

AI FLIGHT PLANNER

... a comprehensive, fully integrated AI flight planning tool for FS9 and FSX

INDEX

- [1.](#) GENERAL
- [2.](#) INSTALLING/UNINSTALLING AI FLIGHT PLANNER
- [3.](#) THE AI FLIGHT PLANNER MAIN WINDOW
 - [3.1](#) Airports
 - [3.2](#) Aircraft
 - [3.3](#) Flight Plans
- [4.](#) OTHER AI CONSIDERATIONS
 - [4.1](#) Traffic File Compilers
 - [4.2](#) Compiled-for-FS9 Traffic Files in FSX
 - [4.3](#) Day-of-Week Encoding
 - [4.4](#) The @ Symbol (User-Specified Arrival Times)
 - [4.5](#) The “37-Minute Problem” when Using @
- [5.](#) A GUIDED TOUR OF AI FLIGHT PLANNER’S MAIN WINDOW
- [6.](#) FLIGHT PLAN OPERATIONS
 - [6.1](#) Loading Flight Plan Data
 - [6.2](#) Saving/Compiling Flight Plan Data
 - [6.3](#) Validating Flight Plans and Traffic Files
 - [6.4](#) Finding FS9 traffic Files on FSX Systems
 - [6.5](#) Converting FS9 Flight Plans for Use with FSX
 - [6.6](#) Creating Flight Plan Subsets
 - [6.7](#) Displaying Arrival/Departure Information
 - [6.8](#) Adjust Flight Plans for Summer/Standard Time
- [7.](#) EDITING FLIGHT PLANS
 - [7.1](#) Arrival and Departure Times
 - [7.2](#) Flight Plan Editor
 - [7.3](#) Departure Airport in Consolidated Leg Mode
 - [7.4](#) New Flight Plans from “Scratch”
 - [7.5](#) Built-In Text Editor
 - [7.6](#) Find/Replace Functions
- [8.](#) MANAGING AIRCRAFT DATA
 - [8.1](#) Loading and Saving Aircraft Data
 - [8.2](#) Creating a New Aircraft List from Aircraft Folders
 - [8.3](#) Adding an Aircraft to the Aircraft List
 - [8.4](#) Editing an Aircraft Already in the Aircraft List
 - [8.5](#) Restoring Cruise Speeds
 - [8.6](#) Deleting Aircraft
 - [8.7](#) Finding Duplicate Aircraft
- [9.](#) AIRPORT AND RELATED DATA
 - [9.1](#) Displaying Airport Information
 - [9.2](#) Collecting Airport Data

9.3	Customizing the Airport List
9.4	Editing and Adding New Airports
9.5	Airport Data Bulk Update
9.6	Saving and Retrieving Lists of Selected Airports
9.7	Entering/Editing Time Zone Data
10.	SUPPORT
11.	ACKNOWLEDGEMENTS
11.	END USER LICENSE AGREEMENT

1. GENERAL

AI Flight Planner provides all the facilities necessary to create, develop, edit and maintain TTools-format AI traffic flight plans for both the FS9 and FSX versions of Microsoft Flight Simulator (MSFS). It includes an integrated compiler/decompiler. Hence, AI Flight Planner can use either TTools-format *.txt* files or compiled MSFS traffic *.bgl* files as “source code” for flight plans and associated aircraft data – or you may create one or both “from scratch”.

In addition to the features expected in any flight planning tool, AI Flight Planner:

- has a leg-oriented flight plan editor that can handle daily and weekly traffic in an identical manner;
- permits entry of arrival and departure times based on any time zone or using local times;
- permits designation of airports with either ICAO or IATA codes;
- displays at any time the arrivals and departures at any airport;
- generates traffic for a subset of the airports found in a flight plan file;
- removes traffic to/from specific airports from default traffic files; and
- allows data from either *.txt* files or compiled traffic files to be appended to previously-loaded flight plan and aircraft data;
- includes in the compiled traffic file only the airport and aircraft information used by the included flight plans, reducing system overhead;
- filters flight plans by airport, thus permitting the generation of efficient AI traffic files pertaining to only a region or single airport from multiple, airline-system-wide flight plan files; and
- finds FS9 traffic files and converts them for use with FSX.

If you are using AI Flight Planner with any of the many “prepared flight plan” packages, such as those from WoAI, MAIW or AIG Alpha-India Group, available from various download sites, you should restore the aircraft cruise speed to their usual values as described in Section 8.5 of this manual before compiling/re-compiling those files.

2. INSTALLING / UNINSTALLING AI FLIGHT PLANNER

AI Flight Planner is installed simply by copying the files from the downloaded archive to the desired destination folder. The complete set of files includes:

- *AI Flight Planner.exe*

- *AirportList_Base.dat*
- *Timezone_Base.dat*
- *ICAO-IATA.dat*
- *FS9Chg_AIFP.txt*
- *FS9Del_AIFP.txt*
- *TimeZones.txt*
- and several information files including *AI Flight Planner.pdf* and *AI Flight Planner.htm* (this manual in two formats).

The distributed archive file for some earlier versions of AI Flight Planner contained several additional *.dat* and *.exe* files which are no longer required.

AI Flight Planner does not affect the system registry.

AI Flight Planner is a VB NET.Framework 2.0 application. If NET.Framework 2.0 is not already installed on your computer, the “redistributable” can be downloaded from the Microsoft website at no charge.

Vista users need full administrative privileges to use some of the features of AI Flight Planner, in particular Collect Airports and Convert FS9 Traffic Files for FSX.

To uninstall AI Flight Planner, simply delete the folder in which it is installed.

3 MAIN AI COMPONENTS

A MSFS AI traffic file comprises three main databases:

- airports – the name, geographic position and elevation of the airports between which the AI operates
- aircraft – the identification and cruise speed of the aircraft involved, and
- flight plans – the schedule and other information controlling aircraft movement.

AI Flight Planner provides the “tools” to manage these three data sets.

It is important to note that most of this data is used only to determine the time at which each AI aircraft arrives at various points along its route, including when it should appear in the vicinity of the destination airport. The performance of an AI aircraft while in the vicinity of the user aircraft is always controlled by the data in the relevant aircraft file of the simulator. The location and other characteristics of airports are always determined from the information in the relevant stock airport or scenery file. And finally, even if the flight plan specifies arrival time, the actual arrival time of an AI is affected by aircraft performance, weather and other traffic (including the “user” aircraft).

- 3.1 Airports – Generally, AI Flight Planner derives the airport information required for the generation of traffic files from a file named *AirportList_Base.dat*. *AirportList_Base.dat* “shipped” with AI Flight Planner includes every stock airport in FS9 and FSX and records which airports exist in only one or the other MSFS version. *AirportList_Base.dat* is copied to a user file named *AirportList.dat* on initial start-up of AI Flight Planner. However, if a flight plan in an existing flight plan file refers to a non-stock airport, AI Flight Planner uses the positional data from the companion airport file.

To address new or modified airports included in scenery add-ons, AI Flight Planner provides:

- an editor to add/update individual airports,
- a bulk update facility which adds/updates airports from TTools-format airport files, and
- an airport collection facility that extracts airport data from add-on scenery files.

Any changes to airport data you make are saved to a file named *AirportList_Updates.dat* and re-applied should you re-build *AirportList.dat*.

If you have a state-of-the-art computer system, the size of *AirportList.dat* should not present a problem. However, if you use an older system or have limited RAM, you may experience unacceptable delays when accessing airport data. So, AI Flight Planner allows you to specify a subset of the available airports for inclusion in a custom airport list and use that list instead.

Using a “tree-view” style window, you can quickly find the ICAO and IATA codes and other details of any airport “known” to the system.

The data for the airports used in the currently loaded set of flight plans are held in the Airport List.

3.2 Aircraft – Like TTools, AI Flight Planner maintains an aircraft “database” for each set of flight plans. This aircraft information may be derived from:

- TTools-format aircraft *.txt* files,
- compiled traffic *.bgl* files, and/or
- MSFS aircraft files.

Data from all sources may be combined and used simultaneously.

Aircraft data may be edited using either AI Flight Planner’s custom editor or in TTools format using the default text editor on the host system.

The aircraft used by the currently loaded set of flight plans are held in the Aircraft List

3.3 Flight Plans – Flight plans may be:

- created from scratch,
- loaded from TTools-format flight plan files, or
- retrieved from compiled traffic files.

Multiple flight plan files, whatever their source, may be appended.

Individual flight plans may be edited with either:

- the leg-oriented editor, or
- the built-in plain-text editor.

A Find/replace function is also provided to allow similar changes to be applied to several flight plans at once. After editing of any kind, flight plans are error-checked for a variety of conditions

The set of flight plans currently being operated on are held in the Flight Plan List.

4. OTHER AI CONSIDERATIONS

There are several common misconceptions and other issues surrounding MSFS AI that it may be helpful to clarify at this point.

4.1 Traffic File Compilers – Three different AI flight plan compilers are in widespread use:

- TTools, a compiler/decompiler for FS9 and earlier versions of MSFS,
 - AIFPC, a TTools-compatible compiler/decompiler for FS9 and FSX (early versions of AI Flight Planner utilized AIFPC), and
 - Traffic DataBase Builder (TDBB), Microsoft's AI traffic generation utility for FS9 and FSX
- and now, AI Flight Planner.

There is a common misconception that flight plans for FS9 must be in TTools format and those for FSX must be developed using TDBB. And, often, you hear "FS9 flight plans can't be used with FSX". But, the reality is, TTools flight plans are compatible with FSX. And TDBB, to the surprise of many, was available in FS9.

The source data formats used by TDBB are very different from those used by TTools and AIFPC and are not discussed further.

4.2 Compiled-for-FS9 Traffic Files in FSX – FSX accepts and displays AI from traffic files compiled for FS9. However, if you attempt to mix TTools-format traffic files compiled for FS9 with traffic files compiled for FSX, the compiled-for-FSX traffic may be suppressed. This has nothing to do with the day-of-week encoding scheme discussed below. Rather, data formats of the two types of files are different and, apparently, incompatible. Surprisingly, FSX gives priority to the compiled-for-FS9 traffic files.

So, each compiled-for-FS9 traffic file on your system that potentially will be "seen" by FSX should be re-compiled for use with FSX – particularly if you wish to see the FSX default AI traffic.

To help you find those files needing to be recompiled, AI Flight Planner provides a capability to locate FS9 traffic files anywhere on your system.

AI Flight Planner requires you specify the target system (FS9/FSX) at compile time. To help you recognize which files have been compiled for FS9 and which for FSX, AI Flight Planner - if you wish - adds a "_FS9" or "_FSX" suffix to the compiled traffic *.bgl* file.

4.3 Day-of-Week Encoding – In FS9, the day-of-week encoding scheme for weekly flight plans uses 0 for Sunday, 1 for Monday and so on to 6 for Saturday. In FSX, Microsoft changed the day-encoding to 0 for Monday and 6 for Sunday. To

avoid potential confusion, AI Flight Planner uses the FS9 scheme irrespective of the target MSFS version, letting the compiler handle any required conversions. In any case, except in the Flight Plan List, day-encoding is transparent. AI Flight Planner's editor provides checkboxes for entry of day-of-week by name.

Should the day-encoding of a TTools flight plan file have previously been adjusted for FSX, the only consequence should that file be compiled for FS9 is that the AI will operate a day early. To avoid having to correct each flight plan in a file individually, AI Flight Planner provides the *Flight Plans / Adjust Day Encoding* menu item, which advances the day-encoding of all flight plans in the currently-loaded flight plan file one day at a time. (If you inadvertently adjust day-encoding or otherwise go too far, adjust another six times.)

- 4.4 The @ Symbol (User-Specified Arrival Times) –TTools allowed arrival times to be specified by the user by prefixing the time with a “@” symbol. Alternately, the arrival time could be left blank, in which case it would be computed by the compiler based on the distance to be travelled and the specified cruise speed of the aircraft used. There is no parallel in TDBB. With TDBB, arrival times are always calculated based on distance and speed.

For AI purposes, the earth is divided into 256 bands north-south and 384 bands east- west, for a total of nearly 100,000 sectors about 30 nm high and an average (depending on latitude) of about 40 nm wide. The arrival time calculated by both the TTools and TDBB compilers is used to “tell” the AI engine when an AI aircraft should arrive in the vicinity of the destination airport (i.e., at the boundary of the sector in which the destination airport resides) and at the boundary of every other sector along the way – based on the specified cruise speed of the aircraft (no allowance for climb/decent profiles or departure routing). Thus, the AI engine has the information necessary to position the aircraft all along the route so as to arrive approximately at the specified arrival time.

AI aircraft only ever materialize in the vicinity of the user aircraft. (Otherwise, processing the thousands of AI aircraft flights scheduled in the default traffic files would place an impossible burden on the system.)

The “@” symbol is not “understood” by the AI engine, so its use in TTools-format flight plan files involved some “trickery”. When arrival time was preceded by the “@” symbol, the TTools compiler subtracted 15 minutes - a notional amount to allow for approach, landing and taxiing - from the specified arrival time and used that value instead to determine sector-arrival times. For the purposes of the next section, please note those times are not related to the designated aircraft cruise speed.

It is important to recognize that a user-specified arrival time does not control the exact time of arrival of an AI aircraft at its parking spot, but rather, the time at which the AI aircraft “materializes” for approach – allowing 15 minutes for the balance of the flight.

- 4.5 The “37-Minute Problem” when Using @ – The foregoing discussion about calculation of time of AI arrival in the vicinity of the destination airport is

somewhat oversimplified. (Those wishing a more detailed explanation should consult the TTools manual and/or related forum discussions.)

Under certain, as yet not-fully understood, circumstances, the AI engine recalculates the sector boundary arrival times. But, this recalculation is always based on the specified cruise speed of the aircraft. Consequently, the arrival-in-the-neighborhood times may not agree with those calculated by the TTools compiler for user-specified arrival times, i.e., using the "@" symbol.

Unfortunately, if the arrival-in-the-neighborhood time specified in the traffic file is more than about 22 minutes later than the time recalculated by the "AI engine", the AI for that leg of the flight plan never materializes for landing. Instead, it spawns in a parking spot at the destination airport some time later in preparation for the next leg. Given the 15 minute offset applied by the compiler, the 22 minute interval equates to 37 minutes later than a user-specified arrival time. Hence, I've coined the name "37-minute problem".

The 37-minute problem is most likely to occur when simulating scheduled airline long-haul operations where the scheduled arrival time - even allowing for the 15 minute buffer - is often substantially later than the simply-calculated (distance/speed) arrival-in-the-neighborhood time.

AI Flight Planner's compiler avoids this problem by halving the aircraft cruise speeds saved in the traffic file, thus assuring that the arrival-in-the-neighborhood time that may be calculated by the AI engine is later than any reasonable user-specified arrival time. These cruise speeds are reconstructed when the traffic file is decompiled (which may result in a 1-knot change in the aircraft cruise speed shown in the Aircraft List). Other than this possible slight change in aircraft cruise speeds, there are no known side-effects of this scheme. (In earlier releases of AI Flight Planner, the solution to the "37-minute problem" resulted in AI departing from distant airports and scheduled to do TNGs arrived late. This is no longer an issue.)

Should you decompile an AI Flight Planner-generated traffic file other than with AI Flight Planner, you will find an unusually-named airport as the first entry in the airports.txt file. This is a "flag" used by AI Flight Planner to indicate that it has taken special measures to address the "37-minute problem" and to indicate that it must double the aircraft cruise speeds derived from companion aircraft data.

There may be situations where you wish the compiled traffic file to reflect exactly the specified aircraft cruise speeds. In such cases, AI Flight Planner's "raw" compile mode should be used.

5. A GUIDED TOUR OF AI FLIGHT PLANNER'S MAIN WINDOW

AI Flight Planner's main window is partitioned as follows.

- The main menu bar is in its traditional position across the top of the window. Used to initiate functions not directly associated with the editing of flight plans, it comprises four sections: *Flight Plans*, *Aircraft*, *Airports* and *Help*; in this manual, references to menu selections are in the form "Menu Name / Item name".

- The left-hand side of the main window holds the *Flight Plan List* - a list box displaying all loaded flight plans - and related buttons and search functions. Each item in the *Flight Plan List* is identified with a line number.
- To the right of the *Flight Plan List*, towards the bottom of the window, is the *Airport List*, a list of all the airports referenced by the flight plans in the *Flight Plan List* and in the editor. Immediately above the *Airport List* is a checkbox entitled "Use IATA Codes". When checked, you may use IATA codes to identify airports.
- The top-center portion of the window is occupied by the *Time Zone Selector*. Flight plan times may be displayed and edited in either local time or a selected time zone. However, for local times to be used, geographic information (city, country, position) on all the airports involved in the flight plan must be included in AI Flight Planner's internal airport list.
- The upper right-hand corner is devoted to the *Aircraft List*, which includes the available aircraft and permits selection of the aircraft to be used in the currently-selected flight plan. The *Aircraft List* includes a field for the entry of the aircraft's tail (registration) number.
- The remainder of the screen is consumed by the *Flight Plan Editor*. Across the top of that area is the *Flight Plan Base Data*, which displays and allows editing of the flight plan data common to all the legs of a flight plan. Below that is the leg editing area and below that again, the *Leg List*

Whenever a change is made to the *Flight Plan List*, the *Aircraft List* or the *Leg List*, a red ### indicator appears to the right of the list name to remind you to save the list.

As a general rule, only those menu items, buttons and data entry fields valid for use in the current context of AI Flight Planner are enabled. In particular, all the fields of the editor are disabled unless a flight plan is being edited. All the fields of the *Flight Plan Base Data* must be completed before the fields and buttons applicable to the *Leg List Editor* are enabled. All leg fields except *Override ETA* (which may be left blank) must be completed before the buttons involved in placing the edited leg data (back) into the *Leg List* are enabled. Buttons pertaining to the insertion of data into or deletion of data from any list are enabled only when an item in the relevant list is selected. And, finally, the buttons for moving a completed flight plan from the editor (back) into the *Flight Plan List* are enabled only when there has been a change to the leg data and there are at least two legs in the *Leg List* – the minimum for a valid flight plan – unless touch-'n-go operation is specified.

So, at any time, if a button, menu item or field data-entry field is disabled, it is because that button, menu item or field is not useable in the current context of AI Flight Planner – probably because some required data is missing.

Many buttons and menu items have keyboard shortcuts. If you don't see the shortcuts, it is because Windows is preventing them from being displayed until you press the <Alt> key. You may override this feature of Windows using Control Panel. In Windows XP, the control to do so is found at Display – Appearance – Effects. In other versions of Windows it may be elsewhere.

6.0 FLIGHT PLAN OPERATIONS

This section describes the main flight plan file-handling operations.

6.1 Loading Flight Plan Data – Flight plan data may be loaded from any of three sources:

- TTools-format *.txt* files - click *Flight Plans / Open File Set* and specify the desired flight plan file; if there a companion aircraft file, it also is loaded;
- compiled MSFS traffic *.bgl* files - click *Flight Plans / Open Traffic .bgl* and specify the traffic file of interest; the traffic file is decompiled and both flight plan and aircraft data is loaded; and
- reload the last loaded or saved file – click *Flight Plans / Reload Last File (Set)*, which discards all changes since the last load or save/compile operation. The last loaded file data is preserved across AI Flight Planner sessions.

The flight plans are saved to the *Flight Plan List* after the associated aircraft data (if any) has been loaded into the *Aircraft List*. Since AI Flight Planner maintains its own airport database (*AirportList.dat*), companion airport data is not routinely used by AI Flight Planner. However, if a flight plan refers to an airport not in *AirportList.dat*, AI Flight Planner looks for that airport in the companion file. If found, it saves the related data.

At the completion of the file loading activity, a summary of any missing airports or aircraft and any other errors is provided – in printable form if more than a single error is detected. In the case of missing airports, you are asked if you wish to update *AirportList.dat*. If you elect to do so, the airport editor opens with all available data (which may be only the ICAO code) shown.

FS9 and FSX flight plans in *.txt* format are identical save for day-of week encoding. As discussed earlier, in FS9, Sunday is encoded as 0, while FSX uses 0 for Monday. When the flight plan data comes from compiled traffic files, AI Flight Planner “knows” which encoding scheme is used. However, there is no way for AI Flight Planner to determine automatically the day-encoding scheme used in TTools-format flight plan files and it issues a message advising you of the problem. AI Flight Planner always uses the FS9 day-encoding scheme. However, it provides a mechanism (*Flight Plans / Adjust Day Encoding* menu item) to adjust the day-encoding in the flight plan should it be for FSX after opening a TTools-format flight plan file.

Previously-loaded flight plan and aircraft data may be supplemented at any time by appending data from additional files using the applicable Main Menu function. Appended data may be derived from either TTools-format files or compiled traffic files, irrespective of the source of the previously loaded data. Before appending TTools-format flight plan files, if it cannot determine the day-encoding scheme of the file to be appended, AI Flight Planner asks for confirmation that the current day-encoding is for FS9. If it's not, it will be necessary to load the second file by itself and adjust its day-encoding before it can be appended.

Please be aware that the times in the flight plan information returned by the *Flight Plans / Open Traffic .bgl* function may not match exactly the times originally specified by you in the flight plan file. This is because of the way times are stored in the traffic file. The “consolidated” mode of operation of the Leg List may

be affected (even though AI Flight Planner allows a 2-minute “window”). This problem is less of an issue with FSX which uses a more precise traffic file storage format. As well, the solution to the “37-minute problem” may result in aircraft cruise speed returned by the de-compiler being in error by 1 knot.

6.2 Saving/Compiling Flight Plan Data – Flight plan data may be saved either:

- in TTools-format plain text (.txt) format – click *Flight Plans / Save* (or *Save As* if you wish to specify a new file name), or
- as a compiled traffic (.bgl) file – click *Flight Plans / Compile Traffic .bgl*, *Flight Plans / Compile Raw* or *Flight Plans / Compile Again*.

In all cases, the full flight plan file is first checked for errors. If errors are found, you are asked if you wish to save anyway. Please note that if you elect to compile with errors, depending on the nature of the error, the compile operation may not succeed, in which case you must either correct the error or save in plain-text format.

For file saves, if there are any unused aircraft in the Aircraft List, you’ll be asked if you want to save them. For compiles, the unused airports are simply discarded.

AI Flight Planner has two compile modes:

- “normal”, where the data is pre-processed to avoid the “37-minute” problem identified earlier, and
- “raw”, where the data is passed to the compiler exactly as entered.

Unless you have reason to do otherwise, it is recommended you use the “normal” compile mode.

Once error checking has been performed, the compile dialog is opened – unless you selected *Compile Again*. In the compile dialog window you may select the filename under which the data is to be saved (which must start with “Traffic”), MSFS version for which the file is to be compiled and whether or not a file name suffix identifying the target version is to be added to the file name.

If you did select *Compile Again*, the data is submitted directly to the compiler using the same compile parameters as for the previous compile without any opportunity to make changes.

If you have checked the “Identify Non-Stock Airports on Compile” item in the Airports menu, prior to compiling, AI Flight Planner will warn you if any non-stock airports are referenced by the flight plans.

6.3 Validating Flight Plans and Traffic Files – The integrity of every flight plan is automatically checked when a flight plan is loaded into AI Flight Planner and prior to the flight plan being saved or compiled. It is also verified whenever an edited flight plan is returned to the Flight Plan List. But, for convenience, AI Flight Planner permits flight plans containing errors to be saved. Consequently, you may wish to validate the flight plans in the Flight Plan List at other times. To do so, simply click *Flight Plans – Validate FP List*

You may also validate one or more traffic files without loading them into AI Flight Planner by clicking the *Flight Plans / Validate Traffic Files* menu item. This opens a directory-tree from which you may select any combination of folders and files for validation. Folders known not to contain traffic files, while still shown, are dimmed. Folders to which you do not have access privileges are not shown at all.

In both cases, a summary of all errors found – in printable form if more than one – is generated.

AI Flight Planner's validation facility detects a number of situations that, while they won't "crash" the simulator, would cause AI not to behave as intended. Corresponding warning messages are output each time the validation routine runs. This may become irritating, especially when dealing with large files. So, there is an option in the Flight Plans menu to suppress these warnings. Messages relating to serious errors are, however, always output. To avoid warning messages from being suppressed without your being aware of it, this suppression feature must be activated each time AI Flight Planner is run.

- 6.4 Finding FS9 Traffic Files on a FSX System – When FSX "sees" both compiled-for-FS9 traffic files and compiled-for-FSX traffic files, the FS9 traffic will be displayed normally; however, the FSX traffic will be suppressed. This situation will exist until the very last FS9 traffic file has been located and either disabled or converted for FSX operation (see next section).

To locate FS9 traffic files, click on the *Flight Plans – Find FS9 Files* menu item. As in the previous section, this opens a directory-tree from which you may select any combination of folders and files to be searched. Folders known not to contain FS9 traffic files, while still shown, are dimmed. Folders to which you do not have access privileges are not shown at all.

Select the folders of interest and click the Find Files button. You could select entire logical disks, but the search time would be excessive.

- 6.5 Converting FS9 Flight Plans for Use with FSX – Conversion of FS9 flight plan and traffic files for use with FSX addresses two main attributes:

- Airports – The ICAO code designator of many FS9 airports changed between FS9 and FSX. AI Flight Planner updates the designators for those airports to their FSX equivalents. If an airport does not exist in FSX and there is no equivalent, an error message is issued.
- Day-Encoding – updated to the FSX scheme.

The conversion function, which is sometimes referred to as "bulk-conversion", allows you to select any combination of folders and files for conversion using a "tree-view" directory-tree. Folders known not to contain FS9 traffic files, while still shown, are dimmed. Folders to which you do not have access privileges are not shown at all. Selected folders may contain a mix of FSX and FS9 traffic files; only the FS9 files are affected. The names of the converted files are suffixed with "_FSX" and backed-up if they exist already.

Following conversion, any missing airports or other problems encountered are noted in a conversion report.

- 6.6 Creating Flight Plan Subsets – AI Flight Planner allows the creation of subsets of the flight plans currently in the Flight Plan List based on the airports selected in the *Airport List*. Subsets may be either inclusive or exclusive. An inclusive subset includes each flight plan that references any airport selected in the *Airport List*. An exclusive subset includes every flight plan that does not reference those airports.

To create a subset, select the airports of interest in the Airport List and click on one of:

- *Flight Plans / Create Subset (Inclusive)*
- *Flight Plans / Create Subset (Exclusive)*

For both items, you are given an opportunity to specify whether or not a subset of the Aircraft List also is to be generated and if comments are to be preserved.

Among other things, this feature allows the generation of regional flight plans (inclusive option) and the exclusion of designated airports from the default traffic files (exclusive option).

- 6.7 Displaying Arrival/Departure Information – A detailed listing of arrivals and departures at any airport contained in the Flight Plan List may be obtained by clicking on *Airports / Arrivals/Departures*. This opens the Arrivals/Departures dialog which lists all the airports referenced by the flight plans – essentially a duplicate of the Airport List. Click on any airport in the list and all the arrivals and departures at that airport are displayed in time sequence.

A hard-copy of the arrivals and departures information can be obtained by clicking on the “dump to Notepad” button and using Notepad’s Print function.

Should you wish to modify any arrival or departure, double-click on it. The Arrivals/Departures dialog closes and the corresponding flight plan is moved to the editor.

- 6.8 Adjust Flight Plans for Summer/Standard Time – Flight plan arrival and departure times are always saved using the UTC equivalents – even if originally specified or edited using local times. Consequently, at airports where daylight savings time is observed, the AI at those airports operates one hour early or late in some seasons.

Rather than requiring re-specification of arrival and departure times to allow for correct operation during the summer season, AI Flight Planner’s menu item *Flight Plans / Advance to Summer Time* advances all those times by one hour except in the case of airports where it is known that daylight savings time is not observed. This simple operation followed by a re-compile of the file results in proper summertime operations.

But, of course, such flight plans will then operate one hour late during the winter. *Flight Plans / Retard to Standard Time* to the rescue! It reverses the effect of the *Advance to Summer Time* feature

7.0 EDITING FLIGHT PLANS

AI Flight Planner offers three alternatives for editing flight plan data:

- the leg editor, which allows editing on a per-leg basis (double-click on the flight plan in the Flight Plan List)
- the built-in text editor, which allows plain-text editing of a flight plan in TTools-like text format (select the flight plan in the Flight Plan List and click on the Open Text Editor button), and
- the Find/Replace function (edits are made to the flight plans *in situ*)

Use of each is described below.

- 7.1 Arrival and Departure Times – Arrival and departure times in all three editors may be specified in either UTC, a time zone selected in the Time Zone combo box or, provided *AIrportList.dat* contains full geographic information for all the airports used in the Flight Plan File, local time - which may be either standard or daylight savings time. However, flight plans are not date-specific, so arrivals and departures specified in local time may be in error by an hour in the transition periods to and from daylight savings time (where applicable).

When using local time, you should appreciate that AI Flight Planner's time zone database reflects real-world time zones – which may not match exactly the time zone calculated by MSFS. MSFS uses a geographic approximation technique to determine the time zone in the area where the user aircraft is located. Hence, there may be disagreement at airports close to the edge of time zones. Since MSFS' calculation of time zones can be affected by add-ons, such discrepancies cannot always be resolved by AI Flight Planner. As well, while significant efforts were expended in making AI Flight Planner's time zone data base complete and accurate, time zone data for smaller airports, especially in developing countries, sometimes is not readily available and, hence, may be in error (but, is correctable using the time zone database editor).

Unlike TTools, AI Flight Planner does not use “@” and “TNG” as prefixes for arrival times. Touch 'n go operation is specified using a checkbox. The function performed by the “@” symbol is automatic with AI Flight Planner. (Any arrival time that differs by more than two minutes from the system-calculated arrival time is assumed to have been specified by you. This tolerance is necessary to accommodate “jitter” introduced by de-compilation.)

Also unlike TTools, all arrival times (not just user-specified ones) reflect nominal arrival at parking. While the actual arrival times will depend on weather, traffic, aircraft performance, etc, the AI Flight Planner compiler makes a 15-minute allowance for approach, landing and taxiing in all cases.

AI Flight Planner determines whether the specified arrival time is for the day (repeat period) of departure or the following day (repeat period) and applies an

appropriate suffix where appropriate. (Any arrival time in 24 hour format that is earlier than departure time is assumed to refer to the following day (repeat period). The suffix indicating a following day/after midnight arrival is “+1” – a notation used in many airline schedules. Where the flight crosses the International Date Line, the suffix may also be “-1” (eastbound flights leaving Asia just after midnight) or “+2” (westbound flight leaving North America just before midnight). You need not enter these suffixes; AI Flight Planner applies them where appropriate. But, if you do, your entry is used.

When a sub-daily repeat period is selected, the hour value of the arrival and departure time entries must be less than the repeat period. For example, for a repeat period of 4 hours, the maximum acceptable arrival or departure time is 03:59. When the repeat period is changed to a smaller value such that previously-entered arrival and departure times in the editor are invalid, the arrival time and predicted ETA suffixes may become what appears to be nonsensical. Such a situation may not be detected and an error message issued until an attempt is made to save the flight plan.

In weekly consolidate mode, the departure times displayed in the *Leg List* are in “day-time” configuration, i.e., d/HH:MM. This is to maintain proper sequencing of the entries. Departure time entries in the editor are always in HH:MM; AI Flight Planner derives its day-of-week information from the day-of-week checkboxes in the *Base Data* area.

- 7.2 Flight Plan Editor – A flight plan in the Flight Plan File is moved into the editor by double clicking on it (the flight plan). The *Aircraft List* combo box highlights the aircraft used by the flight plan (if it is in the *Aircraft List*), the other base data of the flight plan is displayed in the base data area across the top of the main window, the flight plan legs are inserted into the *Leg List* and, lastly, the top item in the *Leg List* is selected, parsed and displayed in the editor.

Flight plan data editing is straightforward. (Nonetheless, if you are not familiar with the individual flight plan fields, it is strongly recommended that you download Lee Swordy's TTools and refer to its user documentation.)

Each leg of the flight plan is displayed in the Leg List, time sequenced. As noted earlier, for weekly flight plans, a leg which is flown on two or more days may be shown in either of two ways:

- “individual”, i.e., each leg in the flight plan shown individually, or
- “consolidated”, i.e., legs that operate on more than one day of the week but are otherwise identical consolidated into a single list item.

In “individual” mode, the legs must be entered in operational sequence. The destination airport of one leg becomes the departure airport for the next, with the destination airport of the last leg being the departure airport for the first. In “consolidated” mode, the position in the *Leg List* at which a new leg is entered determines the departure airport for that leg - for the purpose of calculating distance, duration and ETA. When the data for a new leg is entered into the Leg List or when edited data is saved, AI Flight Planner automatically positions the edited leg based on departure time. The *Consolidate Weekly FPs* checkbox located in the bottom right-hand corner of the window controls this mode.

An individual leg is moved into the leg editor by double-clicking on it in the *Leg List*. Leg data in the editor is displayed in a combination of text boxes, check boxes and radio buttons. In general, error checking, where applicable, is performed when you move the cursor away from an edited text box. Error checking of the leg as a whole is performed when the edited data is saved.

When a leg is loaded into the editor, AI Flight Planner attempts to identify the departure airport and calculate the distance between it and the destination airport, as well as the expected duration of the flight (based on the cruising speed of the selected aircraft) and anticipated arrival time. Of course, if AI Flight Planner picks the wrong departure airport in a complex weekly flight plan (see next section), these calculations are of little use. Should that be the case, you should enter the proper departure airport for the leg in the editor. (If this problem occurs at all, it likely will occur every time a leg is moved into the editor. Please be tolerant.)

While every airport in MSFS has an ICAO identifier, only large airports that handle commercial flights are likely to have IATA designators. To allow broader use of the "Use IATA codes" mode, AI Flight Planner allows entry of ICAO codes when in the IATA mode by prefacing ICAO codes with "**".

When you enter or edit an airport designator (either IATA or ICAO code), the newly-entered code is validated as soon as you move the cursor to another field. If the code is valid, the airport is entered into the *Airport List* if it is not already there and the name of the city shown. If invalid, an error message is issued and you are given an opportunity to enter the airport into the system.

If unsure of the ICAO/IATA code for the intended airport you may enter "?" optionally preceded by a character string. If:

- no character string is entered, i.e., just "?", the Get Airport Information dialog box is opened; locate the ICAO code of interest, select (click on) it and close the dialog box;
- the character string contains ">", a list of ICAO codes where the city name or airport name contains the entered string pops-up; or
- otherwise, a list of ICAO codes which start with the entered string pops-up.

In the latter two cases, double-click on the airport of interest in the pop-up list. This closes the list and places the selected ICAO or IATA code in the relevant airport field in the editor.

When editing an existing flight plan file, should the ICAO code of an airport that is not known to the system be entered but the airport is contained in the companion TTools-format airports file, you may direct AI Flight Planner to use the data from the airports file. You are also given the opportunity to add the airport to *AirportList.dat* using the Airport Editor.

An AI aircraft is known to MSFS Air Traffic Control (ATC) by either its flight number or its registration number. If the *ATC Callsign FN* radio-button is selected, a flight number must be entered for the leg. If the *Reg* button is

selected, the aircraft associated with the flight plan must have a tail (registration) number.

When all intended edits have been applied to the leg, use the *Save Edited Leg Data* button to update the item to which the edits apply in the Leg List. The original copy of the leg in the *Leg List* is updated, irrespective of which leg might be selected when the *Save Edited Leg Data* button is clicked.

New legs may be inserted into or added to the *Leg List* as necessary. To insert a new leg, select the leg in the *Leg List* above which the new leg is to be inserted and click on the *Insert Leg* button. A blank leg is inserted into the *Leg List* at that point as a "place-holder". To add a new leg at the bottom of the *Leg List*, click on the *Add Leg* button. Similar to *Insert Leg*, a blank leg is appended to the leg list and the departure airport set. Please note that the *Save Edited Leg Data* button is not enabled until all fields except arrival time have been entered.

When a new leg is added to/inserted in a weekly flight plan, there is no day of the week information available. So initially, AI Flight Planner assumes that the leg originates at the destination of the leg immediately above it in the *Leg List*. However, once the days of operation are specified, AI Flight Planner attempts to refine its earlier choice by selecting a departure airport from an earlier leg that operates on the same day(s).

When all edits have been made to a flight plan, the edited flight plan may update the original copy of the flight plan in the *Flight Plan List* or be added to the bottom of, or inserted immediately above the selected flight plan in, the list using the *Update FP in List*, *Insert FP in List* or *Add FP to List* button, as applicable.

- 7.3. Departure Airport in Consolidated Leg Mode - A flight plan leg specifies only the destination airport. The departure airport for any leg is the destination of the previous leg in time. For the first leg of a flight plan, the originating airport is the destination of the final leg – with one exception. If, when starting a new flight plan, you enter a departure airport for the first leg, AI Flight Planner continues to report that airport as the departure airport for the first leg irrespective of the following legs. When it comes time to save the flight plan, if the destination of the final leg does not match this airport, AI Flight Planner alerts you. You then have the option to either save the flight plan as it exists, with the destination airport of the last leg becoming the departure airport for the first leg, or to further edit the flight plan.

Otherwise, AI Flight Planner uses the departure airport (solely) to calculate the distance and duration of the flight specified in the leg and to predict the arrival time. When a flight plan leg is moved into the editor, AI Flight Planner attempts to determine the airport from which that leg departs. For a daily (or more frequent) flight plan, this is a simple task – it is the destination of the leg immediately previous in the *Leg List* to the leg of interest. However, when in the consolidated display mode, the previous leg (in a weekly flight plan) is not necessarily the leg immediately previous in the *Leg List*.)

Consider, for example, the simple case of a flight plan that operates between airports A and B from Monday to Friday but makes an intermediate stop at airport

C on Wednesday. Such a situation requires three legs to be specified; A to B on M/T/T/F and A to C and then C to B on Wednesday. In the *Leg List*, the sequence of the legs is A to B, A to C and then C to B. So, there are two intervening legs between the A to B leg and the ongoing leg from B. To accommodate such situations, AI Flight Planner looks back up the *Leg List* to find an earlier flight plan scheduled for the same day of the week and assumes that to be the departure airport.

However, depending on the complexity of the flight plan, AI Flight Planner may not correctly identify the departure airport. (This is a trade-off for being able to specify a leg only once irrespective of how many days of the week it operates, and is of little consequence, since the departure airport is only used to calculate the distance and duration and to predict the ETA of the flight.) Should AI Flight Planner not determine the correct departure airport, you may override its selection to allow the calculation of distance, duration and ETA for the leg. This action has no effect on the saved/compiled flight plan, which always reflects the leg sequence.

- 7.4 New Flight Plans from “Scratch” – To start a new flight plan “from scratch”, click on the *Start New FP* button. This clears the *Leg List* if there is any data in it, places a blank entry in the *Leg List* and also clears the editor fields. Initially, only the fields for the base data are enabled. Once all the base data for the flight plan has been entered and the aircraft selected, the *Leg List Editor* fields/controls are enabled. This sequence ensures that all required data is available when needed. In the *Leg List* editor, all fields must be completed except for Override ETA, which may be left blank. When the required data for the first leg has been entered, save it to the *Leg List* using the *Save Edited Leg Data* button. Create additional legs as necessary using either the *Add Leg* or *Insert Leg* buttons. (In all cases, please note that the *Save Edited Leg Data* button is not enabled until all fields except arrival time have been entered.)

Before attempting to enter flight plan leg data, it is recommended to confirm that the associated aircraft exist in the Aircraft List. (The leg editor is not enabled until an aircraft has been selected.)

Once all the legs of the new flight plan have been entered, save the flight plan to the *Flight Plan List* using the *Add to FP List* or *Insert in FP List* buttons as appropriate. These buttons are not enabled unless there are sufficient legs in the *Leg List*, i.e., two for all but TNG operation.

- 7.5 Built-In Text Editor – TTools formatting supports embedded comments. AI Flight Planner provides a simple text editor to allow insertion of comments into, and editing of comments already in, the *Flight Plan List*.

To open the text editor, double click on a comment line in the *Flight Plan List*. Alternately, select an item in the *Flight Plan List* and click the *Open Text Editor* button. The text editor supports both single-line and multi-line comments. Each line in a multi-line comment should be terminated using the keyboard <Enter> key and the succeeding line commence with either “,” or “//”.

This text editor also supports editing of flight plans in a TTools-like format. For simple changes to flight plans, it may be more convenient to use the text editor rather than the flight plan editor. To edit a flight plan in the text editor, select it in the *Flight Plan List* and click the *Open Text Editor* button. The data is formatted to make each leg readily identifiable. Should edits destroy this clarity, it may be restored with the *Format* button.

In addition to replacing the *Flight Plan List* item selected when the text editor was opened, the contents of the text editor may be added at the end of the *Flight Plan List* or inserted immediately above the selected item. Full validation of flight plan edited in the text editor is performed when an attempt is made to place it back into the Flight Plan List.

- 7.6 Find/Replace Functions – AI Flight Planner's flight plan Find/Replace function allows similar changes to be applied to several flight plans. Select the field of interest in the Field combo box and the "find mode" as: less than (<) equal to (=) or greater than(>). Then enter the value to be found/replaced in the Find text box.

To select the top-most flight plan meeting this criteria, click on the "Find" button. Subsequent flight plans may be selected using the "Next" button. To select all flight plans meeting the criteria, click on the "All" button.

If the value in the designated field is to be replaced, enter the replacement value in the "Replace with" textbox and click on the Replace button after selecting the flight plan in which the field is to be replaced, or on Replace All if all occurrences are to be replaced.

The Replace function may also be used independently of the Find function, by selecting the field of interest, entering the replacement value and manually selecting the flight plan(s) where the replacement is to occur prior to clicking on the "Replace" button.

8.0 MANAGING AIRCRAFT DATA

AI Flight Planner does not restrict you to using aircraft data associated-by-name with the flight plan data ("companion" file). When a TTools-format flight plan file or a compiled traffic file is loaded into AI Flight Planner, "companion" aircraft data (if it exists) is also loaded. However, you may replace this data or supplement it with data from other sources.

- 8.1 Loading and Saving Aircraft Data – Loading and saving of aircraft information is handled in an identical manner to flight plans. The corresponding items in the *Aircraft* main menu are:
- *Load New Aircraft File*
 - *Append Aircraft File*
 - *Save Aircraft File*
 - *Save As ...*

- 8.2 Creating a New Aircraft List from Aircraft Folders – Creation of a new *Aircraft List* including some or all of the aircraft in FS9 and/or FSX aircraft files is initiated from the *Create Aircraft List* menu item. Once the desired aircraft folders have been selected, an initial list including all aircraft in the selected folders is created. That list is in alphabetical order. You then have the opportunity to “massage” that list by deleting items from it and moving items up or down. The *Save New Aircraft List* button copies the temporary list to the main Aircraft List, assigning aircraft reference numbers sequentially, replacing the previous contents.

So, if you’ve made changes to the Aircraft List, you should save it before creating a new one.

- 8.3 Adding New Aircraft to the Aircraft List – Individual aircraft may be added to the Aircraft List by clicking on the *Aircraft / Add Aircraft* menu item. This opens the Aircraft Editor dialog box and places the next highest aircraft reference number in the *AC#* text box. However, you may assign any other unused reference number. The aircraft to be added may be specified in any of three ways:

- Specify a FS9 or FSX aircraft folder using *Aircraft Folder Select*. The titles of all the aircraft variants available from that folder are listed in the *Aircraft Title* combo box. The *Cruise Speed* text box is filled in automatically with the cruising speed specified in the relevant *aircraft.cfg* file, but it may be edited. If *Cruise Speed* is edited at this time, the cruising speed in the *aircraft.cfg* file is updated and, hence, will affect the performance of the aircraft in the simulator. Then to enter:
 - one aircraft from the list, select the title of the desired aircraft in the list and click on the Add button, or
 - all the aircraft in the list, just click on the Add All button.
- Enter the title of the new aircraft (which must be unique in the *Aircraft List*) into the *Aircraft Title* text/combo box and a cruising speed into the *Cruise Speed* text box. Once all three fields have been entered, the *Add* button is enabled. Click it to add the new aircraft to the Aircraft List.

The dialog box remains open until you close it.

- 8.4 Editing an Aircraft Already in the Aircraft List – To edit an aircraft in the Aircraft List, select the aircraft and click on the *Aircraft / Modify Selected Aircraft* menu item. This action opens the Aircraft Editor dialog box.

Edit the *Aircraft Name*, *AC#* and/or *Cruise Speed* fields as necessary, bearing in mind that if either the title or reference number is changed, the new value must be unique in the *Aircraft List*. When ready, click on the *Replace* button. Please note that, in this case, a change to *Cruise Speed* is not recorded in the *aircraft.cfg* file (since all association with the aircraft folder from which the aircraft may have been added has been lost). The dialog box is automatically closed when the replacement is accepted. If you change the *AC#*, AI Flight Planner offers to update all references to the old number.

- 8.5 Restoring Cruise Speeds – As noted in Section 4, if an AI aircraft is scheduled to arrive more than about 37 minutes later than it would based on distance/cruising speed (that specified in the *aircraft...txt* file), it does not materialize for landing.

Instead, it spawns in a parking spot at the destination airport prior to departure for the next leg. To address this problem, some suppliers of complete AI add-on packages, such as World of AI (WoAI), and of AI flight plans, such as AIG Alpha-India Group, specify a cruise speed of 200 kts for all aircraft.

This artificial cruising speed is problematic when used in conjunction with AI Flight Planner.

- For jet passenger aircraft, the calculated duration of each flight plan leg is much longer than (2-3 times) the real-world value and, consequently, the calculated ETA is very late. Hence, you'll have to specify arrival time in all cases.
- As discussed in Section 4, AI Flight Planner has its own solution to the 37-minute problem (i.e., halving the specified cruise speed). This results in a cruising speed in the traffic file being further reduced to only 100kts. Fortunately, with the exception noted in section 4, this does not materially affect AI arrival times – but you must override system-calculated arrival time.

So, when using such prepared flight plan information, you should restore the aircraft cruise speed to the value in the relevant *aircraft.cfg* file using the *Aircraft / Restore Cruise Speeds* menu item. This menu item allows you to select the aircraft folders to be used for this restoration.

- 8.6 Deleting Aircraft – To delete an aircraft from the *Aircraft List*, simply select the aircraft and click on the *Aircraft / Delete Selected Aircraft* menu item. All aircraft not used by the flight plans in the Flight Plan List may be deleted using the *Aircraft / Delete Unused Aircraft* menu item.
- 8.7 Finding Duplicate Aircraft – From time to time as you add more AI, you may find that a given flight plan has been duplicated in another traffic file. To find which one, select the aircraft of interest in the Aircraft List and click on the *Aircraft / Find Aircraft in Traffic File* menu item. You'll be presented with a directory tree on which you may select the drives/folders to be searched. Then click on the Find Aircraft button. At the completion of the search, a list of the traffic files that use the selected aircraft will be displayed.

9.0 AIRPORT AND RELATED DATA

AI Flight Planner uses its own internal airport information – even if there is companion airport data for the flight plan file(s) loaded.

For your convenience, AI Flight Planner “ships” with:

- the data for both FS9 and FSX stock airports already collected in a file named *AirportList_Base.dat*,
- time zone information (UTC offsets) for every country and region where a stock airport exists in a file named *Timezone_Base.dat*.

For new installations, AI Flight Planner automatically creates *AirportList.dat* and *Timezone.dat* from these files.

- 9.1 Displaying Airport Information – Information on any airport known to the system is but a few mouse-clicks away. Clicking on the *Airports / Get Airport Information*

menu item displays a list of countries for which airport data exists in tree-view form. “Expanding” a country results in the constituent state/provinces/regions (if any) or a list of cities to appear. Expanding a city shows all its airports, including IACO code, IATA code if it has one, position and UTC offsets.

- 9.2 Collecting Airport Data - AI Flight Planner includes a facility to collect data for FS9 and or FSX stock airports, as well as add-on airports for both. The collected airport data is displayed in a standard list box and saved to *AirportList.dat* under user control.

The scope of the collection process is controlled by the four checkboxes located beneath the top-level folder path displays. Data may be collected for any combination of FS9 stock airports, FSX stock airports and add-on airports for either version.

For stock airport data, you may specify use of *AirportList_Base.dat* (check *Use provided base data* checkbox) or have AI Flight Planner collect the stock airports from your system. You would use the latter alternative if, for example, you had modified your default airport data.

When you indicate that add-on data also is to be collected, AI Flight Planner displays a directory-tree from which you may select the add-on folders of interest. (Only the folders to which your Windows operating system gives you access are shown.) Please note that only those add-on airports that have at least one runway or helipad are collected. (Depending on the approach used by the developer, there may be several instances of an updated airport in the add-on) If you want the positional data (latitude, longitude and elevation) of your add-on airports to replace that of the corresponding stock airports, check the *Update from Add-on* checkbox.

The airport data collection function attempts to locate the FS9 and FSX top folders using registry entries. If it is successful, the path(s) are displayed at the top of the dialog box and the associated stock and add-on airport checkboxes enabled. If FS9 or FSX exists on your computer and is not located automatically, you may indicate the relevant top folder using the *Select* buttons. The FS9 and/or FSX checkboxes and applicable, checkboxes remain disabled until the path to the corresponding top folder is known.

Each time stock airport data is collected, any updates you have made (saved in the file *AirportList – Updates.dat*) are re-applied.

Because of the extent of control over the collection process, this airport data collection feature may find other uses. For example, if you wish to know which airports are installed in a given add-on folder, the airport collector will answer the question. Hence, airport data for purposes other than updating *AirportList.dat* may be explored. Data is not copied to *AirportList.dat* until you click on the *Save “AirportList.dat”* button, so you may safely experiment.

- 9.3 Customizing the Airport List – The airport data included with AI Flight Planner includes every stock airport in both FS9 and FSX – which number over 25,000. Thus, the file *AirportList.dat* is huge – over 2mb. For those with state-of-the-art

computer systems, this should not present a problem. However, if you have an older computers or limited RAM, you may wish to work with only a subset of the available airports.

To generate a custom airport list - which is selected automatically on subsequent start-ups of AI Flight Planner unless you direct otherwise:

- click on the *Airport / Customize Airport List* menu item,
- select the airports/cities/regions/countries of interest in the “tree-view” display, and
- click on the *Generate Subset* button.

The newly generated airport data file is named *AirportList-Custom.dat*. If custom airport data already exists, you are warned before overwriting it.

Generally, if you have created a custom internal airport list, AI Flight Planner uses this custom list as its source of airport information. But, you may revert to the master *AircraftList.dat* at any time by clicking on the *Airports / Use Master Airport List* menu item.

- 9.4 Editing and Adding New Airports – Errors have been noted in MSFS airport data. As well, the original airport data may not match that contained in add-on scenery installed on your system. While small changes are unlikely to have any noticeable effect on AI operation, you may wish to have *AirportList.dat* reflect the corrected data. As well, since AI Flight Planner cannot compile a flight plan file containing an airport for which it does have positional information, any new airport created in add-on scenery must be added to *AirportList.dat* (and its custom counterpart, if used) before AI can be programmed for that airport.

To modify the data for any airport or add a new airport, click on the *Airports / Edit/Add Airport* menu item, which opens the Airport Editor dialog box. The airport editor is also automatically opened if you respond affirmatively to a system enquiry as to whether you wish to enter/update the data for a specific airport or if you double-click on any item in the Airport List

Proceed as follows:

- if custom airport list exists, designate whether the update is to be applied to the custom or the original data;
- enter the airport ICAO or IATA code and click the *Open* button to edit an airport or click the *New* button and then enter the new ICAO code for a new airport
- enter/update the remaining data as necessary, and
- click the *Apply Update* button.

Please note that an IATA code may be assigned to only one airport. There is one situation where this becomes problematic. When a FS9 airport has been re-designated and replaced in FSX, only one or the other may be assigned the IATA code.

To allow you to abort a series of updates without affecting *AirportList.dat*, any changes you make do not become permanent until you click on the *Save File and Exit* button. If you wish to exit without saving, click the *Exit* button.

Where both custom and original data is to be updated, the change must be entered twice or the custom file re-created.

Whenever an airport is entered or updated manually, a record of the updated airport is saved in a file named *AirportList – Updates.dat*. If *AirportList.dat* ever needs to be regenerated, these updates are re-applied automatically

- 9.5 **Airport Data Bulk Update** – Despite the large number of stock airports included in FSX, some small local airports, grass strips, water airports and military fields are missing. Scenery developers often model these missing airports. Where flight plans for such airports have previously been prepared, AI Flight Planner is able to update *AirportList.dat* based on the data in the corresponding TTools-format airport file.

To update AI Flight Planners airport data from a TTools-format airport file:

- click on the *Airports / Bulk Update* menu item
- check the *Update Existing* checkbox if you wish any airports already in the *AirportList.dat* to be updated based on the data in the TTools file; and
- if custom airport data exists, designate whether the update is to be applied to the custom or the original data; and
- specify the TTools-format airport file using *TTools Airport File for Update Select*.

The first airport in the file, or the first one not already in *AirportList.dat*, as applicable, is loaded into the editor. If the airport was previously in *AirportList.dat*, the full record as updated by the TTools-format file data is shown. Make whatever further changes are necessary and then click on the *Apply Update* button. Scroll forwards or backwards through the airports in the file meeting the *Update Existing* criteria using the *Next Airport* and *Previous Airport* buttons. If you wish all airports to be processed a single step without any manual updates, simply click on *All* after checking one or both of FS9 and/or FSX (the same selections are used for all airports updated using *All*). Please note that once you click *All* and the related processing is complete, all airports in the file are then known to the system; hence, the *Update Existing* checkbox must be checked in order for any airports to be displayed.

When finished click on the *Save File and Exit* button. To exit without saving, click the *Exit* button.

Whenever an airport is entered or updated using the bulk update feature, a record of the change is saved in a file named *AirportList – Updates.dat*. If *AirportList.dat* ever needs to be regenerated, these updates are re-applied automatically

- 9.6 **Saving and Retrieving Lists of Selected Airports** – From time to time you may find it convenient to be able to save and later retrieve a list of a subset of the airports

in the Airport List. This would be the case if you are interested in generating multiple flight plan files for a selected set of airports and wish to avoid having to manually reselect all the airports of interest each time.

To save such a list, select the airports of interest in the Airport list and click on the *Airports / Save Selected Airports List* menu item. You then must specify the file name for a text file to hold the list (which should start with *AirportList*).

To reselect these same airports in the Airport List, click on the *Airports / Select Airports from List* menu item and select the file holding the list. Each airport in the selected list which is also in the Airport List is selected.

- 9.7 Entering/Editing Time Zone Data – The file *Timezone_Base.dat* in AI Flight Planner's distribution archive file includes time zone information (UTC offsets) for every country and region where a stock airport exists. But reliable (free) time zone data for some of the more remote airports is difficult to obtain. And, even though certain countries notionally lie entirely within a single time zone, there are instances where a certain city in or a small region of such countries use a different time zone. If you develop AI flight plans for such areas using local times, you may find it necessary to update AI Flight Planners time zone data.

To do so, click on the *Airports / Change Local Time Offsets* menu item. This opens the time zone editor. Enter the ICAO or IATA code of the airport of interest into the designated text box and click the *Open* button. If sufficient information is known about that airport, its location information is displayed.

Enter the standard and daylight savings time offsets from UTC and click the *Apply Update* button. This action results in an updated entry in *Timezone.dat* which then established the UTC offsets not only for the designated airport but also for any other airport existing in the same country/region/city combination. As with *AirportList.dat* updates, such changes are not made permanent until you click the *Save File and Exit* button

10 SUPPORT

AI Flight Planner's support forum is located in the "Tools support" area at <http://fsdeveloper.com>. Please direct your problem reports, suggestions for improvement and other comments there. When you report problems, please include relevant details. In particular, the version number, the exact error message and a summary of what you were doing at the time are likely to be particularly helpful

I have also creating a support website at <http://members.shaw.ca/aifp>. The most recent release of AI Flight Planner will be available from that site.

While I can't promise to resolve every issue you report or include every feature addition you propose, I will undertake to support and enhance AI Flight Planner in a manner consistent with it becoming and remaining the AI Flight Planning tool of choice for Microsoft Flight Simulator.

11 ACKNOWLEDGEMENTS

I wish to thank the following people whose prior efforts or contributions made AI Flight Planner possible:

- Lee Swordy for TTools
- Winfried Orthmann for bglAnalyze (which was used in the early versions of AIFP in the generation of the airport lists) and for his documentation of airport .bgl file formats,
- Peter van der Veen for AIFPC and for permission to include its traffic file compiler/decompiler in the early versions of AI Flight Planner,
- Karl Swartz for his Great Circle Mapper website (<http://gc.kls2.com>) which seems to be the most comprehensive source of airport time zone data available at no cost ,
- Arno Gerretson for hosting the AI Flight Planner support forum on FS Developer.com, and
- last but not least, my beta team members, who expended a lot of effort in testing AI Flight Planner and/or advising me.

12 END USER LICENSE AGREEMENT

You are granted a free, non-exclusive right solely to install and use AI Flight Planner on your computer system(s) for your own personal enjoyment and, subject to what follows and the rights of others, to use and distribute flight plan and aircraft data in TTools text or MSFS compiled format files created or modified with AI Flight Planner ("derivative files").

You may not:

- upload AI Flight Planner to any file distribution system,
- reverse engineer, disassemble or decompile any part of AI Flight Planner,
- incorporate AI Flight Planner in whole or in part into any commercial product or facility, "shareware" or "freeware", or
- use AI Flight Planner or any derivative files in the development, marketing or support of, or incorporate derivative files in, any commercial product

without the express written permission of me, the author. Use for commercial purposes may be subject to a license fee.

In this end user license agreement, any reference to AI Flight Planner includes the files *AirportList.dat*, *Timezone.dat*, all other files included in the downloaded archive (.zip) file and any derivatives thereof.

Your use of AI Flight Planner is entirely at your own risk. I accept no liability whatsoever for any damage arising from its use no matter how caused.

By installing this software, you are deemed to have agreed to the foregoing.

AI Flight Planner - © 2008 - Don Grovestine