FATM Message Translator

AFTN ATS/OLDI Message Translator

Contract Number: **TBD**

Prepared for:

Prepared By:

Flight ATM Systems Ltd.

Referenced Documents

No.	Identification	Rev Date	Edition	Name	Source
[1]	Doc 4444, ATM 501,	22 Nov 2007	15th Edition	Procedures for Air Navigation Services	ICAO
[2]	EUROCONTROL - SPEC- 0107	20 Oct 2007	Edition 3.0	EUROCONTROL SPECIFICATION for ATS Data Exchange Presentation (ADEXP)	EUROCONTROL
[3]	EUROCONTROL - SPEC-0106	16 Dec 2010	Edition 4.2	EUROCONTROL Specification for On-Line Data Interchange (OLDI)	EUROCONTROL
[4]	IFPS/DOM/IFPS	18 Nov 2011	Edition 2.003	IFPS and RPL Dictionary of Messages; ICAO 2012 Special edition	EUROCONTROL
[5]	BASIC CFMU HANDBOOK	20 Apr 2009	Edition 13.0	AIR TRAFFIC FLOW & CAPACITY MANAGEMENT OPERATIONS ATFCM USERS MANUAL	EUROCONTROL
[6]	TBD			Interface Design Document	
[7]	TBD			Translator Tests	
[8]	TBD			Software Requirement Specification	
[9]	TBD			CADF Message Description	EUROCONTROL
[10]	URB/URD/2012_ REQ	25 June 2012	Edition 1.42	CFMU 2012 REQUIREMENTS	EUROCONTROL
[11]	DPS.ET1.ST06-STD-01-01	Dec 2001	Edition 2.1	EUROCONTROL Specification for On-Line Data Interchange (OLDI)	EUROCONTROL
[12]	DPS.ET1.ST06-STD-01-01	Dec 2001	Edition 2.3	EUROCONTROL STANDARD DOCUMENT FOR ON-LINE DATA INTERCHANGE (OLDI)	EUROCONTROL
[13]	Doc 4444-RAC/501	7 Dec 1996	13th Edition	PROCEDURES FOR AIR NAVIGATION SERVICES RULES OF THE AIR AND AIR TRAFFIC SERVICES	ICAO

Table of Contents

List of Figures

List of Figures

1 Document Overview

1.1 Identification

Program:	FATM Message Translator
Document Name:	AFTN ATS/OLDI Message Translator
Document Number:	FATM-TRANS01
Revision:	A0.02
Revision Date:	Monday, 21 September 2015
Document Owner:	Peter Venton – Flight ATM Systems Ltd.
File Name:	FATM AFTN Translator V2.0.doc

1.2 Purpose

This document provides a description of the Flight ATM Systems AFTN ATS/OLDI Message Translator.

This document provides a functional description of the Flight ATM Systems AFTN ATS/OLDI Message Translator that is able to translate:

- ATS ICAO format messages into ADEXP format;
- ATS ADEXP format messages into ICAO format;
- OLDI ICAO format messages into ADEXP format;
- OLDI ADEXP format messages into ICAO format;
- Pre-FPL 2012 version messages into Post-FPL 2012 format (irrespective if message format is OLDI or ADEXP);
- Post-FPL 2012 version messages into Pre-FPL 2012 format (irrespective if message format is OLDI or ADEXP);

1.3 Conventions

1.3.1 Requirement Numbering

The software requirements are numbered according to the following schema:

<Requirement Number> - <Short Requirement Description>

<Requirement Text>

<End of Requirement Marker>

The requirement numbering shown above are described in the following table:

Item	Description
Requirement Number	A unique atomic requirement in the form 'SRS-xxx' where 'xxx' is a number in the range 001 to 999. Where requirements relate to identical functionality but are referring to a different set of messages, a suffix 'a', 'b', 'c' etc. will be added to indicate that the functionality is applicable to 'n' messages where the requirement has an identical number.
Short Requirement Description	A short description of a requirement.
End of Requirement Marker	Indicates the end of a requirement with a hash '#' character.

Example

SRS-001 – AFTN Connection

The AFTN Translator shall be able to connect to the customers AFTN in order to receive and transmit AFTN ATS messages onto the AFTN.

#

1.3.2 Acronyms

Acronyms are never defined in line within the text; a complete of definitions for the acronyms used in this document can be found in the last appendix of this document.

1.4 Document Overview

This document contains a description of the ATS and OLDI Message Translator (AOMT) functionality. The AOMT is implemented using two primary software packages:

- The FATM ATS Message Parser software library;
- Proprietary software utilizing the message parser that implements the AOMT functionality;

This document provides a description for each of these components. The description of the parser library functionality does not go into great detail, as the software library is treated as a functional block that implements message parsing according to the relevant standards.

The proprietary software implementing the AOMT which uses the parser software library is described in more detail and includes comprehensive information about the supported messages and the types of translation that are supported by the AOMT as a whole.

1.5 Translator Overview

The AOMT is able to parse both OLDI and AFTN messages that are exchanged on the AFTN and/or peer-to-peer OLDI links and translate these into a target message format. In the ATM environment, there are two message formats in common use:

- **ICAO** – this format consists of messages with a fixed structure represented by a number of fields separated with a hyphen character. The structure and number of fields is fixed for a given message title. The ICAO format is defined in [1].
- **ADEXP** – this format supports a message structure with variable numbers of fields where the placement of the fields is irrelevant. The ADEXP format is defined in [2].

There exist a number of different message types that can occur in either ICAO and/or ADEXP formats, namely:

- **ICAO ATS Messages** – These represent a basic set of messages defined by ICAO to support the exchange of flight plan data around the world on the AFTN. The complete set of ATS messages is defined in [1] and consists of 16 messages in total.
- **CFMU IFPS ATS Messages** – The CFMU expanded the ICAO message set in both the number of messages and their content. These messages are defined in [4].
- **CFMU ETFMS Messages** – The CFMU added an entirely new set of messages that deal with slot allocation for aircraft departures to avoid traffic congestion. These messages are defined in [5].
- **CFMU CADF Messages** – The CFMU CADF administer the opening and closing of airspace and send this data using a number of specialised ADEXP format messages to ATC units and AO's. These messages are defined in [9].
- **OLDI Messages** – These messages deal with inter-ATC unit flight plan coordination. These messages are defined in [3].

The following table defines the message formats supported for the different message types.

Message Type	ADEXP	ICAO
ICAO ATS	No	Yes
CFMU IFPS ATS[1]	Yes	Yes
CFMU ETFMS	Yes	No
CFMU CADF	Yes	No
OLDI	Yes[2]	Yes

Table 1 - Message Types and Formats

In 2012, the ICAO specification [1] was the subject of a major change to modify the content of ICAO field 10 to include support for more detailed descriptions of aircraft navigation and communication equipment carried on board aircraft – this change is commonly referred to as the FPL 2012 change. This change introduced the concept of two different versions of flight plan data, namely pre and post FPL 2012 versions. Messages are only impacted by this change if they include ICAO field 10 or the ADEXP equivalent (CEQPT and SEQPT). In the OLDI messages, field 10 data is given in OLDI fields 80 and 81. There are additional impacts related to ICAO field 18 caused by the field 10 change, but the field 18 changes are a result of the changes made to field 10 and not a change in their own right.

[1] These include the ICAO ATS messages plus all additional message titles defined by EUROCONTROL CFMU IFPS
[2] Not all OLDI messages are supported in both formats, in general, messages defined earlier in the OLDI specification were defined in ICAO formats, with later editions of the specification adding ADEXP definitions for the earlier messages as well as adding messages defined only in the ADEXP format.

The AOMT is able to translate messages from one message format to another (ADEXP ⇔ ICAO) and translate message versions (pre ⇔ post FPL 2012). The translator is able to perform combinations of translation; for example, a pre-FPL 2012 ICAO format message can be translated to a post-FPL 2012 in ADEXP format.

When translating OLDI messages, there are cases where the translation must also include translating from a newer to older version. In general, newer versions of messages contained more data as well as more message titles. Translating from older versions to newer version may not always be possible if the data required for a newer version is not present in an older version of a message.

The translator uses the Flight ATM ATS Message Parser to decompose a message into its constituent fields; the parser parses all ADEXP and ICAO messages specified in [1], [2], [3], [4], [5] and [9]. The parser input is a message and its output is an XML document containing the individual message fields. The translator software implements a wrapper around the message parser that builds an output message according to a format and version specified by a caller. The output also includes any errors detected by the parser while parsing a message. It is a caller's responsibility to handle any errors in an appropriate manner. The following figure provides an overview of the translator internal processing steps.

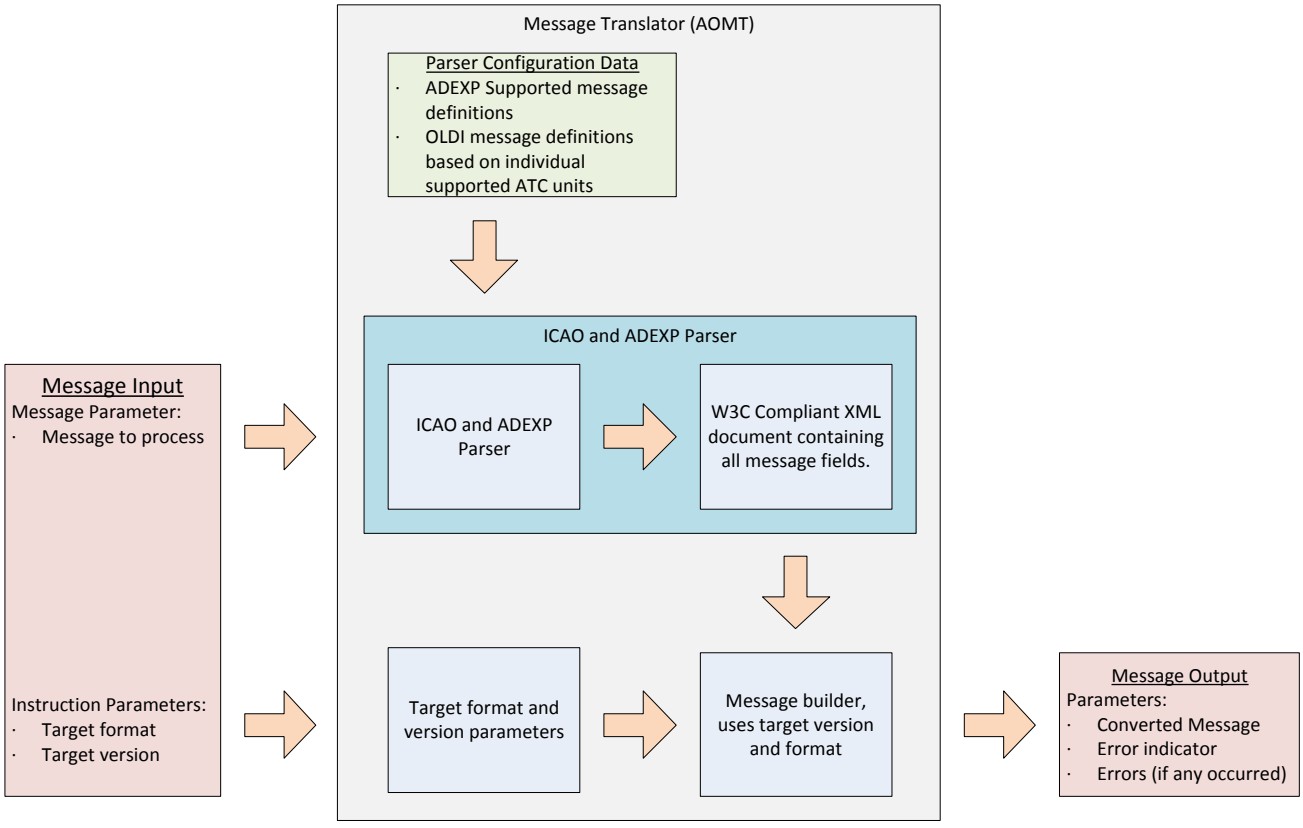

Figure 1 - Translator Logical Data Flow

The translator is supplied as a Java™ software library (.jar file) that runs as a network service with a TCP/IP interface. Refer to the IDD [6] for details on the interface and the API.

The translator library is tested using JUnit tests; the tests are described in [7].

1.5.1 Limitations

It is to be noted that the SVC and channel check messages are not processed by the AOMT as these are messages that the protocol handler deals with; these messages should never be past up to the application layer being handled by the AOMT.

2 Requirements

2.1 AOMT Overview

As mentioned above, the AOMT software is implemented using the Flight ATM ATS Message Parser to parse both ATS and OLDI messages. The AOMT implements a translation wrapper around the message parser that takes as input, instructions on the type of translation to perform along with the message to process. The translator software sends the message to the parser and re-builds the output by using the parser output and the processing instructions to build a message as requested by a caller.

To summarise, there are three types of translation to consider:

- Format translation:
  - Translate an ADEXP format message into an ICAO format message;
  - Translate an ICAO format message into an ADEXP format message;
- FPL 2012 version translation:
  - Translate a pre-FPL 2012 version message into a post-FPL 2012 version message;
  - Translate a post-FPL 2012 version message into a pre-FPL 2012 version message;
- OLDI message version translation:
  - Translate an OLDI version 2.x message into a new version 3.x message;
  - Translate an OLDI version 3.x message into a new version 2.x message;

Combinations of the three translation modes listed above can occur simultaneously when translating a message. Note that a further possibility of old/new ATS versions is not listed as the content of the ATS ICAO messages has not change significantly apart from the FPL 2012 changes and for the ADEXP formats, the standards specify a more tolerant approach to unknown fields in a message and hence they need not be considered for additional translation processing.

In order to understand how the AOMT works, the Flight ATM ATS Message Parser functionality must be understood and is described in the following section. Section 2.1.2 describes how the AOMT works as a whole in conjunction with the Flight ATM ATS Message Parser to generate the required output messages.

2.1.1 Flight ATM ATS Message Parser Description

This section describes the Flight ATM ATS Message Parser and provides a broad overview of the parser functionality. This parser overview provides necessary background information on the concepts used to implement the parser as well as describing the configuration data needed to correctly configure the parser.

This section is not written as a requirement document and does not attempt specify topics in an atomic form, it is included in order to understand the AOMT functionality as a whole.

The parser is able to process OLDI, ATS, ETFMS and CADF messages in either ICAO or ADEXP formats (not all messages are supported in both formats).

The parser can be configured to process any type of ADEXP message. The ATS ICAO format message support is fixed and cannot be modified using configuration data. The content of OLDI ICAO format message is bilaterally agreed between adjacent units, as a result of which, the parser supports the definition of OLDI message content per adjacent unit.

By default, the parser is delivered with a pre-configured set of ADEXP and ICAO messages. The message titles supported for a given message type and format are given in the appendices as shown below.

Message Type	Message Format	Message titles describe in Appendix…
ATS	ICAO	A
CFMU IFPS ATS	ADEXP	B
CFMU ETFMS	ADEXP	C
OLDI	ADEXP & ICAO	D
CFMU CADF	ADEXP	E

Table 2 - Supported Messages

2.1.1.1 Flight ATM ATS Message Parser Overview

The parser consists of two distinct parsers, one processes the ICAO format, the other, the ADEXP format messages. Upon receiving a message, arbitration software determines the message format based on the presence of an open bracket or a hyphen as the first character in the message body.

If there is a hyphen found before any open bracket then the message is flagged as an ADEXP message, if an open bracket is found before any hyphens then it is flagged as an ICAO format message. Based on this decision process, a message is handed off to the appropriate parser.

The ICAO parser is implemented as a finite state machine; for the ICAO ATS messages the state machine is not configurable as it was decided during the design phase that the ICAO ATS message format is relatively stable and few custom messages are implemented within the industry at large in the ICAO format. However, within the software the parser is highly configurable and can be supplied with custom messages if required.

For the OLDI messages in ICAO format, the message structure is dependant on agreements between any two adjacent units; hence the message content is different per adjacent unit. The parser contains configuration data that supports the definition of OLDI message content per adjacent unit. The supported adjacent units are specified in the same configuration data as the OLDI message content.

The ADEXP parser is implemented as a data driven parser with the supported messages and content defined in a set of offline configuration files. Within the ATM domain, custom messages are often implemented in the ADEXP format; hence at the design stage it was decided that a flexible approach had to be made in order to support custom message definition.

The ADEXP message descriptions are defined in four XML configuration files referred to in this document as the descriptor files. A fifth file defines the error messages reported by the parser.

The parser produces a W3C complaint XML document with its internal structure based on the ICAO fields contained within a message. The ADEXP format contains many fields that have no ICAO equivalent; when the ADEXP parser has completed parsing, the result from the ADEXP parser is ‘copied’ into the W3C output document. The definition of the ADEXP fields to copy to the output document is also defined in the ADEXP configuration files. Configuration options exist to copy all ADEXP fields in a message to the output or individually selected fields.

Any errors detected during processing are included in the W3C XML output document.

2.1.1.2 Support for FPL 2012

The parser supports processing for messages in the FPL 2012 format. The parser is able to determine the message 'version' (either pre or post FPL 2012) based on message content. It is important to note that there are situations where the version cannot be determined. The detected version is written out to the XML document; it is a caller's responsibility to check the version and take appropriate action as required. The parser includes errors in the output relating to FPL 2012 errors.

The AOMT translator wrapper uses the FPL 2012 information in the W3C output document to take appropriate action when constructing the output message by translating the message as necessary.

2.1.1.3 Parser Technical Features

The ATS message parser requires a JRE 1.8.x to run. Other technical features include:

- Written in Java™;
- Platform independent;
- Output generated in an open standard W3C XML document;

2.1.1.4 Parser Data Flow

The following figure illustrates the data flow through the parser and the association of the configuration data with the various internal software components.

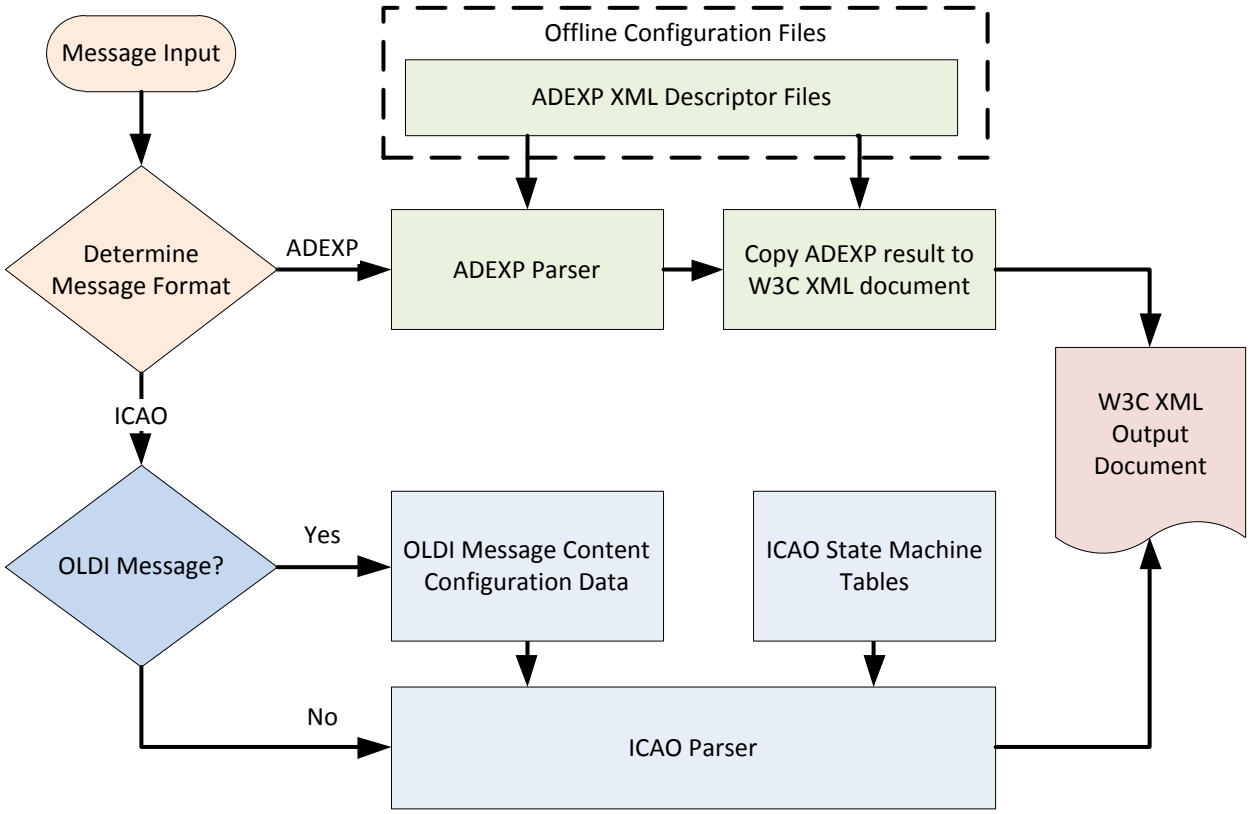

Figure 2 - Parser Data Flow

2.1.1.5 Error Handling

Errors detected by the parser are included in the W3C XML output document; it is the caller's responsibility to check if errors are present. The AOMT retrieves any errors and includes them in a translated message returned to a caller. When an error is detected, and the error is ***not*** listed in configuration data as 'to be ignored for translation', translation will be inhibited and the erroneous message returned to the caller.

2.1.1.6 ADEXP Parser Configuration Files

There are four configuration files that define the supported ADEXP message titles and their content. These files can be used to configure custom ADEXP messages[3].

There is also a fifth XML file that contains the text for the global errors reported by the ADEXP parser, further information about these error messages can be found in section 2.1.1.6.6 titled 'Global Errors'.

An overview of the ADEXP parser configuration files is provided in the following table, a more detailed description of each file is provided in the following sub-sections:

File Name	Description
`ADEXP_Supported_Messages.xml`	Contains a list of the ADEXP message titles and for each title, the ADEXP fields that make up the message;
`ADEXP_primary_fields.xml`	Contains a list of ADEXP primary fields that are currently specified by EUROCONTROL;
`ADEXP_subfields.xml`	Contains a list of ADEXP sub-fields that are currently specified by EUROCONTROL;
`ADEXP_auxiliary_terms.xml`	Contains a list of ADEXP auxiliary terms that are currently specified by EUROCONTROL;
`Global_Error_Text.xml`	Contains global errors reported by the parser;

Table 3 - Descriptor Files Overview

Refer to the IDD [6] for information on the location of the ADEXP descriptor files within the delivered software deployment.

[3] Extreme caution must be taken when altering these files, as inconsistencies in the data will lead to run time exceptions being raised in the parser.

The relationship between these files is shown in the following diagram.

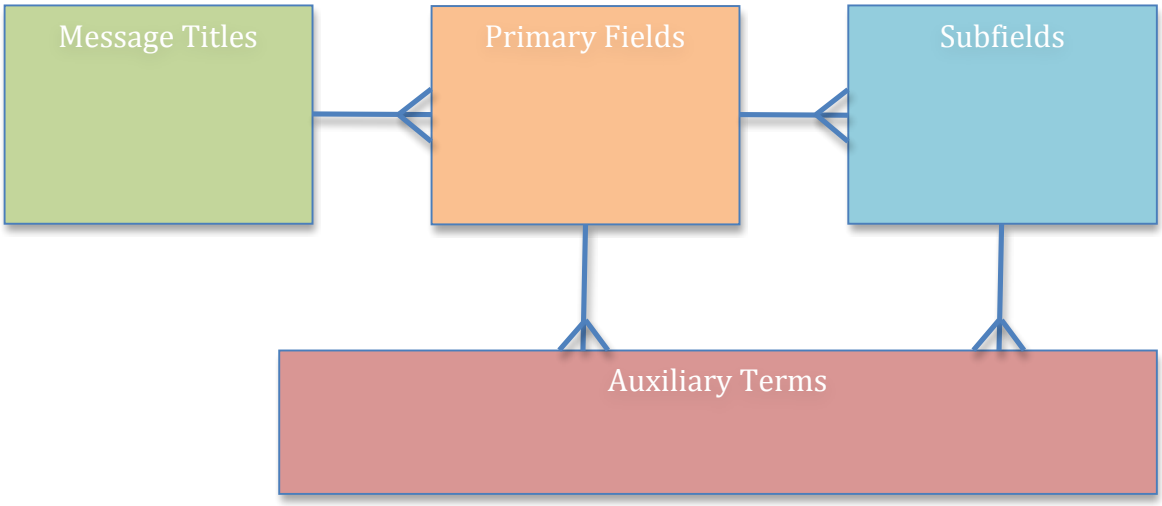

Figure 3 - Descriptor Files Relationship

2.1.1.6.1 Supported Messages File

This file specifies the message titles to be supported by the ADEXP parser. For each message title a list of primary fields is specified from which a message is comprised.

This file has the structure shown in the following figure.

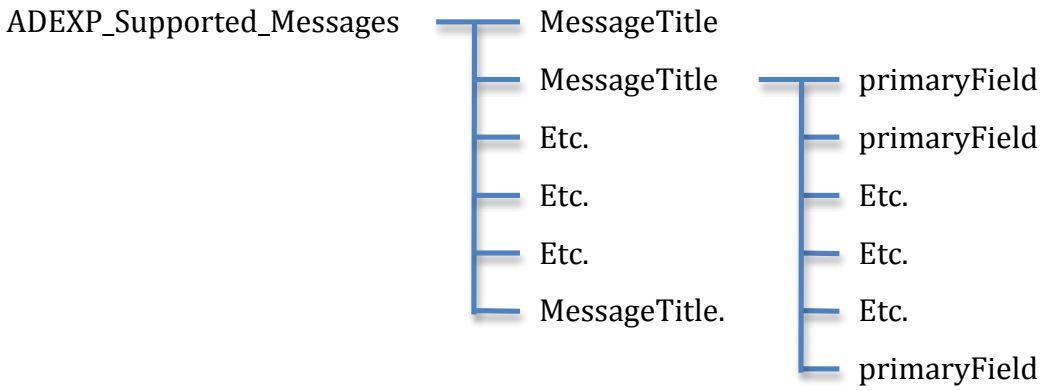

Figure 4 - Message Title Descriptor File Structure

Each of the primary fields specified must have a corresponding entry in the **`ADEXP_primary_fields.xml`** descriptor file.

The order in which the fields appear is not important, although for clarity it is recommended that they be entered in alphabetical order. Some of the messages contain many fields and it is easier to locate a field when sorted alphabetically. A number of attributes can also be specified on the XML nodes, these are described in the following table.

Node Name	Attributes	Description	Valid Values
MessageTitle	name	The name of a message title such as ‘ACK’ or ‘IFPL’.	The name of a message title as a string;

Node Name	Attributes	Description	Valid Values
	adexpOutput	Informs the parser to output all field of the ADEXP message to the W3C XML output. If this is 'false' then ADEXP fields that do not have an ICAO equivalent field name, are ignored and will not appear in the W3C XML output. This attribute is optional.	True or False
	isOLDImessage	Specifies the message type, true indicates that the message is an OLDI message, false indicates an ATS message. If the attribute is omitted, false is assumed and the message is flagged as an ATS message. This attribute is optional.	True or False
	icao	Provides the ICAO message title, IFPS ATS messages in ADEXP format typically have an 'I' preceding the ICAO message title, i.e. FPL is IFPL. This attribute provides a means to specify the ICAO name for any IFPS ATS message. This attribute is optional.	An ICAO message title
	oldiversion	Specifies the OLDI version of a message. Functionality to process this attribute is currently not implemented, the attribute is reserved for later use and definition. Any OLDI message omitting this attribute is considered to be OLDI 2.x.	An OLDI version number given as 4.x.
primaryField	name	The name of a primary field; this name must be defined in the descriptor file containing the primary field descriptions. If the primary field is a primary list field the 'BEGIN' keyword must not be entered, the parser knows which primary fields are list fields, (more about this is described in the primary field file description).	Any primary field name that exists in the Primary Field descriptor file.
	min	The minimum number of occurrences that this field is allowed to appear in a message, typical values are: 0 – Indicates that the field is optional; 1 – Indicates that the field is compulsory and must appear at least once, if it is missing the parser will report an error. If the 'max' attribute is set to 1 then this implies a single compulsory occurrence of the field is required;	Integers in the range 0…n
	max	The maximum number of occurrences that this field is allowed to appear in a message, typical values are: 1 – The field cannot occur more then once, if the 'min' attribute is also set to 1 then this implies a single compulsory occurrence of the field is required; -1 – The field has no maximum occurrence limit and can appear any number of times;	Integers in the range -1, 0…n

Node Name	Attributes	Description	Valid Values
mutuallyExclusive	rule	Specifies fields that are mutually exclusive; such fields must be defined as optional and then be listed in this section as mutually exclusive. Any message definition can optionally include an additional tag called <mutuallyExlusive> with an embedded tag <rule> used to specify one or more mutually exclusive fields in a message. For example, if a field is mutually exclusive with another field, i.e. MACH | SPEED then the message definition should include: <mutuallyExlusive> <rule>SPEED | MACH</rule> </mutuallyExlusive> The <rule> tag can be repeated to specify more than one set of mutually exclusive fields. Any field specified in a <rule> tag must be specified with min=0 amd max=1 or max=-1 (i.e. as an optional field, one to 'n' times).	A logically ‘or’ set of primary field names.

Table 4 - Supported Messages File Nodes and Attributes

2.1.1.6.2 Primary Fields

This file specifies the primary fields supported by the ADEXP parser. The primary fields come in 3 different types:

- Basic – Key value pair fields, this is the simplest type of field;
- List – Begins with the ‘BEGIN’ keyword and ends with the ‘END’ keyword. The order of the items contained within the BEGIN/END is important and must be maintained. The fields specified within a BEGIN/END block can be any primary and/or subfields; this includes nested list fields, (recursive).
- Structured – Contains an un-ordered set of sub-fields

This file is structured in three sections to reflect the 3 different types of primary fields. The basic fields are the simplest, represented as key-value pairs in a message. These are defined in this file with one or more pointers to the auxiliary terms file. The auxiliary term file specifies the regular expression with which to parse the value part of a key-value pair; typically only one term is specified, but in some cases an ‘option’ is provided to cover more than one possible definition.

The list and structured fields contain pointers to the fields that they themselves contain. It is to be noted that recursive structures can be defined where a list or structured field my include another list or structured field. This implementation is according to the EUROCONTROL ADEXP specification, which does specify nested levels of field descriptions.

is file has the structure shown in the following figure.

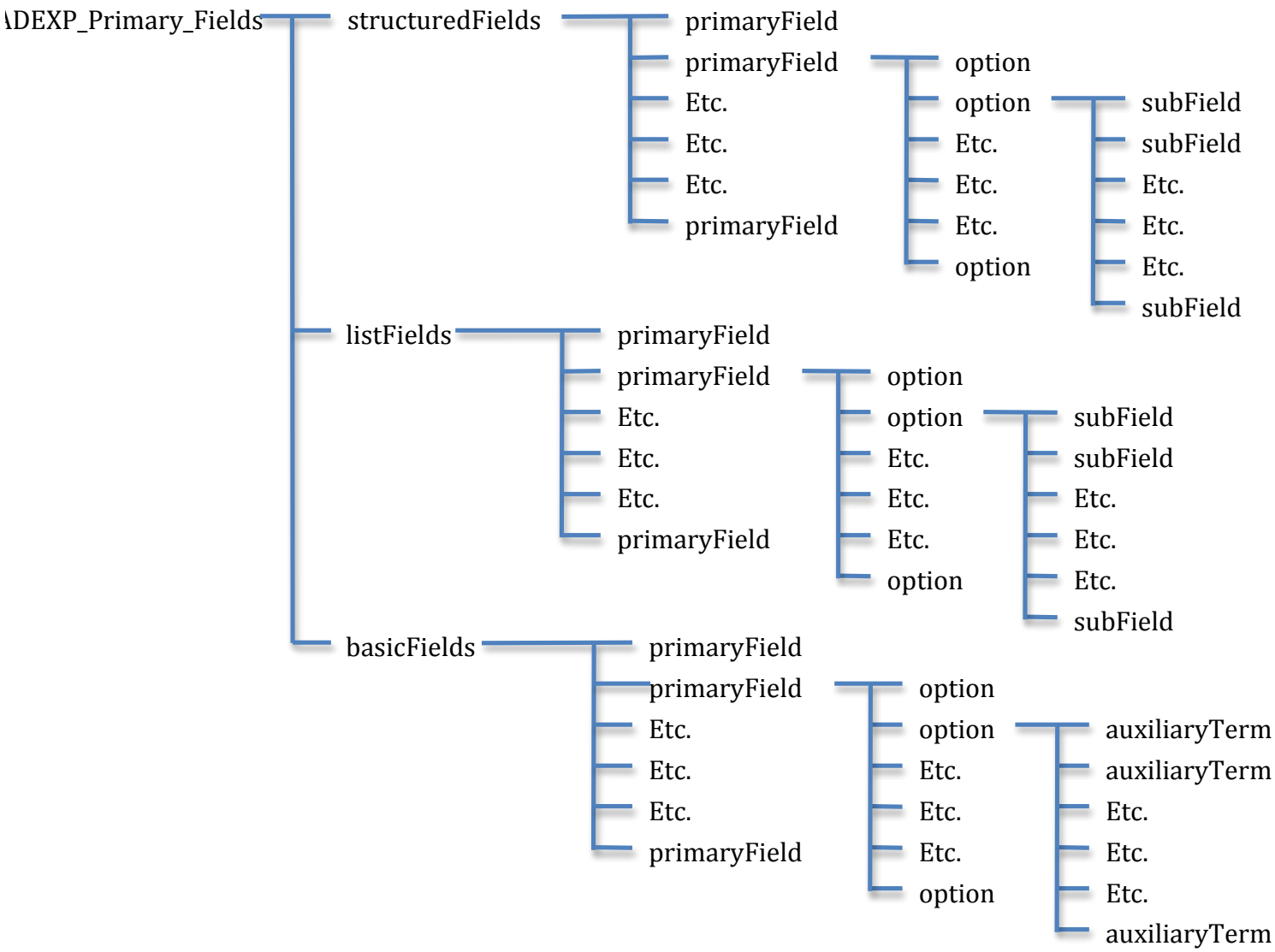

Figure 5 - Primary Field Descriptor File Structure

Each of the subfields specified must have a corresponding entry in the **`ADEXP_subfields.xml`** descriptor file. Each of the auxiliary terms specified must have a corresponding entry in the **`ADEXP_auxilliary_terms.xml`** descriptor file.

The order in which the fields appear is not important, although for clarity it is recommended that they be entered in alphabetical order to make it easier to locate a field. A number of attributes can also be specified on the XML nodes, these are described in the following table.

Node Name	Attributes	Description	Valid Values
structuredFields	None	This is a single node that contains all the primary structured fields as child nodes.	N/A
listFields	None	This is a single node that contains all the primary list fields as child nodes.	N/A
basicFields	None	This is a single node that contains all the primary basic fields as child nodes.	N/A

Node Name	Attributes	Description	Valid Values
primaryField	name	The name of a primary field; If the primary field is a primary list field the 'BEGIN' keyword must not be entered, the parser knows which primary fields are list fields because of its placement within the 'structuredFields' or 'listFields' nodes. The primary field nodes contain one or more option nodes, which in turn contain either a subfield or auxiliary term. The option nodes provide a means to: Offer one or more alternative combination of subfields (this is particularly true for the structured fields as the ADEXP specification often describes several variations on the content of a structured field). Offer one or more alternative auxiliary terms.	Any field name specified as: [A-Z0-9]+
	icaoname	If this attribute is present, the parser will copy the data from the primary field to the W3C output document to a tag with the name as entered for this attribute. For example, if the attribute is set to 'icaoname="F13,a"' then the value is copied to the 'fields/F13/a' node in the output document. By default the parser is delivered with this attribute set on all ADEXP primary fields that have an equivalent ICAO field to ensure that irrespective of the format of a message, that the ICAO content is contained in the output document. If no ICAO subfield is specified (the comma and last letter(s) is not present) then a node is created as a child of the 'fields' node. This can be a useful mechanism to copy data from ADEXP fields that have no equivalent ICAO field. For example, the CTOT field from an ETFMS message may be wanted in the W3C output document. If the attribute 'icaoname="CTOT"' is set for the CTOT primary field then a node 'fields/CTOT' is created in the output document containing the data received for the CTOT field. The 'icaoname' attribute has no effect when it is defined on structured or list fields. If all ADEXP fields are required in the W3C output then the 'adexpOutput' attribute defined for the message title should be used as opposed to labelling each primary field with this attribute. *Note: If a basic primary field is modified to include the 'icaoname' attribute the XSD must be updated to ensure the schema remains valid; this is necessary as a new node is being added to the output document and the schema must also reflect this change.*	The name for an XML node that will appear as a child node of the 'fields' node.
	containsmessage	Indicates that this field contains a complete ADEXP message as its data. This causes the parser to abandon further parsing. If the attribute is omitted it is the same as specifying 'no' (the default case).	'yes' or 'no'

Node Name	Attributes	Description	Valid Values
option	None	These nodes contain one or more sets of subfields (for list and structured fields) or one or more auxiliary terms for the basic fields. Each set contained within one option node represents a particular combination of subfields or auxiliary terms that are valid for the primary field. This concept has been implemented to support the multiple combinations specified by the ADEXP standard for the list and structured fields. Typically for the auxiliary terms, only one definition is defined. However, there are a few instances where a field's possible semantics cannot be defined in a single regular expression, in such cases a list of one or more possibilities can be defined.	N/A
subField	name	The name of the subfield; the subfield must be present in the Subfield descriptor file.	Any field name that exists in the Subfield descriptor file.
	optional	Indicates if a subfield is optional or compulsory, the definition is 'yes' or 'no';	'yes' or 'no' in lower case letters.
	multiple	Indicates that the subfield may occur more than once, the definition is '-1' or a positive integer value. If '-1' then the field can occur 1 to 'n' times, if a positive integer value is defined then the field must occur the exact number of times specified.	'-1' or 0..n
auxilliaryTerm	term	The name of the auxiliary term that defines the syntax and possibly the semantics of the data associated with a basic field. The name must be a name defined in the auxiliary term descriptor file.	Any auxiliary term name that exists in the Auxiliary Term descriptor file.

Table 5 - Primary Field File Nodes and Attributes

Appendix F – ADEXP to ICAO Field Mapping, contains a complete list of the ADEXP fields mapped to the equivalent ICAO field using the 'icaoname' attribute as delivered with the parser.

2.1.1.6.3 Sub-fields

This file specifies the subfields supported by the ADEXP parser. The subfields, similar to the primary fields, come in 3 different types:

- Basic – Key value pair fields, this is the simplest type of field;
- List – Begins with the 'BEGIN' keyword and ends with the 'END' keyword. The order of the items contained within the BEGIN/END is important and must be maintained. The fields specified within a BEGIN/END block can be any subfield; this includes nested list fields, (recursive).
- Structured – Contains an un-ordered set of subfields

This file is structured in three sections to reflect the 3 different types of subfields fields. The basic fields are the simplest, represented as key-value pairs in a message. These are defined in this file with one or more pointers to the auxiliary terms file. The auxiliary term file specifies the regular expression with which to parse the value part of a key-value pair; typically only one term is specified, but in some cases an 'option' is provided to cover more than one possible definition.

The list and structured subfields contain pointers to the subfields that they themselves contain. It is to be noted that recursive structures can be defined where a list or structured subfield my include another list or structured subfield. This implementation is according to the EUROCONTROL ADEXP specification, which does specify nested levels of subfield descriptions.

This file has the structure shown in the following figure.

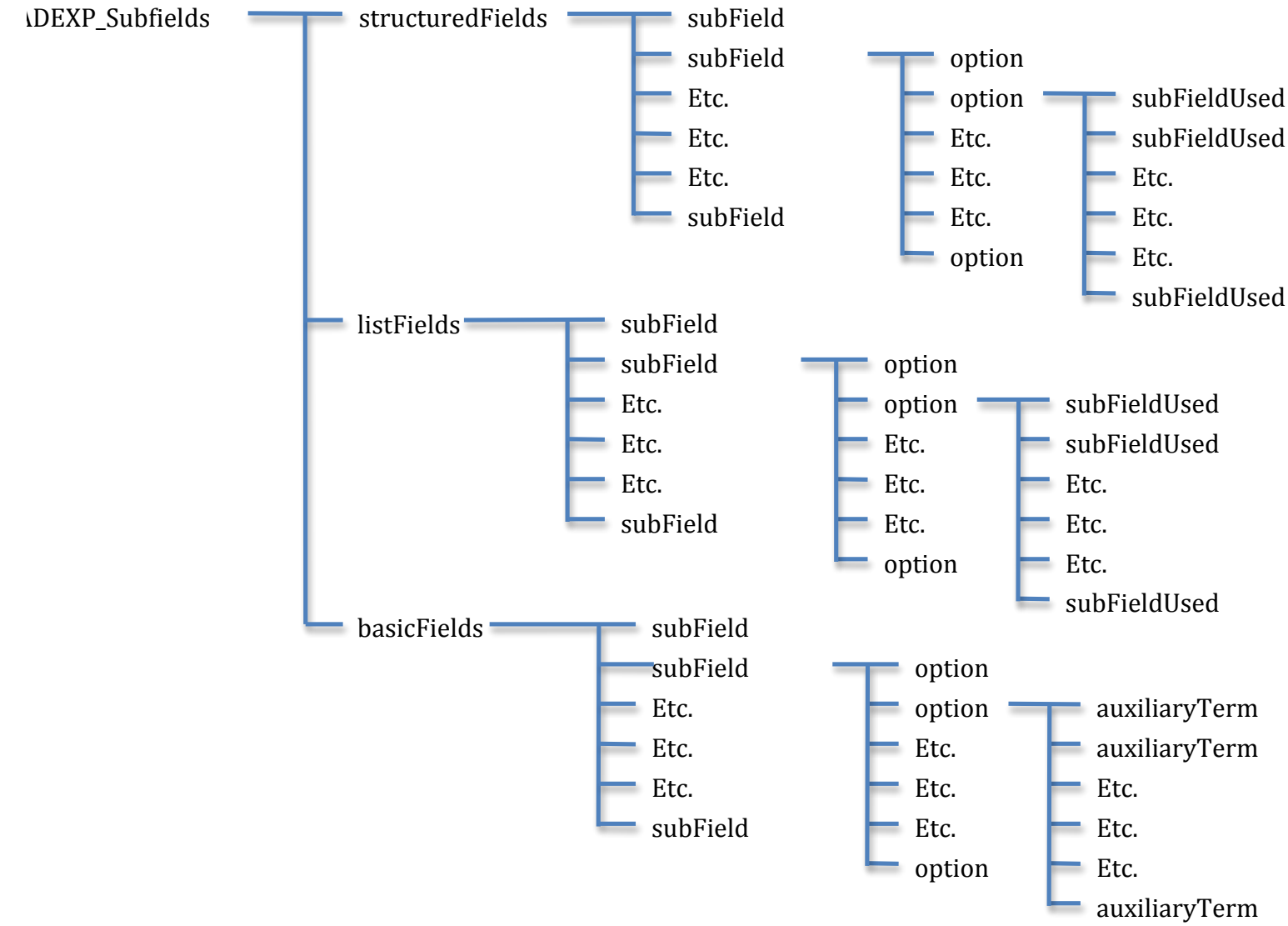

Figure 6 - Subfield Descriptor File Structure

Each of the subfields specified must have a corresponding entry in the **`ADEXP_subfields.xml`** descriptor file. Each of the auxiliary terms specified must have a corresponding entry in the **`ADEXP_auxilliary_terms.xml`** descriptor file.

The order in which the fields appear is not important, although for clarity it is recommended that they be entered in alphabetical order to make it easier to locate a field when sorted alphabetically. A number of attributes can also be specified on the XML nodes, these are described in the following table.

Node Name	Attributes	Description	Valid Values
structuredFields	None	This is a single node that contains all the structured subfields as child nodes.	N/A

Node Name	Attributes	Description	Valid Values
listFields	None	This is a single node that contains all the list subfields as child nodes.	N/A
basicFields	None	This is a single node that contains all the basic subfields as child nodes.	N/A
subField	name	The name of a subfield; this name must be defined in the descriptor file containing the subfield descriptions. If the field is a primary list subfield the 'BEGIN' keyword must not be entered, the parser knows which subfields are list subfields because of its placement within the 'structuredFields' or 'listFields' nodes. The subfield nodes contain one or more option nodes, which in turn contain either a subfield or auxiliary term. The option nodes provide a means to: Offer one or more alternative combination of subfields (this is particularly true for the structured fields as the ADEXP specification often describes several variations on the content of a structured field). Offer one or more alternative auxiliary terms.	Any subfield name that exists in the Subfield descriptor file given as: [A-Z0-9]+
option	None	These nodes contain one or more sets of subfields (for list and structured subfields) or one or more auxiliary terms for the basic subfields. Each set contained within one option node represents a particular combination of subfields or auxiliary terms that are valid for the subfield. This concept has been implemented to support the multiple combinations specified by the ADEXP standard for the list and structured subfields. Typically for the auxiliary terms, only one definition is defined. However, there are a few instances where a subfields possible semantics cannot be defined in a single regular expression, in such cases a list of one or more possibilities can be defined.	N/A
subFieldUsed	name	The name of the subfield; the subfield must be present in the Subfield descriptor file.	Any subfield name that exists in the Subfield descriptor file.
	optional	Indicates if a subfield is optional or compulsory, the definition is 'yes' or 'no';	'yes' or 'no' in lower case letters.
auxiliaryTerm	term	The name of the auxiliary term that defines the syntax and possibly the semantics of the data associated with a basic subfield. The name must be a name defined in the auxiliary term descriptor file.	Any auxiliary term name that exists in the Auxiliary Term descriptor file.

Table 6 - Subfield File Nodes and Attributes

2.1.1.6.4 Auxiliary Terms

The auxiliary terms file is a ‘flat’ file that contains a list of auxiliary terms where each term has a name and a number of other attributes. The terms in this file are referenced from the primary and subfield descriptor files. Any term referenced from the primary and subfield files must be present in the auxiliary term file. If this is not the case then the parser with fail with an exception when it tries to initialise its internal data storage.

The parser software is delivered with all auxiliary terms currently defined by EUROCONTROL in the ADEXP standards given in [2].

The nodes and attributes in the auxiliary term file are described in the following table.

Node Name	Attributes	Description	Valid Values
auxilliaryTerm	name	The name of an auxiliary term;	A name comprised of the following characters: [a-zA-Z0-9]+ They are typically all lower case letters.
	regexp	The regular expression that defines a given ‘value’ of a key-value pair. For example “[A-Z]{4}” is the regular expression that defines valid syntax for an aerodrome location indicator.	Any regular expression
	errmsg	The text to be displayed in an error message if a value does not match the specified regular expression, (see section 2.1.1.6.6 also).	Text describing the error.
	multi	Indicates that the multiple auxiliary terms are present, i.e. ADES requires that only a single data item/auxiliary term is present, but the ROUTE field contains a complete ICAO field 15 route description and hence contains more than one term. Such fields must be specified with ‘yes’. Omitting this field is the same as specifying ‘No’ (the default value)	‘Yes’ or ‘No’

Table 7 - Auxiliary Term File Node and Attribute Descriptions

2.1.1.6.5 Auxiliary Term FPL 2012 Impact

The FPL 2012 changes have imposed a number of limitations on the ability to express the syntax and semantics in the auxiliary term file due to the syntax complexity of a number of the FPL 2012 fields. The following auxiliary terms have additional hard coded syntax and semantic checks performed that cannot be controlled from the auxiliary terms file:

- aidequipment
- ssrequipment
- pbn

The regular expressions for these fields are deliberately lax to only check the allowable characters for each field. The ICAO syntax and semantic rules for these fields is hardcoded in the software implementation. The syntax and semantic checks are carried out according to the ICAO FPL 2012 changes.

2.1.1.6.6 Global Errors

A fifth XML file is delivered with the software that contains ‘global’ errors that may reported for any field, i.e. if a field contains too many auxiliary terms then this error is not related to any specific field or subfield, it’s a ‘global’ error that is applicable for any field or subfield.

Field specific errors relating to syntax and semantics are defined in the auxiliary term file, (described in the previous section).

The **`GlobalErrorText.xml`** file contains tag names that are referenced from the software and must not be changed. This XML file is provided in order to support internationalisation and is not foreseen to be modified; hence no further description of this files content are contained in this document, although the structure of the file is simple enough to understand if it is felt that the error messages need to be changed.

2.1.1.7 OLDI Configuration Data

When parsing OLDI ICAO format messages, the message content is typically defined between two adjacent units, hence the message structure, i.e. the ICAO fields included in a message, are dependant on the adjacent unit from which an OLDI message is received.

In order to support the OLDI message content definition, the message content for each supported adjacent unit is defined in a Java property file. When an OLDI message is received by the parser, the parser analysis ICAO field 3 to obtain the sending unit identifier. The sending unit identifier is then used to obtain the field list for the message. The field list is used by the ICAO message parser to validate the message content. OLDI ADEXP messages are treated as per the definition in the ADEXP configuration data.

The ICAO OLDI content is specified in terms of ICAO field numbers, for example F13 is ICAO field 13 'a' (ADEP) and 'b' (EOBT).

If only part of an ICAO field is required then this must be specified as such, for example F16A implies only the ADES is included as a field.

With the OLDI property file, an adjacent unit name is prefixed to the message title separated by a dot ('.'), for example BU.ACT=... where 'BU' is the adjacent unit name as it appears in the OLDI message title field 3 'b' and/or 'c'. If no adjacent unit name prefixes a message title then such an occurrence of a message title is considered the default occurrence.

Each message also contains a bilaterally agreed field 22 content that has to be specified in a similar manner. The F22 content is only used when building a message for output, the ICAO OLDI message parser processes any and all F22 amendment fields received in a message.

The default specifications contain all the compulsory fields for the ICAO format messages as specified in the OLDI 4.x standard.

The following is an example for the default and adjacent units ‘AA’ and ‘BB’ configuration data for the OLDI ACT message as it appears in the Java property file:

```
! ********* ACT START *********************
ACT=F3,F7,F13a,F14,F16a,F22
ACTf22=F9,F80,F81

AA.ACT=F3,F7,F13a,F14,F16a,F22
AA.ACTf22=F8a,F9,F15,F18,F80,F81

BB.ACT=F3,F7,F13a,F14A,F16a,F22
BB.ACTf22=F8a,F9,F80,F81
! ********* ACT END ***********************
```

One section as shown above is defined for each supported ICAO format OLDI message title. There is always a default definition and any number of specific adjacent unit definitions.

2.1.1.8 W3C XML Output Document Structure

The software is provided with an XSD document that describes, in detail, the structure and content of the W3C XML output document. This section provides a simplified overview of the output documents content without going into the level of detail provided in the XSD.

The overall structure is shown in the following figure, the names used for the various nodes depicted in the diagram are the names used for the XML tags in the output document.

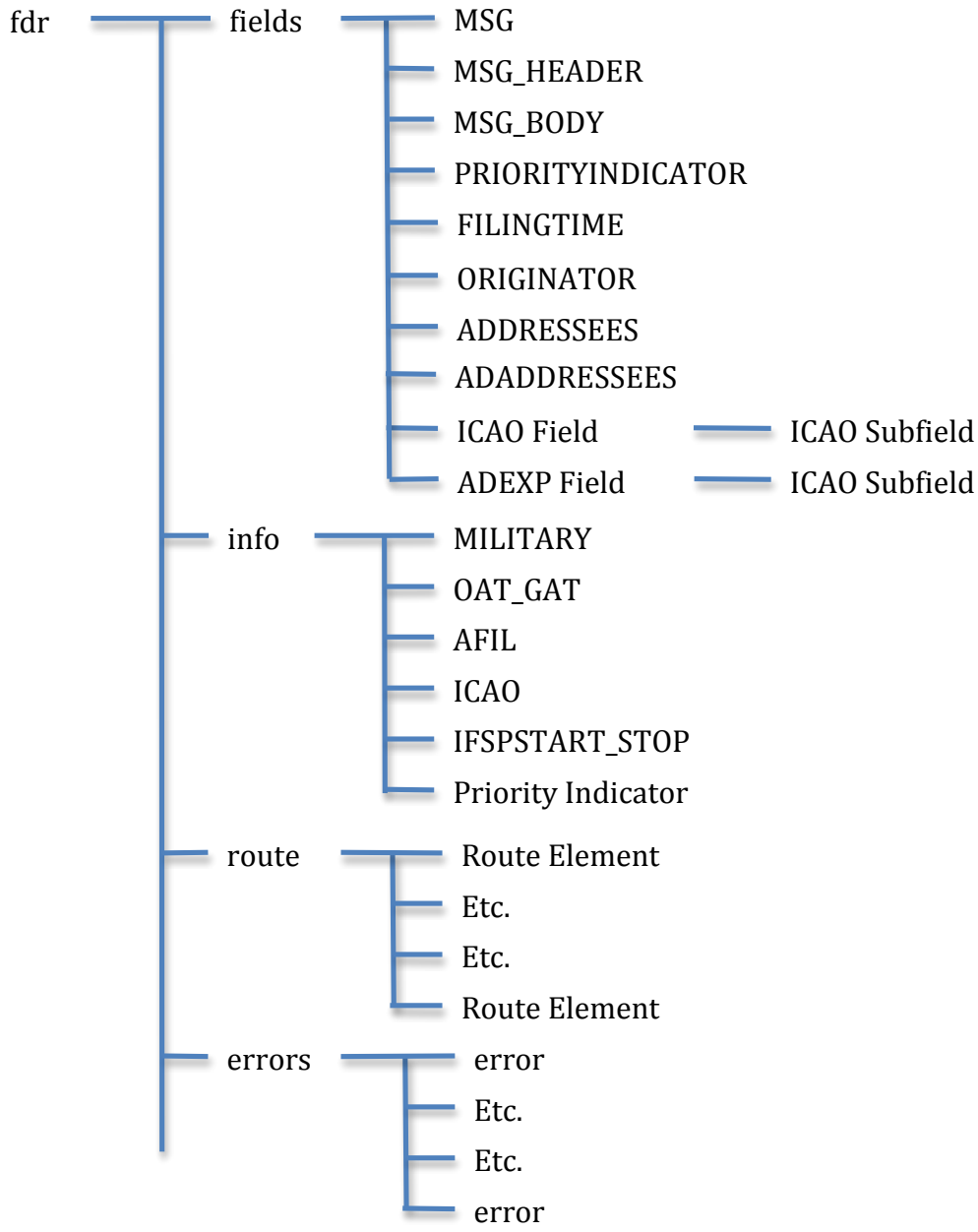

Figure 7 - W3C Output Document File Structure

The node names shown in the figure are described in the following table. One way with which to access the data in the W3C output document would be to use Xpath with the path names as described in the XSD.

Node Name	Description
fdr	The root node of the document;
fields	This node has one or more child nodes, where each child node contains the ICAO field data as extracted from a message; the node naming conventions are given in the XSD.
info	The child nodes in this node contain global data applicable to the message as a whole;
errors	This node contains one or more error nodes, one for each error generated by the parser. If no errors are present this node has no children. Each error node is in itself a complex structure that contains additional information about an error, this includes: The token causing the error; The start position of the erroneous token in a given field; The end position of the erroneous token in a given field; The ICAO name of the field with the error; The offset of the erroneous field with respect to the start of the message This structure is not described further in this document. If an index to an erroneous token is needed then the start position in a field must be added to the field offset, this yields a zero based index of the first character of the erroneous token within the message as a whole.
route	This node contains the extracted route derived from field 15. The node ‘fields/F15’ contains the route data as supplied in field 15 of a message and never contains any child nodes as for the ‘ICAO Field’ node. The extracted route is output to a dedicated structure as two or more ‘Route Element’ child nodes to the ‘route’ node. This document does not describe this structure further here and the reader is advised to examine the XSD for more details. The sequence always starts and ends with the ADEP and ADES with any number of intermediate items in between. A ‘Route Element’ item is created for each token in field 15 apart from the speed/level token, these have their values ‘applied’ to a previous point token. Note that the parser does not contain any AIP data with which to complete the extracted route sequence; hence the extracted route is an initial extracted route that requires further processing. However, the fields set aside in this output document are sufficient to represent a complete trajectory should further processing be carried out.
ICAO Field	This node contains the actual field as extracted from a message; the syntax is always a capital ‘F’ followed by the ICAO number, e.g. F13. The data contained in these nodes is the actual data present in a field such as F13, e.g. LOWW0800. This node also contains child nodes where a given ICAO field is decomposed into its subfields. This however, does not apply for field 15, which is handled a little differently, (refer to the ‘route’ node description above).
ADEXP Field	If a basic ADEXP primary field has been configured in the descriptor file ‘Primary Field’ to be copied to the W3C output document, then the ADEXP field will also appear as a child node of the ‘fields’ nodes The node name assigned will be as specified in the ‘Primary Field’ descriptor file.

Node Name	Description
ICAO Subfield	This node contains an ICAO subfield such as the EOBT from F13. The syntax is simply the letter assigned by ICAO. For example, F13 has an ‘a’ and a ‘b’ subfield so the child nodes for F13 would be ‘a’ and ‘b’ where ‘a’ = LOWW and ‘b’ = 0800 if F13 contained LOWW0800. For field 18 and 19 the subfield names are used, e.g. if a message had a field 18 containing ‘RMK/THIS IS A REMARK’ the ‘fields’ node would have a child node called F18 with the data ‘THIS IS A REMARK’ and a child node called ‘rmk’ also containing ‘THIS IS A REMARK’. For ADEXP fields the subfield node name assigned will be as specified in the ‘Primary Field’ descriptor file.
MILITARY	Set to ‘true’ if ICAO field 8b = ‘M’, false otherwise;
OAT_GAT	Set to ‘true’ if field 15 contains the tokens OAT/GAT in the route description, if not present in field 15 then this node is not present.
AFIL	Set to ‘true’ if ICAO field 13a = ‘AFIL’, false otherwise;
IFPSTART_STOP	Set to ‘true’ if field 15 contains the tokens IFPSTOP/IFPSTART in the route description, if not present in field 15 then this node is not present.
ICAO	Set to ‘true’ if a message was processed in ICAO format and false if the message was in ADEXP format;
ADJ_UNIT_NAME	The name of the sending unit if a message contained F3b – this is used for OLDI message processing.
ATS_MESSAGE	The name of the sending unit if a message contained F3b – this is used for OLDI message processing.
F10a_VERSION	Indicates if the ICAO field 10a is pre or post FPL 2012; the content is an enumeration (FPL_2012_VERSION) defined at the end of this table.
F10b_VERSION	Indicates if the ICAO field 10b is pre or post FPL 2012; the content is an enumeration (FPL_2012_VERSION) defined at the end of this table.
F10_VERSION	Indicates if the ICAO field 10 is pre or post FPL 2012; the content is an enumeration (FPL_2012_VERSION) defined at the end of this table.
F18_VERSION	Indicates if the ICAO field 18 is pre or post FPL 2012; the content is an enumeration (FPL_2012_VERSION) defined at the end of this table.
Priority Indicator	This node name is the priority indicator value, which can be one of the following when the priority indicator is a recognised value – ‘DD’, ‘EE’, ‘FF’, ‘GG’ or ‘SS’. If the priority indicator is unknown ‘__UNKNOWN__’ is inserted, the value is always ‘true’.
Primary_Basic_Fields	Contains one or more tags with names equal to the ADEXP basic primary field in a message. Each tag has three attributes: data – Contains the data provided with the ADEXP field key value pair; start – A zero based index where the ADEXP and data tokens start in the message. end – A zero based index where the ADEXP and data tokens end in the message. This tag is only present if the message has been flagged for ADEXP output using the attribute ‘adexpOutput’ in the supported message title configuration file.

Node Name	Description
Primary_List_Fields	Contains a list of ADEXP list fields with the BEGIN and END field delimiting a given list structure. Within in each list field, there is one or more list, basic and/or structured fields. This tag is only present if the message has been flagged for ADEXP output using the attribute 'adexpOutput' in the supported message title configuration file.
Primary_Structured_Fields	Contains a list of ADEXP structured fields with field name delimiting a given structure. Within in each structured field, there is one or more basic and/or structured fields. This tag is only present if the message has been flagged for ADEXP output using the attribute 'adexpOutput' in the supported message title configuration file.
Undefined_for_this_Message	This tag contains a list of fields that exist in the parser configuration data, (i.e. they are 'known' to the parser) but are not defined for a given message title. The ADEXP specification states that unknown fields are to be ignored and should not generate an error. This tag is only present if the message has been flagged for ADEXP output using the attribute 'adexpOutput' in the supported message title configuration file.
Unknown_Content	This tag contains a list of fields that are not defined in the parser configuration data, (i.e. they are 'unknown' to the parser). The ADEXP specification states that unknown fields are to be ignored and should not generate an error. This tag is only present if the message has been flagged for ADEXP output using the attribute 'adexpOutput' in the supported message title configuration file.

The version of a message with respect to the FPL 2012 specification can fall into one of four categories as defined by the enumeration type FPL_2012_VERSION:

- PRE_FPL_2012 – Indicates that a field is in the format of a pre FPL 2012 message;
- POST_FPL_2012 – Indicates that a field is in the format of a post FPL 2012 message;
- PRE_AND_POST_FPL_2012 – Indicates that a field contains data that is both pre and post FPL 2012 format;
- UNKNOWN_FPL_2012 – Indicates that the FPL 2012 version cannot be determined from the data in the field, i.e. the data is allowed in both pre and post FPL 2012 messages.

It is a caller's responsibility to decide on how to process messages that indicate mixed versions. When the parser detected FPL 2012 related errors, these are included in the XML output. There is no 'global' flag indicating if a message is in pre or post FPL 2012 format. It is recommended that the F10_VERSION and F18_VERSION tags be used to ascertain if a message is pre or post FPL 2012. Note that one or both of these tags may be absent if they were not included in a message or an error occurs that prevents these fields from being created.

If the parser detected that there are mixed pre and post data in the F10 and F18 fields an error is raised and included in the output.

2.1.2 AOMT Wrapper Description

The AOMT receives a message along with a callers translation requirements, the AOMT sends the message to the message parser, then uses the parser output to rebuild the message in the target translation format and version.

The supported translation instructions are defined in the following table.

Translation Instruction	Description
PRE_2012	Instructs the AOMT to output the message in the Pre-FPL 2012 format.
POST_2012	Instructs the AOMT to output the message in the Post-FPL 2012 format.
ICAO	Instructs the AOMT to output the message in the ICAO format.
ADEXP	Instructs the AOMT to output the message in the ADEXP format.
OLDI_2.x[4]	Instructs the AOMT to output the message as an OLDI version 2.x message. This instruction is only used when an OLDI message is processed.
OLDI_3.x	Instructs the AOMT to output the message as an OLDI version 3.x message. This instruction is only used when an OLDI message is processed.

Table 8 - AOMT Translation Instructions

2.1.2.1 FPL 2012 Version Translation

The parser output includes data indicating if a message is pre or post FPL 2012; it is not always possible to determine the FPL 2012 classification, which leads to one of the following possibilities being determined by the parser:

- **PRE_FPL_2012** - Indicates that a field is pre FPL 2012. For a message to be considered as pre FPL 2012, all subfields must indicate this mode or, a mixture of 'PRE_FPL_2012' and 'UNKNOWN_FPL_2012' modes.
- **POST_FPL_2012** - Indicates that a field is post FPL 2012. For a message to be considered as post FPL 2012, all subfields must indicate this mode or, a mixture of 'POST_FPL_2012' and 'UNKNOWN_FPL_2012' modes.
- **PRE_AND_POST_FPL_2012** - This is an error condition and the message will not be translated; the message will contain an error that indicates to an end user what the problem is. This will occur if any of the fields are indicated as 'mixed' pre and post FPL 2012.
- **UNKNOWN_FPL_2012** - Indicates that the format is valid for both pre and post FPL 2012 - no translation is required.

A message will not be translated, but passed through as is, when any of the following error conditions exist:

- If any fields indicate a mixture of pre and post FPL 2012 formats;
- If a combination of fields indicates a mixture of pre and post FPL 2012 formats

Messages are also left untranslated if the 'unknown' status is indicated; this state implies that a field is in a format that is not specifically pre or post FPL 2012 and can therefore be used for both pre and post formats.

The translation is carried out according to rules specified by EUROCONTROL in [10].

[4] It is currently not know if this instruction is necessary – will be determined when analyzing the difference between version 2.x and 3.x.

2.1.2.1.1 AOMT Configuration Data

The configuration data available for configuring the AOMT are the parser specific data described in sections 2.1.1.6 (ADEXP Parser) and 2.1.1.7 (OLDI Messages) plus an additional configuration file that specifies the following:

ADEXP message titles that are:

- Totally ignored for any processing and sent back to the caller;
- Cannot be converted to ICAO format as no corresponding ICAO format message exists;

This configuration file is a Java property file with the following structure:

```
unsupportedTitles=ACK,ACP,BFD…
ignoredTitles=AUP,CRAM…
```

The two configuration items contain the following:

- Item 'unsupportedTitles' – Contains a list of messages that cannot be converted from ADEXP to ICAO but can be version translated.
- Item 'ignoredTitles' – Contains a list of message titles that are not processed by the AOMT parser as there is no version of format translation possible at all.

To support translation of messages with errors, an additional configuration item is provided that contains a list of text strings that are interpreted by the software as regular expressions and used to 'ignore' the error and translate a message that generates the error. This ensures that messages that have sufficient data to translate are translated, even if they contain errors. The configuration item is:

```
ignoredErrors=CONSISTENCY.*;F15
```

This example will cause the AOMT to ignore any errors containing 'CONSISTENCY' or 'F15' in their text and perform the translation even though errors are present.

2.1.2.1.2 F18 Translation

The following table indicates translation for the various F18 fields. The translation can be used in both directions, pre to post and post to pre FPL 2012. Any fields not explicitly mentioned in the following table will not be translated and left as they are.

Post 2012 F18 Subfield	Post Data	Pre 2012 F18 Subfield	Pre Data
EUR	EXM833	STS	EXM833
EUR	PROTECTED	STS	PROTECTED
EUR	RNAVINOP	STS	RNAVINOP
EUR	RNAVX	STS	NONRNAV
EUR	CPDLCX	STS	CPDLCX
STS	ATFMX	STS	ATFMEXEMPTAPPROVED
SUR	nnnn	RMK/SUR	nnnn
DOF	nnnnnn	DOF	nnnnnn – The DOF will only be output for the FPL, not for other ATS messages.
DLE		No Output – not defined for pre 2012 versions.	
ORGN		No Output – not defined for pre 2012 versions.	

Post 2012 F18 Subfield	Post Data	Pre 2012 F18 Subfield	Pre Data
TALT	nnnn	RMK	TALT nnnn

Table 9 - F18 Translation

The F18 NAV, DAT and COM subfields are also impacted by the FPL 2012 translation, but the output of these fields is determined by the content of fields 10 ‘a’ and ‘b’ which are described below. The translation of these fields is not bi-directional.

2.1.2.1.3 Post to Pre FPL 2012 Translation

2.1.2.1.3.1 F10a Post to Pre Translation

The translation of post to pre FPL 2012 for various F10a items and the update's to the F18 subfields required for the items being added in F18 is shown in the following tables.

Post F10a	Pre F10a	Pre-F18 Subfield	Pre-F18 Subfield Data	Pre-F18 Subfield	Pre-F18 Subfield Data
A	Z	NAV	GBAS		
B	Z	NAV	LPV		
E1	Z	COM	E1	RMK	FMC WPR ACARS
E2	Z	COM	E2	RMK	DFIS ACARS
E3	Z	COM	E3	RMK	PDC ACARS
J1	J	COM	J1	DAT	V
J2	J	COM	J2	DAT	H
J3	J	COM	J3	DAT	V
J4	J	COM	J4	DAT	V
J5	J	COM	J5	DAT	S
J6	J	COM	J6	DAT	S
J7	J	COM	J7	DAT	S
M1	Z	COM	M1	RMK	INMARSAT
M2	Z	COM	M2	RMK	MTSAT
M3	Z	COM	M3	RMK	IRIDIUM
P1 to P9	No Output	No Output	No Output	No Output	No Output
R[5]					
S	VOL				
SF	S				
Z	Z	COM	nnnn	COM	nnnn
Z	Z	NAV	nnnn	NAV	nnnn
Z	JZ	DAT	S, H, V or M	DAT	S, H, V or M
Z	Z	DAT	nnnn	COM	nnnn

Table 10 - Post FPL 2012 to Pre FPL 2012 Translation for F10a

The 10 'a' characters C, D, F, G, H, I, K, L, N, O, T, U, V, W, X, and Y require no translation.

[5] Refer to the dedicated table Table 11 - in section 2.1.2.1.3.2 for a description abnout the 10a 'R' processing.

2.1.2.1.3.2 F10a Post to Pre for ‘R’ Translation

The table shown below describes the translation (post to pre FPL 2012) for the ‘R’ in F10a. In FPL 2012 messages the 'R' is supplemented with data in the F18 PBN field. The PBN data must be transferred to the pre FPL 2012 F18 NAV field and the meaning of the F10 R transferred to the pre F18 RMK field.

In all cases the PBN field is removed;

The translation taking place is shown below in the following table:

Post 2012 PBN Data	Pre 2012 F10a	Pre 2012 F18 RMK Data
A1	RZ	RNAV10 RNP10
B1	RZ	RNAV5
B2	RZ	RNAV5
B3	RZ	RNAV5
B4	RZ	RNAV5
B5	RZ	RNAV5
B6	RZ	RNAV5
C1	RZ	RNAV2
C2	RZ	RNAV2
C3	RZ	RNAV2
C4	RZ	RNAV2
D1	PRZ	RNAV1
D2	PRZ	RNAV1
D3	PRZ	RNAV1
D4	PRZ	RNAV1
L1	RZ	RNP4
O1	PRZ	RNP1
O2	PRZ	RNP1
O3	PRZ	RNP1
O4	PRZ	RNP1
S1	GZ	RNP APRCH
S2	GZ	RNP APRCH BARO VNAV
T1	GZ	RNP AR APRCH RF
T2	GZ	RNP AR APRCH

Table 11 - Post FPL 2012 to Pre FPL 2012 for ‘R’ in F10a

2.1.2.1.3.3 F10b Post to Pre Translation

The translation of post to pre FPL 2012 for various F10b items and the updates to the F18 subfields required for the items being added in F18 are shown in the following tables.

Post F10b	Pre F10b	F18 Subfield	F18 Subfield Data
E	SD	COM	E
H	S	COM	H
L	SD	COM	L
B1	D	COM	B1
B2	D	COM	B2
U1	D	COM	U1
U2	D	COM	U2
V1	D	COM	V1
V2	D	COM	V2
D1	D	COM	D1
G1	D	COM	G1

Table 12 - Post FPL 2012 to Pre FPL 2012 Translation for F10b

The 10 ‘b’ characters A, C, I, N, P, S and X require no translation.

2.1.2.1.4 Pre to Post FPL 2012 Translation

This section describes the translation from a pre FPL 2012 to post FPL 2012 version.

2.1.2.1.4.1 F10a Pre to Post Translation

This section contains a description of how the pre to post FPL translation will be carried out for field 10a. To simplfly the description, each character is taken in turn.

The 'Z' Character

Post F10a	Pre F10a	Pre-F18 Subfield	Pre-F18 Subfield Data	Pre-F18 Subfield	Pre-F18 Subfield Data
A	Z	NAV	GBAS		
B	Z	NAV	LPV		
E1	Z	COM	E1	RMK	FMC WPR ACARS
E2	Z	COM	E2	RMK	DFIS ACARS
E3	Z	COM	E3	RMK	PDC ACARS
M1	Z	COM	M1	RMK	INMARSAT
M2	Z	COM	M2	RMK	MTSAT
M3	Z	COM	M3	RMK	IRIDIUM
Z	Z	COM	nnnn		
Z	Z	NAV	nnnn		
Z	JZ	DAT	S, H, V or M		
Z	Z	DAT	nnnn		

Table 13 - Pre to Post FPL 2012 Translation Field 10a, 'Z'

If the post FPL 2012 data for field 10a cannot be determined from the field 18 subfields shown in Table 13 - , the 'A', 'B', 'Ex', 'Mx' and 'Z' characters will not be included in the post FPL 2012 field 10a.

The 'J' Character

Post F10a	Pre F10a	Pre-F18 Subfield	Pre-F18 Subfield Data	Pre-F18 Subfield	Pre-F18 Subfield Data
J1	J	COM	J1	DAT	V
J2	J	COM	J2	DAT	H
J3	J	COM	J3	DAT	V
J4	J	COM	J4	DAT	V
J5	J	COM	J5	DAT	S
J6	J	COM	J6	DAT	S
J7	J	COM	J7	DAT	S

Table 14 - Pre to Post FPL 2012 Translation Field 10a, 'J'

If the post FPL 2012 data for field 10a cannot be determined from the field 18 subfields shown in Table 14 - , the 'Jx' characters will not be included in the post FPL 2012 field 10a.

The ‘S’ Character

Post F10a	Pre F10a	Pre-F18 Subfield	Pre-F18 Subfield Data	Pre-F18 Subfield	Pre-F18 Subfield Data
S	VOL				
SF	S				

Table 15 - - Pre to Post FPL 2012 Translation Field 10a, ‘S’

The ‘S’ character will be put into field 10a if the characters ‘V’, ‘O’ and ‘L’ are all present in the pre 2012 field 10a.

The characters ‘S’ and ‘F’ will be put into the post 2012 field 10a if the character ‘S’ is found in the pre 2012 field 10a.

The ‘Px’ Characters

The values P1 to P9 require no further consideration when translation from pre to post 2012.

The ‘R’ Character

The pre 2012 characters ‘P’, ‘R’, ‘Z’ and ‘G’ will only be output into a post FPL 2012 field 10a if they can be identified by the associated PBM and RMK data. If a unique identification cannot be made, the characters as shown in Table 11 - , will not be created in the post FPL 2012 field 10a.

Characters not Impacted by FPL 2012

Any characters not specified for translation will be transferred from the pre to post FPL 2012 fields. The characters requiring no translation are C, D, F, G, H, I, K, L, N, O, T, U, V, W, X, and Y.

2.1.2.1.4.2 F10b Pre to Post Translation

This section describes the mapping of pre FPL 2012 field 10b to the post 2012 field 10b.

Note: There is little to no information available as to how this direction of translation should be done.

Pre F10b	Pre F18 Subfield	Pre F18 Subfield Data	Post F10b
SD	COM	E	E
S	COM	H	H
SD	COM	L	L
D	COM	B1	B1
D	COM	B2	B2
D	COM	U1	U1
D	COM	U2	U2
D	COM	V1	V1
D	COM	V2	V2
D	COM	D1	D1
D	COM	G1	G1

Table 16 - Pre FPL 2012 to Post FPL 2012 Translation for F10b

The 10 ‘b’ characters A, C, I, N, P and X require no translation.

If the field 18 COM subfield is not present, it is impossible to determine what should be entered for the second character in field 10b. If none of the characters A, C, I, N, P, S or X are present and none of the data items listed in Table 16 - can be derived, it is in fact somewhat difficult to determine a valid translation from pre to post 2012. The following truth table describes the action taken under various pre 2012 field 10b content.

Pre F10b Content			Pre F18 COM		
1st Char	2nd Char	‘H’ Available	‘E’ or ‘L’ Available	Other than ‘E’, ‘H’ or ‘L’ Available	Post 10b Content
‘N’	Empty	Don’t care	Don’t care	Don’t care	‘N’
‘A’, ‘C’, ‘I’, ‘P’ or ‘X’	Empty	Don’t care	Don’t care	Don’t care	‘A’, ‘C’, ‘I’, ‘P’ or ‘X’
‘S’	Empty	Yes	Don’t care	Don’t care	‘H’
‘S’	‘D’	Don’t care	No	No	Empty
‘S’	‘D’	Don’t care	No	Yes	Empty
‘S’	‘D’	Don’t care	Yes	Don’t care	‘E’ and/or ‘L’
‘A’, ‘C’, ‘I’, ‘P’ or ‘X’	‘D’	Don’t care	Don’t care	No	‘A’, ‘C’, ‘I’, ‘P’ and/or ‘X’
‘A’, ‘C’, ‘I’, ‘P’ or ‘X’	‘D’	Don’t care	Don’t care	Yes	‘A’, ‘C’, ‘I’, ‘P’, ‘X’ and/or one or more of ‘B1’, ‘B2’, ‘U1’, ‘U2’, ‘V1’, ‘V2’, ‘D1’, ‘G1’.

Table 17 - Pre to Post FPL 2012 Translation for Field 10b

2.1.2.2 OLDI Translation

The OLDI translation may be carried out between messages defined by two different OLDI standards versions, namely version 2.x and 4.x, (given in [11], [12] and [3]] respectively). This section lists the differences between the two versions and provides comments as to how the translator deals with translation of OLDI messages that may have no counterpart in the target version, or no counterpart field in a target version.

The table shown below lists the message titles supported in 2.x and 3.x.

Message Title	Defined in Version 2.x	Defined in Version 3.x	Supported in ICAO Format	Comment
ABI	Yes	Yes	Yes	
ACP	Yes	Yes	Yes	
ACT	Yes	Yes	Yes	
AMA	No	Yes	Yes	If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
BFD	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. This message can only be translated for the FPL 2012 content.
CDN	Yes	Yes	Yes	
CFD	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. This message can only be translated for the FPL 2012 content.
COD	Yes	Yes	Yes	
COF	Yes	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so.
CRP	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
CRQ	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.

Message Title	Defined in Version 2.x	Defined in Version 3.x	Supported in ICAO Format	Comment
HOP	Yes	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so.
INF	Yes	Yes	Yes	
LAM	Yes	Yes	Yes	
LOF	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
MAC	Yes	Yes	Yes	
MAS	Yes	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so.
NAN	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
OCM	No	Yes	Yes	If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
PAC	Yes	Yes	Yes	
PNT	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
RAP	Yes	Yes	Yes	
REV	Yes	Yes	Yes	
RJC	Yes	Yes	Yes	

Message Title	Defined in Version 2.x	Defined in Version 3.x	Supported in ICAO Format	Comment
RLS	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
ROC	No	Yes	Yes	If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
ROF	Yes	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so.
RRQ	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
RRV	Yes	Yes	Yes	
RTI	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
SBY	Yes	Yes	Yes	
SCO	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
SDM	Yes	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so.

Message Title	Defined in Version 2.x	Defined in Version 3.x	Supported in ICAO Format	Comment
SKC	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
TIM	Yes	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so.
TIP	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
XAP	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
XCM	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.
XIN	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.

Message Title	Defined in Version 2.x	Defined in Version 3.x	Supported in ICAO Format	Comment
XRQ	No	Yes	No	Will never be translated from ADEXP into ICAO, the AOMT will report an error if a request is made to do so. If the AOMT is employed at an OLDI connection that receives such a message, it is assumed that the connection must be an OLDI 3.0 connection, hence the AOMT will translate between ADEXP and ICAO formats without generating any errors.

Table 18 - OLDI Version Message Title Differences

Of the OLDI ICAO format messages supported in both version 2.x and 3.x, the message content with respect to the OLDI version is shown in the following table.

Message Title	F3	F7	F13 a	F13	F14a	F14	F16a	F22	Comment
ABI (2.x)	C	C	C			C	C	C	
ABI (3.x)	C	C	C			C	C	C	
ACP (2.x)	C	O	O				O	O	
ACP (3.x)	C	O	O				O	O	
ACT (2.x)	C	C	C			C	C	C	
ACT (3.x)	C	C	C			C	C	C	
CDN (2.x)	C	C	C			O	C	O	
CDN (3.x)	C	C	C			O	C	O	
COD (2.x)	C	C	C				C		
COD (3.x)	C	C	C				C		
INF (2.x)	C	C	C		M[6]	M	C	C	
INF (3.x)	C	C	C		M[7]	M	C	C	
LAM (2.x)	C								
LAM (3.x)	C								
MAC (2.x)	C	C	C		C		C	O	
MAC (3.x)	C	C	C		C		C	O	
PAC (2.x)	C	C	M[8]	M		M	C	C	
PAC (3.x)	C	C	M[9]	M		M	C	C	
RAP (2.x)	C	C	C			C	C	C	
RAP (3.x)	C	C	C			C	C	C	
REV (2.x)	C	C	C			C	C	O	
REV (3.x)	C	C	C			C	C	O	

[6] The F14a and F14 fields are mutually exclusive;
[7] The F14a and F14 fields are mutually exclusive;
[8] The PAC contains either F13a and F14 or only F13 (F13a, F14 and F13 are mutually exclusive);
[9] The PAC contains either F13a and F14 or only F13 (F13a, F14 and F13 are mutually exclusive);

Message Title	F3	F7	F13 a	F13	F14a	F14	F16a	F22	Comment
RJC (2.x)	C								At the time of writing this document, it is not clear if these content differences are a problem.
RJC (3.x)	C							O	At the time of writing this document, it is not clear if these content differences are a problem.
RRV (2.x)	C	C	C			C	C		
RRV (3.x)	C	C	C			C	C		
SBY (2.x)	C								
SBY (3.x)	C								

Table 19 - OLDI Message Content for ICAO Format Version 2.x and 3.x

The following table contains the F22 content for any messages listed above that are specified with F22.

Message Title	F8a	F9	F15	F18	F80	F81	Comment
ABI (2.x)	O	C	O	O	C	C	
ABI (3.x)	O	C	O	O	C	C	
ACP (2.x)				O			
ACP (3.x)				O			
ACT (2.x)	O	C	O	O	C	C	
ACT (3.x)	O	C	O	O	C	C	
CDN (2.x)			O	O			At the time of writing this document, it is not clear if these content differences are a problem.
CDN (3.x)			O				At the time of writing this document, it is not clear if these content differences are a problem.
INF (2.x)		C	C	C			
INF (3.x)		C	C	C			
MAC (2.x)				O			
MAC (3.x)				O			
PAC (2.x)	O	C	O	O	O	O	
PAC (3.x)	O	C	O	O	O	O	
RAP (2.x)		C					At the time of writing this document, it is not clear if these content differences are a problem.
RAP (3.x)	O	C	O	O	C	C	At the time of writing this document, it is not clear if these content differences are a problem.
REV (2.x)						O	
REV (3.x)						O	
RJC (3.x)				O			At the time of writing this document, it is not clear if these content differences are a problem. The F22 is not defined for the 2.x version.

Table 20 - OLDI Message Content for F22

If an attempt is made to convert any of the following OLDI messages to ICAO format, the translation will be rejected with an error.

BFD, CFD, COF, CRP, CRQ, HOP, LOF, MAS, NAN, PNT, RLS, ROF, RRQ, RTI, SCO, SDM, SKC, TIM, TIP, XAP, XCM, XIN and XRQ.

2.1.2.3 ETFMS Translation

None of the ETFMS messages can be translated into ICAO equivalents as no ETFMS messages are defined in ICAO format. In addition, there are a set of ETFMS messages that are AO specific and these messages are not processed by the ATS parser at all.

The ETFMS message titles supported by the ATS parser are:

ACK, BFD, CFD, DES, DPI, EFD, ERR, FLS, FUM, SAM, SDM, SLC, SMM, SRM and TIP

The ETFMS AO messages that are passed straight back to the caller without processing are:

FCM, RDY, RJT, RRP, SIP, SKC, SPA, SRJ and SRR

2.1.2.4 CADF Translation

None of the CADF messages can be translated into ICAO equivalents as no CADF messages are defined in ICAO format. The ATS parser is configured to process these messages but they are sent back to the caller without any processing, these message titles are:

AUP, CRAM and UUP;

2.2 Required States and Modes

No special states or modes are applicable;

2.3 CSCI Capability

This section of the document contains atomic requirements specifying the AOMT functionality.

2.3.1 FPL 2012 Version Translation

The data that the FPL 2012 version is related to must be present and valid in a message for a correct assessment to be made by the AOMT. Some messages do not contain FPL 2102 version information, in other cases it may be that the version cannot be determined, either because the version data is corrupt (in which case the parser will report an error) or because there are some data combinations that are valid for both pre and post FPL 2012, hence the version cannot be determined.

SRS-001 – Message version translation pre to post

The AOMT shall translate a pre-FPL 2012 to a post-FPL 2012 message if the following conditions are all true:

- The message is recognised as a pre-FPL 2012 message (not all messages are impacted by the FPL 2012 changes);
- The message does not contain any errors that would prohibit identification of the version;
- The caller specifies that the message should be translated to a post-FPL 2012 version;

#

SRS-002 – Message version translation post to pre

The AOMT shall translate a post-FPL 2012 to a pre-FPL 2012 message if the following conditions are all true:

- The message is recognised as a post-FPL 2012 message (not all messages are impacted by the FPL 2012 changes);
- The message does not contain any errors that would prohibit identification of the version;
- The caller specifies that the message should be translated to a pre-FPL 2012 version;

#

Most messages are not impacted by the FPL 2012 changes and contain no version specific data and there is no need to make any changes to such message if one is received.

SRS-003 – Message version no translation

If the AOMT receives a message where the version cannot be determined or the version is identical to the target version specified by the caller, the AOMT shall return the message unchanged to the caller without any error.

#

SRS-004 – FPL 2012 Translation

The AOMT shall support FPL 2012 translation for the messages listed in Table 21 - . These messages all contain FPL 2012 version information.

#

SRS Requirement	Message Title	Format	Type
SRS-004a – ARR 2012 Translation	ARR	ICAO	ICAO ATS
SRS-004b – CNL 2012 Translation	CNL	ICAO	ICAO ATS
SRS-004c – DEP 2012 Translation	DEP	ICAO	ICAO ATS
SRS-004d – DLA 2012 Translation	DLA	ICAO	ICAO ATS
SRS-004e – RQP 2012 Translation	RQP	ICAO	ICAO ATS
SRS-004f – RQS 2012 Translation	RQS	ICAO	ICAO ATS
SRS-004g – CHG 2012 Translation	CHG	ICAO	ICAO ATS
SRS-004h – FPL 2012 Translation	FPL	ICAO	ICAO ATS
SRS-004i – CPL 2012 Translation	CPL	ICAO	ICAO ATS
SRS-004j – SPL 2012 Translation	SPL	ICAO	ICAO ATS
SRS-004k – BFD 2012 Translation	BFD	ADEXP	OLDI
SRS-004l – CFD 2012 Translation	CFD	ADEXP	OLDI
SRS-004m – IACH 2012 Translation	IACH	ADEXP	CFMU IFPS ATS
SRS-004n – IAFP 2012 Translation	IAFP	ADEXP	CFMU IFPS ATS
SRS-004o – IAPL 2012 Translation	IAPL	ADEXP	CFMU IFPS ATS
SRS-004p – ICHG 2012 Translation	ICHG	ADEXP	CFMU IFPS ATS
SRS-004q – IDEP 2012 Translation	IDEP	ADEXP	CFMU IFPS ATS
SRS-004r – IDLA 2012 Translation	IDLA	ADEXP	CFMU IFPS ATS
SRS-004s – IFPL 2012 Translation	IFPL	ADEXP	CFMU IFPS ATS

Table 21 - Supported Messages for FPL 2012 Translation

2.3.2 Format Conversion

The AOMT is able to translate the ICAO format messages to ADEXP format messages and vice versa.

SRS-005 – Message format translation ICAO to ADEXP

The AOMT shall translate an ICAO format message to an ADEXP format message if the following conditions are all true:

- The message is recognised as ICAO format;
- The message does not contain any errors that would prohibit identification of the format;
- The caller specifies that the message should be translated to an ICAO format;

#

SRS-006 – Supported Messages for ICAO ADEXP translation

The AOMT shall support translation of the ICAO format messages to the ADEXP format for the messages shown in Table 22 - .

#

SRS Requirement	Message Title	Format	Type
SRS-006a – ARR ICAO to ADEXP Translation	ARR	ICAO	ICAO ATS
SRS-006b – CHG ICAO to ADEXP Translation	CHG & ACH	ICAO	ICAO ATS
SRS-006c – CNL ICAO to ADEXP Translation	CNL	ICAO	ICAO ATS
SRS-006d – CPL ICAO to ADEXP Translation	CPL,[10] AFP & APL	ICAO	ICAO ATS
SRS-006e – DEP ICAO to ADEXP Translation	DEP	ICAO	ICAO ATS
SRS-006f – DLA ICAO to ADEXP Translation	DLA	ICAO	ICAO ATS
SRS-006g – EST ICAO to ADEXP Translation	EST	ICAO	ICAO ATS
SRS-006h – FPL ICAO to ADEXP Translation	FPL	ICAO	ICAO ATS
SRS-006i – RQP ICAO to ADEXP Translation	RQP	ICAO	ICAO ATS
SRS-006j – RQS ICAO to ADEXP Translation	RQS	ICAO	ICAO ATS
SRS-006k – SPL ICAO to ADEXP Translation	SPL	ICAO	ICAO ATS
SRS-006l – ABI ICAO to ADEXP Translation	ABI	ICAO	ICAO OLDI
SRS-006m – ACP ICAO to ADEXP Translation	ACP	ICAO	ICAO OLDI
SRS-006n – ACT ICAO to ADEXP Translation	ACT	ICAO	ICAO OLDI
SRS-006o – AMA ICAO to ADEXP Translation	AMA	ICAO	ICAO OLDI
SRS-006p – CDN ICAO to ADEXP Translation	CDN	ICAO	ICAO OLDI
SRS-006q – COD ICAO to ADEXP Translation	COD	ICAO	ICAO OLDI
SRS-006r – INF ICAO to ADEXP Translation	INF	ICAO	ICAO OLDI
SRS-006s – LAM ICAO to ADEXP Translation	LAM	ICAO	ICAO OLDI
SRS-006t – MAC ICAO to ADEXP Translation	MAC	ICAO	ICAO OLDI
SRS-006u – OCM ICAO to ADEXP Translation	OCM	ICAO	ICAO OLDI
SRS-006v – PAC ICAO to ADEXP Translation	PAC	ICAO	ICAO OLDI
SRS-006w – RAP ICAO to ADEXP Translation	RAP	ICAO	ICAO OLDI
SRS-006x – REV ICAO to ADEXP Translation	REV	ICAO	ICAO OLDI
SRS-006y – RJC ICAO to ADEXP Translation	RJC	ICAO	ICAO OLDI
SRS-006z – ROC ICAO to ADEXP Translation	ROC	ICAO	ICAO OLDI
SRS-006aa – RRV ICAO to ADEXP Translation	RRV	ICAO	ICAO OLDI
SRS-006ab – SBY ICAO to ADEXP Translation	SBY	ICAO	ICAO OLDI

Table 22 - Supported ICAO Messages for ADEXP Format Translation

SRS-007 – Message format translation ADEXP to ICAO

The AOMT shall translate an ADEXP format message to an ICAO format message if the following conditions are all true:

- The message is recognised as ADEXP format;
- The message does not contain any errors that would prohibit identification of the format;
- The caller specifies that the message should be translated to an ICAO format;

[10] This message is translated to a BFD message, not an ADEXP CPL;

#

SRS-008 – Supported Messages for ICAO ADEXP translation

The AOMT shall support translation of the ADEXP format messages to the ICAO format for the messages shown in Table 23 - .

#

SRS Requirement	Message Title	Format	Type
SRS-008a – ABI ADEXP to ICAO Translation	ABI	ADEXP	ADEXP OLDI
SRS-008aa – ACP ADEXP to ICAO Translation	ACP	ADEXP	ADEXP OLDI
SRS-008b – ACT ADEXP to ICAO Translation	ACT	ADEXP	ADEXP OLDI
SRS-008c – AMA ADEXP to ICAO Translation	AMA	ADEXP	ADEXP OLDI
SRS-008d – CDN ADEXP to ICAO Translation	CDN	ADEXP	ADEXP OLDI
SRS-008e – COD ADEXP to ICAO Translation	COD	ADEXP	ADEXP OLDI
SRS-008f – IACH ADEXP to ICAO Translation	IACH	ADEXP	CFMU IFPS ATS
SRS-008g – IAFP ADEXP to ICAO Translation	IAFP	ADEXP	CFMU IFPS ATS
SRS-008h – IAPL ADEXP to ICAO Translation	IAPL	ADEXP	CFMU IFPS ATS
SRS-008i – IARR ADEXP to ICAO Translation	IARR	ADEXP	CFMU IFPS ATS
SRS-008j – ICHG ADEXP to ICAO Translation	ICHG	ADEXP	CFMU IFPS ATS
SRS-008k – ICNL ADEXP to ICAO Translation	ICNL	ADEXP	CFMU IFPS ATS
SRS-008l – IDEP ADEXP to ICAO Translation	IDEP	ADEXP	CFMU IFPS ATS
SRS-008m – IDLA ADEXP to ICAO Translation	IDLA	ADEXP	CFMU IFPS ATS
SRS-008n – IFPL ADEXP to ICAO Translation	IFPL	ADEXP	CFMU IFPS ATS
SRS-008o – INF ADEXP to ICAO Translation	INF	ADEXP	ADEXP OLDI
SRS-008p – IRQP ADEXP to ICAO Translation	IRQP	ADEXP	CFMU IFPS ATS
SRS-008q – LAM ADEXP to ICAO Translation	LAM	ADEXP	ADEXP OLDI
SRS-008r – MAC ADEXP to ICAO Translation	MAC	ADEXP	ADEXP OLDI
SRS-008s – OCM ADEXP to ICAO Translation	OCM	ADEXP	ADEXP OLDI
SRS-008t – PAC ADEXP to ICAO Translation	PAC	ADEXP	ADEXP OLDI
SRS-008u – RAP ADEXP to ICAO Translation	RAP	ADEXP	ADEXP OLDI
SRS-008v – REV ADEXP to ICAO Translation	REV	ADEXP	ADEXP OLDI
SRS-008w – RJC ADEXP to ICAO Translation	RJC	ADEXP	ADEXP OLDI
SRS-008x – ROC ADEXP to ICAO Translation	ROC	ADEXP	ADEXP OLDI
SRS-008y – RRV ADEXP to ICAO Translation	RRV	ADEXP	ADEXP OLDI
SRS-008z – SBY ADEXP to ICAO Translation	SBY	ADEXP	ADEXP OLDI

Table 23 - Supported ADEXP Messages for ICAO Format Translation

If the message format cannot be determined, it is due to the message being unrecognised.

SRS-009 – Message format translation failure

If the AOMT receives a message where the format cannot be determined, the AOMT shall return the message unchanged to the caller with an error.

#

SRS-010 – Message format translation inhibited

If the AOMT receives a message already in the target format, the AOMT shall return the message unchanged to the caller. There may or may not be errors present as detected by the parser.

#

SRS-011 – Invalid translation requests

If the AOMT receives a message that is only defined as an ADEXP format message with no equivalent ICAO definition and the translation instruction is given to translate to the ICAO format, the AOMT shall return the message to the caller unchanged with an error indicating that it is not possible to translate the message to the ICAO format.

The messages that are only in ADEXP format are listed in Table 24 - .

#

SRS Requirement	Message Title	Format	Type

SRS-011a – ACK ADEXP to ICAO Translation	ACK	ADEXP	CFMU IFPS ATS
SRS-011c – BFD ADEXP to ICAO Translation	BFD	ADEXP	CFMU IFPS ATS
SRS-011d – CFD Block ICAO Translation	CFD	ADEXP	ADEXP OLDI
SRS-011e – COF Block ICAO Translation	COF	ADEXP	ADEXP OLDI
SRS-011f – CRP Block ICAO Translation	CRP	ADEXP	ADEXP OLDI
SRS-011g – CRQ Block ICAO Translation	CRQ	ADEXP	ADEXP OLDI
SRS-011h – DES Block ICAO Translation	DES	ADEXP	CFMU ETFMS
SRS-011i – DPI ADEXP to ICAO Translation	DPI	ADEXP	CFMU IFPS ATS
SRS-011j – EFD ADEXP to ICAO Translation	EFD	ADEXP	CFMU IFPS ATS
SRS-011k – ERR Block ICAO Translation	ERR	ADEXP	CFMU ETFMS
SRS-011l – FLS Block ICAO Translation	FLS	ADEXP	CFMU ETFMS
SRS-011m – FUM ADEXP to ICAO Translation	FUM	ADEXP	CFMU IFPS ATS
SRS-011n – HOP Block ICAO Translation	HOP	ADEXP	ADEXP OLDI
SRS-011o – IRQS ADEXP to ICAO Translation	IRQS	ADEXP	CFMU IFPS ATS
SRS-011p – LOF Block ICAO Translation	LOF	ADEXP	ADEXP OLDI
SRS-011q – MAN ADEXP to ICAO Translation	MAN	ADEXP	CFMU IFPS ATS
SRS-011r – MAS Block ICAO Translation	MAS	ADEXP	ADEXP OLDI
SRS-011s – NAN Block ICAO Translation	NAN	ADEXP	ADEXP OLDI
SRS-011t – PNT Block ICAO Translation	PNT	ADEXP	ADEXP OLDI
SRS-011u – RDY Block ICAO Translation	RDY	ADEXP	CFMU ETFMS
SRS-011v – REJ ADEXP to ICAO Translation	REJ	ADEXP	CFMU IFPS ATS
SRS-011w – RLS Block ICAO Translation	RLS	ADEXP	ADEXP OLDI
SRS-011x – ROF Block ICAO Translation	ROF	ADEXP	ADEXP OLDI
SRS-011y – RRQ Block ICAO Translation	RRQ	ADEXP	ADEXP OLDI
SRS-011z – RTI Block ICAO Translation	RTI	ADEXP	ADEXP OLDI
SRS-011aa – SAM Block ICAO Translation	SAM	ADEXP	CFMU ETFMS
SRS-011ab – SCO Block ICAO Translation	SCO	ADEXP	ADEXP OLDI
SRS-011ac – SDM Block ICAO Translation	SDM	ADEXP	ADEXP OLDI
SRS-011ad – SKC Block ICAO Translation	SKC	ADEXP	ADEXP OLDI
SRS-011ae – SLC Block ICAO Translation	SLC	ADEXP	CFMU ETFMS
SRS-011af – SMM Block ICAO Translation	SMM	ADEXP	CFMU ETFMS
SRS-011ag – SRM Block ICAO Translation	SRM	ADEXP	CFMU ETFMS
SRS-011ah – SRR Block ICAO Translation	SRR	ADEXP	CFMU ETFMS
SRS-011ai – TFD Block ICAO Translation	TFD	ADEXP	CFMU ETFMS
SRS-011aj – TIM Block ICAO Translation	TIM	ADEXP	ADEXP OLDI
SRS-011ak – TIP Block ICAO Translation	TIP	ADEXP	ADEXP OLDI
SRS-011al – XAP Block ICAO Translation	XAP	ADEXP	ADEXP OLDI
SRS-011am – XCM Block ICAO Translation	XCM	ADEXP	ADEXP OLDI

SRS-011an – XIN Block ICAO Translation	XIN	ADEXP	ADEXP OLDI
SRS-011ao – XRQ Block ICAO Translation	XRQ	ADEXP	ADEXP OLDI

Table 24 - ADEXP Messages that cannot be translated to ICAO Format

SRS-012 – CADF message translation requests

The AOMT shall ignore CADF messages and return all CADF messages to the caller.

The CADF message set is CRAM, AUP and UUP.

#

2.3.3 Erroneous Message Processing

The AOMT checks all messages it receives to ascertain if it is a recognised and supported message. If a message is recognised, the AOMT checks the message for the correct number of fields and that each field is syntactically correct. When a message title is not recognisable and/or a message contains syntax errors, an error is generated by the parser.

Broadly speaking there are two types of error:

- Errors that result in insufficient data being extracted from a message to perform a translation. For example, if a message title cannot be recognised then parsing cannot be carried out at all and it will not be possible to translate a message.
- Errors that are less severe such as consistency checking errors, for example a message may have been successfully parsed but when the parser checks if the rules match the associated field 15 route description, an error may be generated if these are inconsistent, (VFR rules but IFR routing given in field 15). Such errors can be effectively ignored and translation can proceed.

In order to support the definition of which erroneous messages can be translated and which cannot, configuration data is provided in which errors can be listed as ‘to be ignored’ for translation.

SRS-014 – Unrecognised messages

If the AOMT recognises that a message is not one of the supported messages defined in this document, the AOMT shall forward the message without modification with an error indicating the message is unrecognised.

Recognition of a message is carried out using the message title, (ICAO field 3a or the ADEXP ‘TITLE’ field).

#

SRS-015 – Erroneous message content

If the content of a message is syntactically incorrect, and the error(s) generated is/are ***not*** listed as an error to ignore for translation, the AOMT shall return the original message without any translation modification with the error(s) to the caller.

#

2.4 CSCI External Interface Requirements

The AOMT runs as a server expecting connections from a client; for additional details on the interface refer to the IDD [6].

2.5 Adaptation Requirements

The AOMT uses the parser to parse messages; the ATS message parser can be configured to support any ADEXP message using configuration data. The ADEXP configuration data is specified in four XML files:

- **Supported messages** – this file contains a list of supported ADEXP message titles, and for each title, a list of the ADEXP primary fields contained in a given message title. For each primary field, the number of occurrences, whether it is optional or compulsory are also specified in this configuration file.
- **Primary fields** – this file contains a list of all primary fields (includes basic, list and structured fields). This file only requires modification if new primary fields are added; the delivered software contains all the primary fields defined in [2] and rarely requires modification unless custom fields are required.

- **Sub-fields** – this file contains a list of all sub- fields (includes basic, list and structured sub-fields). This file only requires modification if new sub-fields are added; the delivered software contains all the sub-fields defined in [2] and rarely requires modification unless custom sub-fields are required.
- **Auxiliary Terms** – this file contains the regular expressions defined for each term defined in the ADEXP specification in [2]. There is also an error message defined for each auxiliary term that is reported if the parser detects that the data for a given field does not comply to that specified by the auxiliary term. This file only requires modification if new auxiliary terms are added; the delivered software contains all the auxiliary terms defined in [2] and rarely requires modification unless custom auxiliary terms are required.

The ICAO ATS messages are fixed format and cannot be specified using configuration data.

For the location of the configuration data refer to the IDD [6].

The ICAO OLDI messages have a structure that is bilaterally agreed which means that the message structure is ATC unit dependant. The OLDI ICAO message parser uses a configuration data set that specifies for a given adjacent unit the message structure.

SRS-016 – ADEXP configuration data

The AOMT shall support the definition of supported ADEXP message titles along with their content in off line configuration data.

#

SRS-017 – ICAO OLDI configuration data

The AOMT shall support the definition of supported ICAO format OLDI message titles along with their content on a per adjacent unit definition in off line configuration data.

#

SRS-018 – Missing configuration data

The AOMT shall throw an exception if any of the configuration files are missing.

#

2.6 CSCI Environment Requirements

The AOMT requires a Java runtime environment 1.8.x;

2.7 Computer Resource Requirements

2.7.1 Computer Hardware Requirements

The AOMT will run on any modern PC or workstation provided the correct Java runtime is installed. The AOMT service is operating system independent and requires only a correct Java runtime be installed on the machine it is to be executed on.

2.7.2 Computer Hardware Resource Utilization Requirements

The AOMT does not consume any appreciable resources of the machine it runs on.

2.7.3 Computer Software Requirements

No specific operating system is required; providing the operating system supports a Java runtime environment of 1.8.x.

2.7.4 Computer Communications Requirements

The network protocol TCP/IP must be installed and running.

2.8 Software Quality Factors

Not applicable

2.9 Design and Construction Constraints

There are a number of constraints related to how some of the ADEXP fields are populated when translating from ICAO to ADEXP. The following ADEXP fields require content that is not normally available in ICAO messages:

- RTEPTS – This contains the extracted route sequence, what is commonly refered to as the trajectory. The information is comprised of a list of points with calculated flight level, speed and estimated time at the point.
- ADDR – This contains addressing information typically found in the ICAO AFTN header, not the message body.
- SRC – This contains the source information for a message.
- IFPLID – This is a unique identifier for a flight plan which is also output in any and all update messages associated to a flight plan.

The solutions on how to deal with these issues is described in the following sections.

2.9.1 ADEXP RTEPTS Field

The AOMT message parser extracts a route based on the information in the ICAO field 15. The extracted route is not as complete as one sent from the IFPS as it does not represent a 'trajectory'. However, the content does expressed the route as a sequence of points with connecting routes; each point is output with a flight level and speed. It is not possible to calculate the ETO's as a trajectory would have to be calculated which the message parser cannot do.

As an example, let us assume that an ICAO field 15 contains the following data:

```
N0458F400 DCT EVX UT300 SENLO UN502 RATKA DCT GUNSO DCT BEDRA/M080F400 DCT
48N020W/M080F400 45N030W 44N040W 44N050W/N0457F400 DCT BOBTU/N0454F430 DCT JAROM…
```

The RTEPTS output generated would be as shown below:

```
-BEGIN RTEPTS
-PT -PTID LFPB -FL F400 -PTSPEED N0458
-PT -PTID EVX -FL F400 -PTSPEED N0458
-PT -PTID SENLO -FL F400 -PTSPEED N0458
-PT -PTID RATKA -FL F400 -PTSPEED N0458
-PT -PTID GUNSO -FL F400 -PTSPEED N0458
-PT -PTID BEDRA -FL F400 -PTMACH M080
-PT -PTID GEO01 -FL F400 -PTMACH M080
-PT -PTID GEO02 -FL F400 -PTMACH M080
-PT -PTID GEO03 -FL F400 -PTMACH M080
-PT -PTID GEO04 -FL F400 -PTSPEED N0457
-PT -PTID BOBTU -FL F430 -PTSPEED N0454
-PT -PTID JAROM -FL F430 -PTSPEED N0454
-PT -PTID TOPPS -FL F430 -PTSPEED N0454
-PT -PTID ENE -FL F430 -PTSPEED N0454
-PT -PTID BOS -FL F430 -PTSPEED N0454
-PT -PTID BDL -FL F430 -PTSPEED N0454
-PT -PTID KBDL -FL F430 -PTSPEED N0454
-END RTEPTS
```

The ‘GEOxx’ items are a reference to an ADEXP ‘GEO’ field that contains information about a latitude/longitude point. Given the information available in field 15 this is the best output the AOMT can generate.

2.9.2 ADEXP ADDR Field

The ADEXP ADDR field contains information about the message originator, addressees and filing time that are all needed as part of the ADDR field. This information is contained in the ICAO AFTN message header. In order that the AOMT CSCI obtains this information, messages sent to the AOMT will contain the ICAO AFTN message header. The AOMT will then use this information to construct the message output when translating from the ICAO to ADEXP format.

There remains an outstanding issue on how to obtain this information for the ADEXP OLDI messages. At the time of producing this document the OLDI header format remains unknown. Further updates will be made to this section once more information about the OLDI message header format is known.

2.9.3 ADEXP SRC Field

When translating from ICAO to ADEXP it is not always possible to obtain the data from the ICAO format message to populate the ADEXP SRC field. There is an ICAO field 18 field item called SRC, if present the AOMT will use it, however, the SRC is not compulsory.

If the SRC is not present in an ICAO format message, the following mapping will be used to construct the SRC output in the ADEXP format message:

ADEXP Message Title	ADEXP SRC Field Content	ADEXP Message Title	ADEXP SRC Field Content
IFPL	FPL	ICHG	FNM
IACH	AFP	ICNL	FNM
IAFP	AFP	IDEP	FNM
IAPL	AFP	IDLA	FNM
IARR	FNM		

Table 25 - ADEXP SRC Field Content Derivation

2.9.4 ADEXP IFPLID Field

ICAO Messages do not contain an IFPLID field; as a result the ADEXP message will be created with a default IFPLID of 'AA00000000'.

1.1.1 ICAO to ADEXP Translation

There are some difficulties translating ICAO format messages to their ADEXP equivalents. Many fields are defined in the ADEXP messages that are not present in the equivalent ICAO messages. This situation is particularly acute when dealing with the OLDI messages. Typically, the bilaterally agreed data is contained in the ICAO F22 amendment field, but in the ADEXP definition, many of the bilaterally agreed fields are 'compulsory' basic primary ADEXP fields.

In order to accommodate this difference in content, the internal message parser rules for many of the ADEXP messages have been relaxed to make them optional. The AOMT will try where it can to output as much data as it can.

The process for F18 works as follows:

- Loop over all primary fields defined for an ADEXP message and check if a field has an ICAO F18 equivalent (this mapping is defined in the ADEXP configuration data).
- For each field that has a F18 ICAO equivalent, check if the F18 item is available in the flight plan data record.
- If a match is found retrieve the data from the flight plan record and write it out to the ADEXP output.

For F22 a similar processing concept is applied:

- Loop over all primary fields defined for a given ADEXP message title and check if a field has an ICAO equivalent that is not F18 (this mapping is defined in the ADEXP configuration data).
- For each field that has an ICAO equivalent, check if the field (based on its field number) is available in the flight plan data record F22 amendment field.
- If a match is found, write this data out to the ADEXP output.

The following example illustrates the process:

- In the ADEXP ABI message definition, there is a mandatory field specified called ARCTYP that contains the aircraft type (e.g. B737). In ICAO, the aircraft type will only ever appear in the F22 amendment field as a part of field 9.
- In the ADEXP configuration data, the field ARCTYP contains the ICAO equivalent field defined as 'F9,b'.
- Using the field number 9, it is possible to search the flight data record to check if there is a field 9 in the F22 amendment list, if there is, then write the data out to the ARCTYP field.

It should be noted that if there is no field 9 present in the F22 amendment field, the ARCTYP field will not be output at all.

2.9.5 TTLEET Field

Many ICAO ATS Update messages do not contain the estimated elapsed time (EET) as shown in the following example:

```
(DEP-CPA234-LIMC1131-VHHH-DOF/150426)
```

There is no EET given in this message (would normally follow the VHHH, i.e. VHHH0200)

In ADEXP, the TTLEET in an IDEP message is specified as compulsory.

The AOMT will insert '0000' if the TTLEET field is required in a target ADEXP message but is missing from the ICAO ATS message.

2.9.6 EOBD Field

Many ICAO message do not contain a field 18 DOF field; when translating to ADEXP the DOF is often compulsory. The AOMT will insert the current date if the DOF field is required in a target ADEXP message but is missing from the ICAO ATS message.

2.9.7 ARR Field 16/17

When an ICAO ARR message is translated into ADEXP format and the message contains F17 rather than F16, the ADEXP ADES will be sourced from field 17a rather than field 16a.

2.9.8 CPL EOBT

ICAO CPL messages do not contain an EOBT; when translating to a BFD message the EOBT is compulsory. The AOMT will insert '0000' into the ADEXP EOBT field.

2.10 Training Related Requirements

Technical training is required to understand how to update and/or modify the configuration data. No special training software is required.

2.11 Logistics Related Requirements

This CSCI forms a part of a larger system and will be deployed with the complete system. No special handling is required. The AOMT is started from Unix/Linux shell script or DOS batch file and will be started as part of the overall system start-up.

2.12 Other Requirements

SRS-019 – Message processing latency

The AOMT shall be able to process a minimum of 100 messages per minute.

#

2.13 Precedence and Criticality of Requirements

Not applicable;

3 Appendix A – Supported ICAO ATS Messages

All the messages listed below are in ICAO format only.

Message Title	Description
ARR	Arrival
CHG	Modification
CNL	Flight cancellation
CPL	Current flight plan
DEP	Departure
DLA	Delay
EST	Estimate
FPL	Filed flight plan
RQP	Request flight plan
RQS	Request supplementary flight plan
SPL	Supplementary flight plan

Table 26 - Supported ICAO Messages

4 Appendix B – Supported CFMU IFPS ATS Messages

All the messages listed below are in ADEXP format only.

Title	Type	Comment
ACK	ACKNOWLEDGE	IFPS Operational Reply Message
IACH	INDIVIDUAL ATC MODIFICATION	
IAFP	INDIVIDUAL ATC FLIGHT PLAN PROPOSAL	
IAPL	INDIVIDUAL ATC FLIGHT PLAN	
IARR	INDIVIDUAL ARRIVAL	
ICHG	INDIVIDUAL MODIFICATION	
ICNL	INDIVIDUAL CANCELLATION	
IDEP	INDIVIDUAL DEPARTURE	
IDLA	INDIVIDUAL DELAY	
IFPL	INDIVIDUAL FLIGHT PLAN	
IRQP	INDIVIDUAL FLIGHT PLAN REQUEST	
MAN	MANUAL PROCESSING PENDING	IFPS Operational Reply Message
REJ	REJECT	IFPS Operational Reply Message

Table 27 - Supported CFMU IFPS ATS Messages

5 Appendix C – Supported CFMU ETFMS Messages

All the messages listed below are in ADEXP format only.

Title	Type	Comment
DES	DE-SUSPENSION	
ERR	ERROR	ETFMS Operational Reply Message
FCM	FLIGHT CONFIRMATION	This is an AO -> ETFMS message, it is not clear why an ATC unit would receive such a message – this is not normally the case?
FLS	FLIGHT SUSPENSION	
RDY	READY MESSAGE	More information to be provided once definition is obtained.
RJT	RE-ROUTEING REJECTION	This is an AO -> ETFMS message, it is not clear why an ATC unit would receive such a message – this is not normally the case?
RRP	RE-ROUTEING PROPOSAL	This is an ETFMS -> AO message, it is not clear why an ATC unit would receive such a message – this is not normally the case?
SAM	SLOT ALLOCATION	
SIP	SLOT IMPROVEMENT PROPOSAL	This is an ETFMS -> AO message, it is not clear why an ATC unit would receive such a message – this is not normally the case?
SLC	SLOT REQUIREMENT CANCELLATION	
SMM	SLOT MISSED	
SPA	SLOT PROPOSAL ACCEPTANCE	This is an ETFMS -> AO message, it is not clear why an ATC unit would receive such a message – this is not normally the case?
SRJ	SLOT PROPOSAL REJECTION	This is an ETFMS -> AO message, it is not clear why an ATC unit would receive such a message – this is not normally the case?
SRM	SLOT REVISION	
SRR	SLOT REVISION REQUEST	Cannot locate message definition

Table 28 - Supported CFMU ETFMS Messages

Design Note: The SRR and RDY messages are unknown – format to be located in order to specify the ADEXP parser configuration data.

6 Appendix D – Supported OLDI Messages

Message Title	Description	Comment	ADEXP Format	ICAO Format
ABI	ADVANCE BOUNDARY INFORMATION		✔	✗
ACP	ACCEPTANCE		✔	✔
ACT	ACTIVATION		✔	✔
AMA	Arrival Manager		✔	✔
BFD	BASIC FLIGHT DATA	Not defined in OLDI 2.0	✔	✗
CDN	CO-ORDINATION		✗	✔
CFD	CHANGE TO FLIGHT DATA	Not defined in OLDI 2.0	✔	✗
COD	SSR CODE ASSIGNMENT		✔	✔
COF	CHANGE OF FREQUENCY	No ICAO format for this message is defined in OLDI 2.0	✔	✗
HOP	HAND-OVER PROPOSAL	No ICAO format for this message is defined in OLDI 2.0	✔	✗
INF	INFORMATION		✔	✔
LAM	LOGICAL ACKNOWLEDGEMENT MESSAGE		✔	✔
MAC	MESSAGE FOR ABROGATION OF CO-ORDINATION		✔	✔
MAS	MANUAL ASSUMPTION OF COMMUNICATIONS	No ICAO format for this message is defined in OLDI 2.0	✔	✗
OCM	Oceanic Message		✔	✔
PAC	PRELIMINARY ACTIVATION		✔	✔
RAP	REFERRED ACTIVATE PROPOSAL		✔	✔
REV	REVISION	Messages presented in OLDI 2.0 but the message format is not given.	✔	✔
RJC	REJECT CO-ORDINATION		✔	✔
ROF	REQUEST ON FREQUENCY	No ICAO format for this message is defined in OLDI 2.0	✔	✗
RRV	REFERRED REVISION PROPOSAL		✔	✔
SBY	STAND-BY		✔	✔
SDM	SUPPLEMENTARY DATA	No ICAO format for this message is defined in OLDI 2.0	✔	✗
TIM	TRANSFER INITIATION	No ICAO format for this message is defined in OLDI 2.0	✔	✗

Message Title	Description	Comment	ADEXP Format	ICAO Format
XAP	CROSSING ALTERNATE PROPOSAL	Message not defined in OLDI 2.0, did not appear until OLDI 3.0. No ICAO format for this message is defined.	✔	✗
XCM	CROSSING CANCELLATION	Message not defined in OLDI 2.0, did not appear until OLDI 3.0. No ICAO format for this message is defined.	✔	✗
XIN	CROSSING INTENTION NOTIFICATION	Message not defined in OLDI 2.0, did not appear until OLDI 3.0. No ICAO format for this message is defined.	✔	✗
XRQ	CROSSING REQUEST	Message not defined in OLDI 2.0, did not appear until OLDI 3.0. No ICAO format for this message is defined.	✔	✗

Table 29 - Supported OLDI Messages

7 Appendix E – Supported CADF Messages

SUPPORTED CADF MESSAGES

Message Title	Description	Comment	ADEXP Format	ICAO Format
CRAM			✔	✗
UUP			✔	✗
AUP			✔	✗

Table 30 - Supported CADF Messages

8 Appendix F – ADEXP to ICAO Field Mapping

The Flight ATM Parser is delivered with all the ADEXP fields that have an equivalent ICAO field, mapped to the relevant ICAO field. In addition the ETFMS ADEXP field CTOT and ADEXP fields IFPLID and ELDT are included as child nodes of the 'fields' node.

When mapping ADEXP fields to an ICAO field, the ICAO field numbers must be expressed as 'F[0-9]{1,2}'; the subfields must also be as used by ICAO, e.g. '[a-e]{1}'. If the fields and/or subfields are named according to a different schema they will not be copied from the ADEXP field to the ICAO output document. F18 and F19 use the equivalent subfield names as described by ICAO in all cases apart from F19, (syntax shown in the following table).

The table shown below defines:

- The ADEXP field name mapped to its ICAO equivalent field;
- The ICAO field name and subfield syntax;
- The Xpath path to use in order to access a given field;

ADEXP Field	ICAO Field	ICAO Subfield	Xpath Path
TITLE	F3	'a'	fdr/fields/F3/a
ARCID	F7	'a'	fdr/fields/F7/a
SSRCODE	F7	'c'	fdr/fields/F7/c
FLTRUL	F8	'a'	fdr/fields/F8/a
FLTTYP	F8	'b'	fdr/fields/F8/b
NBARC	F9	'a'	fdr/fields/F9/a
ARCTYP	F9	'b'	fdr/fields/F9/b
WKTRC	F9	'c'	fdr/fields/F9/c
CEQPT	F10	'a'	fdr/fields/F10/a
SEQPT	F10	'b'	fdr/fields/F10/b
ADEP	F13	'a'	fdr/fields/F13/a
ATD, ATOT, EOBT	F13	'b'	fdr/fields/F13/b
COORDATA basic subfield PTID	F14	'a'	fdr/fields/F14/a
COORDATA basic subfield TO	F14	'b'	fdr/fields/F14/b
COORDATA basic subfield STO	F14	'b'	fdr/fields/F14/b
COORDATA basic subfield TFL	F14	'c'	fdr/fields/F14/c
COORDATA basic subfield SFL	F14	'd'	fdr/fields/F14/d
ROUTE	F15	None[11]	fdr/fields/F15
ADES	F16	'a'	fdr/fields/F16/a
TTLEET	F16	'b'	fdr/fields/F16/b
ALTRNT1	F16	'ca'	fdr/fields/F16/ca
ALTRNT2	F16	'cb'	fdr/fields/F16/cb
ATA	F17	'b'	fdr/fields/F17/b

[11] The extracted route is placed in the 'route' node – refer to the XSD for details;

ADEXP Field	ICAO Field	ICAO Subfield	Xpath Path
ALTNZ	F18	‘altn’	fdr/fields/F18/altn
AWR	F18	‘awr’	fdr/fields/F18/awr
ARCADDR	F18	‘code’	fdr/fields/F18/code
COM	F18	‘com’	fdr/fields/F18/com
DAT	F18	‘dat’	fdr/fields/F18/dat
DEPZ	F18	‘dep’	fdr/fields/F18/dep
DESTZ	F18	‘dest’	fdr/fields/F18/dest
DLE	F18	‘dof’	fdr/fields/F18/dle
EETFIR	F18	‘eet’	fdr/fields/F18/eet
EOBD	F18	‘dof’	fdr/fields/F18/dof
EUR	F18	‘eur’	fdr/fields/F18/eur
IFP	F18	‘ifp’	fdr/fields/F18/ifp
NAV	F18	‘nav’	fdr/fields/F18/nav
OPR	F18	‘opr’	fdr/fields/F18/opr
ORGN	F18	‘orgn’	fdr/fields/F18/orgn
PBN	F18	‘pbn’	fdr/fields/F18/pbn
PER	F18	‘per’	fdr/fields/F18/per
RALT	F18	‘ralt’	fdr/fields/F18/ralt
REG	F18	‘reg’	fdr/fields/F18/reg
RFP	F18	‘rfp’	fdr/fields/F18/rfp
RIF	F18	‘rif’	fdr/fields/F18/rif
RMK[12]	F18	‘rmk’	fdr/fields/F18/rmk
RVR	F18	‘rvr’	fdr/fields/F18/rvr
SEL	F18	‘sel’	fdr/fields/F18/sel
SRC	F18	‘src’	fdr/fields/F18/src
STAYINFO[1-9]	F18	‘stayinfo[1-9]’	fdr/fields/F18stayinfo[1-9]
STS[13]	F18	‘sts’	fdr/fields/F18/sts
SUR	F18	‘sur’	fdr/fields/F18/sur
TALT	F18	‘talt’	fdr/fields/F18/talt
TYPZ	F18	‘typ’	fdr/fields/F18/typ
SPLA	F19	‘a’	fdr/fields/F19/a
SPLC	F19	‘c’	fdr/fields/F19/c
SPLDCAP[14]	F19	‘d2’	fdr/fields/F19/d2

[12] This subfield can occur more than once;
[13] This subfield can occur more than once;
[14] This field is part of ICAO F19 ‘d’ but is split into 4 component parts in ADEXP; if any of these subfields are found they are concatenated and ‘put back together’ as field ‘d’ and can be accessed using ‘fdr/fields/F19/d’.

ADEXP Field	ICAO Field	ICAO Subfield	Xpath Path
SPLDCOL[15]	F19	‘d4’	fdr/fields/F19/d4
SPLDCOV[16]	F19	‘d3’	fdr/fields/F19/d3
SPLDNB[17]	F19	‘d1’	fdr/fields/F19/d1
SPLE	F19	‘e’	fdr/fields/F19/e
SPLJ	F19	‘j’	fdr/fields/F19/j
SPLN	F19	‘n’	fdr/fields/F19/n
SPLP	F19	‘p’	fdr/fields/F19/p
SPLR	F19	‘r’	fdr/fields/F19/r
SPLS	F19	‘s’	fdr/fields/F19/s

Table 31 - ADEXP to ICAO Field Mapping

The following table lists the ADEXP fields that have no ICAO equivalent that are also included in the W3C output file.

ADEXP Field	Xpath path
CTOT	fdr/fields/CTOT
ELDT	fdr/fields/ELDT
IFPLID	fdr/fields/IFPLID
CFL	fdr/fields/CFL

Table 32 - Additional ADEXP Fields

[15] This field is part of ICAO F19 ‘d’ but is split into 4 component parts in ADEXP; if any of these subfields are found they are concatenated and ‘put back together’ as field ‘d’ and can be accessed using ‘fdr/fields/F19/d’.
[16] This field is part of ICAO F19 ‘d’ but is split into 4 component parts in ADEXP; if any of these subfields are found they are concatenated and ‘put back together’ as field ‘d’ and can be accessed using ‘fdr/fields/F19/d’.
[17] This field is part of ICAO F19 ‘d’ but is split into 4 component parts in ADEXP; if any of these subfields are found they are concatenated and ‘put back together’ as field ‘d’ and can be accessed using ‘fdr/fields/F19/d’.

9 Appendix G - Acronyms

	Acronym	Description
A		
	ABI	Advanced Boundary Information message (OLDI)
	ACARS	Air Ground Data Link VHF Airline Communications and Reporting System
	ACK	EUROCONTROL Operational Reply Message indicating a message was successfully accepted by EUROCONTROL and automatically processed
	ACP	Acceptance message (OLDI)
	ACT	Activation message (OLDI)
	ADEP	Departure Aerodrome
	ADES	Destination Aerodrome
	ADEXP	ATS Data Exchange Presentation
	AFIL	Air Filed Flight Plan
	AFTN	Aeronautical Fixed Telecommunications Network
	AIP	Aeronautical Information Publication
	ALTNZ	ICAO F18 subfield, this is the ADEXP equivalent to the F18 subfield ALTN;
	AMA	Arrival Manager Constraint message (OLDI)
	AO	Airline Operator
	AOMT	ATS and OLDI Message Translator
	API	Application Programming Interface
	ARCID	Aircraft Identification (callsign)
	ARR	ICAO ATS Arrival Message
	ATA	Actual Time of Arrival
	ATC	Air Traffic Control
	ATD	Actual Time of Departure
	ATFCM	Air Traffic Flow and Capacity Management
	ATM	Air Traffic Management
	ATMGW	ATM Gateway
	ATOT	Actual Take Off Time
	ATS	Air Traffic Service
	AUP	Airspace Use Plan
	AWR	ICAO F18 subfield, indicates AO What-if Re-routing
B		
	BFD	Basic Flight Data (DFS version of an IFPL)
C		
	CADF	Centralised Airspace Data Function
	CDN	Coordination Message (OLDI)
	CDRL	Contract Deliverables Requirements List
	CFD	Change to Flight Data (DFS version of ICHG)

	Acronym	Description
	CFL	Cleared Flight Level
	CFMU	Central Flow Management Unit
	CHG	ICAO ATS Change Message
	CNL	ICAO ATS Cancel Message
	COD	SSR Code Assignment Message (OLDI)
	CODE	ICAO F18 subfield, specifies the aircraft code
	COF	Change of Frequency (OLDI)
	COM	ICAO F18 sub-field containing information about communication equipment
	CPL	ICAO ATS Current Flight Plan message title
	CRAM	Conditional Route Availability Message
	CRP	Clearance Response Message (OLDI)
	CRQ	Clearance Request Message (OLDI)
	CSCI	Computer Software Configuration Item
	CTOT	Calculated Take Off Time
D		
	DAT	ICAO F18 subfield, indicating the data link capability of an aircraft.
	DEP	ICAO ATS Departure Message
	DEPZ	ICAO F18 sub-field, ADEP plain text name
	DES	De-Suspension Message (ETFMS)
	DESTZ	ICAO F18 sub-field, ADES plain text name
	DLA	ICAO ATS Delay Message
	DLE	ICAO F18 subfield, indicates a delay on a point of the route;
	DOM	Dictionary of Messages, EUROCONTROL ICD for the CFMU;
	DOS	Disk Operating System
	DPS	Data Playback System
E		
	ELDT	Estimated Landing Time
	EOBD	Estimated Off Block Date
	EOBT	Estimated Off Block Time
	ERR	ETFMS Operational Reply Message
	ETFMS	Enhanced Tactical Flow Management System
	EUR	Euro Currency
	EUROCONTROL	European Organisation for the Safety of Air Navigation
F		
	FATM	Flight ATM Systems Ltd.
	FCM	Flight Confirmation, ETFMS message for AO’s
	FLS	Flight Suspension Message (ETFMS)
	FMC	Flight Management Computer

	Acronym	Description
	FPL	Flight Plan Message (ICAO)
G		
	GAT	General Aviation Traffic
H		
	HOP	Hand Over Proposal
I		
	IACH	EUROCONTROL ATC Change Message
	IAFP	EUROCONTROL ATC Flight Plan Proposal Message
	IAPL	EUROCONTROL ATC Flight Plan Message
	IARR	EUROCONTROL ATS Arrival Message
	ICAO	International Civil Aviation Organisation
	ICHG	EUROCONTROL ATS Change Message
	ICNL	EUROCONTROL ATS Cancel Message
	IDD	Interface Design Document
	IDEP	EUROCONTROL ATS Departure Message
	IDLA	EUROCONTROL ATS Delay Message
	IFP	ICAO F18 sub-field, Indication of known errors within a FPL
	IFPL	EUROCONTROL ATS Flight Plan Message
	IFPLID	Unique Flight Plan Identifier originated by the CFMU
	IFPS	Integrated Initial Flight Plan Processing System
	INF	Information Message (OLDI)
	IP	Internet Protocol
	IRQP	EUROCONTROL Request Flight Plan Message
	IT	Information Technology
J		
	JRE	Java Runtime Environment
	JUnit	Java test environment/tool
L		
	LAM	Logical Acknowledgement Message (OLDI)
	LOF	Logon Forward message (OLDI)
	LOWW	ICAO Location Indicator for Vienna Airport
M		
	MAC	Message for Abrogation of Coordination (OLDI) or Media Access Control
	MAN	EUROCONTROL Operational Reply Message indicating a message received by EUROCONTROL required manual processing
	MAS	Manual Assumption of Communication Message (OLDI)
	MFG	Manufacturing
N		
	NAN	Next Authority Notified Message (OLDI) or National Administration Network (GACA)

	Acronym	Description
	NATO	North Atlantic Treaty Organization (Brussels, Belgium)
	NAV	ICAO F18 sub-field, contains additional information about navigation equipment on board an aircraft;
O		
	OAT	Operational Air Traffic
	OCM	Oceanic Message (OLDI)
	OLDI	On-Line Data Interchange
	OPR	ICAO F18 sub-field, contains name of the airline operator
	OR	Operational Release
	ORGN	ICAO F18 sub-field, contains AFTN originator address
P		
	PAC	Preliminary Activation Message (OLDI)
	PBN	ICAO F18 subfield, used to specify the ICAO code for RNAV and RNP capabilities;
	PC	Planning Controller or Personal Computer
	PER	ICAO F18 sub-field, contains aircraft performance data
	PNT	Point Message (OLDI)
Q		
R		
	RALT	ICAO F18 sub-field, used to specify the name of en-route alternative aerodromes;
	RAP	Referred Activate Proposal Message (OLDI)
	RDY	Ready Message (OLDI)
	REG	ICAO F18 sub-field, contains aircraft registration markings
	REJ	EUROCONTROL Operational Reply Message indicating a message was rejected by EUROCONTROL and returned to sender
	REV	Revision Message (OLDI)
	RFP	ICAO F18 sub-field, repetitive flight plan indicator
	RIF	ICAO F18 sub-field, used to specify a revised route subject to clearance in flight, and terminating with the ICAO designator of the revised aerodrome of destination.
	RJC	Reject Coordination Message (OLDI)
	RJT	Re-Routing Rejection Message (ETFMS) from CFMU to AO
	RLS	Release Message (OLDI)
	RMK	ICAO F18 sub-field, contains free text remarks
	RNP	Required Navigation Performance
	ROC	Rate of Climb
	ROF	Request on Frequency Message (OLDI)
	RPL	Repetitive Flight Plan
	RQP	Request Flight Plan (ICAO)
	RQS	Request Supplementary Information (ICAO)

	Acronym	Description
	RRP	Re-Routing Proposal Message (ETFMS) from CFMU to AO
	RRQ	Release Request Message (OLDI)
	RRV	Referred Revision Proposal Message (OLDI)
	RTI	Request Tactical Instruction Message (OLDI)
	RVR	Runway Visual Range
S		
	SAM	Slot Allocation Message (ETFMS)
	SBY	Standby Message (OLDI)
	SCO	Skip Communication Message (OLDI)
	SDM	Supplementary Data Message (OLDI)
	SEL	ICAO F18 sub-field, used to specify a SELCAL code.
	SFL	Supplementary Flight Level
	SIP	Software Integration Plan
	SKC	Skip Cancellation Message (OLDI)
	SLC	Slot Requirement Cancellation Message (ETFMS)
	SMM	Sector Monitor Message (RMCDE)
	SPA	Slot Proposal Acceptance Message (ETFMS) from CFMU to AO
	SPL	ICAO ATS Supplementary Flight Plan Message
	SRC	ICAO F18 subfield, specifies the source of a message
	SRJ	Slot Proposal Rejection Message (ETFMS) from CFMU to AO
	SRM	Slot Revision Message (ETFMS)
	SRR	Slot Revision Request Message (ETFMS)
	SRS	System Requirements Specification
	SSR	Secondary Surveillance Radar
	SSS	System Segment Specification
	STAYINFO	A token in ICAO field 15 specifying holding in a given airspace;
	STD	Standard
	STS	ICAO F18 sub-field, contains free text (priority flights)
	SUR	ICAO F18 sub-field, specifies surveillance applications or capabilities not in SEQPT
	SVC	Service Message (AFTN)
T		
	TALT	ICAO F18 sub-field, contains the name of alternate take-off aerodromes;
	TBD	To Be Decided
	TBP	To Be Provided
	TCP	Transmission Control Protocol
	TFL	Coordinated entry level
	TIM	Transfer Initiation Message (OLDI)
	TIP	Tactical Instruction Proposal Message (OLDI)

	Acronym	Description
	TYPZ	ICAO F18 sub-field, contains the type of aircraft when no ICAO code exists.
U		
	URB	User Requirement Brief
	URD	User Requirement Document
	UUP	Updated Airspace Use Plan
V		
W		
	W3C	World Wide Web Consortium
X		
	XAP	Crossing Alternate Proposal Message (OLDI)
	XCM	Crossing Cancellation Message (OLDI)
	XIN	Crossing Intention Notification Message (OLDI)
	XML	Extensible Mark up Language
	XRQ	Crossing Clearance Request Message (OLDI)
	XSD	XML Schema Definition
Y		
Z		

Table 33 - Acronyms

LAST PAGE OF DOCUMENT