

I believe that that the format will change from the way it was handled on the Surface will be changed. The best way to enable this feature in iOS will be by Jeppesen creating a custom URL scheme. Many apps have done this. The most relevant example is ForeFlight, which allows the loading of the route using a link which can then be endcoded as a QR code. If you have ForeFlight and want some examples, you can look here: http://www.johnfbell.com/JeppExamples/ExampleXML_routes.html.

ETP format change

The ETP format in the route section has been changed so that you can paste the ETP directly into Jeppesen FD Pro. This is cumbersome, but less cumbersome than some of the other techniques floating around out there. The technique is to:

1. Copy the ETP text from the route extract portion of the flight plan.
2. Tap to create a new point anywhere in Jepp FD.
3. Paste the ETP text from the html flight plan in the name section.
4. Cut the latitude and longitude from the full text in the name section and paste it into the latitude section.
5. Cut the longitude from the full coordinates and paste it into the longitude section.
6. Choose "Add to Route." This should also select "Delete When Route is Cleared" in Jepp FD 3. You must tap on the newly made ETP and select "Delete When Route is Cleared" in Jepp FD 4. This difference has been reported and hopefully will be fixed.

I have a video that needs to be updated, but shows the basic technique. https://youtu.be/zK916zn7UPg