ATMS/ETMS Software Requirements

<u>Name:</u>	Ken Howard
<u>Date:</u>	~~11/21/00~~7/6/04
<u>Feature Described:</u>	Simplified substitutions – Rev 7~~6~~
<u>ETMS Version:</u>	7.9~~CDM~~
<u>Remarks:</u>	
<u>Reviewers/CSC:</u>	
<u>Reviewers/Volpe:</u>	
<u>Reviewers/FAA:</u>	

1. Overview

Simplified Substitutions (a.k.a. simplified subs) is a work activity to re-implement the way airlines do flight/slot substitutions in a ground delay program. A System Specification for simplified subs has been developed collaboratively with the airlines and has been agreed to by the airlines and the FAA (see *Requirements for Simplified Subs – Rev ~~9~~13*, dated ~~10/16/00~~7/6/04, by Ken Howard). It is necessary to read this document prior to attempting to understand these software requirements. The most relevant requirements are summarized as follows:

- The format of the EDCT lists and CTA lists (a.k.a. re-synch lists) sent out by FAA will be changed. We refer to the new format as *slot lists*.

- The slot lists will be available to the airlines over CDMNET (via socket) in addition to ARINC.

- Airline will assign flights to slots using FM messages grouped in a new packet type: SS.

- Canceling flights and assigning flights to slots will no longer be coupled.

- Error checking will continue to enforce the one-flight to one-slot rule.

- A mechanism will be built to allow the airline to "hold" open slots from compression, but to ensure that the slots be used.

To help ensure a smooth transition for the airlines, these additional requirements must be met:

- ETMS must accept subs in the new or old formats. An airline will be required to do a re-synch request prior to switching formats.

- ETMS must be able to send both the old EDCT lists and the new slot lists to the airlines when a GDP is issued.

Simplified subs will require implementation changes for the several ETMS hub-site processes, ETMS field-site processes, FSM, and the airlines. These software requirements provide a detailed breakdown of the hub processing architecture and requirements. The requirements for FSM and the ETMS field site software are included here for completeness. It is left to the airlines to determine their own requirements.

2. Functional Decomposition

System architecture

Simplified subs requires architectural changes in two areas: how GDPs are issued, and how subs are processed.

Issuing a GDP

Figure 1 shows a data flow diagram of how EDCTs will be issued under simplified subs.

The EDCTs are generated by FSM in the FADT file. For simplified subs, FSM will generate two additional sections (a.k.a. reports) in the FADT for the new slot lists: an airline-sorted section in the new format, and a center-sorted section in the new format. FSM will still generate all the existing sections as well; that is, the current format EDCTs and the new format slot lists will both be represented in the FADT file.

Autosend will then be invoked to send out the EDCTs and/or slot lists. Autosend/RTR will be configurable so that they know where to send the EDCTs (current format) and where to send the slot lists (new format). Autosend/RTR will transmit both the old and new formats via IFCN.FE as they currently do. Data destined for ARINC, NADIN, or the hub EDCTs will be forwarded by IFCN.FE to the appropriate process as it currently does. Data destined to the airlines over the CDMNET will be sent to the FD_FE process.
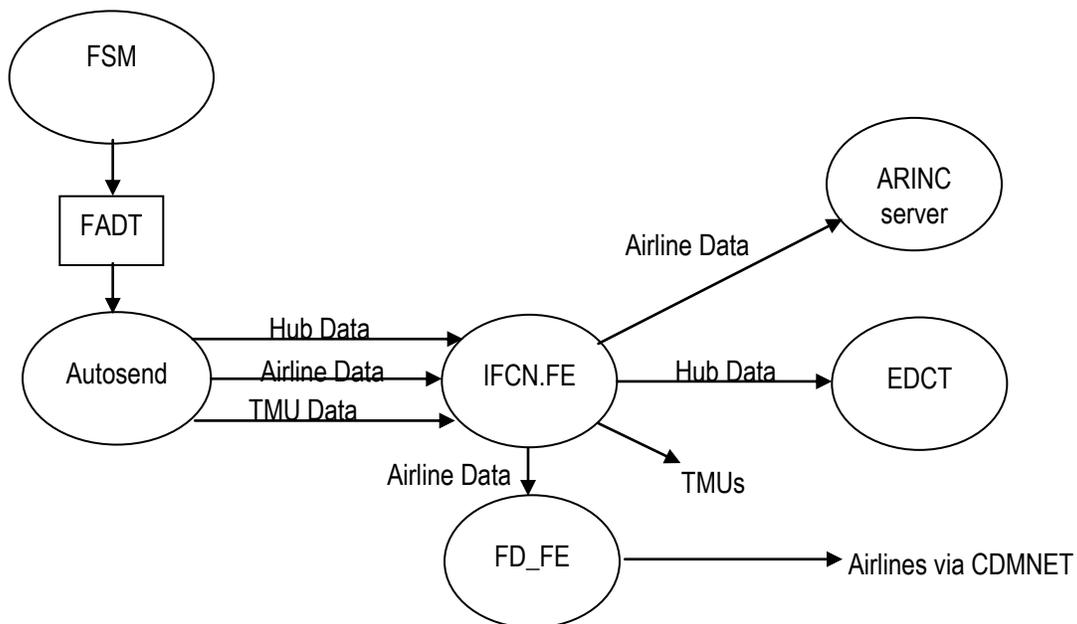
Figure 1. EDCT/Slot List Data Flow

It is critical for test and deployment that the data format and transmission paths be easily configurable. Autosend/RTR and IFCN.FE will together allow the configuration to be controlled as follows:

- Hub EDCTs and slot lists will be controlled by a combination of Autosend/RTR and IFCN.FE, as follows:

  - Autosend/RTR will be configurable to send only the old format EDCTs, only the new format slot lists, or both.

- IFCN.FE will be configurable to select whether each hub EDCT process is getting the old or new format.

- NOTE: This allows both old and new hub strings to receive operational GDP data, which will be especially useful during OT&E. Once all hubs are transitioned to the new format, the transmission of the old format from the ATCSCC to the hub can be turned off.

- Airline EDCTs and slot lists will be completely controlled by Autosend/RTR. For each individual airline, Autosend/RTR will provide the following configuration choices:

  - Whether the airline should get the old format EDCTs, the new format slot lists, or both.

  - The ARINC addresses to which the old format EDCTs should be sent (current functionality).

  - The ARINC addresses and/or CDMNET client names to which the new format slot lists should be sent.

- The transmission of EDCTs and slot lists to the TMUs will be completely controlled by Autosend/RTR. A single selection will control the format to all TMUs. The choices will be old format EDCTs only, new format slot lists only, or both.

The master-slave configuration will work as follows.

- Autosend will send EDCTs and/or slot lists to the IFCN.FE processes on both master and slave strings.

- Only the master IFCN.FE will forward data on to the destinations: ARINC, NADIN, FD_FEs, TMUs, hub EDCTs.

- The master IFCN.FE will send to both the master and slave EDCT processes.

- The master IFCN.FE will send to the FD_FE processes running on both master and slave strings.

- Each FD_FE will send airline data on to the airline regardless of its string's master/slave status. This allows an airline to be connected to either FD_FE.

Processing Airline Messages

Figure 2 shows a data flow of how the airline messages will be processed.

SS messages can be sent either over the ARINC ADNS line or over CDMNET via a socket connection to the FD_FE process. The ARINC SS messages are sent by the ARINC server to the IFCN.FE process, which relays them to the EDCT process for error checking. The CDMNET SS messages are sent by the FD_FE directly to the EDCT process (NOTE: this is the existing data flow for airline flight data messages). If the EDCT process detects an error, it will generate a message back to the airline indicating the type of cause of the error. The error message may be directed back either via the FD_FE and CDMNET, or via the IFCN.FE and ARINC. If EDCT accepts the message, it will update its own database, the Controls Database (CDB), and pass the updates to the Flight Database Process (FBDP). EDCT will also send a confirmation message back to the airline, again either via ARINC or CDMNET. The Flight Database Process (FDBP) will update its own database, the Flight Database (FDB), and propagate the updates to TDB and FTM through the normal mechanisms.

There are several other requirements for new data flows to the airlines.

A slot list request (re-synch) can be made by the airline over the CDMNET. These requests will be sent directly from the FD_FE to the EDCT Request Server on the same string. The reply will come from an EDCT Request Server back to the FD_FE for transmission to the airlines.

The airlines can also receive automated messages when subs are turned on or off, or when a GDP is purged. These messages will be generated by the EDCT process and directed to a configurable set of addresses (the same addresses from which a sub is allowed?). Messages destined for ARINC will be sent to the IFCN.FE to be relayed to the ARINC server; messages destined for CDMNET clients will be sent directly to the FD_FE.
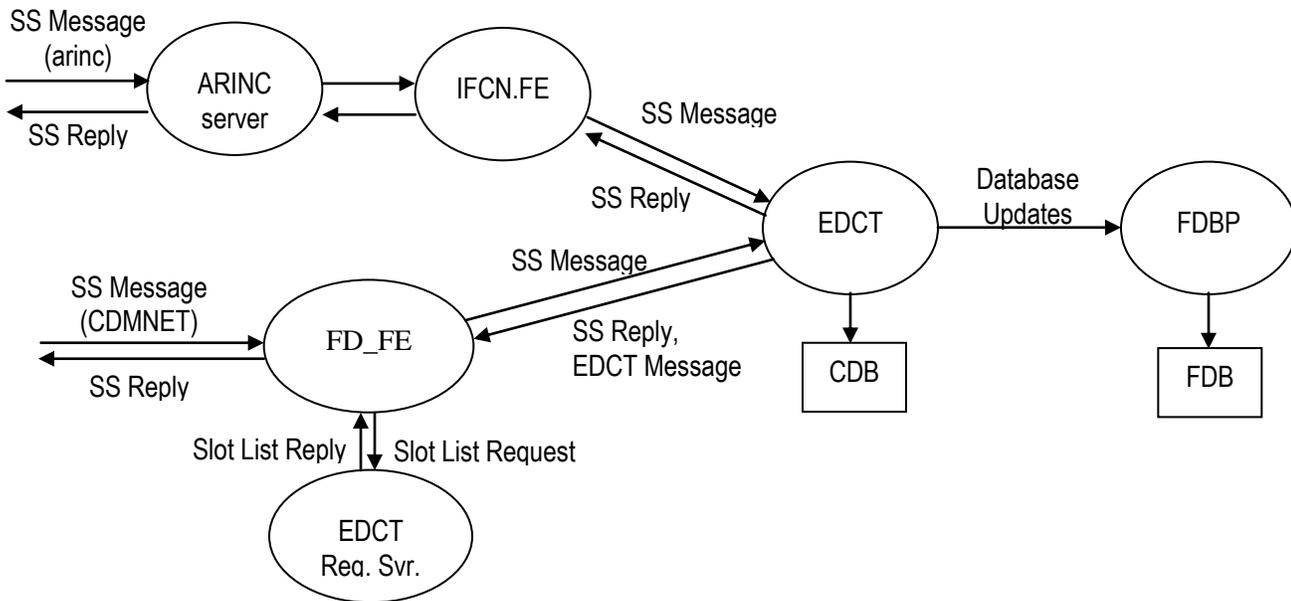

Figure 2. Simplified Sub Message Data Flow

The master-slave configuration will work as follows.

- The ARINC Server will send messages to the IFCN.FE processes on both master and slave strings.

- Only the master IFCN.FE will send the messages on to the EDCT processes; it will send to both the master and slave EDCTs.

- CDMNET SS messages may come in on either the master or slave string. Each FD_FE will send messages to both the master and slave EDCTs.

- Each EDCT will check the sub message and if accepted, update its CDB.

- Only the master EDCT will send updates to the FDBP on its string.

- Only the master EDCT will send error or confirmation messages back to the sender.

- Only the master EDCT will send SUB ON, SUB OFF, and purge messages to the airlines; it will send to both FD_FE processes.

- The FDBPs will process updates in the normal master-slave method.

4

FSM Requirements

1) When sending EDCTs, FSM will generate slot lists in the new format and include them in the FADT file in addition to the current EDCT format.

2) FSM will generate a center-sorted slot list that will appear in the FADT as a new "report" (similar to the B6 report).

3) FSM will generate an airline-sorted slot list that will appear in the FADT as a new "report" (similar to the B8 report).

4) FSM will recognize the Hold Flag variable for each flight in the ADL and display it as part of the Flight Detail report.

5) During RBS++ or compression, FSM will preserve open slots associated with cancelled flights for which the Hold Flag is true.

Autosend/RTR Requirements

6) Autosend/RTR will process and transmit FA delays and EDCTs (old format) in the same manner as in the current ETMS.

7) Autosend will accept new sections in the FADT for center and airline sorted slot lists (new format).

8) Autosend will make "sub files" for the slots lists in a similar manner to the EDCT lists, but using the slot list preface files and the slot list data. (NOTE: The hub slot list file should include the first line of the FADT as the EDCT file does.)

9) Autosend will provide separate, user-configurable, preface files for the slot lists (i.e., separate from the EDCT preface files).

10) Autosend/RTR will optionally send TMU slot lists to the same address as the TMU EDCTs.

11) Autosend/RTR will optionally send ETMS hub slot lists to the same IFCN.FE processes to which it sends the hub EDCT files.

12) Autosend/RTR will optionally send airline slot lists out to the airlines over ARINC using the current path through IFCN.FE. If a message is destined for more than one ARINC address, IFCN.FE will send one packet with multiple address fields.

13) Autosend/RTR will optionally send airline slot lists to the hub site IFCN.FE functions for transmittal to the airlines over the CDMNET. If a message is destined for more than one CDMNET client, IFCN.FE will send one packet with multiple client names. [NOTE: Although it is not a requirement, Autosend will also be able to send the old-format EDCT lists over the CDMNET.]

14) Autosend/RTR will provide new configuration capabilities to control the sending of data to the hub EDCT processes:

   a) whether to send only EDCTs to the ETMS hub(s)

   b) whether to send only slot lists to the ETMS hub(s)

   c) whether to send both EDCTs and slot lists to the ETMS hub(s)

15) Autosend/RTR will provide new configuration capabilities to control the sending of data to each individual airline:

   a) whether to send only EDCTs to the airline

   b) whether to send only slot lists to the airline

c) whether to send both EDCTs and slot lists to the airline

d) what ARINC address(es) and/or CDMNET client names to send the airline EDCTs to

e) what ARINC address(es) and/or CDMNET client names to send the airline EDCTs to

16) Autosend/RTR will provide new configuration capabilities to control the sending of data to the TMUs (both via NADIN and ETMS):

a) whether to send only EDCTs to the TMUs

b) whether to send only slot lists to the TMUs (this will be the operational choice)

c) whether to send both EDCTs and slot lists to the TMUs

17) Autosend will use the same communications protocol (e.g., handshaking, queuing, buffering) with IFCN.FE for slot lists as it does for the EDCTs with one exception:

a) Autosend will break CDMNET-bound airline messages into sections based on the ETMS network addressing packet-size limit rather than the ARINC packet-size limit.

IFCN.FE Requirements

18) IFCN.FE will continue to process EDCT and SI messages as it does in the current ETMS.

19) IFCN.FE will pass airline slot lists destined for ARINC to the ARINC server for transmittal to the airlines in the same manner as airline EDCT lists (no change required for this).

20) IFCN.FE will pass airline slot lists destined for CDMNET to one or more FD_FE processes for transmittal to the airlines.

21) IFCN.FE will use a similar protocol with FD_FE as it does with the ARINC server (one packet with multiple addresses).

22) IFCN.FE will process hub slot list files in the same manner as the hub EDCT files with one exception:

a) IFCN.FE will name the slot list files using the keyword "slot" rather than "edct".

23) IFCN.FE wil perform site authorization checking for the slot list files in the same manner as the EDCT files; that is, it will only accept a slot list from an authorized site.

24) IFCN.FE will pass SS messages received from the ARINC driver to the EDCT process in the same manner as the SI sub messages (no change).

25) IFCN.FE will send SS replies and other messages from EDCT out to ARINC via the ARINC server (in the same manner as the current SI replies).

26) IFCN.FE will apply the same master/slave logic to the slot lists and SS messages as it does for the EDCTs and SIs; that is, the IFCN.FE will only transmit the files if in master mode.

27) IFCN.FE will perform the same communications protocol (i.e., handshaking and queuing) for the slot lists and simplified subs as it does for the EDCT files and SI messages.

28) IFCN.FE will provide the following configuration control regarding the addressing of messages:

a) the site for the ARINC server

b) the site for the NADIN server

c) one or more sites for FD_FE servers

d) one or more sites for EDCT processes that should get EDCT files (old format)

e) one or more sites for EDCT processes that should get slot list files (new format)

FD_FE Requirements

29) FD_FE will accept socket connections from airlines on a designated port from authorized IP addresses.

30) FD_FE will validate the client tag on the first message received. If the tag is authorized for the IP address, FD_FE will allow the connection and create a table entry for that client. Other wise FD_FE will close the connection.

31) FD_FE will associate a unique client name with each client tag. The client names will be defined in a configuration file.

32) FD_FE will associate a client type with each client tag. The client type will be "FD" for a CDM flight data client or "SS" for a simplified sub client.

33) FD_FE will transmit slot lists in the following manner:

a) FD_FE will accept slot lists messages from IFCN.FE. A slot list will have one or more client names associated with it

b) If valid, FD_FE will attempt to send a copy of the slot list to each connected client whose tag starts with the same three letters as the name(s) associated with the list.

c) If the client for whom the slot list is intended is connected, the FD_FE will send the slot list out to the client over the socket.

d) If the client is not connected, the FD_FE will discontinue any attempt to send the slot list to that client (e.g., there will be no queuing, re-sending, or error messages sent).

34) FD_FE will processed simplified sub messages in the following manner:

a) FD_FE will accept SS messages from the client and transmit them to one or more EDCT processes through ETMS network addressing.

b) FD_FE will perform handshaking with the EDCT processes as for the FD messages.

c) FD_FE will allow both handshaking and non-handshaking EDCT clients top be defined in the configuration file. FD_FE will support only two hand-shaking clients.

d) FD_FE will accept SS replies from EDCT and transmit them in the same manner as FD replies.

35) FD_FE will processed GDP requests in the following manner:

a) FD_FE will accept GDP requests from the client and transmit them to the EDCT Request Server on the same string through ETMS network addressing.

b) FD_FE will accept GDP replies from EDCT and transmit them in the same manner as FD replies; that is, the reply will only be sent to the specific client that made the request.

c) If the client that made the request is no longer connected when the reply is ready, FD_FE will discard the reply.

36) FD_FE will transmit EDCT SUB OFF, SUB ON, PURGE, and other messages in the following manner:

a) FD_FE will accept EDCT messages from EDCT. An EDCT message will have one or more client names associated with it

b) FD_FE will validate that the EDCT message came from an authorized site, as defined in a static_data file.

c) If valid, FD_FE will attempt to send a copy of the EDCT message to each client of type "SS" that is currently connected.

EDCT Requirements

The EDCT Requirements are generally written for either the master or slave EDCT. Notes are inserted where there is a difference in the master/slave functionality. The differences mainly involve suppressing the slave from sending out data.

Communications

37) EDCT will accept slot lists from IFCN.FE using the same protocol as for EDCT files. EDCT will apply the same site authorization checking as for EDCT files. (file: site_authorization_file.edct)

38) EDCT will accept and acknowledge SS messages from IFCN.FE (originating from ARINC). EDCT will apply the same site authorization checking as for SI messages. (file: airlines.dat)

39) EDCT will accept and acknowledge SS messages from FD_FE (originating from CDMNET). EDCT will apply the same site authorization checking as for the IFCN.FE SS messages. (file: airlines_cdmnet.dat)

40) EDCT will send SS replies as follows:

a) For message received over ARINC, EDCT will send SS replies to the sending ARINC address via IFCN.FE.

b) For messages received over CDMNET, EDCT will send replies to FD_FE clients via FD_FE.

41) EDCT will send EDCT PURGE messages to the airlines whenever a PURGE is successfully performed, as follows:

a) EDCT will generate a PURGE message for each GDP airport even when processing an "ALL" transaction type. For example, if EDCT processes an "EDCT PURGE ALL PROGRAMS" message and there are current GDPs for BOS and DFW, EDCT will send two messages: EDCT PURGE BOS and EDCT PURGE DFW.

b) EDCT will send a single copy of a PURGE message to the FD_FE process on both master and slave strings, as defined in the EDCT parameter file. (NOTE: We can use the site names defined for ifcn.fe in the parameter file. There may be one or two addresses here.) EDCT will wild-card the client name, telling FD_FE to broadcast to all clients.

c) EDCT will send PURGE messages to each ARINC address configured in the static_data/airline_mapping file. (NOTE: The airline_mapping file is also used by Autosend.)

d) NOTE: Only the master EDCT will send any of these messages.

42) EDCT will send a message to the airlines whenever an EDCT UPDATE or FA DELAY is successfully applied to one of its flights. These messages will be sent as follows:

a) EDCT will send a copy of a message for each client to the FD_FE process on both master and slave strings. The FD_FE site names are defined in the EDCT parameter files; there may be one or two addresses here. The clients who should get notified for each three-letter code will be defined in the airline_mapping file. (NOTE: The airline_mapping file is also used by Autosend.)

b) EDCT will send a copy of the message to each ARINC client via the IFCN.FE. The ARINC clients who should get notified for each three_letter code will be defined in the airline_mapping file. (NOTE: The airline_mapping file is also used by Autosend.)

c) Only the master EDCT will send these messages.

43) EDCT will accept SS data requests (EDCT LIST, EDCT SUB SHOW, EDCT SLIST) from ARINC and pass them to EDCT Request Server

44) EDCT will accept messages from EDCT Request Server and pass them to IFCN.FE for transmission to ARINC. (This is the mechanism used to get SUB ON, SUB OFF, and SS data replies out to SS clients using ARINC addresses.)

Slot List Processing

45) EDCT will process a slot list file in a similar manner to an EDCT file; that is,

a) create an entry for each flight in the slot list in the Controls Database (CDB)

b) remove any previously issued controls for each flight in the slot list

c) preserve any open slot data associated with an already controlled flight for non-CDM participants

d) reset any open slot data associated with an already controlled flight for CDM participants

e) reset the SI-cancelled status for a previously controlled flight belonging to a CDM participant

46) EDCT will accept the following fields for each flight in the slot list (NOTE: all times will be 6 digits, DDHMM):

a) Flight ID

b) Assigned arrival slot

c) Departure airport

d) CTD

e) CTA

f) Control type

g) Exempt flag

h) Cancel flag

i) Hold flag

j) ERTA

k) Initial departure date/time

47) EDCT will save all fields for each flight in the slot list (all fields must be saved to replicate the slot list in a re-synch request).

48) EDCT will send a transaction to FDB for each flight in the slot list. The transactions will include all fields in the slot list. This transaction will be referred to as a "slot assignment".

Simplified Sub (SS) Message Parsing

49) EDCT will parse the new SS message format and perform syntax and limited semantic checking according to the interface specifications. Specific changes include:

a) FM format will be enhanced to allow CTD (T5), CTA (T6), Assigned Arrival Slot (A2), and Slot Hold (A6) fields.

b) FX format will be enhanced to allow Slot Hold (A6) field.

c) SC (slot create) message will now be accepted.

50) EDCT will reject any SS packet in which it finds a syntax or semantic error and return an error message to the sender.

51) EDCT will perform authorization checking as follows:

a) For an SS message received over ARINC, EDCT will verify that the sending ARINC address is authorized to sub for all of the carrier codes contained in the SS packet.

b) For an SS message received over CDMNET, EDCT will verify that the sending client is authorized to sub for all of the carrier codes contained in the SS packet. The authorizations will be defined in the static_data/cdm_airlines.dat file.

NOTE: We need to add this checking to regular FD packets as well!

52) EDCT will reject any SS packet that fails the authorization checking without making any changes to the CDB or sending any updates to the FDB. (SS ERROR: NOT AUTHORIZED TO SUB FOR THESE FLIGHTS)

53) EDCT will accept only those individual messages in an SS packet that meet the following criteria:

a) A Flight Modify message (FM) that contains at least the CTD (T5), CTA (T6), and ASLOT (A2) fields.

b) A Flight Cancel message (FX).

c) A "HOLD ALL" message.

d) A "RELEASE ALL" message.

e) A Slot Create (SC) message.

54) EDCT will reject any FM or FC received in an FD packet that modifies either the CTD (T5), CTA (T6), or ASLOT (A2) field.

55) EDCT will accept a new optional "Hold Flag" (A6) field on an FX message. (Note for clarity: EDCT will continue to accept an FX message received in an FD packet for a flight that is controlled as it already does.)

56) EDCT will reject an FD packet that contains either a "HOLD ALL", "RELEASE ALL", or Slot Create (SC) message.

Simplified Sub (SS) Message Processing

57) EDCT will perform error checking on every message in a packet and will report all errors found back to the sender. EDCT will only commit any updates to the CDM or FDB from an SS packet if it is found to be error-free.

58) EDCT will accept an FX message in an SS packet only if the flight being cancelled is controlled in the CDB. If not, it will reject the packet and return the following error message (SS ERROR: CANNOT CANCEL A NON-CONTROLLED FLIGHT)

59) EDCT will only accept an FM message in an SS packet if the following conditions are met:

a) The CTA for each flight is no earlier than the slot time and no more than 20 minutes later than the slot time. (SS ERROR: CTD NOT WITHIN 20-MINUTE WINDOW)

b) The ASLOT is currently assigned in the CDB to a flight that the sending carrier is authorized to sub. (SS ERROR: CAN'T SUB INTO SLOT NOT OWNED BY THIS CARRIER)

c) The ETE (CTA – CTD) is no more than 50% greater or less than the previous ETE (CTA – CTD). (SS ERROR: ETE CHANGE TOO BIG)

60) EDCT will reject an FM message under the following conditions:

a) The FM contains an A6 field and the flight is not cancelled in the CDB. (SS ERROR: CAN'T MODIFY HOLD FLAG; FLIGHT NOT CANCELLED)

b) The flight is a pop-up (i.e., has a control type of "FA")

c) The FM assigns a flight to an earlier ASLOT that is currently occupied by a former pop-up flight.

61) EDCT will only accept FMs in an SS packet that meets the following aggregate criteria:

a) Each ASLOT in the slot list appears in only one FM. (SS ERROR: CAN'T SUB TWO FLIGHTS INTO ONE SLOT)

b) Each flight in the slot list appears in only one FM. (SS ERROR: CAN'T SUB ONE FLIGHT INTO TWO SLOTS)

c) Each flight in the slot list already has an ASLOT in the CDB.. (SS ERROR: CAN'T SUB NON-CONTROLLED FLIGHT)

d) Each ASLOT in the SS packet is already occupied by a flight in the SS packet.. (SS ERROR: SLOT ALREADY OCCUPIED)

62) EDCT will only accept an SC message if:

a) the flight identified in the message is not already controlled

b) the slot identified in the message is not already defined

c) the slot is in the future.

63) EDCT will only accept a "HOLD ALL" message if the airport identified in the message is controlled. (SS ERROR: AIRPORT IS NOT CONTROLLED)

64) EDCT will only accept a "RELEASE ALL" message if the airport identified in the message is controlled. (SS ERROR: AIRPORT IS NOT CONTROLLED)

65) Once EDCT has accepted a SS packet, it will update the CDB and FDB as follows:

a) For each flight in an FX message,

    i) if the Hold Flag is modified, save the new value in the CDB

    ii) send a transaction to the FDB to cancel the flight

b) For each flight in an FM message, EDCT will:

    i) update the flight in the CDB with the new CTD, CTA, and ASLOT

    ii) optionally update the ERTA or Hold Flag, if sent

    iii) send a transaction to FDB to update the flight data (this should include all data sent in the FM)

c) For each flight in an SC message, EDCT will:

   i) Create an entry in the CDB and save all the flight data

   ii) Send an EDCT transaction to the FDB as if the flight has just been controlled (control type should be "SUB")

d) EDCT will process a "HOLD ALL" message as follows:

   i) Find all the cancelled flights that have slots in the future.

   ii) For each of these flights, EDCT will:

      (1) Set the Hold flag to TRUE in the CDB.

      (2) Send an update to the FDB telling the FDB to set the Hold Flag to TRUE. The transaction to the FDB should look exactly like an FM transaction that only sets the Hold Flag.

e) EDCT will process a "RELEASE ALL" message as follows:

   i) Find all the cancelled flights that have slots in the future.

   ii) For each of these flights, EDCT will:

      (1) Set the Hold flag to FALSE in the CDB.

      (2) Send an update to the FDB telling the FDB to set the Hold Flag to FALSE. The transaction to the FDB should look exactly like an FM transaction that only sets the Hold Flag.

66) Once EDCT has successfully processed a simplified sub message, it will:

a) Send a reply to the airline. The reply will consist of a header line indicating the message was processed successfully followed by a slot list including all the flights modified by the message

b) Log the results to the EDCT log file. The format of the log entry will exactly match the reply sent to the airline. (NOTE: Old-format subs will continue to be logged as they always have.)

c) Send a message to the printer at the command center. The format of this message will include the message sent by the airline as well as the reply sent to the airline. (NOTE: The message for old-format subs will continue to be generated in the old format.)

d) Send a message to the TMU. The format of the message will exactly match the reply sent to the airline. (NOTE: The message for old-format subs will continue to be generated in the old format.)

Compatibility with Old Format Subs

67) EDCT will continue to process old format sub messages as in the current ETMS.

68) When swapping flights using old format subs, EDCT will swap the new data fields related to the slot and/or control times. At this time, this appears to be only the new controlled arrival date field.

69) When a controlled arrival time is adjusted using an old format sub, EDCT will ensure that the controlled arrival date is adjusted for any day boundary changes.

70) When processing simplified sub messages, EDCT will NOT maintain the fields specific to supporting the old format subs (e.g., sub flag, rpl flag, open slot, etc.)

Other Messages

71) EDCT will accept "HOLD ALL SLOTS" or "RELEASE ALL SLOTS" commands from EDCT Request Server as process them as follows:

    a) If the command does not specify an airline name, EDCT will apply the command to all carriers in the GDP.

    b) If the command includes an airline name, EDCT will update the CDB and FDB as described for an airline-generated message.

    c) EDCT will send a notice back to EDCT Request Server indicating the results of the processing. (e.g., "@ 12 slots held for EWR", "@ 3 slots released for LGA AAL", "? No open slots for BOS")

72) The processing of the EDCT UPDATE command will be modified as follows:

    a) The format will be changed to require the IGTD as the sixth field of the message.

    b) EDCT will use the IGTD to find the matching entry in the CDB.

    c) EDCT will allow the following fields to be modified: controlled time of departure (CTD), controlled time of arrival (CTA), cancel flag (CX), slot hold flag (SH), earliest runway time of arrival (ERTA).

    d) All fields following the IGTD will require an identifying prefix.

    e) All fields following the IGTD will be optional.

    f) EDCT will accept and apply an update for a cancelled flight.

    g) If the cancel flag is modified by the update command, EDCT will change the status of the flight in the CDB.

    h) EDCT will pass any data changes, including a change in cancel status, to the FDB.

73) EDCT will process a simplified sub data request in an ARINC message, as follows:

    a) EDCT will accept data requests with a header line of "RQ".

    b) EDCT will pass the text of the requests to EDCT Request Server (as it does now for a CTALIST received in an SI message), along with the ARINC address from which it came.

    c) EDCT will accept messages from EDCT Request Server and pass them to IFCN.FE for transmission to ARINC (as it does now for a CTALIST reply).

Updates from FDB

74) EDCT currently receives transactions for controlled flights from FDB under certain conditions: when a controlled flight is activated, when a flight plan is processed for a controlled flight, when a flight is created for which an FA delay might apply, and when a controlled flight is diverted (is there also one for re-instating a cancelled flight?). In addition, EDCT will process the following a new data update from FDB updating the following fields for controlled flights:

    a) ERTA

    b) Slot Hold flag

EDCT Request Server Requirements

The first set of requirements pertain to the airlines ability to get GDP data.

75) EDCT Request Server (EDCT-R) will accept a EDCT SLIST, EDCT LIST, or EDCT SUB SHOW message sent from either ARINC (via EDCT) or CDMNET.

76) EDCT-R will validate that the ARINC address or CDMNET client name that sent the request is authorized to get data, and which carriers that address is authorized for. (These will be referred to as the authorized carriers.)

77) If the requesting address or client name is not authorized to get data, EDCT-R will return an error message (similar to what it uses for the REQ CTALIST).

78) If the requesting address or client name is authorized, EDCT-R will extract all data for that GDP for the authorized carriers, format the data into a slot list, and send it back to the requestor (via EDCT for an ARINC address).

79) EDCT-R will continue to generate the "REQ CTALIST" report in its current format.

80) EDCT-R will reject the following messages from ARINC or CDMNET: EDCT CHECK, EDCT LOG, EDCT PURGE, EDCT SHOW, EDCT SUB OFF, EDCT SUB ON, EDCT SUB PRINT, ECCT SUMMARY, EDCT UPDATE.

81) EDCT-R will send a message to SS clients whenever a SUB ON or SUB OFF is successfully performed, as follows:

   a) A single message will be sent to FD_FE for broadcast to all SS clients. (NOTE: EDCT-R must wild-card the client name.)

   b) A message will be sent to each ARINC address defined for SS data in the airline_mapping file. These messages will be passed to EDCT, which will relay them to IFCN.FE for transmission to ARINC.

   c) Only the master EDCT_R will send these messages.

The second set of requirements pertain to the FAA's ability to get GDP data (e.g., through TSD).

82) EDCT-R will accept an EDCT SLIST request from TSD and generate a report as follows:

   a) If the SLIST does not identify an airline, the report will include all flights in the GDP.

   b) If the SLIST includes and airline name, the report will be generated as if that carrier requested the report.

83) EDCT-R will accept new EDCT commands "HOLD ALL SLOTS" and "RELEASE ALL SLOTS". These commands will be passed to the master and slave EDCTs in the same manner as an EDCT PURGE command. Authorization checking should also be done as for an EDCT PURGE.

84) EDCT-R will modify the format of the EDCT CHECK response as follows (see sample in system requirements document):

   a) All fields from the slot list will be included.

   b) All time fields will be in a 6-digit date-time format (ddhhmm).

   c) All fields except for the flight ID will be preceded by the field identifier.

85) EDCT-R will modify the format of the EDCT LIST response as follows (see sample in system requirements document):

   a) The sub status will be included as a new column.

   b) The printer location will be included as a new line.

86) EDCT-R will accept EDCT UPDATE commands in the new format and send them through to EDCT for processing as it does now.

FDB Requirements

87) The FDB will process an EDCT transaction (edctfdb_msg_t) as follows:

   a) FDB will look for a matching flight using the following criteria:

      i) The flight ID must match.

      ii) The origin and destination airport must both match.

      iii) The initial gate departure date and time from the message must be within +/- one hour of the create departure date/time in the FDB record.

   b) FDB will update any flight that meets the above matching criteria.

   c) FDB will assign the values from the transaction for the following fields:

      i) Slot name

      ii) CTD (minutes after midnight and julian)

      iii) CTA (minutes after midnight and julian)

      iv) CTL_TYPE (based on the ctrl_type of the transaction)

      v) CTL_EXMPT

      vi) CTL_ELEM

      vii) GS delay flag (if ctrl type is gs_ctrl)

      viii) GDP delay flag (if ctrl type is gdp_ctrl)

      ix) If the OCTD field is blank and the ctrl type is gdp_ctrl, comp_ctrl, or blkt_ctrl, FDB will assign the CTD value to the OCTD. (OCTD is not set if CTL_TYPE is GS.)

      x) If the OCTA field is blank and the CTL_TYPE is gdp_ctrl, comp_ctrl, or blkt_ctrl, FDB will assign the CTA value to the OCTA. (OCTA is not set if CTL_TYPE is GS.)

      xi) Only if the transaction type is update (upd_ctrl), FDB will also update:

         (1) Slot Hold flag

         (2) ERTA

         (3) Cancel status, as follows

            (a) When a flight is cancelled from an update message, FDB will:

               (i) Set the flight status to cancelled

               (ii) Set the si_cancel flag to true

            (b) When a flight is re-instated from an update message, FDB will:

               (i) Set the flight status to controlled

               (ii) Clear all the cancel flags

   d) The FDB will use the CTD/CTA to update the flight times as follows (NOTE: the new controlled departure date field should be used in all time comparisons):

i) If the flight is not active, and the CTD is later than the current time, FDB will:

    (1) set the first event time to the time from new CTD

    (2) set the current departure date to the new controlled departure date

    (3) update the remaining events by adding the "delta T" (the difference between the new first event time and the old first event time)

ii) If the flight is not active and the CTD is earlier than the current time and the current time minus the CTD is less than the time-out interval, FDB will update the flight so that:

    (1) the first event time is approximately the current time

    (2) the flight appears time-out delayed

    (3) the flight will time-out cancel 90 minutes after the new CTD (if the new CTD is more than 90 minutes in the past

iii) If the flight is not active and the CTD is earlier than the current time and the current time minus the CTD is greater than or equal to the time-out interval, FDB will update the flight so that:

    (1) the first event time is the CTD + the time-out interval

    (2) the flight appears time-out cancelled

e) When FDB processes a slot assignment transaction for a cancelled flight, FDB will set the value of the Hold Flag for that flight to FALSE.

f) For each matching flight, FDB will generate an update transaction to EDCT if any of the following fields do not match what was already in the FDB:

    i) Hold Flag

    ii) Cancel Status

    iii) ERTA

g) FDB will log the transactions in the "CDM" log file using the existing message types.

88) The FDB FM message processing will be enhanced as follows:

a) FDB will process an FM for a cancelled flight.

b) FDB will assign the values from the transaction for the following new fields:

    i) Slot name

    ii) CTD

    iii) CTA

    iv) Controlled departure date

    v) ERTA

    vi) Hold Flag

c) If an FM message modifies a slot name, CTD, or CTA, FDB will set the following additional fields:

    i) CTL_TYPE will be set to SUB

    ii) CTL_EXMPT will be set to FALSE

d) When FDB processes an FM that updates the CTD/CTA, FDB will update the flights event list as follows (NOTE: the new controlled departure date field should be used in all time comparisons):

 i) If the CTD is later than the LRTD (or LGTD, if no LRTD defined), the FDB will update the flight times as follows:

  (1) The first event time will be set to the time from new CTD

  (2) The current departure date will be set to the new controlled departure date

  (3) The event list will be remodeled as it currently is.

e) When FDB processes an FM for a cancelled flight, and the FM does NOT update the CTD/CTA, and the cancel type is SI, FX, DV, or RS, FDB will NOT update the event list.

f) When FDB processes an FM for a cancelled flight and the cancel type is SI, FX, DV, or RS, FDB will NOT change the status of the flight.

g) When FDB processes an FM for a cancelled flight, and the cancel type is TO or RZ, FDB will update the event list and re-instate the flight according to the current logic.

h) When FDB processes an FM that is not a "simplified sub" FM for a flight that is "controlled" or "active" with control times, it will send an update to EDCT whenever one of the following fields is changed:

 i) Hold Flag

 ii) ERTA

89) The FDB FX message processing will be enhanced as follows:

a) FDB will update the value of the Hold Flag whenever a flight is cancelled according to the following rules:

 i) If a new value is provided in the FX transaction, the FDB will use that value.

 ii) If no value is provided in the FX transaction, the FDB will set the Hold Flag to FALSE.

b) FDB will include the value of the Hold Flag when it notifies EDCT that a controlled flight is cancelled.

90) The FDB SI CNX message processing will be enhanced as follows:

a) FDB will look for a matching flight using the following criteria (NOTE: same as for an edctmsg_t):

 i) The flight ID must match.

 ii) The origin and destination airport must both match.

 iii) The initial gate departure date and time from the message must be within +/- one hour of the create departure date/time in the FDB record.

b) The FDB will update every flight that meets those criteria.

c) The FDB will set the Hold Flag to FALSE

d) The FDB will include the value of the Hold Flag when it notifies EDCT that a controlled flight is cancelled.

91) If a flight is cancelled by an OAG cancel message (RS), a NAS Flight plan cancel (RZ), or a time-out (TO), the FDB will set the Hold Flag to ~~TRUE~~FALSE.

92) When an old-style EDCT or sub is processed, FDB will:

    a) Include the same fields in the transaction to EDCT as it does for a slot assignment (does it send a transaction when a sub is done?).

93) The FDB FC message processing will be enhanced as follows:

    a) FDB will notify EDCT whenever an FC re-instates a cancelled flight that was previously controlled (already does this?).

    b) FDB will notify EDCT when an FC creates a flight that might have an FA delay.

    c) FDB will notify EDCT when an FC updates any of the following fields:

        i) Hold Flag

        ii) ERTA

94) Every FDB transaction to EDCT will include the following fields:

        i) Flight ID

        ii) Origin

        iii) Destination

        iv) IGTD

        v) Hold Flag

        vi) Cancel Status

        vii) ERTA

95) Whenever FDB creates any new flight in the database, it will set the Slot Hold flag to false.

96) Whenever FDB re-instates a previously cancelled flight, it will set the Slot Hold flag to false.

3. External Systems

4. Databases

CDB will include the following new fields (Melia please give me a list).

FDB will include the following new fields:

- controlled departure date
- Kevin – anything else?

5. Data Interfaces

Configuration Files

Simplified subs include several new functions that must be governed by configuration data. Wherever possible these will be built on existing configuration files. These functions, and the associated configuration files are as follows.

97) airline_mapping

a) Contents: for each airline as identified by 3-letter code,

   i) whether to send slot lists, edct files, or both.

   ii) the ARINC addresses to which to send an airline's simplified sub data

   iii) the FD_FE client names to which to send an airline's simplified sub data

b) Used by:

   i) Autosend, for where to send slot lists. (/etms/ats/data/address_files/airline_mapping)

   ii) EDCT, for where to send the simplified subs messages for SUB OFF, SUB OFF, EDCT PURGE, FA DELAY, and EDCT UDPATE. (/etms/edct/static_ data/address_files/airline_mapping):

c) Format:

   i) each line will consist of:

      (1) three-letter airline code

      (2) keyword saying what kind of data to send: ORIGINAL for old-style EDCTs only; SLOT for new slot lists and other SS messages; BOTH for both

      (3) optional ARINC and/or CDMNET addresses to send SS messages to. CDMNET client names are preceded by "*" (e.g., *AAL).

      (4) sample: AAL SLOT *AAL SOCHQAA

98) /etms/edct/static_data/airlines.dat

a) Contents: for each airline as identified by 3-letter code,

   i) the ARINC addresses which are authorized to sub for the airline

b) Used by (applied only to messages received over ARINC):

   i) EDCT, to check authorization of SI messages

   ii) EDCT, to check authorization of FD messages

   iii) EDCT, to check authorization of SS messages

   iv) EDCT, to determine what flights should be included in a HOLD ALL SLOTS or RELEASE ALL SLOTS command

   v) EDCT Request Server, to determine what flights to include in an SLIST or CTALIST request

c) Format:

   i) each line will consist of:

      (1) three-letter airline code

      (2) one or more ARINC addresses authorized to send data for this carrier

      (3) sample: AAL ANPVPAA SOCHQAA

99) /etms/edct/static_data/airlines_cdmnet.dat

a) Contents: for each airline as identified by 3-letter code,

   i) the CDMNET client names which are authorized to send sub messages

b) Used by (applied only to messages received over CDMNET):

i) EDCT, to check authorization of FD messages

ii) EDCT, to check authorization of SS messages

iii) EDCT, to determine what flights should be included in a HOLD ALL SLOTS or RELEASE ALL SLOTS command

iv) EDCT Request Server, to determine what flights to include in an SLIST or CTALIST request

c) Format:

i) each line will consist of:

(1) three-letter airline code of sender/requester

(2) one or more three-letter codes of the carriers for which the sender/requester can submit or request data

(3) sample: AAL EGF

(4) NOTE: It is always assumed that a carrier can send and see data for its own flights. That is, in the sample, AAL can modify and see data for AAL and EGF.

6. User Interfaces

General Flow

For FAA, the user interfaces changes for simplified subs are limited to several enhancements to the data displays. The methods the specialist uses for running GDPs and requesting data will be unchanged.

Commands

Should the UPDATE command be enhanced to include the hold flag? The slot name? Exempt flag? Anything else?

Screens

Additional data fields have already been added as part of FSM 1.8. The only additional data that will start to appear are the hold flags.

Reports

Add fields to the EDCT SHOW? Any others?

7. Implementation Issues/Suggestions

Airline issues and questions:

1. We will need to provide the capability of generating EDCTs in both the new and old formats, at least for the airlines. The airlines will need to have an extended period where they can get either the both formats over ARINC, or the old over ARINC and the new over the socket. They will need to do a re-synch whenever they switch at their end.

2. Airlines will only be allowed to switch from new to old format subs during a particular GDP. If a new GDP is issued, airlines can start that GDP using the new format.

3. The new "broadcast" messages for SUB OFF, SUB ON, and EDCT PURGE could be broadcast to all the same addresses configured to get the EDCT lists. Is this what makes sense, or should there be a separate file of addresses to send these to.