

**Traffic Flow Management
Data to Industry:
Interface Control Document**

Version 1.7

9 May 2007

Prepared by

Volpe Center
55 Broadway
Cambridge, MA 02142

Table of Contents

1.	Introduction	4
1.1.	Background	4
1.2.	Purpose of this Document	4
1.3.	The TFMDI Servers	5
1.4.	References	5
2.	Overview of the TFM Data	6
2.1.	Introduction	6
2.2.	XML File Format Ground Rules	6
2.3.	How to Read the Tables that Describe the Data	7
3.	TFMDI Index File	8
4.	Public Reroutes and Initial Reroute Flight Lists	12
5.	Public Flow Evaluation Areas/Flow Constrained Areas	20
6.	Flow Evaluation Area/Flow Constrained Area Flight Lists	33
7.	Alert Data	33
8.	Collaborative Convective Forecast Product (CCFP) Data	33
9.	National Convective Weather Forecast (NCWF) Data	33
Appendix A. Color Table		34

Revision History			
Version	Date	Author's Initials	Description of Change
1.0	8/10/04	KH	Initial version .
1.1	9/20/04	KH	Documentation added for FCAs and public FEAs in Section 5.
1.2	12/14/04	RO	Section 2.4 on dealing with redundancy added. Description of zipping of files added.
1.3a	6/30/05	MH	Updated TFMDI index file description in Section 4 to v.2. Updated reroute data description in Section 4 to ETMS 8.0. Corrected FCA data descriptions in Section 5 as necessary. Minor changes to formats and phraseology throughout.
1.3	7/14/05	MH	After extensive reviews, many corrections and clarifications.
1.4	12/9/05	MG	Modified Index, FEA, and Reroute definitions to reflect changes for ETMS 8.2.
1.5	6/16/06	MG	Modified Table 2 to reflect new/modified data elements that will be included in the Reroute definition file beginning with the ETMS 8.3 release.
1.6	12/13/06	MG	Modified Table 3 to reflect new/modified data elements that will be included in the FEA/FCA definition file beginning with the ETMS 8.4 release.
1.7	5/9/07	MG	Modified Tables 3 and 5 to indicate that COLOR_ID is an optional tag within the <PRIMARY_FILTER> definition,

			and its presence should not be expected.
--	--	--	--

1. Introduction

1.1. Background

Currently, the FAA provides a great deal of data to Collaborative Decision Making (CDM) participants on the Common Constraint Situation Display (CCSD). The CCSD, in effect, provides a map that graphically displays current information relevant to traffic flow management (TFM) such as flow evaluation areas (FEAs), flow constraint areas (FCAs), reroutes, and alerts. In addition, the CCSD provides textual data about these items, such as reroute advisories and lists of flights that are affected. This data is jointly referred to in this document as TFM data.

The distribution of this data on the CCSD has gone part of the way toward providing common situational awareness to NAS users, but there are two shortcomings.

- The data on the CCSD is “display-only” in that the user can, for example, see an FCA on the CCSD but does not have access in computer-readable format to the data that defines the FCA, e.g., the lat/lon coordinates of the vertices. This means that the user cannot, in an automated way, grab this data and use it in internal applications.
- The data is not available even in display-only mode to the Aircraft Situation Display to Industry (ASDI) vendors, so these vendors are not able to include this data in any of the products that they supply to the aviation community.

To deal with these two shortcomings, the FAA decided to undertake the program called TFM Data to Industry (TFMDI). In short, this program provides the TFM data in a machine-readable format to Industry, which can then incorporate it into their tools. For example, since the data that defines an FEA/FCA (the lat/lon of the vertices, upper and lower altitudes, filtering, and so forth) is provided, then Industry can incorporate FEA/FCA data into their own displays and other tools. This should improve the tools, promote common situational awareness, and perhaps provide this data to a much wider audience.

1.2. Purpose of this Document

The purpose of this document is to provide to a software developer with the technical information needed to obtain and interpret the TFM data:

- It tells how an application would need to be written to access the TFM data. That is, it describes what a program needs to do to obtain the data.
- It tells how the data is formatted. That is, it tells what a program needs to do to interpret the data.

Once a program can obtain and interpret the data, the question remains of what should be done with it. This question is left to the private sector, but some guidelines can be found in an accompanying document that describes the meaning and use of the data; see Section 1.4.

At present, only the data for public reroutes and FEA/FCAs is available, and this version of this document only covers that data. As more types of TFM data become available, this document will be expanded and updates will be issued.

1.3. The TFMDI Servers

To provide redundancy of the data feed, there are two TFMDI servers. Each of these servers is fed by one of the two operational ETMS systems that provide redundancy for ETMS; moreover, these TFMDI servers are on different boxes on different networks with different power supplies so that there is no single point of failure. (If an Industry user has a single communications line to the ETMS hubsite, then this is a single point of failure; an Industry user can, if desired, eliminate this single point of failure by acquiring a second line.)

Industry users will be given two operational IP addresses, one for each of the operational TFMDI servers. The servers that are accessed by these two IP addresses are equivalent; it is not the case that one is primary and one is back-up. Industry users can access either server and will get the same data, apart from minor differences due to timing.

The ETMS goal is that at least one of these TFMDI servers will always be operational; in practice, both will be operational far more than 99 percent of the time. If Volpe needs to install a security patch or an operating systems upgrade, we will take one server down, and only after it is back in service will we take the other server down. If only one server is up, it is the responsibility of the Industry user to find it and fetch data from it. If routine maintenance is being done that requires one server at a time to be taken down, Volpe will not announce this since Industry users are assumed to be able to deal with this situation. Volpe will only announce outages if both servers are expected to be down at the same time; such outages are expected to be extremely rare. If software upgrades are installed, Volpe will announce this so that users can watch for any resulting problems.

1.4. References

“Traffic Flow Management Data to Industry: Overview,” Volpe Center, report no. ETMS-TFMDI-001, ver. 1.0, 10 August 2004. This document explains the rationale for the TFM Data to Industry program and spells out the principles that guide it.

“Traffic Flow Management Data to Industry: Explanation of the Data,” Volpe Center, report no. ETMS-TFMDI-002, ver. 1.0, 10 August 2004. This document explains the data from the operational point of view, as opposed to the ICD, which explains the data from the technical point of view. That is, this document explains how the data is generated and what its significance is for operational personnel.

To download the latest version of this ICD, the other documents listed here, and other documents such as sample data files, go to the Traffic Flow Management Data to Industry web page:

<http://www.fly.faa.gov/ASDI/asdi.html>

2. Overview of the TFM Data

2.1. Introduction

The TFM data information structure is composed of a index file and individual data files. The index file gives the names and creation times of each of the data files; the index file can be thought of as providing pointers to the data files. The general method by which the TFM data is provided to Industry is as follows.

- Volpe has set up a TFM Data to Industry web server. (Actually, there are two independent and redundant web servers.)
- A directory exists on this web server for the TFM data that is to be distributed.
- Whenever there is an update to any type of data, ETMS writes an updated file to the relevant directory.
- There is a directory that contains an index file. This index file shows the latest update time for each type of data.
- An Industry user would use http to fetch any desired files.
- Access to the web server is controlled so that only the authorized users can fetch the data.

It is expected that an Industry user would operate in the following way.

- Once a minute (or at whatever time interval the Industry user desired) a program written by the Industry user would use HTTP to fetch the index file.
- A program written by the Industry user would inspect the index file to see if there had been any updates to the data types that are of interest. Note that the TFMDI system sets the timestamp given in the index file for a data file (which is also the write time of the data file) to *the most recent change to the data in the file*. Thus, if the timestamp given in the index file has not changed, the data in the file has not changed.
- A program written by the Industry user would use HTTP to fetch the desired data files.
- Once the data files had been received, the Industry user could do whatever was desired with them.

2.2. XML File Format Ground Rules

The data files are in a subset of XML, with which the reader is assumed to have some familiarity. The following are the basic rules for the TFM data files.

- The files contain only printable ASCII characters.
- The file format follows XML structural conventions.
- In XML terminology, the files are guaranteed to be "well-formed." This means that every opening tag `<TAG>` has a corresponding closing tag `</TAG>`, opening and closing tag pairs are correctly matched and nested, and consistent capitalization is used.
- The files are guaranteed to be "valid", meaning that the content matches the current documentation (this document).
- Only a simple subset of XML is used. All data is between matching start and end tags:
`<TAG>data</TAG>`
- The files do NOT contain data in any other XML form. For example, composite tags (e.g., `<TAG="data"/>`) and attributes are never used.
- A pair of matching start and end tags, together with the data between them, is known as a *data element* or simply an *element*.

- The file has *structure*, i.e., elements can contain other elements. An element that contains other elements is referred to as a *container*. The container element is considered to be the parent to the elements contained within. Example:

```
<CTR_ROUTE>
  <CTR_SEG>...</CTR_SEG>
</CTR_ROUTE>
```

- Characters that are not between matching start and end tags are to be ignored, and may be used occasionally for comments or to improve readability. Example:

```
<TAG1>
  <TAG2>This is data </TAG2>
  This is a comment.
  <TAG2>This is data</TAG2>
</TAG1>
```

- New-line characters between matching start and end tags are part of the element's data.
- Data elements can be in any order within their parent element's tag pair (if there is one) or within the file.

2.3. How to Read the Tables that Describe the Data

For each data type there is a table that describes the data file format. The developer should be aware of a couple of features of the data file format.

- The first line of every file is the standard "<?XML . . .>" tag, identifying the XML version number. There is no corresponding end tag.
- The table is organized according to the hierarchy of the data. That is, each tag that defines a container is followed by the elements that belong in that container. This nesting is explicitly described in the **Parent** column of each table, defined below.

The table for each data type has the following columns.

- **Tag** – The name of the tag as it appears in the file.
- **Lv** – The nesting depth of the element in the file structure. Example: VERSION is at level 1, the highest level. ID is at level 2, because it appears within the CTR_HEAD tag pair.
- **#** – The number of these elements that can be present within the parent. For example, there can be only one INCLUDE in each CTR_SEG. However, since there can be many CTR_SEG elements in a CTR_ROUTE, there can be many INCLUDE elements in a file.
- **Rq?** – Describes whether the element is required, optional etc.:
 - Y** Required.
 - N** Optional.
 - *Y*** This element must be present if its parent is present.
 - %** Used to flag elements that may or may not be defined within a parent container. If the parent is present, at least one of those elements will be defined.
- **Parent** – The container element inside which this element appears. For example, in a reroute data file CTR_HEAD is the parent of ID.
- **Description** – Describes the element. If appropriate, all the allowed data values are listed and CAPITALIZED.

3. TFMDI Index File

The most current index file is always named "TFMDIndex". The contents are described using the following example. Only reroute and FEA/FCA files are shown in this example.

```
<?xml version="2.1" standalone="yes" ?>
<TFMDI_INDEX_FILE>
  <VERSION>2.1</VERSION>
  <TIMESTAMP>20040330133815</TIMESTAMP>
  <REROUTE_FILES>
    <REROUTE_FILE>
      <REROUTE_NAME>BNA</REROUTE_NAME>
      <FILE_NAME>RR.BNA.Dat.gz</FILE_NAME>
      <FILE_TIMESTAMP>20040330121546</FILE_TIMESTAMP>
    </REROUTE_FILE>
    <REROUTE_FILE>
      <REROUTE_NAME>FCA002</REROUTE_NAME>
      <FILE_NAME>RR.FCA002.Dat.gz</FILE_NAME>
      <FILE_TIMESTAMP>20040330123641</FILE_TIMESTAMP>
    </REROUTE_FILE>
  </REROUTE_FILES>
  <FCA_FILES>
    <FCA_FILE>
      <FCA_NAME>FCA001</FCA_NAME>
      <FILE_NAME>FCA.FCA001.Dat.gz</FILE_NAME>
      <FILE_TIMESTAMP>20040330121546</FILE_TIMESTAMP>
      <AFP_STATUS>TRUE</AFP_STATUS>
      <SECONDARY_FILTER>
        <SECONDARY_FILTER_NAME>ORD_DEP</SECONDARY_FILTER_NAME>
      </SECONDARY_FILTER>
      <SECONDARY_FILTER_FILE_NAME>FCA.FCA001.ORD_DEP.Filter.gz</SECONDARY_FILTER_FILE_NAME>
      <SECONDARY_FILTER_FILE_TIMESTAMP>20040330121546</SECONDARY_FILTER_FILE_TIMESTAMP>
    </FCA_FILE>
  </FCA_FILES>
</TFMDI_INDEX_FILE>
```

At present, the only data that is provided is data on public reroutes, FCAs, and public FEAs. (NOTE: Public FEAs and FCAs are treated interchangeably in this document and in the data feed). The index file gives the names, data file names, and data file creation timestamps for all active public reroutes and FEAs/FCAs. The detailed information for each reroute is contained in a separate data file; that is, there is one data file for each reroute. The primary definition of each FEA/FCA is also in a single data file; however, an FEA/FCA may also have secondary filters associated with it. If so, each secondary filter is stored in a separate data file.

Note that the name of an FEA/FCA can be modified, and at present this cannot be detected using only data from the index file. To detect an FEA/FCA whose name has changed, refer to the FCA_ID value that can be obtained from the FEA/FCA definition file, as detailed in Table 3. The FCA_ID value is assigned by ETMS when an FEA/FCA is first created, is unique, and never changes. So, if an FEA/FCA with a new name appears in the index file, but the FEA/FCA has the same FCA_ID value as an FEA/FCA that was previously in the index file, then this means this is the same FEA/FCA with a modified name.

The files in the index file have the '.gz' extension to indicate that they have been zipped. Industry users will want to unzip these files after fetching them. The unzipped files are also available in the same directory; these unzipped versions have the same filenames except that the extension '.gz' is omitted. A user could fetch these unzipped versions if necessary; however, we strongly recommend that the zipped files be fetched. Some of the TFMDI files will be quite large, and unzipped XML is very wasteful of bandwidth.

The URL of the Index file has the format <http://IPAddress/TFMDI/TFMDIndex> . Industry users will be given the test and operational IP addresses upon request. A program can fetch this file using an HTTP GET command.

The data elements in the TFMDI index file are described in the following table.

Table 1. TFMDI Index File Data Elements

Tag	L v	#	Rq?	Parent	Example	Description
<?xml version="1.0" standalone="yes" ?>	1	1	Y	None	<?xml version="1.0" standalone="yes"?>	First line, to make the file XML compliant.
TFMDI_INDEX_FILE	1	1	Y	None	<TFMDI_INDEX_FILE> everything </TFMDI_INDEX_FILE>	Container for the entire index file.
VERSION	2	1	Y	TFMDI_INDEX_FILE	<VERSION>2.1</VERSION>	Index file format version number. The version at the time of this document is 2.1.
TIMESTAMP	2	1	Y	TFMDI_INDEX_FILE	<TIMESTAMP>data</TIMESTAMP>	Date and time ¹ the index file was last updated.
REROUTE_FILES	2	1	N	TFMDI_INDEX_FILE	<REROUTE_FILES>data</REROUTE_FILES>	Container for the reroute file definitions.
REROUTE_FILE	3	1+	*Y*	REROUTE_FILES	<REROUTE_FILE>data</REROUTE_FILE>	Container for an individual reroute file definition.
REROUTE_NAME	4	1	*Y*	REROUTE_FILE	<REROUTE_NAME>data</REROUTE_NAME>	The name of the reroute as defined by the traffic management specialist.
FILE_NAME	4	1	*Y*	REROUTE_FILE	<FILE_NAME>data</FILE_NAME>	The name of the zipped version of the reroute data file.
FILE_TIMESTAMP	4	1	*Y*	REROUTE_FILE	<FILE_TIMESTAMP>data</FILE_TIMESTAMP>	Date and time ¹ the data file was last updated.
FCA_FILES	2	1	N	TFMDI_INDEX_FILE	<FCA_FILES>data</FCA_FILES>	Container for the FEA/FCA file definitions.
FCA_FILE	3	1+	*Y*	FCA_FILES	<FCA_FILE>data</FCA_FILE>	Container for an individual FEA/FCA file definition.
FCA_NAME	4	1	*Y*	FCA_FILE	<FCA_NAME>data</FCA_NAME>	The name of the FEA/FCA as defined by the traffic management specialist.
FILE_NAME	4	1	*Y*	FCA_FILE	<FILE_NAME>data</FILE_NAME>	The name of the FEA/FCA data file.
FILE_TIMESTAMP	4	1	*Y*	FCA_FILE	<FILE_TIMESTAMP>data</FILE_TIMESTAMP>	Date and time ¹ the data file was last updated.
AFP_STATUS	4	1	*Y*	FCA_FILE	<AFP_STATUS>TRUE</AFP_STATUS>	TRUE if there is an AFP associated with this FCA, otherwise FALSE.
SECONDARY_FILTER	4	1+	N	FCA_FILE	<SECONDARY_FILTER>data</SECONDARY_FILTER>	Container for a secondary filter definition associated with the FEA/FCA. There can be none to many secondary filters.
SECONDARY_FILTER_NAME	5	1	*Y*	SECONDARY_FILTER	<SECONDARY_FILTER_NAME>data</SECONDARY_FILTER_NAME>	The name of the secondary filter as defined by the traffic management specialist.
SECONDARY_FILTER_FILE_NAME	5	1	*Y*	SECONDARY_FILTER	<SECONDARY_FILTER_FILE_NAME>data</SECONDARY_FILTER_FILE_NAME>	The name of the zipped version of the secondary filter data file.
SECONDARY_FILTER_FILE_TIMESTAMP	5	1	*Y*	FCA_FILE	<SECONDARY_FILTER_FILE_TIMESTAMP>data</SECONDARY_FILTER_FILE_TIMESTAMP>	Date and time ¹ the secondary filter data file was last updated.

1. Dates and times are in 14-digit date and time format: year (4 digits) month (2) day (2) hour (2) minute (2) and second (2).

4. Public Reroutes and Initial Reroute Flight Lists

Reroute data file names are of the form `RR.reroute_name.Dat` (example: `RR.BOS_ORD.Dat`).

Reroute data files contain all the data associated with the reroute definition, including the initial flight list if one was generated. Sample reroute data files can be found on the TFM Data to Industry web site referred to in Section 1.3.

The URL of a reroute data file is <http://IPAddress/TFMDI/filename> . Industry users will be given the test and operational IP addresses upon request. The filename comes from the Index file. A program can fetch a reroute data file using an HTTP GET command.

The following table describes the structure of the files, including all defined tags and their associated data.

Table 2. Reroute Data File Data Elements

Tag	Lv	#	Rq?	Parent	Example	Notes
<?xml version="1.0" standalone="yes" ?>	1	1	Y	None	<?xml version="1.0" standalone="yes"?>	First line, to make the file XML compliant.
REROUTE	1	1	Y	None	<REROUTE>everything</REROUTE>	Container for the entire reroute data file.
INDEX_INFO	2	1	Y	REROUTE	<INDEX_INFO>rr.vntsc..ux24.20030910101010, BOS ORW, PUBLIC, 20031010101010,vntd,ux20</INDEX_INFO>	Comma-delimited string with reroute_id, name, domain, update_time, update_site, update_wkstn. Used by legacy code. Up to 128 characters. Can be ignored, as all this data is contained in other individual elements.
VERSION	2	1	Y	REROUTE	<VERSION>8.3</VERSION>	File version -- this is 8.3.
CTR_HEAD	2	1	Y	REROUTE	<CTR_HEAD>data</CTR_HEAD>	Container for the reroute header data.
ID	3	1	Y	CTR_HEAD	<ID>rr.vntscf.ux24.20021107095003</ID>	System-generated reroute identifier. Up to 64 characters.
NAME	3	1	Y	CTR_HEAD	<NAME>TEST_BOS_ORW</NAME>	Reroute name. Up to 64 characters. Beginning in ETMS 8.2, name of an FCA-based reroute can have the form "<FCA Name>:<suffix>". For example, FCA001:EWR_FLOW.
DOMAIN	3	1	Y	CTR_HEAD	<DOMAIN>PUBLIC</DOMAIN>	Generally, possible values are PUBLIC, LOCAL, or PRIVATE, but only PUBLIC reroutes are available on TFMDI.
LASTUPDATE	3	1	Y	CTR_HEAD	<LASTUPDATE>20021107101623</LASTUPDATE>	Date and time reroute was last updated. ⁵
ORIGINAL_CREATETIME	3	1	Y	CTR_HEAD	<ORIGINAL_CREATETIME >20021107095003 <ORIGINAL_CREATETIME >	Date and time the reroute was first created. yyymmddhhmmss
SITE	3	1	Y	CTR_HEAD	<SITE>vntscf</SITE>	Site where reroute was created. Up to 8 characters.
WKSTN	3	1	Y	CTR_HEAD	<WKSTN>wkstn01</WKSTN>	Workstation where reroute was created. Up to 8 characters.
STATUS	3	1	Y	CTR_HEAD	<STATUS>ACTIVE</STATUS>	One of ACTIVE, PLANNED.
SHOW	3	1	Y	CTR_HEAD	<SHOW>1111</SHOW>	Four digits (each 0 or 1) specifying whether to draw fixes, nav aids, reroute label, and/or the route by default.
COLOR_ID	3	1	Y	CTR_HEAD	<COLOR_ID>5</COLOR_ID>	Color index, value 0 – 35. See Appendix A.
REROUTE_TIMETYPE	3	1	Y	CTR_HEAD	<REROUTE_TIMETYPE>ETA</REROUTE_TIMETYPE>	One of ETA, ETD, FCA, indicating the method used to select flights involved in the reroute.

<i>Tag</i>	<i>Lv</i>	<i>#</i>	<i>Rq?</i>	<i>Parent</i>	<i>Example</i>	<i>Notes</i>
REROUTE_STARTTIME	3	1	Y	CTR_HEAD	<REROUTE_STARTTIME>20021107112004</REROUTE_STARTTIME>	Reroute start time. ⁵
REROUTE_ENDTIME	3	1	N	CTR_HEAD	<REROUTE_ENDTIME>20021107122004</REROUTE_ENDTIME>	Reroute end time. ⁵
FCA_NAME	3	1	N	CTR_HEAD	<FCA_NAME>FCA001</FCA_NAME>	Name of FCA if reroute is based on FCA. Present if REROUTE_TIME_TYPE is FCA. Up to 30 characters.
FCA_STARTTIME	3	1	N	CTR_HEAD	<FCA_STARTTIME>20021107112004</FCA_STARTTIME>	Present if REROUTE_TIME_TYPE is FCA. Currently same as REROUTE_STARTTIME. ⁵
FCA_ENDTIME	3	1	N	CTR_HEAD	<FCA_ENDTIME>20021107112004</FCA_ENDTIME>	Present if REROUTE_TIME_TYPE is FCA. Currently same as REROUTE_ENDTIME. ⁵
AIRBORNE	3	1	Y	CTR_HEAD	<AIRBORNE>ALL_FLIGHTS</AIRBORNE>	One of AIRBORNE, NOT_AIRBORNE, ALL_FLIGHTS, indicating which flights are included in the reroute by flight status.
CTR_ROUTE	2	1	Y	REROUTE	<CTR_ROUTE> other data </CTR_ROUTE>	Container for one or more route segments.
CTR_SEG	3	1+	Y	CTR_ROUTE	<CTR_SEG>data</CTR_SEG>	Container for route segment data.
INCLUDE	4	1	*Y*	CTR_SEG	<INCLUDE>INCLUDE</INCLUDE>	One of INCLUDE, EXCLUDE. Indicates whether this segment is included in the reroute or excluded.
ORIGIN	4	1	*Y*	CTR_SEG	<ORIGIN>ZHU BOS</ORIGIN>	List of origin airports/centers, separated by spaces. Up to 128 characters.
DESTINATION	4	1	see note	CTR_SEG	<DESTINATION>BOS JFK</DESTINATION>	List of destination airports/centers, separated by spaces. Up to 128 characters.
FLTR_SUMMARY	4	1	N	CTR_SEG	<FLTR_SUMMARY>xxx</FLTR_SUMMARY>	System-generated text summarizing any filter conditions defined for this reroute segment. Up to 100 characters.
FULL_ROUTE	4	1	*Y*	CTR_SEG	<FULL_ROUTE>FULL</FULL_ROUTE>	One of FULL, ORIG, DEST. ^{1,2}
ROUTE_STR	4	1	*Y*	CTR_SEG	<ROUTE_STR>BNA J42 BKW</ROUTE_STR>	Route string ³ , up to 256 characters. Note that route elements are separated by spaces.
WAYPOINT	4	2+	Y	CTR_SEG	<WAYPOINT>3871,7824,BELMA,F</WAYPOINT>	Gives <i>lat</i> , <i>long</i> , <i>name</i> , and <i>type</i> (separated by commas, up to 32 characters total) for each waypoint or center entry point on the route segment. Each point is in a separate tag. Type is one of F=fix, N=navaid, W=waypoint, C=center entry. For an un-named waypoint, <i>name</i> is "?".
ROUTE_TYPE	4	1	Y	CTR_SEG	<ROUTE_TYPE>CDR RTE</ROUTE_TYPE>	User-specified route-segment type. One of NONE, CDR RTE, RERTE, UNKN RTE, UPT RTE

<i>Tag</i>	<i>Lv</i>	<i>#</i>	<i>Rq?</i>	<i>Parent</i>	<i>Example</i>	<i>Notes</i>
SEG_SOURCENAME	4	1	N	CTR_SEG	<SEG_SOURCENAME>BOS_ORW</SEG_SOURCENAME>	If this route segment was retrieved from a playbook, this is the name of the play. Up to 64 characters.
SEG_REMARKS	4	1	N	CTR_SEG	<SEG_REMARKS>Over water</SEG_REMARKS>	User-entered remark. Up to 512 characters.
SEG_ID	4	1	Y	CTR_SEG	<SEG_ID>D0</SEG_ID>	The route grid row in which this segment appears. Rows in the upper (origin) grid begin with "O", and rows in the lower (destination) grid begin with "D". Within each grid, rows are sequentially numbered beginning with 0. That is, Origin segments rows are O0, O1 etc., while Destination segment rows are D0, D1 etc. Up to 6 characters.
CTR_FLTR	4	0-1	N	CTR_SEG	<CTR_FLTR>data</CTR_FLTR>	Container for a set of filter conditions that constrain the flights to which this reroute segment is applied. Only FULL route or ORIGIN segments can have filters. ⁴
NOT_FROM	5	1	%	CTR_FLTR	<NOT_FROM>DCA IAD BWI</NOT_FROM>	List of excluded departure airports separated by spaces. Up to 128 characters.
NOT_TO	5	1	%	CTR_FLTR	<NOT_TO>DCA IAD BWI</NOT_TO>	List of excluded destination airports separated by spaces. Allowed only if the DESTINATION for this segment contains a center or UNKN. Up to 128 characters.
THRU	5	1	%	CTR_FLTR	<THRU>ZDC ZATL</THRU>	List of centers separated by spaces. If present, flight must traverse at least one of the listed centers. Up to 128 characters.
AIRWAY	5	1	%	CTR_FLTR	<AIRWAY>J78</AIRWAY>	List of airway names separated by spaces. If present, flight must fly on at least one of the listed airways. Up to 128 characters.
USE_FIX	5	1	%	CTR_FLTR	<USE_FIX>WHITE</USE_FIX>	List of fix names separated by spaces. If present, flight must cross at least one of the listed fixes. Up to 128 characters.
AC_TYPE	5	1	%	CTR_FLTR	<AC_TYPE>MD80 B752</AC_TYPE>	List of aircraft types separated by spaces. If present, flight must be one of the listed types. Up to 64 characters.
FLIGHTLVLOW	5	1	%	CTR_FLTR	<FLIGHTLVLOW>450</FLIGHTLVLOW>	Minimum flight level (cruising altitude), 0-600. Default = 0.
FLIGHTLVHIGH	5	1	%	CTR_FLTR	<FLIGHTLVHIGH>550</FLIGHTLVHIGH>	Maximum flight level (cruising altitude), 0-600. Default = 600.

Tag	Lv	#	Rq?	Parent	Example	Notes
AIR_CAT	5	1	%	CTR_FLTR	<AIR_CAT>J P T</AIR_CAT>	Aircraft category. One or more of J, P, T, separated by spaces. If present, flight must be one of the listed categories.
WEIGHT_CLASS	5	1	%	CTR_FLTR	<WEIGHT_CLASS>L H</WEIGHT_CLASS>	Aircraft weight class. One or more of L, H, S, separated by spaces. If present, flight must be one of the listed weight classes.
USER_CAT	5	1	%	CTG_FLTR	<USER_CAT>T F C</USER_CAT>	User type. One or more of T, F, C, G, M, separated by spaces. G=GA T=AirTaxi F=Cargo C=Commercial M=Military. If present, flight must be one of the listed user categories.
RVSM	5	1	%	CTR_FLTR	<RVSM>COMPLIANT</RVSM>	One of ALL, COMPLIANT, NON_COMPLIANT. Indicates which flights to include by their RVSM status. Default is ALL. Note there is a CTR_FLIGHT tag with the same name.
FLTR_TIMETYPE	5	1	%	CTR_FLTR	<FLTR_TIMETYPE>DEP</FLTR_TIMETYPE>	One of DEP, ARR, indicating that flights are filtered by departure time or arrival time. FLTR_TIMETYPE, FLTR_STARTTIME and FLTR_ENDTIME tags are all present or all absent.
FLTR_STARTTIME	5	1	%	CTR_FLTR	<FLTR_STARTTIME>20021129124500</FLTR_STARTTIME>	Filter start date and time. ⁵ If present, only flights that depart or arrive (depending on FLTR_TIMETYPE) within the specified start and end time are included.
FLTR_ENDTIME	5	1	%	CTR_FLTR	<FLTR_ENDTIME>20021129134500</FLTR_ENDTIME>	Filter end date and time. ⁵ If present, only flights that depart or arrive (depending on FLTR_TIMETYPE) within the specified start and end time are included.
CTR_FLIGHT_LIST	2	0-1	N	None	<CTR_FLIGHT_LIST>data</CTR_FLIGHT_LIST>	Container for list of flights involved in the reroute. Present only if flight list was generated when reroute was created.
FLIGHT_LIST_LASTUPDATE	3	1	Y	CTR_FLIGHT_LIST	<FLIGHT_LIST_LASTUPDATE>20021129134500</FLIGHT_LIST_LASTUPDATE>	Date and time of last flight list update. ⁵
CTR_FLIGHT	3	1+	*Y*	CTR_FLIGHT_LIST	<CTR_FLIGHT>data</CTR_FLIGHT>	Container for a flight. There is at least one if CTR_FLIGHT_LIST is given.
ACID	3	1	*Y*	CTR_FLIGHT	<ACID>COM218</ACID>	Aircraft ID.

Tag	Lv	#	Rq?	Parent	Example	Notes
RVSM	3	1	N	CTR_FLIGHT	<RVSM>NC</RVSM>	One of NC, -. NC = RVSM non-conformant, - = RVSM conformant. Note there is a CTR_FILTER tag with the same name but a different value set.
E_TYPE	3	1	N	CTR_FLIGHT	<E_TYPE>B738</E_TYPE>	Equipment type.
DCENTR	3	1	N	CTR_FLIGHT	<DCENTR>ZNY</DCENTR>	Departure center.
ACENTR	3	1	N	CTR_FLIGHT	<ACENTR>ZBW</ACENTR>	Arrival Center.
ORIG	3	1	N	CTR_FLIGHT	<ORIG>PHL</ORIG>	Origin airport.
DEST	3	1	N	CTR_FLIGHT	<DEST>SFO</DEST>	Destination airport.
ETD	3	1	N	CTR_FLIGHT	<ETD>P2140</ETD>	Estimated time of departure. ⁷
ETA	3	1	N	CTR_FLIGHT	<ETA>E2206</ETA>	Estimated time of arrival. ⁸
PGTD	3	1	N	CTR_FLIGHT	<PGTD>281925</PGTD>	Proposed gate time of departure. ⁶
ALT	3	1	N	CTR_FLIGHT	<ALT>360</ALT>	Requested flight level 000 - 600.
IGTD	3	1	N	CTR_FLIGHT	<IGTD>282050</IGTD>	Initial gate time of departure. ⁶
AC_CAT	3	1	N	CTR_FLIGHT	<AC_CAT>J</AC_CAT>	Aircraft category. One of J, P, T.
SPEED	3	1	N	CTR_FLIGHT	<SPEED>485/SPEED>-	Speed in knots.
CTR_ASSIGNED_RTE	3	1+	N	CTR_FLIGHT	<CTR_ASSIGNED_RTE> other data </CTR_ASSIGNED_RTE>	Container for assigned route.
ASSIGNED_RTE	4	1	Y	CTR_ASSIGNED_RTE	<ASSIGNED_RTE> ROD MIE SHM TARNE1</ASSIGNED_RTE>	Assigned route of flight. Up to 1024 characters. Note that route elements are separated by spaces.
ASSIGNED_RTE_TYPE	4	1	Y	CTR_ASSIGNED_RTE	<ASSIGNED_RTE_TYPE>CDR RTE </ASSIGNED_RTE_TYPE>	Route type of the assigned route. One of NONE, CDR RTE, RERTE, UNKN RTE, UPT RTE.
INCLUDE_RTE	4	1	Y	CTR_ASSIGNED_RTE	< INCLUDE_RTE>INCLUDE</ INCLUDE_RTE>	One of INCLUDE, EXCLUDE. Indicates whether this flight and this assigned route are included in the reroute or excluded.
DBKEY	3	1	N	CTR_FLIGHT	<DBKEY>20030305123</DBKEY>	Oracle key of flight in database. Used internally to match assigned route and flights. Up to 32 characters.
CTR_ADV_LIST	2	0-1	N	None	<CTR_ADV_LIST> data </CTR_ADV_LIST>	Container for Advisory data.
ADV_NUMBER	3	1	*Y*	CTR_ADV_LIST	<ADV_NUMBER> 005</ADV_NUMBER>	Advisory number. Up to 6 characters. “---“ if no advisory number.
ADV_TYPE	3	1	*Y*	CTR_ADV_LIST	<ADV_TYPE>MESSAGE</ADV_TYPE>	One of MESSAGE, ADVISORY. Indicates a message was sent or an advisory was issued.
ADV_TEXT	3	1	N	CTR_ADV_LIST	<ADV_TEXT>Advisory text</ADV_TEXT>	Advisory text sent by e-mail. Unlimited length.
CTR_ADV	3	1-10	*Y*	CTR_ADV_LIST	<CTR_ADV>data</CTR_ADV>	Container for an advisory field. ⁹

<i>Tag</i>	<i>Lv</i>	<i>#</i>	<i>Rq?</i>	<i>Parent</i>	<i>Example</i>	<i>Notes</i>
ADV_NAME	4	1	*Y*	CTR_ADV	<ADV_NAME>REASON</ADV_NAME>	Advisory field name: one of CATEGORY, ACTION, CONSTRAINED AREA, REASON, INCLUDE TRAFFIC, FACILITIES INCLUDED, PROBABILITY OF EXTENSION, REMARKS, ASSOCIATED RESTRICTIONS, MODIFICATIONS.
ADV_DATA	4	1	*Y*	CTR_ADV	<ADV_DATA>Snowing and blowing in BOS</ADV_DATA>	Advisory field value. Free form, up to 512 characters.

NOTES:

1. A reroute segment can be a full end-to-end route (FULL), in which case the segment container defines everything you need to know about that segment, or a partial segment. A partial segment can be an origin segment (ORIG) or a destination segment (DEST).
2. An ORIG segment can be matched with one or more DEST segments by comparing the last fix in the route string (ROUTE_STR) of the ORIG segment with the first fix in the route string (ROUTE_STR) of the DEST segment. When a matching ORIG/DEST pair is found, a composite route should be created by combining the two routes. This composite route should be applied only to flights that depart from the ORIGIN of the ORIG segment and arrive at the DESTINATION of the DEST segment. For a fuller discussion of full and split routes, see the Explanation of the Data document referred to in Section 1.4.
3. If the ROUTE_TYPE is NONE, the route segment is a specific route of flight defined by a sequence of fixes and airways. Such a reroute segment must contain waypoints. If the ROUTE_TYPE is any other value, the route string is free-form text and the reroute segment will not contain waypoints.
4. Because ORIG and DEST segments must be paired up to be used, it would create ambiguities or conflicts if both segments contained filters. Therefore, only ORIG segments (and FULL routes) have filters.
5. A 14-digit date and time is composed of the year (4 digits) month (2) day (2) hour (2) minute (2) and second (2)..
6. A 6-digit date and time is composed of the day (2 digits) hour (2) and minute (2).
7. ETD consists of a prefix followed by a 4-digit time. The possible values of the prefix are:
 - a. S – Scheduled. Flight data is based on OAG data and/or CDM messages.
 - b. R – Reroute. Indicates that flight has been modeled based on its assigned reroute rather than its historical route.
 - c. N – Early Intent. ETMS has received an Early Intent message for the flight.
 - d. P – Proposed. Flight data is based on flight plan.
 - e. T – Taxi. ETMS has received a message indicating the flight has pushed off the gate.

- f. A – Flight is active and the ETD is an actual departure time.
 - g. E – Flight is active and the ETD is an estimated departure time.
8. ETA consists of a prefix followed by a 4-digit time. The possible values of the prefix are:
- a. E –The ETA is an estimated arrival time.
 - b. A – The ETA is an actual arrival time; that is, the flight has landed.
9. An advisory is composed of many fields. Each field is saved in a CTR_ADV. Each CTR_ADV contains two parts. The ADV_NAME is the label for that field. The ADV_DATA is the content of that field.

5. Public Flow Evaluation Areas/Flow Constrained Areas

FEA/FCA data file names are of the form `FCA.FCA_name.Dat` or `FEA.FEA_name.Dat`. Examples are `FCA.FCA001.Dat` and `FEA.DC_METROS.Dat`. FEA/FCA definition files contain the primary data associated with the FEA/FCA definition. Unlike reroute data files, there are no flight lists embedded in the FEA/FCA definitions; a discussion of the data on flights that are in an FEA/FCA will be provided later (in Section 6). If there are secondary filters associated with an FEA/FCA, each secondary filter is defined in a separate file.

The primary filter is an essential part of the definition of an FEA/FCA and is included in the FEA/FCA data file; the primary filter is used to decide which flights are in the FEA/FCA. The secondary filters, in effect, partition the flights in the FEA/FCA into subsets; this is useful, for example, if one wants to display different flows in different colors. Each secondary filter defines one subset and is in a separate file; the Index file identifies these files.

The URL of an FEA/FCA file is <http://IPAddress/TFMDI/filename>. Industry users will be given the test and operational IP addresses upon request. The filename comes from the Index file. A program can fetch an FEA/FCA file using an HTTP GET command.

The following tables describe the structure of the FEA/FCA files, including all defined tags and their associated data. The FEA/FCA primary definition data file format is presented in Table 3 and the FEA/FCA secondary filter data file format is presented in Table 4. Each of these definitions can include a filter definition. To reduce redundancy in this document, the filter definition format is given separately, in Table 5.

Table 3. FEA/FCA Definition File Data Elements

Tag	Lv	#	Rq?	Parent	Example	Description
<xml version="1.0" standalone="yes"?>	1	1	Y	None	<xml version="1.0" standalone="yes"?>	First line, to make the file XML compliant.
FCA	1	1	Y	None	<FCA> everything</FCA>	Container for the entire FEA/FCA definition
INDEX_INFO	2	1	Y	FCA	<INDEX_INFO>fca.cdmb.lxpc103.20040713161706%TEST 1%PUBLIC%20040713161739%lxpc103% %cdmb%lxpc103%cdmb%NONE%FEA%25%20040713161 500%20040713211500%#FSM%</INDEX_INFO>	ETMS legacy data, up to 128 characters. Can be ignored, as all this data is contained in other better-defined elements. (An FSM indicator has been added to the end of this field in ETMS 8.2, but it can also be ignored here. Its value is #FSM if the FEA/FCA is FSM eligible, and otherwise contains a single blank character. The equivalent information is contained in the new FSM_ELIGIBLE tag defined below.)
FCA_ID	2	1	Y	FCA	<FCA_ID>fca.cdmb.lxpc103.20040713161706</FCA_ID>	System-generated unique identifier of FEA/FCA
NAME	2	1	Y	FCA	<NAME>FCA007</NAME>	Name of FEA/FCA as created by the traffic management specialist.
DOMAIN	2	1	Y	FCA	<DOMAIN>PUBLIC</DOMAIN>	Generally, possible values are PUBLIC, SHARED, LOCAL, or PRIVATE, but only public FEA/FCAs are available on TFMDI.
LASTUPDATE	2	1	Y	FCA	<LASTUPDATE>20021107101623</LASTUPDATE>	Date and time ¹ the FEA/FCA was last updated.
UP_WKSTN	2	1	Y	FCA	<UP_WKSTN>lxpc103</UP_WKSTN>	ETMS workstation from which the FEA/FCA was last updated
UP_SITE	2	1	Y	FCA	<UP_SITE>cdmb</UP_SITE>	ETMS site from which the FEA/FCA was last updated
CR_WKSTN	2	1	Y	FCA	<CR_WKSTN>lxpc103</CR_WKSTN>	ETMS workstation from which the FEA/FCA was created
CR_SITE	2	1	Y	FCA	<CR_SITE>cdmb</CR_SITE>	ETMS site from which the FEA/FCA was created.
SHARE_SITES	2	1	N	FCA	<SHARE_SITES>ccsd</SHARE_SITES>	Space-separated list of ETMS sites with which the FEA/FCA was shared.
REASON	2	1	Y	FCA	<REASON>NONE</REASON>	Reason for the FEA/FCA; one of NONE, WEATHER, VOLUME, RUNWAY, EQUIPMENT, OTHER.
TYPE	2	1	Y	FCA	<TYPE>FEA</TYPE>	One of FEA, FCA.

Tag	Lv	#	Rq?	Parent	Example	Description
COLOR_ID	2	1	Y	FCA	<COLOR_ID>17</COLOR_ID>	Color index indicating the color that ETMS is using to draw the FEA/FCA polygon and residual flights (residuals are flights that are in the FEA/FCA but do not match any of the secondary filters). 0 – 35. (NOTE: See Appendix A for a definition of the COLOR_ID values.)
START	2	1	Y	FCA	<START>20040713161500</START>	Date and time ¹ for starting the FEA/FCA.
END	2	1	Y	FCA	<END>20040713211500</END>	Date and time ¹ for ending the FEA/FCA .
EXTENDED	2	1	Y	FCA	<EXTENDED>TRUE</EXTENDED>	Flag to indicate whether the FEA is an Extended FEA. Content is “TRUE” or “FALSE”.
LOOK_AHEAD	2	1	Y	FCA	<LOOK_AHEAD>12</LOOK_AHEAD>	Time range in hours from current time over which ETMS will determine the FCA flights. Content is 1-15 hours.
FSM_ELIGIBLE	2	1	Y	FCA	<FSM_ELIGIBLE>TRUE</FSM_ELIGIBLE>	Flag to indicate whether the FEA/FCA is eligible to be monitored by FSM. Content is “TRUE” or “FALSE”.
POLYGON	2	1	%	FCA	<POLYGON>data</POLYGON>	Container for data elements for a custom-polygon FEA/FCA. ³
LINE	2	1	%	FCA	<LINE>data</LINE>	Container for data elements for a multi-segment line FEA/FCA. ⁴
CIRCLE	2	1	%	FCA	<CIRCLE>data</CIRCLE>	Container for data elements for a circle FEA/FCA. ⁵
NAS	2	1	%	FCA	<NAS>data</NAS>	Container for data elements for a NAS-element FEA/FCA. ⁶
CEILING	3	1	*Y*	CIRCLE or POLYGON or LINE	<CEILING>600</CEILING>	Upper altitude of FEA/FCA (100s feet). 000 to 600.
FLOOR	3	1	*Y*	CIRCLE or POLYGON or LINE	<FLOOR>240</FLOOR>	Lower altitude of FEA/FCA (100s feet). 000 to 600.

Tag	Lv	#	Rq?	Parent	Example	Description
POINTS	3	1	*Y*	CIRCLE or POLYGON or LINE	<POINTS>4285,7633 4285,7597 4203,7595 4202,7627</POINTS>	The points of the polygon that defines the shape of the FEA/FCA. Each point is a comma-delimited latitude and longitude in hundredths of degrees (i.e., degrees times 100). Latitudes are assumed to be North and longitudes West. Points are separated by spaces. Up to 60 points.
DIRECTION	3	1	*Y*	POLYGON or LINE	<DIRECTION>270</DIRECTION>	Direction of movement, for a moving FEA/FCA. (0-360 degrees).
SPEED	3	1	*Y*	POLYGON or LINE	<SPEED>18</SPEED>	Speed of movement (knots), for a moving FEA/FCA.
DRAWING	3	1	*Y*	POLYGON or LINE	<DRAWING>>false</DRAWING>	For moving FEA/FCAs: = "false" if the points specified for the custom polygon or line define its location at the time it was drawn, = "true" if the points define its location at the start time."
CENTER	3	1	*Y*	CIRCLE	<CENTER>4225N/07610W</CENTER>	Center point of circle. Latitude and longitude are separated by a slash. Latitude is two-digits of degrees, two-digits of minutes, one letter of direction (N or S). Longitude is three-digits of degrees, two-digits of minutes, one letter of direction (E or W).
RADIUS	3	1	*Y*	CIRCLE	<RADIUS>30</RADIUS>	Radius of circle (nautical miles).
AP_ARR	3	1	%	NAS	<AP_ARR>BOS</AP_ARR>	Indicates that NAS-element type is arrival airport. Content is airport name. No longer used as of ETMS 8.2.
AP_DEP	3	1	%	NAS	<AP_ARR>BOS</AP_ARR>	Indicates that NAS-element type is departure airport. Content is airport name. No longer used as of ETMS 8.2.
AIRPORT	3	1	%	NAS	<AIRPORT>BOS</AIRPORT>	Indicates that NAS-element category is an airport. Content is airport name.
SECTOR	3	1	%	NAS	<SECTOR>ZBW24</SECTOR>	Indicates that NAS-element type is sector in its dynamic state. Content is sector name.
BASE_SECTOR	3	1	%	NAS	<BASE_SECTOR>ZBW24</BASE_SECTOR>	Indicates that NAS-element category is a sector in its baseline state. Content is sector name.
TRACON	3	1	%	NAS	<TRACON>ZMPMSP</TRACON>	Indicates that NAS-element type is Tracon. Content is tracon name.

<i>Tag</i>	<i>Lv</i>	<i>#</i>	<i>Rq?</i>	<i>Parent</i>	<i>Example</i>	<i>Description</i>
ARTCC	3	1	%	NAS	<ARTCC>ZMP</ARTCC>	Indicates that NAS-element category is ARTCC. Content is ARTCC name.
FIX	3	1	%	NAS	<FIX>VUZ</VUZ>	Indicates that NAS-element type is fix. Content is fix name.
SUA	3	1	%	NAS	<SUA>MOA-DEEPWOODS</SUA>	Indicates that NAS-element type is special use area. Content is SUA name.
FCA_DISPLAY_PREFERENCES	2	1	N	FCA	<FCA_DISPLAY_PREFERENCES>data</FCA_DISPLAY_PREFERENCES>	Container for the preferences definitions
FCA_TIMELINE	3	1	N	FCA_DISPLAY_PREFERENCES	<FCA_TIMELINE>data</FCA_TIMELINE>	Container for FCA Timeline preferences
VISIBLE_TIME_RANGE	4	1	N	FCA_TIMELINE	<VISIBLE_TIME_RANGE>300</VISIBLE_TIME_RANGE>	Default visible time range of the Timeline, in minutes.
FLIGHT_COUNT_UNIT	4	1	N	FCA_TIMELINE	<FLIGHT_COUNT_UNIT>TOTAL</FLIGHT_COUNT_UNIT>	One of PEAK, TOTAL, or ENTRY. Default selection rule for flight count values.
SELECTED_TIME_RANGE	4	1	N	FCA_TIMELINE	<SELECTED_TIME_RANGE>300</SELECTED_TIME_RANGE>	Default time range of selected intervals for displaying flights and flight data, in minutes.
AUTO_EXAMINE	4	1	N	FCA_TIMELINE	<AUTO_EXAMINE>FALSE</AUTO_EXAMINE>	Flag to indicate whether the FEA/FCA is to be auto-examined when it is created or edited.
LIST_SORT	3	1	N	FCA_DISPLAY_PREFERENCES	<LIST_SORT>data</LIST_SORT>	Container for the FCA Dynamic List Sort options.
CTR_SORT_LEVEL	4	3	N	LIST_SORT	<CTR_SORT_LEVEL>data</CTR_SORT_LEVEL>	Container for the sorting options for a sort level. There are always containers for each of the 3 levels, even if some or all are empty. Level 1 is the "highest" level.
SORT_LEVEL	5	1	N	CTR_SORT_LEVEL	<SORT_LEVEL>2</SORT_LEVEL>	Sort level, one of 1, 2, 3.
COLUMN_HDR	5	1	N	CTR_SORT_LEVEL	<COLUMN_HDR>Aircraft ID</COLUMN_HDR>	Column to be sorted, or blank if this sort level is unused.
SORT_ORDER	5	1	N	CTR_SORT_LEVEL	<SORT_ORDER>DESCENDING</SORT_ORDER>	One of ASCENDING, DESCENDING. Order for sorting the column.
CTR_COLUMN_DISPLAY	3		N	FCA_DISPLAY_PREFERENCES	<CTR_COLUMN_DISPLAY>data</CTR_COLUMN_DISPLAY>	Container for defining all of the columns to be displayed by default in the FEA/FCA dynamic list.

Tag	Lv	#	Rq?	Parent	Example	Description
COLUMN_HDR	4		N	CTR_COLU MN_DISPL AY	<COLUMN_HDR>Aircraft ID<COLUMN_HDR>	Name of the column data item. Columns are listed in the order to be displayed. One of "Aircraft ID", "Secondary Filter", "Aircraft Type", "NRP Indicator", "FCA Indicator", "Airline", "Departure Time", "Arrival Time", "Controlled Arrival Time", "Controlled Departure Time", "Control Element", "Filed Speed", "Origin", "Destination", "Full Route", "Abbreviated Route", "Time to Intersect", "Entry Time", "Exit Time", "Requested Altitude", "Current Altitude", "Departure Center", "RVSM", "Current Center", "OGTD", "Flight Status", "Centers Traversed", "Arrival Center".
PRIMARY_FILTER	2	1	N	FCA	<PRIMARY_FILTER>data</PRIMARY_FILTER>	Container for the primary filter definition.
<i>Filter Definition</i>	-	-	-	-	-	<i>See Table 5.</i>

NOTES:

1. Dates and times are in 14-digit date and time format: year (4 digits) month (2) day (2) hour (2) minute (2) and second (2).
2. There are four fundamental types of FEAs/FCAs: polygon, circle, line, and NAS.
3. A polygon FEA/FCA is a free-form shape drawn by hand by the creator of the FEA/FCA. It is defined by the vertices of a polygon.
4. A line FEA/FCA is a free-form shape that consists of one or more contiguous line segments drawn by hand by the creator of the FEA/FCA. It is defined by the end points of the line segments.
5. A circle FEA/FCA has a center-point and radius, again selected by the creator. A circle FEA/FCA also contains the vertices of a polygon that approximates the circle.
6. A polygon, circle, or line FEA/FCA can be drawn on a map based solely on the data in the FEA/FCA file.
7. A NAS element FEA/FCA contains the type and name of the NAS element on which the FEA/FCA is based; for example: an airport (BOS) or a TRACON (ZMPMSP). The FEA/FCA file does not contain the location information for a NAS element FEA/FCA. That is, to draw a NAS-element FEA/FCA, the application will need an independent source of the NAS element locations and boundaries.

Table 4 defines the content of a secondary filter file. A secondary filter is always associated with an FEA/FCA.

The URL of a secondary filter file is <http://IPAddress/TFMDI/filename>. Industry users will be given the test and operational IP addresses upon request. The filename comes from the Index file. A program can fetch a secondary filter file using an HTTP GET command.

Table 4. FEA/FCA Secondary Filter File Data Elements

Tag	Lv	#	Rq?	Parent	Example	Description
<xml version="1.0" standalone="yes"?>	1	1	Y	None	<xml version="1.0" standalone="yes"?>	First line, to make the file XML compliant.
SECONDARY_FILTER	1	1	Y	None	<SECONDARY_FILTER> everything </SECONDARY_FILTER>	Container for the entire secondary filter definition.
INDEX_INFO	2	1	Y	SECONDARY_FILTER	<INDEX_INFO> fca.cdmb.lxpc103.20040713161706.cdmb.lxpc103. 20040713162403%fca.cdmb.lxpc103.20040713161706% FILTER_TEST1%PUBLIC%20040713162403%lxpc103% %cdmb%lxpc103%cdmb</INDEX_INFO>	ETMS legacy data, up to 128 characters. Can be ignored, as all this data is contained in other individual elements.
FILTER_ID	2	1	Y	SECONDARY_FILTER	<FILTER_ID>filter. fca.cdmb.lxpc103.20040713161706.cdmb.lxpc103. 20040713162403</FILTER_ID>	System-generated unique filter identifier.
FCA_ID	2	1	Y	SECONDARY_FILTER	<FCA_ID>fca.cdmb.lxpc103.20040713161706</FCA_ID>	Identifier of FEA/FCA with which this filter is associated.
NAME	2	1	Y	SECONDARY_FILTER	<NAME>FILTER_TEST1</NAME>	User-assigned filter name (must be unique among all filters associated with the same FEA/FCA).
DOMAIN	2	1	Y	SECONDARY_FILTER	<DOMAIN>PUBLIC</DOMAIN>	Domain of filter Generally, possible values are PUBLIC, SHARED, LOCAL, or PRIVATE, but only public filter definitions are available on TFMDI.
FCA_DOMAIN	2	1	Y	SECONDARY_FILTER	<FCA_DOMAIN>PUBLIC</FCA_DOMAIN>	Domain of FCA. Generally, possible values are PUBLIC, SHARED, LOCAL, or PRIVATE, but only public FEA/FCAs are available on TFMDI.
LASTUPDATE	2	1	Y	SECONDARY_FILTER	<LAST_UPDATE>20040713162403</LAST_UPDATE>	Date and time ¹ filter was last updated.
CR_SITE	2	1	Y	SECONDARY_FILTER	<CR_WKSTN>lxpc103</CR_WKSTN>	ETMS workstation from which the FEA/FCA was created

Tag	Lv	#	Rq?	Parent	Example	Description
CR_WKSTN	2	1	Y	SECONDARY_FILTER	<CR_SITE>cdmb</CR_SITE>	ETMS site from which the FEA/FCA was created
UP_SITE	2	1	Y	SECONDARY_FILTER	<UP_SITE>cdmb</UP_SITE>	ETMS site from which the FEA/FCA was last updated
UP_WKSTN	2	1	Y	SECONDARY_FILTER	<UP_WKSTN>lxpc103</UP_WKSTN>	ETMS workstation from which the FEA/FCA was last updated
SHOW	2	1	Y	SECONDARY_FILTER	<SHOW>Y</SHOW>	Flag indicating whether the filter should be shown by default (Y or N).
PRIORITY	2	1	Y	SECONDARY_FILTER	<PRIORITY>1</PRIORITY>	Indicates the priority of this filter (integer, or -1 if no priority is assigned). The integer value corresponds to the default order in which the filters are displayed and applied. ²
<i>Filter Definition</i>	-	-	-	-	-	<i>See Table 5.</i>

NOTES:

1. Dates and times are in 14-digit date and time format: year (4 digits) month (2) day (2) hour (2) minute (2) and second (2).
2. Filters are displayed from the top down starting with filter number 1. If a particular flight matches more than one filter, it should be displayed as belonging to the higher filter; that is, the filter with the lower number.

Table 5 defines the data associated with an FEA/FCA filter. These can appear within the PRIMARY_FILTER container in an FEA/FCA definition file, or in the SECONDARY_FILTER container of a FEA/FCA secondary filter file. Because the level depends on the usage, this column has been omitted.

Table 5. FEA/FCA Filter Definition Data Elements

<i>Tag</i>	<i>#</i>	<i>Rq?</i>	<i>Parent</i>	<i>Example</i>	<i>Description</i>
COLOR_ID	1	N	PRIMARY_FILTER or SECONDARY_FILTER	<COLOR_ID>17</COLOR_ID>	Color index indicating the color that ETMS is using to draw the filter. 0 – 35. (NOTE: See Appendix A for a definition of the COLOR_ID values.). Beginning in 8.4, present only in a SECONDARY_FILTER definition, not the PRIMARY_FILTER definition.
CONDITIONS	1	Y	PRIMARY_FILTER or SECONDARY_FILTER	<CONDITIONS> data </CONDITIONS>	Container for conditions ^{1,2,3} that comprise the filter.
ANY	1	N	CONDITIONS	<ANY>data</ANY>	Container for all of the conditions that should be “OR’ed” when filtering the flights. ¹
ALL	1	N	CONDITIONS	<ALL>data</ALL>	Container for all of the conditions that should be “AND’ed” when filtering the flights. ¹
DEPARTS_ANY	1	N	ANY or ALL	<DEPARTS_ANY>ZBW LGA JFK</DEPARTS_ANY>	List of center and/or airport names, separated by spaces. Only flights departing one of these are included in the FEA/FCA.
DEPARTS_NONE	1	N	ANY or ALL	<DEPARTS_NONE>ZBW LGA JFK</DEPARTS_NONE>	List of center and/or airport names, separated by spaces. Flights departing any of these are excluded from the FEA/FCA.
ARRIVES_ANY	1	N	ANY or ALL	<ARRIVES_ANY>ZBW LGA JFK</ARRIVES_ANY>	List of center and/or airport names, separated by spaces. Only flights arriving one of these are included in the FEA/FCA.
ARRIVES_NONE	1	N	ANY or ALL	<ARRIVES_NONE>ZBW LGA JFK</ARRIVES_NONE>	List of center and/or airport names, separated by spaces. Flights arriving any of these are excluded from the FEA/FCA.
TRAVERSE_ANY	1	N	ANY or ALL	<TRAVERSE_ANY>ZOB ZID</TRAVERSE_ANY>	List of center, sector and/or fix names, separated by spaces. Only flights that fly through any of these are included in the FEA/FCA.

Tag	#	Rq?	Parent	Example	Description
TRAVERSE_ALL	1	N	ANY or ALL	<TRAVERSES_ALL>ZOB ZID</TRAVERSES_ALL>	List of center, sector and/or fix names, separated by spaces. Only flights that fly through all of these are included in the FEA/FCA.
TRAVERSE_NONE	1	N	ANY or ALL	<TRAVERSES_NONE>ZOB ZID</TRAVERSES_NONE>	List of center, sector and/or fix names, separated by spaces. Flights that fly through any of these are excluded from the FEA/FCA.
USE_AIRWAY_ANY	1	N	ANY or ALL	<USE_AIRWAY_ANY>J60 J45</USE_AIRWAY_ANY>	List of airway names, separated by spaces. Only flights that fly on any of these are included in the FEA/FCA.
USE_AIRWAY_ALL	1	N	ANY or ALL	<USE_AIRWAY_ALL>J60 J45</USE_AIRWAY_ALL>	List of airway names, separated by spaces. Only flights that fly on all of these are included in the FEA/FCA.
USE_AIRWAY_NONE	1	N	ANY or ALL	<USE_AIRWAY_NONE>J60 J45</USE_AIRWAY_NONE>	List of airway names, separated by spaces. Flights that fly on any of these are excluded from the FEA/FCA.
LOCATED_ANY	1	N	ANY or ALL	<LOCATED_ANY>ZOB ZID</LOCATED_ANY>	List of center and/or sector names, separated by spaces. Only flights currently located within any of these centers or sectors are included in the FEA/FCA.
LOCATED_NONE	1	N	ANY or ALL	<LOCATED_NONE>ZOB ZID</LOCATED_NONE>	List of center and/or sector names, separated by spaces. Flights currently located within any of these centers or sectors are excluded from the FEA/FCA.
TYPE_ANY	1	N	ANY or ALL	<TYPE_ANY>B737 B757</TYPE_ANY>	List of aircraft types, separated by spaces. Only flights that match one of these are included in the FEA/FCA.
TYPE_NONE	1	N	ANY or ALL	<TYPE_NONE>B737 B757</TYPE_NONE>	List of aircraft types that are excluded from the FEA/FCA, separated by spaces.
REMARKS_ANY	1	N	ANY or ALL	<REMARKS_ANY>NRP DVRSN</REMARKS_ANY>	List of remarks, separated by spaces. Only flights that file any of these remarks are included in the FEA/FCA.
REMARKS_ALL	1	N	ANY or ALL	<REMARKS_ALL>NRP DVRSN</REMARKS_ALL>	List of remarks, separated by spaces. Only flights that file all of these remarks are included in the FEA/FCA.
REMARKS_NONE	1	N	ANY or ALL	<REMARKS_NONE>NRP DVRSN</REMARKS_NONE>	List of remarks, separated by spaces. Flights that file any of these remarks are excluded in the FEA/FCA.

Tag	#	Rq?	Parent	Example	Description
HEADING_IS	1	N	ANY or ALL	<HEADING_IS>270 45</HEADING_IS>	A heading followed by a tolerance, separated by spaces. Only flights whose heading is within the specified tolerance of the specified heading are included in the FEA/FCA. The heading is degrees from 0 to 360. The tolerance is degrees from 0 to 180.
HEADING_NOT	1	N	ANY or ALL	<HEADING_NOT>270 45</HEADING_NOT>	A heading followed by a tolerance, separated by spaces. Flights whose heading is within the specified tolerance of the specified heading are excluded from the FEA/FCA. The heading is degrees from 0 to 360. The tolerance is degrees from 0 to 180.
ACID_ANY	1	N	ANY or ALL	<ACID_ANY>B737 B757</ACID_ANY>	List of aircraft IDs (call signs), separated by spaces. Only flights with one of these IDs are included in the FEA/FCA.
ACID_NONE	1	N	ANY or ALL	<ACID_NONE>B737 B757</ACID_NONE>	List of aircraft IDs (call signs), separated by spaces. Flights with one of these IDs are excluded from the FEA/FCA.
FLIGHT_LEVEL	1	N	ANY or ALL	<FLIGHT_LEVEL>600 240</FLIGHT_LEVEL>	Upper flight level and lower flight level, separated by spaces. Only flights that are within the specified range at some time while they are in the FEA/FCA are included. Flight levels are in hundreds of feet.
AIRCRAFT_CATEGORY_ANY	1	N	ANY or ALL	<AIRCRAFT_CATEGORY_ANY>J T</AIRCRAFT_CATEGORY_ANY>	List of single-character aircraft type abbreviations (J=Jet, P=Prop, T=Turbo), separated by spaces. Only flights that match one of these are included in the FEA/FCA.
WEIGHT_CLASS_ANY	1	N	ANY or ALL	<WEIGHT_CLASS_ANY>H L</WEIGHT_CLASS_ANY>	List of single-character weight class abbreviations (H=Heavy, L=Large, S=Small), separated by spaces. Only flights that match one of these are included in the FEA/FCA.
USER_CATEGORY_ANY	1	N	ANY or ALL	<USER_CATEGORY_ANY>T F C</USER_CATEGORY_ANY>	List of single-character user category abbreviations (T=Air Taxi, F=Cargo, C=Commercial, G=General Aviation, M=Military), separated by spaces. Only flights that match one of these are included in the FEA/FCA.

Tag	#	Rq?	Parent	Example	Description
STATUS	1	N	ANY or ALL	<STATUS>ACTIVE</STATUS>	One of ACTIVE, PROPOSED. If ACTIVE, FEA/FCA includes only airborne flights. If PROPOSED, FEA/FCA includes only flights currently on the ground.
RVSM	1	N	ANY or ALL	<RVSM>COMPLIANT</RVSM>	One of COMPLIANT, NON-COMPLIANT. If COMPLIANT, the FEA/FCA includes only flights that are RVSM compliant at the time they are in the FEA/FCA. If NONCOMPLIANT, the FEA/FCA includes only flights that are RVSM non-compliant at the time they are in the FEA/FCA.
DEPARTURE_TIME_RANGE	1	N	ANY or ALL	<DEPARTURE_TIME_RANGE>20040714160000 20040714175900</DEPARTURE_TIME_RANGE>	Begin-time and end-time ⁴ , separated by spaces. Only flights whose departure times fall within this time range are included in the FEA/FCA.
ARRIVAL_TIME_RANGE	1	N	ANY or ALL	<ARRIVAL_TIME_RANGE>20040714160000 20040714175900</ARRIVAL_TIME_RANGE>	Begin-time and end-time ⁴ , separated by spaces. Only flights whose arrival times fall within this time range are included in the FEA/FCA.

NOTES:

1. Filter conditions may be “AND’ed” or “OR’ed” to define the desired set of flights. Conditions that should be “AND’ed” are grouped together in the ALL container; conditions that should be “OR’ed” are in the ANY container. For example, the conditions:

```
<ANY>
  <DEPARTS_ANY>BOS</DEPARTS_ANY>
  <ARRIVES_ANY>LGA</ARRIVES_ANY>
</ANY>
```

include a flight that either departs BOS *or* arrives LGA; that is, the FEA/FCA includes all flights departing BOS plus all flights arriving LGA. However, the conditions:

```
<ALL>
  <DEPARTS_ANY>BOS</DEPARTS_ANY>
  <ARRIVES_ANY>LGA</ARRIVES_ANY>
```

</ALL>

include a flight only if it departs BOS *and* arrives LGA; that is, only flights that were operating from BOS to LGA.

The ANY or ALL distinction can also apply to an individual filter condition. For example,

<TRAVERSESES_ANY>ZID ZOB</TRAVERSESES_ANY>

includes a flight that either traverses ZID *or* ZOB, while

<TRAVERSESES_ALL>ZID ZOB</TRAVERSESES_ALL>

includes a flight only if it traverses both ZID *and* ZOB.

If the “NONE” option is used, a flight is excluded if it matches any of the listed criteria.

2. All of the filter conditions allow multiple entries within the tags. In all cases, these will be space-delimited.
3. If a filter is omitted, no filtering should be applied for that criteria; for example, if WEIGHT_CLASS_ANY is not present, flights of any weight class would be included in the FEA/FCA.
4. Dates and times are in 14-digit date and time format: year (4 digits) month (2) day (2) hour (2) minute (2) and second (2).

6. Flow Evaluation Area/Flow Constrained Area Flight Lists

To be provided later.

7. Alert Data

To be provide later.

8. Collaborative Convective Forecast Product (CCFP) Data

To be provided later.

9. National Convective Weather Forecast (NCWF) Data

To be provided later.

Appendix A. Color Table

The following table provides information for interpreting the color indices included in the various data files. The indices are defined using the COLOR_ID XML tag.

COLOR_ID	Motif Color Name	RGB Value
0	Black	0 0 0
1	Gray40	102 102 102
2	Gray60	153 153 153
3	azure3	193 205 205
4	White	255 255 255
5	AntiqueWhite2	238 223 204
6	red4	139 0 0
7	red1	255 0 0
8	OrangeRed	255 69 0
9	Coral	255 114 86
10	LightPink	255 182 193
11	bisque	255 228 196
12	purple3	125 38 205
13	Plum	197 72 155
14	magenta3	205 0 205
15	VioletRed1	255 62 150
16	plum1	255 187 255
17	LightCyan	224 255 255
18	RoyalBlue4	39 64 139
19	Blue	0 0 255
20	SteelBlue3	79 148 205
21	SkyBlue	114 159 255
22	Cyan	0 255 255
23	PaleTurquoise	175 238 238
24	OliveDrab4	105 139 34
25	LimeGreen	0 175 20
26	DarkSeaGreen	143 188 143
27	khaki3	205 198 115
28	LightSeaGreen	32 178 170
29	chartreuse	127 255 0
30	chocolate3	205 102 29
31	Orange	255 135 0
32	Gold	218 170 0
33	LightSalmon1	255 160 122
34	Yellow	255 255 0
35	LightGoldenrodYellow	250 250 210