

September 11, 2008

## Report of the AOC **Teleconference 41** held on **14 October 2008**

In attendance were:

Frederic Durand, Airbus	Ulf Jahr, LH Systems
Beka Teklu, Boeing	Mike Russo, IA Staff
Doug Mikus	

### **Introduction**

Mike Russo moderated a 90-minute teleconference concerning the development of draft Supplement 1 to ARINC Standard 633 with a focus on Chapter 10, eFF.

### **Agenda**

<b>Item</b>	<b>Topic</b>
1	Admin/Review of Previous Meeting
2	Discuss Working Papers (Boeing/Jeppesen)
3	Strawman 21 of ARINC 631 and Related Schema
4	Close

### **Working Papers**

Doug Mikus introduced a paper recommending revision to Chapter 10 to improve its clarity. He asked a series of related questions. The questions were imbedded as comments in the file distributed prior to the telecon. See attachment 2 to this report.

Generically, Doug questioned the mixed use of EFF and eFF. Supplement 1 should be edited to only use the eFF designation.

10.1.1: The intent of the original text was to minimize over-specifying functionality because it could lead to a limitation of the ingenuity of the software designers. Consensus was to replace the bulleted items 2, 3, and 4 with text appended to bullet 1.

10.2.1.: ARINC 633 has single figures. Doug increased the figures to two each to make the distinction of the initial and unsolicited modification through an EFUSUB. He also generated a second figure to illustrate the eFF uplink requested by the aircraft and an eFF sent to a ground system at flight closure (sent within an EFDREP message).

10.2.2: Editorial changes for clarity.

10.2.4: It is not yet clear whether or not the operational flight plan is always optional. The discussion also involves the Mandatory entry in Table 2. New text was added to the table to indicate that Mandatory at the level of the subfolder implies all documents in such a subfolder are mandatory. Frederic Durand took an action item to clarify.

10.3.2: The most contentious issue was the addition of aircraftRegistration (and delimiters) to Table 7, eFF Filename. While there was unanimous agreement that the inclusion of aircraft registration in the filename would simplify the identification of an aircraft (including David O'Kelley via e-mail), Frederic Durand indicated that this deviation from the original filename would present a conflict for aircraft and ground systems fielded with the original filename. The telecon discussed repositioning the entry to minimize the rework, but noted that the number of Supplement 1 Schema has already led to the need for a major software upgrade. Airbus accepts changes to the flight plan, but not to the core of eFF.

David O'Kelley submitted a comment supporting Doug on the flight date issue. See Attachment 3.

The revised Chapter 10 is included as Attachment 4.

### **Strawman 22 to Supplement 1 of ARINC 633**

Beka Teklu noted that there, indeed, have been a large number of variations. He asked that the next strawman be published such that the reader could identify the difference among the original ARINC 633 and the currently-proposed ARINC 633-1.

### **Future Activity**

AOC Telecon 42 is scheduled to be held on 11 November 2008. There will be no telecon in December.

The next scheduled AOC SC meeting is scheduled for 3-5 December 2008 in Albuquerque, New Mexico. Ultramain will host the meeting.

## Action Item List AOC Meeting (21- 23 July 2008)

<b>Tables Closed: Action Items</b>		
<b>#</b>	<b>DESCRIPTION</b>	<b>STATUS</b>
27-31	<b>Table 2: Telecon 11-12 Action Items</b>	Closed
32 - 40	<b>Table 3: December Meeting Action Items</b>	Closed
41- 57	<b>Table 4: Telecons 15 - 17 Action Items</b>	Closed
58-67	<b>Table 5: Meeting 25-27 April 2006 Action Items</b>	Closed
68-71	<b>Table 6: Telecons 19 - 20 Action Items</b>	Closed
90	<b>Table 8: Telecon 23 Action Items</b>	Superseded
101-102	<b>Table 10: Telecon 25 Action Items</b>	Closed
103 - 108	<b>Table 11: Telecon 26 Action Items</b>	Closed
109 - 121	<b>Table 12: February 2007 Meeting Action Items</b>	Closed
122 - 124	<b>Table 13: Telecon 27 Action Items</b>	Closed
125	<b>Table 14: Telecon 28 Action Items</b>	Closed
126-127	<b>Table 15: Telecon 29 Action Items</b>	Closed
128-130	<b>Table 16 Telecon 30 Action Items</b>	Closed
152-154	<b>Table 19: Action Items AOC Telecon 32 (August 2007)</b>	Closed
155	<b>Table 20: Action Items AOC Telecon 34</b>	Closed
156	<b>Table 21: Action Items AOC Meeting 11 (March 2008)</b>	Closed

Table 1: Action Items Meeting (August 2005)		
#	DESCRIPTION	STATUS
01-06, 08-14, & 16-26		Closed
07	<b>Frederic Durand and Mike Russo</b> will designate end-to-end or air-ground coverage in paragraph to Section 3.2. The input will assign proposed dates when the security function should become available through the supporting infrastructure.	Closed
15	<b>Frederic Durand</b> and <b>Luc Rochard</b> will propose a Database Activation Service message and associated processes. Delays in initiating the development effort suggests that a mature definition may need to be incorporated into ARINC 633 with Supplement 1. However, due to the urgency of the availability of this application, work will start as soon as possible. The subcommittee asked that Air France provide further clarification of the mechanism that is needed to provide the benefits they identified in their original request.	Open

Table 7: Action Items Meeting (August 2006)		
#	DESCRIPTION	STATUS
72 – 75, 77, 79 81-83, 85 – 86, 89		Closed 02/20/07
76	<b>Frederic Durand</b> will check for consistency among the eFF schema signature and the ATA e-LogBook signature definition.	Open
78	<b>Ulf Jahr</b> will provide a detailed definition of the proposed enc.xml file for OpenPGP. Pending NIS SC disposition.	Open
80	<b>Mike Russo and Franklyn Young-Martos</b> should investigate with AEEC whether requirements could be tagged in future documents. Frederic Durand will provide tag examples.	Ongoing
84	<b>Steve Acheson</b> et al were asked to propose a list of applications to be considered for standardization for the next AOC SC meeting.	Ongoing
87	<b>Frederic Durand</b> will work with <b>Dirk Zschunke</b> and <b>Andreas Ritter</b> to develop a Loadsheet and Trimsheet schema to include in Chapter 5. <b>Dirk Zschunke</b> will provide basic definitions for NOTOC XML schemas for integration into Chapter 5. Dirk recommended to reference IATA development when available.	Ongoing

<b>Table 9: Action Items Meeting (Nov 2006)</b>		
<b>#</b>	<b>DESCRIPTION</b>	<b>STATUS</b>
91-96, 99, 100-102		Closed
97	Ulf Jahr and Andreas Ritter will create new examples of eFF schema. Frederic Durand has the action to verify the accuracy of the preliminary eFF schema developed by Ulf.	Open
98	Andreas Ritter and Jocelyn Descaillot volunteered to prepare a more specific proposal for the GACS protocol for consideration at the January 2007 Telecon. <b>To be considered by NIS SC.</b>	Open

<b>Table 12: Action Items Meeting (February 2007)</b>		
<b>#</b>	<b>DESCRIPTION</b>	<b>STATUS</b>
109 – 120		Closed
121	Frederic Durand will submit a modified (supplement 1) eFF schema with the new attributes that have been approved by the subcommittee (including the input from Ludovic Lengrand). <b>See also # 141.</b>	Open

<b>Table 17: Action Items AOC Meeting (June 2007)</b>		
<b>#</b>	<b>DESCRIPTION</b>	<b>STATUS</b>
130 - 137		Withdrawn
138	Dave Knerr will provide additional flight plan examples including a re-clearance.	Open
139 – 140, 143, 146		Closed
141	Frederic Durand will make modifications to the FlightPlan schema to reflect the recommendations of June 2007 AOC SC meeting. Expect an input to the Oct 2007 TWG.	Open
142	Dave Knerr will gather additional information concerning the constructs that have been developed within FAA, US Navy, NOAA, and WMO to define elements and attributes applicable to aviation weather products. He will also look into RTCA SC-206 program. Robert Popov reported that a Eurocontrol activity exists, but will not have a weather product definition for at least 2 years. David O’Kelley to inquire as to definitions by WSI.	Open

144	<b>Ulf Jahr</b> will prepare edits to Chapter 10 (Ref. Item 11 of Airbus list of AFR issues) to propose an expanded request service downlink which allows request of specific data for the various elements of the Topic schema.	Open
145	<b>Airlines and eFF vendors</b> are asked to validate the existing flight plan schema and provide flight plan examples for review by the October 2007 TWG. An initial input was supplied by LH Sys. The work is still ongoing.	Ongoing

<b>Table 18: Action Items AOC/ELPT ad hoc Meeting (June 2007)</b>		
#	DESCRIPTION	STATUS
147-148		Closed
149	Tim Anstey and Steve Yukawa will define some typical message sequence diagrams for a typical flight.	Open
150	Tim Spears will prepare a proposal for the constraints to message sequence relative to technical keys.	Open
151	Dirk Zschunke will prepare edits to Chapter 4 to define submit (SUB) and report (REP) messages.	Open

<b>Table 22: Action Items AOC Meeting 12 (July 2008)</b>		
#	DESCRIPTION	STATUS
157	<b>David O'Kelley</b> will research flight plan enroute diversion route information content to be added to the FlightPlan schema.	Open
158	<b>David O'Kelley</b> will research the format of latitude and longitude used in NDBX.	Open
159	<b>Dirk Zschunke</b> Waypoint Identification (id) needs further clarification. Recommend that the attribute be designated as optional or mandatory.	Open
160	<b>David O'Kelley</b> Explain the use of order of Waypoints in arrays to designate the sequence of waypoint as an attribute.	Open
161	<b>Dirk Zschunke</b> volunteered to prepare a draft ATS Flight Plan schema for review.	Open
162	<b>Dirk Zschunke</b> reply to Ulf on XML encoding and version identification.	Open
163	<b>Beka Teklu</b> to provide use cases for sending eFF on ACARS	Open

**Table 23: Action Items Telecon 39 (Aug 2008)**

#	DESCRIPTION	STATUS
164	The equation used to derive a LeakDetection (Table in Section 5.4.7) currently produces the wrong result. An alternate equation has been proposed. <b>Frederic Durand</b> took an action item to resolve the issue by recommending the removal of the code or repairing it.	Open
165	<b>Beka Teklu</b> took an action to reconfirm that the contents of the CriticalPositionType Condition element (DC, 1X, 2X, DX) in the SuitableAptType Element Definition Table in Section 5.4.10, EROPS, is unique and has a different value than the Condition value in the header. <b>Robert Popov</b> was asked to comment on the text definition of the CriticalPositionType Condition element.	Open
166	The AOC SC proposal to change the definition of Potable Water PSWQ from a 1 to 3 digit percent to a 1 to 4 digit percent in tenths will have an impact on Airbus. <b>Frederic Durand</b> took an action to coordinate the change (confirm) with Airbus software designers.	Open

**Table 24: Action Items Telecon 40 (Sept 2008)**

#	DESCRIPTION	STATUS
167	Placeholder	Open
168	Placeholder	Open
169	Placeholder	Open

**Table 25: Action Items Telecon 41 (Oct 2008)**

#	DESCRIPTION	STATUS
170	<b>Frederic Durand coordinate Airbus position on the Mandatory/Optional status of Operational Flight Plan (Section 10.2.4). Also, doublecheck if the new last paragraph is needed. Clarify if required.</b>	Open
171	<b>Frederic Durand will coordinate an Airbus position on the addition of aircraftRegistration to the eFF filename in Table 7 of Section 10.3.2.</b>	Open



1. 10.1.1: What is the difference between “eFF” and “EFF”? It appears they might have different meanings but are being used interchangeably.
2. 10.1.1: There’s really more than just the documents and the word ‘content’ is used in subsequent paragraphs.
3. 10.1.1: What is the context of “Update”? Who’s doing the updates (e.g. ground tool or EFF apps)? Are these trying to explain full or partial updates the ground tool may send to the EFF apps? Perhaps these three bullets can be replaced by “Update whole or in-part, the documents within the EFF data package (i.e. pilot inputs/modifications)”?
4. 10.1.2: I changed this to match the definition above. Or does application better fit?
5. 10.1.2: Can a flight folder ever be delivered via UPDATEs only (never with a FULL package)?
6. 10.1.2: This is confusing. It appears that pilot updates are re-packaged into a new EFF data package and marked as FULL. I thought only the Mainbase tool would create the EFUSUB packages.  
  
It would appear this document does not differentiate between a Mainbase data package and a pilot “updated” package prior to transporting it to the aircraft.  
  
Recall that the EFF may get transported to the aircraft via USB by the pilot, so the scenario considers the case where the pilot assembles the package.
7. 10.1.2: Does this statement reflect a specific implementation – the word “generally” might imply this?
8. 10.1.2: How can a EFF package be marked FULL if there are pending documents OR does this sentence just need to be clarified?  
  
Clarification of the concepts FULL and UPDATE should be explicitly stated. For instance, what is the difference between a FULL package with pending documents and an UPDATE package?
9. 10.1.2: This suggests we can we have multiple FULL packages for a flight folder with pending documents can be sent from the ground. Is that true?
10. 10.1.2: Can an UPDATE be received prior to a FULL EFF data package?
11. 10.1.2: These should be broken into two figures so they can be properly captioned and referenced.
12. 10.1.2: These should be broken into two figures so they can be properly captioned and referenced.
13. 10.1.2: I would remove or clarify to avoid ambiguity. “generally all communications” has a lot of room for interpretation and may be implementation specific.
14. 10.2.1: I can’t find this specific reference. I just removed it.
15. 10.2.3: What’s the difference between these two?
16. 10.2.4: Do all EFF data packages marked as UPDATE has the OFP included? If so, why? If not, then we’ll need to reword this a bit.
17. 10.2.6: Isn’t that meaningless for the airborne side? Mandatory is yet a flag with “document”. According to Air France, there are two levels of mandatory information. Mandatory at the level of the subfolder means that “all” the documents in such a subfolder are mandatory.



**From:** david.okelley@jeppesen.com [mailto:david.okelley@jeppesen.com]  
**Sent:** Wednesday, October 01, 2008 12:10 PM  
**To:** Russo, Michael (MPR)  
**Subject:** Re: [AOC] 633 Section 10 (Filename Update Proposal)

All,

I have to agree with what Doug is saying. On the flight date issue, it is important to decide which date we are going to use and make it clear to all. In scheduling and reservation systems FlightOriginDate is used, which is the scheduled date of departure for the first leg of a flight (multi-leg). A lot of operational systems do not understand this date and instead use scheduled leg departure date. I would suggest, as Doug does, that we are very clear on the date that we are using in this field.

The addition of the aircraft registration to the file name is probably not a bad idea either. This will provide a quick check that file is really meant for this aircraft.

David C. O'Kelley  
Jeppesen Technology Services  
CMA Operation Services, Solution Architect  
Office: 303-328-6512  
Cell: 720-299-2398

## 10 Electronic Flight Folder

### 10.1 General

The purpose of this section is to standardize how operational information regarding a flight segment is technically organized in a flight folder that is accessed by the pilot **electronically**. This does not intend to provide a standard organization of the documents in the flight folder, but rather a technical mean that would allow carrying a structured organization of flight documents.

As the electronic flight folder is also the support of the operational flight plan, and as operational flight plan contain many important data for the flight that can be passed to other applications, it was deemed necessary to standardize data in an operational flight plan. As there are many different ways to display an operational flight plan, the aim here is to try to have a generic model to handle the data that are available in the operational flight plan, but not on this format. This could be easily handled by XML, as XML is able to separate the data and the way it is displayed.

Future version of this document may define data other than the operational flight plan ~~Other data than operational flight plan may be defined in future versions of this document.~~

This section defines:

- An overview of eFF operational concept (**i.e.** how the data package is elaborated, updated, and finalized)
- The structure of eFF data package, which parameters are defined and how they are used
- **The packaging of eFF data** into an '.eff' file ~~of the eFF data package is covered.~~
- The rules of transportation via an IP network or with physical media (e.g. USB media) are described. The transportation subsection also covers the option to request and transport updates to an existing eFF. Updates can apply either to a part of the eFF which the user wants to update or the whole eFF.

The definition of some document contents can be found in section 5.2 'Common Data'.

Note: **The** way eFF is transported is outside the scope of this section and is addressed in section 2.3.

#### 10.1.1 Review of the eFF workflow

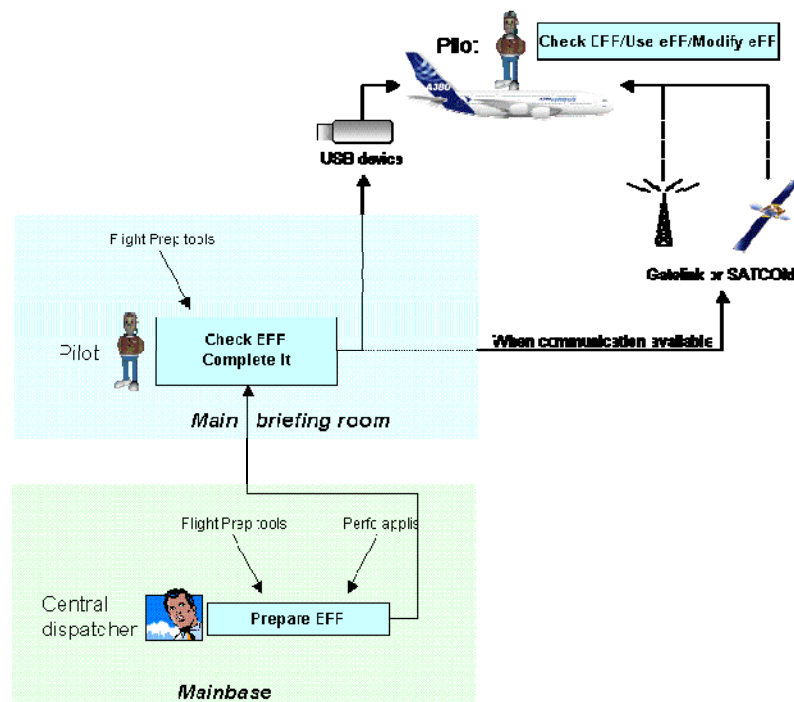
Once the ~~documents contents~~ for the eFF are gathered or created and the eFF is packed properly, **the question of the eFF data package usability** arises. The following actions are possible with an eFF **data package**:

- Transport an eFF **data package** from one System to another (e.g. from ground to an airborne system and vice versa). **Update whole or in-part, the documents within the eFF data package (i.e. pilot inputs/modifications)**
- ~~Update the whole eFF which was fetched before.~~
- ~~Update documents marked with specific topic identifiers in an eFF which was fetched before.~~
- ~~Update single documents in an eFF which was fetched before.~~
- Use content of the eFF **data package**.
  - **Note: This** is out of scope of this document. It is the responsibility of the Applications working with the eFF.

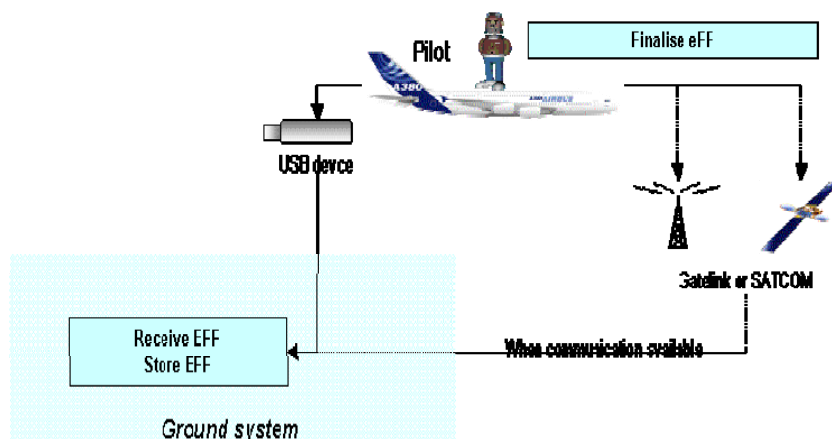
The eFF data package is exchanged between ground and airborne applications. These applications are known as:

- Ground tool at mainbase: (ground application used by dispatcher to generate the eFF)
- eFF ground application (used by the flight crew on ground at main briefing room)
- eFF airborne application (used by the crew onboard the aircraft)

During its lifecycle, the eFF data package (known as 'eFF') is produced, modified and processed by **any of** the above mentioned applications. **Figure 1 shows an example of the process for uplinking of the eFF data package during preflight operations. Figure 2 shows an example process for downlinking fo the eFF data package during post-flight operations.**



**Figure 1: Preflight Process Example**



**Figure 2: Post-Flight Process Example**

## Typical Process For Downlinking Of The Eff Data Package (Postflight)

### 10.1.2 The eFF Lifecycle and associated communication services

The eFF has the following life cycle:

- A ground ~~based application tool~~, e.g. an Airline Flight Data Processing System, generates **an eFF data package**.
- **The initial eFF data package is marked as FULL ('fullPackage' attribute of the root element is set)** ~~here complete (marked as 'FULL')~~. **This initial eFF data package may have all needed documents, with all needed documents, or documents or folders marked as pending to be sent in subsequent eFF data package updates marked as UPDATE ('fullPackage' attribute of the root element is not set)**. ~~to be later sent (in this case these documents are identified in the eFF data package but marked as 'pending' in the eFF data package)~~. The system which composes the eFF decides when not to include a document **in the eFF data package**. **The decision to not include a document may be based on (but not limited to) performance reasons, timed out request or unavaible document. An UPDATE avoids sending all documents that were previously sent to the aircraft.**

~~This option can be used in case of a timed out request to a subsystem or for performance reasons also.~~

- ~~During flight preparation, the pilot may complete the eFF data package using his ground eFF application. The result of this is generally a FULL eFF data package.~~

~~□ Preflight, the whole eFF data package is transported from the eFF ground application to the EFB by the use of described later. Transport can be physical (USB memory stick, CD/DVD..) or via IP networks..~~

~~□ Preflight or in-flight updates may be sent from the eFF ground application to the eFF airborne application, again by the use of the eFF Uplink service.~~

~~o In this case, the full eFF with new documents added may be sent to the aircraft (eFF data package marked as 'FULL' with all documents or pending documents). In this case, previous eFF data package for the same flight will be replaced by the newly received one (it may be useful to keep track of the eventual modification performed on the previous eFF data package by the crew on board the aircraft and copy them on the new eFF data package).~~

~~□ Or only an eFF update is sent to the aircraft (eFF data package marked as 'UPDATE'): this avoids sending again all documents that were previously sent to the aircraft. In this case, the eFF airborne application will add the update to the previous updates and the last FULL eFF data package received.~~

~~□ In-flight and post flight the eFF is enriched by flight crew or airborne systems~~

~~□ Parts of or the entire eFF data package is sent from airborne eFF application to the ground application using the eFF Downlink service.~~

~~<replace the following bullets — to here>~~

**with**

- **Preflight / In-Flight**
  - **eFF data packages (FULL and UPDATES) are transported from the eFF ground application to the EFB using the eFF Uplink Service.**
    - **An eFF data package marked as FULL replaces all previous eFF data package(s) for the same flight. In this case it may be useful to keep track of the modifications performed on the previous eFF data package(s) by the crew and apply them to the new eFF data package.**

- An eFF data package marked as an UPDATE will contain replacement document(s) and/or document(s) that were previously marked as “pending” and/or new documents. It may be useful to keep track of the modifications performed on the previous documents(s) by the crew and apply them to the new documents.
  - The eFF document(s) are enriched by flight crew or airborne systems.
- Post-Flight
  - The eFF document(s) are enriched by flight crew or airborne systems.
  - Parts of or the entire eFF data package is sent from airborne eFF application to the ground application using the eFF Downlink Service.

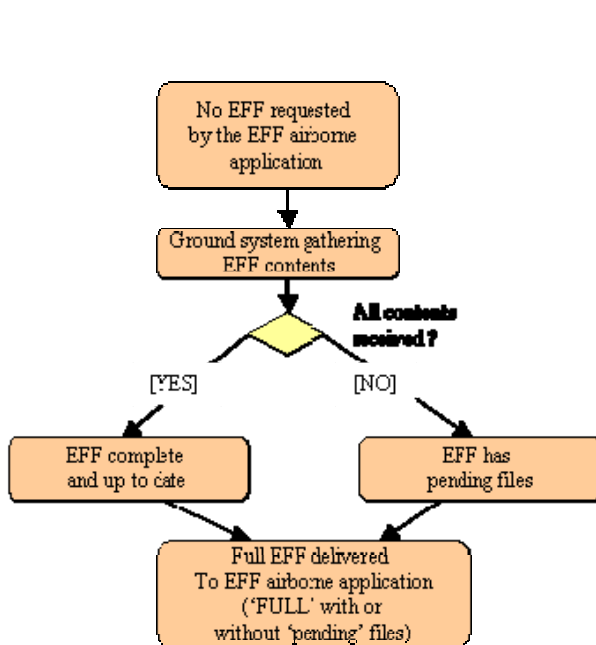


Illustration of Initial eFF generation and uplink (sent in EFUSUB message)

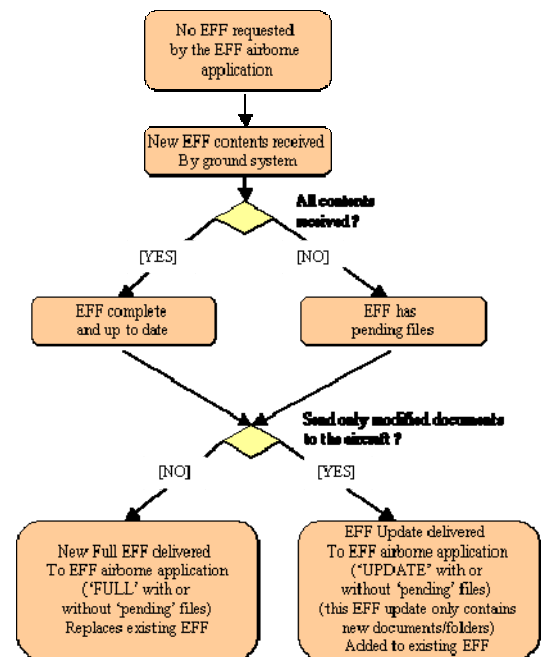


Illustration of Unsolicited eFF modification (sent in EFUSUB message)

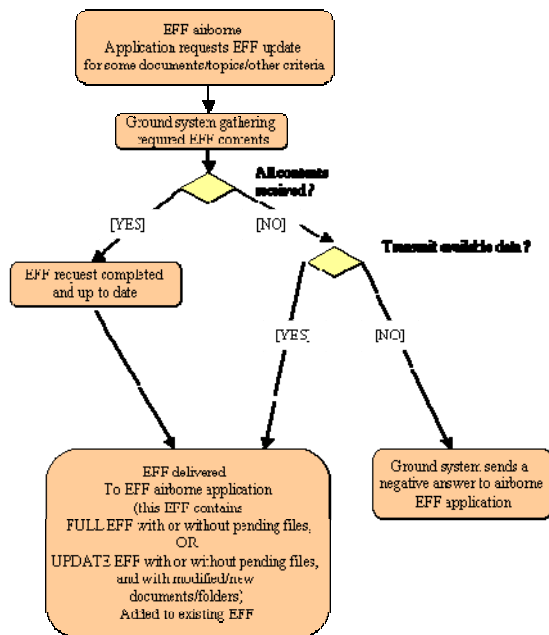


Illustration of eFF requested by Aircraft  
(EFUSUB message sent following  
eFFREQ message)

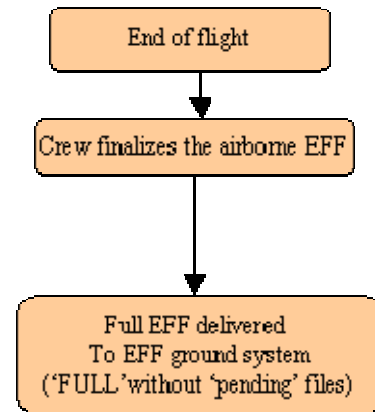


Illustration of eFF sent to  
ground system at flight closure  
(sent in EFDREP message)

## COMMENTARY

‘Unsolicited’ means ‘the eFF **airborne** application **has not sent a request to the ground tool**’ – see section 2 for details on how the transport level exchanges are managed. .

## 10.2 eFF structure

### 10.2.1 Introduction

The **individual** documents of a briefing are not sufficient to generate a usable system for a crew member who has to operate a flight leg. **To achieve useability, the documents have been assigned a structure with optional parameters and various parameters have to be provided to turn a heap of documents into a usable data package.**

The main idea is to add an XML-File with the essential structural information and metadata to the documents.

The interrelation between the documents and this XML-file (called “eff.xml” ~~in the following text also referenced as~~ “structure file”) is described in the following subsections.

See electronic document ‘ARINC633 AOC.zip’ for eFF structure specification in XML and examples. eFF structure XML schema is defined in schema/eFF/eFF.xsd. This section is intended to provide global overview of eFF structure as well as information complementary to the eFF.xsd schema. If information contained in this section is contradictory to the eFF.xsd schema definition, the eFF.xsd schema should have precedence.

## 10.2.2 Documents storable in an eFF

The standard presently does not **define** the structure of **any** documents that can be stored in the eFF **data package**. The standard **only describes** how a document **is stored within** the eFF **data package** structure. The format of the document is dependent of the tool used to **create it and the eFF application which displays the document**. **Even** so, one eFF tool can allow storing PDF documents, **while another eFF application may not support the display of PDF documents**. **To eliminate this problem, a MIME type is associated with each document.** ~~for instance, whereas another one may not allow it as it may not be able to display them.~~

The enclosed definition of eFF XML schemas provide a predefined list of common MIME types that should be supported. (See definition of DocumentMimeType **in the** XML schemas)

## 10.2.3 Documents / Operational Flight Plan documents

Documents as they are described in the XML structure file should be of one of the following types:

- Basic documents
  - A basic document is made of
    - The document file itself (ex : A text file, a graphic file, a PDF file)
    - Optional comment file(s) which contains annotations **to** the document.
  - One optional parameter file. These parameters can be used by other applications, e.g.: result of a take-off performance computation.
- Operational Flight plan documents
  - ~~○ Operational Flight plan documents are data that use a style sheet to be displayed. They are not associated to a static representation to be displayed.~~
  - Operational Flight plan documents are associated with a set of style sheets (called a display model). Display models are part of the configuration of the application. Display models shall be managed in configuration by the operator to ensure the display of information after a long period of time.
  - Further information for the Operational Flight plan can be found in paragraph 5.4, Operational Flight Plan.

## 10.2.4 Contents of the Electronic Flight Folder

~~The following files can be found in an eFF.~~ There are three files that are mandatory **within the eFF data package**: The XML structure file, the operational flight plan(??)<sub>[drm1]</sub> and the integrity check file. The rest of the contents of the eFF are highly depended on the operational procedures of the airline that operates the flight.

An Electronic Flight Folder should be made of the following files:

- One XML structure file ('eff.xml' file)
  - XML elements called subfolders are used to organize the documents in the structure file.
- All the document files.
  - Type is arbitrary.
  - Each of them is referenced in the structure file.
  - There can be several operational flight plans (one can be declared as 'active')
- Comment files
  - Comment files are Optional.
  - Comments are associated to documents by the structure file.
- Parameters files
  - For documents that have associated parameters

- Parameter files are associated to documents by the structure file.
- Integrity check file
  - This file references all the files in the packages and provides a hashcode for each of them to ensure their integrity.
  - The modification of any of these file will modify their hashcode so that it is no more compliant with the stored hashcode in the integrity check file

## COMMENTARY

These files will be zipped together in a single 'eff.xml' file – see Section 10.3 for details.

### 10.2.5 Topic concept

The eFF brings the notion of a topic concept.

The topic represents a functional identifier for one or more document(s) of the same kind. **The functional identifier** is a technical identification of the subject of the document **which provides a common means of identification**. This identification can be attached to each document mentioned in the XML structure file (eff.xml). This may help other applications that need to get data from the eFF **data package** to search for a technical identifier instead of a human readable identifier.

For example, an ATC flight plan can be named “Flight Plan” or “Plan de vol”, or something else. If we identify the ATC Flight plan with a Topic “FlightPlan/ATC”, then any application that requires getting the ATC Flight Plan can look in the eFF for the relevant topic “FlightPlan/ATC”. Flight briefing tool providers are free to define additional topics. **Table 1** shows the predefined topics which should be used when applicable.

The topic identifiers consists of one or more parts divided by a slash. The first part of the topic identifier describes the general scope (e.g. “FlightPlan” or “Weather”). Additional parts of the topic identifiers are applied cumulative. For example the topic “Weather” could be applied to all documents that contain weather information, “Weather/Text” is applied to all documents that contain textual information about weather conditions and “Weather/Text/METAR” is applied to textual METAR information.

Please note that a topic identifier can be associated to several documents but a document can refer only to one topic identifier.

See electronic document ‘ARINC633 AOC.zip’ for Topic specification in XML and examples. Topic XML schema is defined in schema/eFF/Topic.xsd. This section is intended to provide global overview on eFF Topics structure as well as information complementary to the Topic.xsd schema. If information contained in this section is contradictory to the Topic.xsd schema definition, the Topic.xsd schema should have precedence.

**Table 1: Predefined Topics**

Topic	Definition
FlightInfo	General flight information summary on the flight
FlightPlan	Gives general information on a flight plan
FlightPlan/ATCNon-ICAO	This topic should be used for the ATC-Flightplan only (for CAAs not compliant with ICAO).
FlightPlan/ATCICAO	This topic should be used for the ATC-Flightplan only (for CAAs compliant with ICAO).
FlightPlan/Operational	This topic should be used for the OFP only.
Weather	General weather information
Weather/Text	General textual weather information

Weather/Text/METAR	The airport weather topic contains the latest METAR information for all airports relevant for the planned flight
Weather/Text/TAF	The airport weather topic contains the latest TAF information for all airports relevant for the planned flight
Weather/Text/SIGMET	The SIGnificant METErological topic contains information about all significant weather phenomena in airspaces (e.g. FIRs or UIRs) relevant for the planned flight
Weather/Chart	The weather chart topic contains wind charts and significant weather charts relevant for the planned flight
Weather/Chart/Profile	Weather profile chart
Weather/Chart/Sat	Satellite image
Weather/Chart/SigWX	Signification weather chart
Weather/Chart/WindTemp	Wind & Temperature chart
NOTAM/ATC	The NOTice To AirMan topic contains other (non-weather) geo-referenced information relevant for the planned flight issued by ATC.
NOTAM/Company	The NOTice To AirMan topic contains other (non-weather) geo-referenced information relevant for the planned flight issued by the company.
NOTOC	The NOTification TO Captain topic contains dangerous goods and other information relevant of the loading of the aircraft. (See also IATA Airport Handling Manual)
NOTOC/DG	Dangerous goods
NOTOC/Others	Other notification to captain (not a dangerous good)
LoadTrimSheet	The Load and Trim Sheet topic contains information about the distribution of masses along the aircraft's longitudinal axes. See also IATA Airport Handling Manual
CrewList	The Crew list topic contains names and other information about the crew member aboard the aircraft
PaxList	The Pax list topic contains information on passengers
OceanicTracks	The Oceanic Tracksystem Track topic contains general information about Oceanic Tracks
OceanicTracks/Chart	The oceanic Tracksystem Track Chart topic contains a chart showing the current organization of the oceanic track system in graphical form
OceanicTracks/Text	The oceanic Tracksystem Track Text topic contains the coordinates of the current oceanic track system in text form
NavChart	The Navigation Chart topic contains navigational charts that are relevant to the planed flight, e.g. updates of charts
AirportInformation	General information regarding airports
SecurityInformation	General information regarding security
AirlineDefined	User defined information – use of additional parts of the topic identifier is recommended
UpperAirData	Upper Air Data information - two wind/temperature types will be supported: spot and segment
RVSMAltimeterCheck	Fields proposed for recording of altimeter checks in RVSM airspace

## 10.2.6 Subfolders Properties

Subfolders are structural information (elements) in the XML structure file **which provide a way to group similar documents (e.g. all operational flight plans)**. The eFF itself contains no folders in terms of a traditional file system. Subfolder elements in the XML structure file can contain:

- Other subfolders
- Documents

A subfolder has several properties **which are described** in the following table.

**Table 2: Subfolder Properties**

Attribute	Definition
title	Title of the subfolder
changed	Change information. This information describes the differences between updates of an eFF. When an eFF is fetched the first time all subfolders will have the changed status New. In the frame of incremental deliveries, it helps to know the difference between data that are New, data that existed but Revised. Unchanged may be used in the case of a complete delivery of

	<p>an eFF, but partially updated. note</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Semantics</th> </tr> </thead> <tbody> <tr> <td>'New'</td> <td>New Subfolder; in the last received eFF package weren't such a subfolder.</td> </tr> <tr> <td>'Unchanged'</td> <td>Unchanged.</td> </tr> <tr> <td>'Revised'</td> <td>The contents of the subfolder have been updated.</td> </tr> <tr> <td>'Deactivated'</td> <td>The subfolder is outdated. Its contents should not be used anymore.</td> </tr> </tbody> </table>	Value	Semantics	'New'	New Subfolder; in the last received eFF package weren't such a subfolder.	'Unchanged'	Unchanged.	'Revised'	The contents of the subfolder have been updated.	'Deactivated'	The subfolder is outdated. Its contents should not be used anymore.
Value	Semantics										
'New'	New Subfolder; in the last received eFF package weren't such a subfolder.										
'Unchanged'	Unchanged.										
'Revised'	The contents of the subfolder have been updated.										
'Deactivated'	The subfolder is outdated. Its contents should not be used anymore.										
documentActivatable	Boolean state. Specifies that we can have an active document in this subfolder. Typically, active document are mostly used for flight plan. See Note [1] below.										
activeDocument	Specifies the identifier of one document which is active in this subfolder (optional). Active notion is similar to the notion of active flight plan in the FMS. Only one document can be active any time. Active documents relates mainly to flight plan documents. An active flight plan is the flight plan known as "active" (meaning currently in use) by the FMS. It is there to be consistent with the FMS philosophy of activation. See Note [1] below.										
mandatory	Specifies that documents in this subfolders or recursive subfolders must be exchanged. This attribute should be set by the ground system to specify to the airborne system which folders are mandatory in eFF downlinks. <b>Mandatory at the level of the subfolder means that "all" the documents in such a subfolder are mandatory.</b>										
constraint	<p>Defines the constraints on these document subfolders. Constraints can be one of the following:</p> <ul style="list-style-type: none"> <li>• "None" : No specific constraint</li> <li>• "CheckOne" : The pilot is required to check at least one document in this subfolder</li> <li>• "CheckAll" : The pilot is required to check all the documents in this subfolder</li> <li>• "Sign" : The pilot is required to sign documents in this subfolder</li> </ul>										
resetConstraint	<p>Tells when to reset the constraint:</p> <ul style="list-style-type: none"> <li>• "Never"</li> <li>• "OnLoadOnBoard" : When loaded on board</li> <li>• "OnLoadOnGround" : When loaded on ground</li> <li>• "OnNewDocument": When a new document is inserted in this folder</li> <li>• "OnLoad" : On any load</li> </ul>										
resetReadStatus	<p>Tells when the read status (by crew operator) is to be reset:</p> <ul style="list-style-type: none"> <li>• "Never"</li> <li>• "OnLoadOnBoard" : When loaded on board</li> <li>• "OnLoadOnGround" : When loaded on ground</li> <li>• "OnNewDocument": When a new document is inserted in this folder</li> <li>• "OnLoad" : On any load</li> </ul>										
transferPending	Declares that the documents in this folder are missing for some reason. The same as if all documents in this folder where marked "transferPending". This attribute may be used for partly delivery of an eFF, e.g. when gathering of some Files would take too much time. The pending files should be delivered with the next update of the eFF. The Pilot should be informed if there is pending information and the eFF therefore is not complete.										

Note[1] The notion of 'active' document was defined mainly for flight plan, Load and Trim sheet, Take Off performance documents – it allows the pilots to identify without ambiguity which document is to be applied for the flight. The attribute 'documentactivatable' is used to specify that this folder can contain a document tagged as 'active'. If documentactivatable' attribute is absent or not set, it means that no document should be declared active in this folder.

## 10.2.7 Subfolder associations

A subfolder has several associations **which are described in the following table:**

**Table 3: Subfolder Associations**

Element	Attribute	Definition
StageOfFlight	@code	@code represents the stage of flight e.g.: <ul style="list-style-type: none"> <li>• Pre-flight</li> <li>• In-flight</li> <li>• Post-flight</li> </ul>
Application	@name	Association with an application. Represents the identifier of the application on the target platform
	@callable	Specifies is the application can be called when on the target platform.

Topic	@name	Represents a topic associated to this subfolder. (See topic concept for definition of list of topics)
-------	-------	---

## 10.2.8 Documents properties

The following definitions apply to all documents

**Table 4: Documents Properties**

Attribute	Definition
id	Technical identifier of the document. It shall be unique in the whole eFF. Even when considering an eFF with multiple updates an id must be unique in the set of files. E.g. when a document is replaced with a newer version the id must not be identical to any id of a file which was ever part of the eFF.
type	Defines the mime type associated with the document. Mime type is used as a recognized international standard for file typing. XML schemas define a standard set of MIME types that shall be supported
file	If in the eff.xml structure, this references the physical file as embedded in the .eFF package
title	Title of the Document – can be different from the filename
priority	Number between 0 and 15 indicating the priority for data transfers or other treatments. The higher the number the more important the Document.
application	This attribute intends to store the application identifier to which this data relates.
mandatory	States that this document is mandatory. Mandatory attributes shall be processed in exchanging eFF information as forcing documents to be exchanged.
Constraint	Defines the constraints on this document. Constraints can be one of the following: <ul style="list-style-type: none"> <li>• “None” : No specific constraint</li> <li>• “Check” : The pilot is required to check one document in this subfolder</li> <li>• “Sign” : The pilot is required to sign documents in this subfolder</li> </ul>
captainOnly	Information stating that only the captain shall be able to display that information
originalDateTime	Date / Time of the first release of the document in ARINC 633 format
updateDateTime	Data / Time of the last update of the document in ARINC 633 format
expectedUpdateAt	TimeDateStamp in ARINC 633 format at which it is expected that an update to the present document may be available
timeToLive	TimeDateStamp in ARINC 633 format from which on the document should no longer be presented as a valid document
status	Status of the document - Possible values are: <ul style="list-style-type: none"> <li>• ‘None’</li> <li>• ‘Read’</li> <li>• ‘Checked’</li> <li>• ‘Signed’</li> </ul>
preliminary	Indicates Data that is provided as information only.
changed	Change information. Allowed values are: ( ‘New’, ‘Unchanged’, ‘Revised’, ‘Deactivated’)
authorProfile	Profile of the author of the document – Predefined values are: <ul style="list-style-type: none"> <li>• Administrator</li> <li>• Dispatcher</li> <li>• CrewMember</li> </ul> Other values can be used
AuthorName	Name of the author of the document (string)
displayModel	Associated reference to a display model. This is valid only for documents of type (Text/XML)
transferPending	Declares that this document is missing for some reason.

## 10.2.9 Document associations

A document has several associations (XML elements) **which are described in the following table:**

**Table 5: Document Associations**

Element	Attribute	Definition
---------	-----------	------------

Parameter	name value	Used by the application to trace the context in which a document was generated (eg Performance calculation input parameters)
ParameterFile	file	Function similar to 'parameter' element except that all parameters are stored in a file
Signature	@date	-
	@type	
		These date are in fact elements and not attributes Specify the type of signature used
	LegalSign, or SimpleSign	Contains: Digest, Cert Contain: Digest, userId
Topic	@name	Represents a topic associated to this subfolder. (See topic concept for definition of list of topics)
CommentDocument		Optional commentary file associated to the document – several commentary files can be specified

### 10.2.10 Flight Folder Properties

Several information entries should be attached at the level of the eFF itself. The following table summarizes this information:

**Table 6: Flight Folder Properties**

Element	Attribute	Definition
EFUSUB Or EFDREP	@fullpackage	Indicates that this eFF is a fullpackage and not an update. If any eFF is found on the target system with the same flight identification, this new eFF should replace the eFF that was found.
	@ initialeFFGenerationTimestamp	Will be added in a future supplement
	@ prevTimestamp	Gives the timestamp (timestamp element of m633header) of the previous eFF. This attribute is mandatory when eFF is an update (i.e. there is a previous eFF) – it should be omitted if eFF is not an update but a full version of an eFF. When an eFF is an update, this data can be used to ensure that previous update is not missing in the update process. The application may choose to discard this update if the previous one does not have this value for timestamp.
	@templateRef	This identifies the reference of the eFF template model. This is to be used when importing an eFF from the ground using the template possibilities
	@state	Optional attribute - Denotes the current state in the Lifecycle of an eff. The following values are defined: <ul style="list-style-type: none"> <li>• “New” - denotes that the eFF has not been sent to an AC.</li> <li>• “Uplink” – the eFF is on transportation to the AC. This value is especially applicable by transport on physical media.</li> <li>• “OnAC” – the eFF is stored on the EFB and ready to be accessed by the applications.</li> <li>• “Active” – the eFF is on the EFB and currently used for operating a flight.</li> <li>• “DownlinkPartial” – the eFF is on transportation back to the ground. This value is especially applicable for transport on physical media and the flight is not finished (eFF will still evolve until flight closure (see below)).</li> </ul>

Element	Attribute	Definition
		<ul style="list-style-type: none"> <li>“DownlinkClosed” – the eFF was used for operating a flight in the past and is is on transportation back to the ground – eventually stored on the ground system.</li> </ul>
Subfolder		The eFF itself contains no folders in terms of a traditional file system. Subfolder elements in XML structure file can contain: <ul style="list-style-type: none"> <li>- Other subfolders</li> <li>- Documents</li> </ul>
M633Header		General information of the eFF message
M633SuppHeader		General information allowing to identify a flight
Signature	@timestamp	Signature generation date/time
	@type	
	LegalSign, or SimpleSign	These date are in fact elements and not attributes Specify the type of signature used  Contains: Digest, Cert  Contains: Digest, userId

The eFF flight identification is defined in [the ARINC 633 supplementary header](#) (see section 4.4.3) and the eFF.xsd schema. By the nature of mobile devices there will be multiple instances of an eFF **data packages** (different versions) at the same time at different locations. It is the responsibility of the involved systems to handle this circumstance correctly. For example, at a given time, the version of eFF **data package** onboard is older than the new eFF **data package** computed onground and pending for transmission. The version of eFF **data package** is identified by the timestamp of its generation. Additionnally, a given eFF **data package** contains the timestamp of the previous version. **The timestamps allow** the system to track all versions of **an eFF data package**.

If parts of an eFF **data package** are to be updated these updates are generated as another eFF **data package** with the same flight identification as the target eFF **data package**. The “Changed” attribute is used to control how the documents are to be merged with existing eFF.

## 10.2.11 eFF Schema

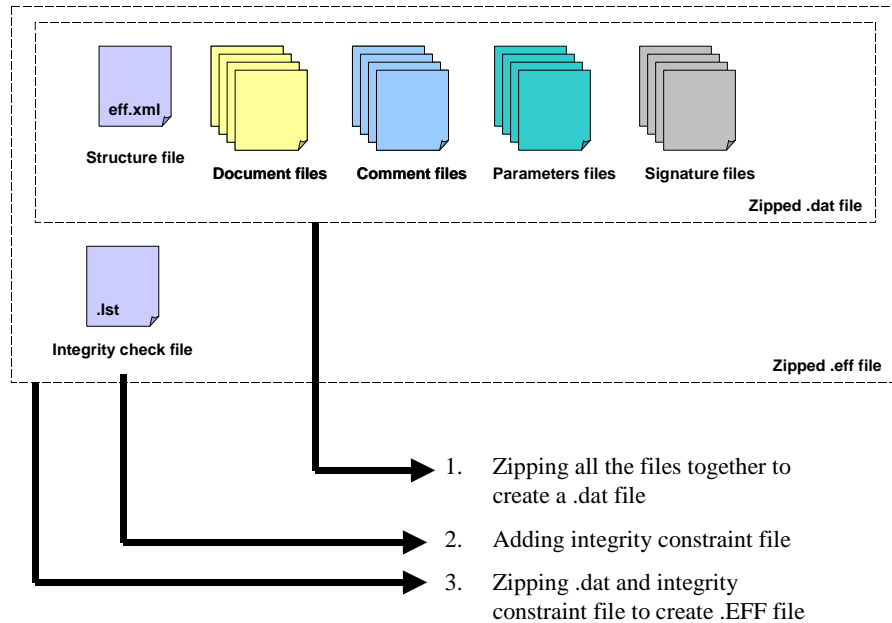
See electronic document ‘ARINC633 AOC.zip’ for details and examples of eff. Schemas are defined in schema/eFF/eFF.xsd.

The definition of the operational flight plan and the M633Header and M633SuppHeader are explained in section 5.2.2.

## 10.3 eFF File Creation

### 10.3.1 eFF Packaging

[Figure 3](#) shows the principle of .eFF packaging:



**Figure 3: eFF Package Format**

#### COMMENTARY

All of these files are embedded in a **single** zipped file (file extension '.eff') **for transportation convenience**. ~~These will easy carrying of flight information as all the files are stored in a single file.~~ This improves consistency by making sure ~~that we are not missing any filefiles are not~~ **missing** during the transportation **process**.

eFF should be transported in the form of a unique file. **The** same conventions apply whether the file is transported over IP connections or via physical media. eFF should be transported through a unique file with extension ".eff".

This file should be generated as follows:

- Creation of structure file: eff.xml file (compliant with eff.xsd' schema)
- All files referenced in eff.xml are gathered
- All these files are Zipped together in a ".dat" file
- An ".lst" integrity xml file is created (encompasses all files)
- A final '.eff' file is created, by zipping together ".dat" and ".lst" files
- The file is named as depicted in **Section 10.3.2**.

#### Zip algorithm

The compression format and algorithm that should be used for eFF compression are specified in RFC 1950 (ZLIB Compressed Data Format Specification version 3.3), and RFC 1951 (DEFLATE Compressed Data Format Specification version 1.3).

## COMMENTARY

At the time of writing ZIP algorithm can be found through the following hyperlink:

<http://java.sun.com/j2se/1.4.2/docs/api/java/util/zip/package-summary.html>

### 10.3.2 Structure of eFF File Name

When the file is created as described above it has to get a name that makes the most important properties of the eFF visible without unpacking. The eFF filename is defined in the following table:

**Table 7: eFF Filename Definition**

Sub field name	Description	Format (see table 4.1.2)	Example
flightNum or flightID	As defined in Table 4.4.2.2.2.	(1-10)C	"DLH1234ABC
Delimiter	One Underscore		" "
departureDate	Identifying flight date	YYMMDD	"050822"
Delimiter	One Underscore		" "
deplCAOCd or deplIATACd	3-letter IATA code or 4-letter ICAO code	XXX or (exclusive) XXXX	"FRA"
Delimiter	One Underscore		" "
destCAOCd or destIATACd	3-letter IATA code or 4-letter ICAO code	XXX or (exclusive) XXXX	"TLS"
Delimiter	One Underscore		" "
<a href="#">aircraftRegistration</a> <sub>(drm2)</sub> <a href="#">(pending Airbus input)</a>	The registration number assigned by the local authority,	Specific format is defined by the local authority and contains a combination of alpha-numeric string and an optional hyphen.	"N12345"
<b>Delimiter</b>	<b>One Underscore</b>		" "
eFF Update or Full eFF	'F' for Full or 'U' for Update	'F' or 'U'	'F'
DateTimestamp	YYMMDDHHMMSS		050822173214
File Type	One period and the letters "eff"		".eff"

Examples:

```
LH436_050822_DUS_ORD_D-ZYXW_U050822122634.eff
UA575_050822_ORD_SEA_N12345_F050821122634.eff
CFG3214D_050401_XRY_PMI_N5647A_U050402052658.eff
```

## COMMENTARY

There is a possibility of inconsistency in the assignment of the Flight Date. When the current flight is the second leg of a multi-leg flight, some airlines use the scheduled date of the initial leg. While the use of the date of the

current flight leg would be equally valid, it is essential that a consistent method be used within an airline.

### 10.3.3 Structure of the Integrity.xml File

This file references all the files in the package and provides a non cryptographic MD5 hash code for each of them to ensure their integrity, providing protection against unintentional modification.

In the integrity.xml file all contents of the eff.dat file are listed and the MD5 hash generated with the according file is given. So every application can easily check if the data in the eFF is corrupted.

The integrity.lst file should follow the structure of the sample given below:

```
<?xml version="1.0" encoding="UTF-8"?>
<hashfilelist version="1.0" toolVersion="1.0">
  <checkcode>bzGFcO/m05v8KFfZgEQ+lg==</checkcode>
  <hashedfile href="file1.xml">bro+2/iN72G15gj/1ZdvgA==</hashedfile>
  <hashedfile href="file2.doc">CbMgaSKuEKYwg5k74cG1IQ==</hashedfile>
  <hashedfile href="file3.txt">Ruxb8AIwEPc+XdtFCiwV4w==</hashedfile>
  <hashedfile href="file4.txt">VqzvypjT7DA2RBBRj2brHA==</hashedfile>
  <hashedfile href="file5.txt">ua0XS0ZpgE8wJcL+GbhXQw==</hashedfile>
  <hashedfile href="file6.txt">VqzvypjT7DA2RBBRj2brHA==</hashedfile>
  <hashedfile href="file7.doc">eFvP4D0RDo0XMxo4JnQIYQ==</hashedfile>
</hashfilelist>
```

where:

'version' attribute of hashfilelist represents the version of the integrity computation function, i.e. version 1 supports only MD5 hash.

'toolversion' attribute of hashfilelist represents the version of the tool that generates the integrity computation

<checkcode> represents the MD5 digest of the .DAT file, represented in base64 (see section 10.3.1).

<hashedfile href="file.txt"> represents the MD5 digest of the file mentioned in href.

## 10.4 eFF Communication Services

This eFF data package can be transmitted between airborne eFF application and ground application using the communication services defined in this section.

Three communication services are defined:

- **EFU**
  - eFF Uplink service allows an eFF data package to be sent by a ground tool to an EFB
  - eFF can be sent asynchronously or upon request (answer to EFR service)
- **EFD**
  - eFF Downlink service allows the eFF data package to be downlinked to the ground or to another EFB
  - eFF is sent asynchronously

- eFF
  - eFF request service allows the eFF airborne updates of an eFF data package
  - This service is synchronous (requires real time interaction between airborne and ground eFF applications)

**COMMENTARY**

Asynchronous means here that there is no guarantee that the receiving application is active and has received the data. However, the sending application may receive indication that the message was properly received by the ground communication system.

**COMMENTARY**

There exists a possibility that an UPDATE may arrive prior to reception of a FULL data package. This possibility will need to be considered in the design.

**COMMENTARY**

The use of 'EFUSUB' as root element denotes that this eFF **data package** was generated by a ground tool for use by an EFB, while the use of 'EFDREP' root element denotes that the eFF **data package** is generated by an EFB.

These services are fulfilled by the exchange of messages (see Table 8):

**Table 8: Service Elements**

Element Name	Gatelink Label	Service Version	DL/UL	Purpose
EFUSUB	N/A	1	UL	Eff Uplink- SUBmit : Ground tool submits a <b>FULL</b> or <b>UPDATE</b> eFF <b>data package</b> to an EFB
EFDREP	N/A	1	DL	Eff Downlink REPort: An EFB exports/sends eFF <b>data package</b> to the ground or other EFB
eFFREQ	N/A	1	DL	eFF uplink- REQuest Requests a <b>FULL</b> or <b>UPDATE</b> (with specific criteria (e.g. some topics only...)) eFF <b>data package</b> to be uplinked.

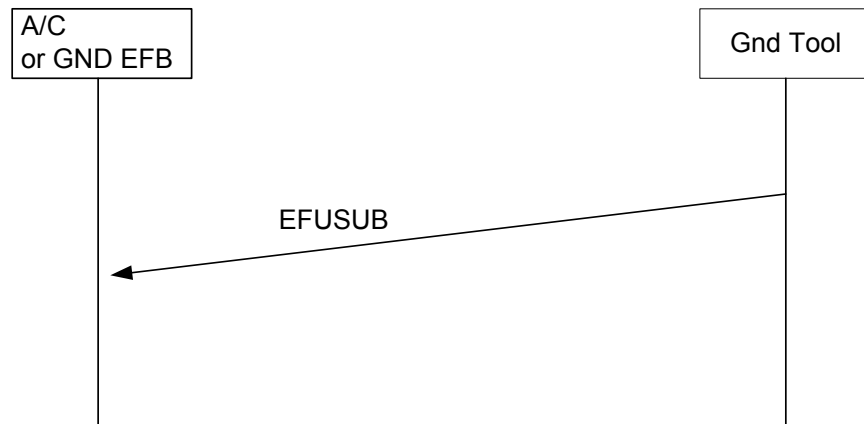
### 10.4.1 eFF Uplink Service (EFU)

This service is used to uplink an eFF data package to an EFB application (i.e. airborne eFF application or to a **Class 1 or 2 EFB application** on ground). The eFF can either be a FULL eFF or an UPDATE. **The EFU service only transmits and EFUSUB message from the ground to the EFB.** ~~consists in transmission from the ground to the aircraft of EFUSUB message.~~

EFUSUB message should contain the eFF data package to be sent to the aircraft.

EFUSUB can either be sent asynchronously, or following a specific request (eFFREQ).

Asynchronous transmission of EFUSUB message should be supported, while the support of synchronous EFUSUB message transmission following EFRREQ reception is optional.



**Figure 4: Typical EFUSUB Exchange**

### COMMENTARY

While eFF application level communications only consist in simple transmission of EFUSUB message, lower communication layers are in charge of ensuring reliable transmission of this message – these lower communication layers will thus involve bidirectional communications (initiated by the aircraft for security reasons).

#### 10.4.1.1 EFUSUB Message

Message: EFUSUB (version 1).

This message should be only sent as a .eff file in the appropriate format for the transport over physical media or IP communication links.

#### 10.4.1.2 Dynamic Aspects

For asynchronous transmission, EFUSUB message should be transmitted either through physical media (see section 2.3) or using IP air-ground communication links (see section 2.2).

Synchronous transmission of EFUSUB message (following reception of EFRREQ message) should be performed only through IP air-ground communication links (see section 2.2).

Several eFF files can be transported simultaneously over the same media (especially when a physical media is used). When the transport system will have sent the '.eff' files to the application, the eFF application should sort out the various files according to the flight segment to which they apply, the generation time stamp, and the 'fullPackage' attribute of the root element.

Upon reception of an EFUSUB message, the eFF airborne application should determine to which flight segment the eFF data package is associated, by using the M633SupHeader element received in eff.xml file.

Reception of a single eFF data package (one EFUSUB reception):

If a single eFF data package is received, the eFF airborne application should compose the eFF data package as follows:

- If the received eFF is FULL ('fullPackage' attribute of the root element is set), then this new eFF data package should replace the preceding one for same flight segment, if timestamp is more recent than last received eFF data package. The

eFF application should keep track of the modifications done on the preceding eFF data package (the way to retain the history of modifications is implementation dependent).

- If the received eFF is an UPDATE ('fullPackage' attribute of the root element is NOT set), then this new eFF data package should be added to the current one without impacting the documents and folders not updated in the received data package. The eFF application should keep track of the modifications done on the preceding eFF data package (the way to retain the history of modifications is implementation dependent).

Reception of multiple eFF data packages (several EFUSUB receptions):

For same flight segment ('flight' attribute of M633 header with same values), if several eFF files are received simultaneously, the eFF airborne application should compose the eFF data package as follows:

- Find the most recent FULL eFF data package received ('fullPackage' attribute of the root element is set)
- Ignore all data packages generated before the last FULL eFF data package
- Then add the more recent eFF data packages if they are 'UPDATE' data packages as defined in the single data package reception processing.

Example of multiple eFF files received (multiple EFUSUB messages received):

```
UA575_050822_ORD_SEA_F050821222634.eff
UA575_050822_ORD_SEA_U050821235712.eff
UA575_050822_ORD_SEA_U050822021246.eff
UA575_050822_ORD_SEA_U050822022355.eff
```

The eFF with the earliest time stamp must be the **FULL** eFF ~~which was fetched first~~ ('F' character in the filename and 'fullPackage' attribute set). The following eFFs are updates. By reading the eFFs **in** the order of their timestamps **an** eFF can be **composed** that has exactly the same status and contents as if all the updates happened via an IP network.

Non nominal cases

Reception of reception of invalid or out of sequence message:

- If the airborne application receives an erroneous message, for example EFDREP instead of EFUSUB, the application should ~~at least log locally the error, and, if supported by the application,~~ send a GERIND (general error message) to the ground system ~~if supported by the application. (e.g. ground administrator log).~~

#### 10.4.1.4 XML Format

The .eff file for EFUSUB message should contain a eff.xml file using the EFUSUB element as root element. This allows unambiguous determination that it is an uplink eFF. ,versionNb' attribute of M633Header element should be set to 1.

See electronic document 'ARINC633 AOC.zip' for details and examples. Schemas are defined in schema/eFF/eFF.xsd.

#### 10.4.1.4.1 Transportation Over IP

The service should here use the IP communication services defined in section 2.2.

#### 10.4.1.4.2 Transportation Via Physical Media

Uplink via physical media is nothing more than copying the eFF on a physical media, carrying the media to the EFB and then copying the eFF from the physical media to the EFB. The service should here use the physical media transportation rules defined in section 2.3.

#### 10.4.2 eFF Downlink Service (EFD)

This service is used to downlink an eFF data package to the ground system or to another EFB (export-import function). The eFF can either be a FULL eFF, or an UPDATE. EFD service only consists in transmission from the aircraft to the ground of EFDREP message.

EFDREP message should contain the eFF data package to be sent to the ground.

EFDREP should always be sent asynchronously.

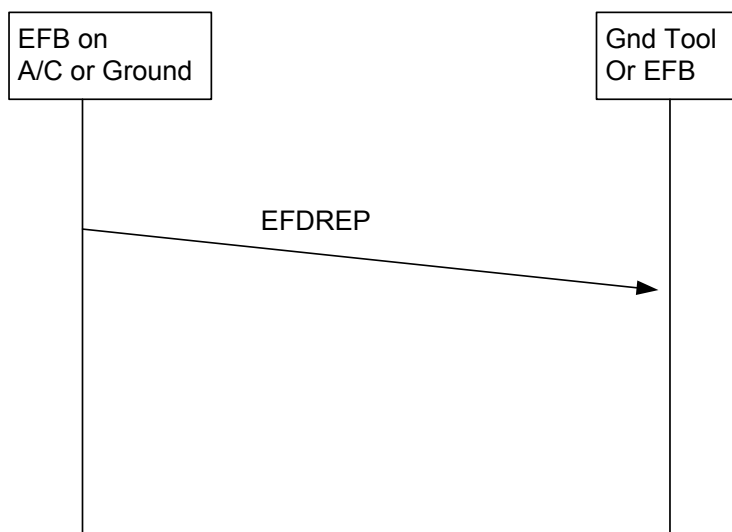


Figure 5: Typical EFDREP Exchange

#### COMMENTARY

#### Typical EFDREP exchange

#### COMMENTARY

While eFF application level communications only consist in simple transmission of EFDREP message, lower communication layers are in charge of ensuring reliable transmission of this message – these lower communication layers will thus involve directional communications (initiated by the aircraft for security reasons).

#### 10.4.2.1 EFDREP Message

Message: EFDREP (version 1).

This message should be only sent as a .eff file (packaged as per Section 10.3) in the appropriate format for the transport over physical media or IP communication links.

#### 10.4.2.2 Dynamic Aspects

EFDREP message should be transmitted asynchronously (i.e. without setting up a transaction with the ground application); either through physical media (see section 2.3) or using IP air-ground communication links (see section 2.2).

#### 10.4.2.3 XML Format

The .eff file for EFDREP message should contain a eff.xml file using the EFDREP element as root element. This allows unambiguous determination that it is a downlink eFF. ,versionNb' attribute of M633Header element should be set to 1.

##### 10.4.2.3.1 Transportation over IP

The service should here use the IP communication services defined in section 2.2.

##### 10.4.2.3.2 Transportation Via Physical Media

Uplink via physical media is nothing more than copying the eFF on a physical media, carrying the media to the EFB and then copying the eFF from the physical media to the EFB. The service should here use the physical media transportation rules defined in Section 2.3.

#### 10.4.3 eFF Request Service

Up-to-dateness is essential for the usage of eFF data or to in other words: The documents in an eFF are aging rapidly. The document properties "ExpectedUpdateAt" and "TimeToLive" should be used to display the pilot when a document is outdated. In various cases the pilot might be willing to update the contents of the eFF without such a notification (e.g. WX).

Thus there **has** to be a possibility to get updates for an eFF that was fetched earlier. This update has to be transferred over potentially slow or expensive connections which demands the possibility to update only parts of the eFF or even single documents.

This service (eFF request service) allows the eFF airborne application to request an eFF data package update. The answer to this request should be an eFF data package tagged 'UPDATE' (i.e. not a FULL eFF), or a FULL eFF package (i.e. tagged 'FULL').

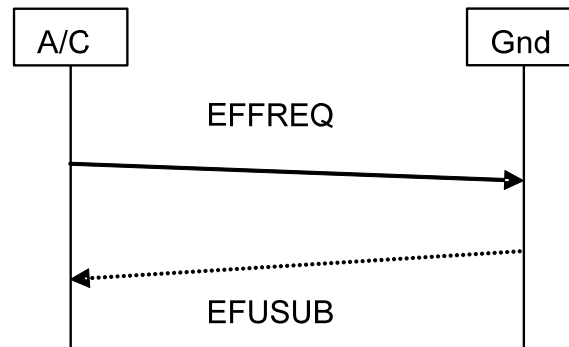
The eFF request service should thus allow to request:

- **Full** eFF data package containing all **available** documents (even those that did not change since generation of last eFF data package on the aircraft). **Some documents may be marked as pending.**
- **Update of ALL modified/new documents** in a given eFF data package (identified by its flight segment and timestamp) – the answer should be a 'UPDATE' eFF data package containing all new or modified documents since the generation of the eFF data package referenced in the request. The answer should be a 'UPDATE' eFF data package containing only the documents required in the request.
- **Selective updates:** updates of documents of a given list of topic, or update of an explicit list of documents of a given eFF data package (identified by its flight segment and timestamp). The answer should be a 'UPDATE' eFF data package containing only the documents required in the request.

The key to partial updates of an eFF is the XML-Structure file – this information is common to the ground system and the EFB – as the ground system deliver this information to the EFB.

Parts of the eFF are referenced using the topic identifier. If a topic identifier is given in a update request all documents marked with this topic identifier are checked if they are outdated.

If a document ID is given only the referenced document is delivered in the update. Other criteria, such as 'mandatory' attribute of a subfolder, can be used.



**Figure 6: Typical eFFREQ Exchange**  
~~Typical eFFREQ exchange~~

#### 10.4.3.1 eFFREQ Message

Message: eFFREQ (version 1).

This message should be only sent as a .xml file (i.e. NOT packaged as per Section 10.3) for the transport over IP communication links.

#### 10.4.3.2 Dynamic Aspects

The answer to an update request should always be an eFF with complete information in the eff.xml structure file (EFUSUB message of EFU service). The eff.xml in this package has the 'changed' flag set properly for all files. If no full eFF is requested, documents are included only when their 'change' status is "new" or "revised".

The aircraft can specify in the request a priority level associated to the requested documents/topics for requested eFF. 3 levels are defined and, when specified, the ground should use the priority attribute as a hint to package eFF data and answer the request (trying to send first higher priority data) .

A 'document' (file tagged as 'document' in the eFF structure by using the 'document' element in the eff.xml file) can be associated with various files (if necessary), in which case these are mentioned in the eFF ('struct.eff' file):

- The document content itself
- The associated parameters (if any)
- The associated signature (if any)
- The associated comments (if any)

With this package the EFB does not only receive a new document or new documents of the requested topic, it receives complete information about the status of all documents in the eFF.

Please note that the server which delivers the updates has to track which file was sent to whom to be able to compose the appropriate updates.

### 10.4.3.3 XML Format

The eFFREQ message should be embedded in an XML file, with eFFREQ as root element only.

See electronic document 'ARINC633 AOC.zip' for details and examples of eFF request. eFF request is defined in schema/eFF/eFF.xsd.

#### 10.4.3.3.1 Request Associations

A request has several associations (see XML schema for more details).

**Table 9: Request Associations**

Element	subelement	attribute	Definition
		@previousTimestamp	Time of last received eFF
FullPackage OR			
UpdateRequest		@pendingDocument	Request only pending documents
		@mandatoryDocument	Requests only mandatory document
	Document	@id	Document = optional element
		@transportPriority	
	Topic	@transportPriority	Topic=Optional element

All criteria are 'AND associated', i.e. if 'mandatory' is set, and topic(s) are requested, then the answer is: all mandatory documents PLUS all documents (mandatory or not) of the requested topics.

#### 10.4.3.3.1 Transportation Over IP

The service should here use the IP communication services defined in section 2.2.

#### 10.4.3.3.2 Transportation Via Physical Media

Not recommended – can be used if needed – however, the EFUSUB message (eFF sent from ground) could be conveyed by physical media. The airborne system should handle these situations.