WarBirds AI Creation Guide

Created by goebel
Revised: 1/11/11

Table of Contents

Keys – Quick Reference

Mouse **left-click** – Places a new waypoint or selects an existing one
Mouse **right-click** – Moves an existing waypoint
Delete – Deletes the currently selected waypoint
Tab – Cycles sequentially through the start position and waypoints
F1 – Toggles the screen map from bird's-eye view to map view
[(or num. keypad **-**) – Zoom in
] (or num. keypad **+**) – Zoom out
Numeric keypad 1-9 – Scrolls the map
Shift-8 – Toggles between chase-view and CMEYE-view
Shift-9 – Cycles through the vehicles when in chase-view

I. Introduction

Welcome. This guide is designed to get you up and running with the creation and implementation of vehicle artificial intelligence (AI) routines for WarBirds. Routines can be used to create a fleet of escorted transport ships, a carrier group, a squadron of fighter planes defending an area, bombers which attack a specific target, a patrolling group of tanks, a supply convoy of trucks and supporting half-tracks, and anything else your imagination can devise. Any vehicle available in WarBirds is able to be used in AI routines. The Mission Editor is used to create new routines and to edit existing ones. While it can be cumbersome at times, the Mission Editor does allow for a great deal of control to be had over your AI routines. The purpose of this guide is to facilitate your understanding of how to make an AI routine, and make using the Mission Editor as painless as possible.

II. Opening the Editor

The first thing to do is to start the program WarBirds. From the Main menu, select "MISSION EDITOR". A popup window will appear asking you which terrain you would like to create an AI routine for. It is important that you select the terrain in which you intend to run your AI routine since each terrain has different coordinates for fields, landmasses and water. If you create a fleet of ships in the Midway terrain, the position of those ships will likely be on the land in the ETO terrain (which, if loaded online, can cause the server to crash). Once you select the terrain you want, the Mission Editor will launch. Note: all of the example screen shots within this guide are from the PTO.

III. Creating an AI Routine

An AI routine is a collection of vehicles which are grouped together and have the same instructions. Each AI routine can have a maximum of eight vehicles. All the information for a routine is contained within a special file called a "mobile" or "MBL" file (since it ends with the suffix ".mbl").

A. Getting Oriented with the Editor

Once the Mission Editor has opened, you will see a screen with controls on the left and bottom, and a bird's-eye view of the terrain in the center. What is displayed on the screen when in the bird's-eye view is the actual terrain (complete with objects), and not simply the in-flight (tower) map, which is simply an artist's rendition of the terrain. You can toggle between the bird's-eye view and the in-flight map by using either the F1 key or the button labeled “M” at the bottom of the screen. The in-flight map is easier to use if the vehicles in the routine are going to be traveling long distances, whereas the bird's-eye view is easer if you only want the vehicles to stay in a fairly localized area of the map. For both the bird's-eye view and the in-flight map the numeric keypad is used to navigate around (up/down//left/right), while the “[” and “]” keys are used to zoom in and out.

Tip: If you have a routine of ships or ground vehicles which have a path near a coastline, the coastlines in the in-flight map are not accurate enough to guarantee that a ship doesn't run aground or that a ground vehicle doesn't take a swim. To be absolutely sure, trace the routine's path while in the bird's-eye view of the terrain, which shows exactly where the coastlines are.

In Figure1, the Mission Editor's main screen is shown with all of the buttons and controls visible. Initially, when it is first opened, only a few of the buttons and controls are visible, with the others becoming visible as waypoints are placed. We'll discuss each of the buttons, toggles, input boxes, and pull-down menus at some point later in this guide. For now, find the vehicle type pull-down menu in the far upper-left of the screen. You use this menu to select the vehicle type that you would like the routine to consist of. Note that the default vehicle type (“FW1908”) is in the middle of the list; you can scroll up or down for more choices. For a complete list of vehicles, see Appendix A. Which vehicle type you choose can affect what options and controls are available later. AI routines consisting of ships or ground vehicles may have different options and controls than those consisting of aircraft. For the example AI routine in this guide, the vehicle type selected is the US carrier type “CV6”.

Below the vehicle type pull-down menu is another menu with which you can select one of the available loadouts for the selected vehicle type. Next to the loadout pull-down menu is an input box for entering the number of vehicles in the AI routine (remember that the maximum is eight). Below this input box is a set of toggles that let you select between a Drone Mission and a Player Mission. These toggle buttons are not shown in Figure 1 since they disappear once the routine is saved. Player missions are meant for offline play in which the player takes control of a vehicle in the routine. Drone missions are meant to be automated vehicles with which players can interact (either offline or online). This guide will be concentrating on the creation of Drone Missions.

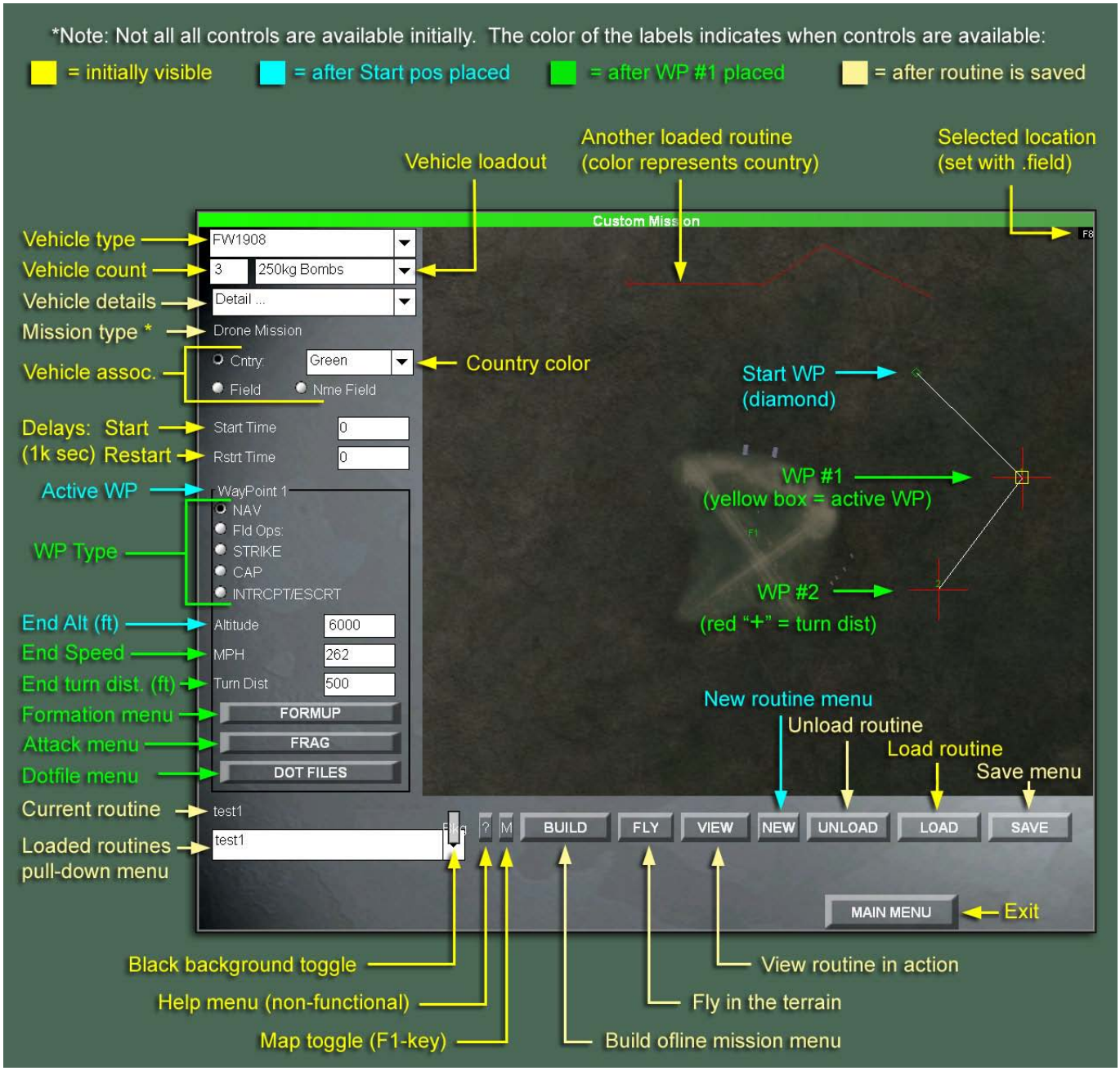


Figure 1 – Buttons and Controls of the Mission Editor: Initially not all of the Mission Editor's buttons and controls are visible. Many only become visible once waypoints have been placed or the routine has been saved.

B. Setting the Association of the Vehicles

The vehicles in the routine must be associated with either a country or a location. This is done using the vehicle association toggle buttons. By selecting "**Cntry**", the vehicles are associated with a particular country color (red, green, gold, or purple), as set by the country color pull-down menu to its right. This is by far the most common association that routines are given. The other two toggle buttons, "**Field**" and "**Nme Field**", make the routine associates with a particular terrain location. The currently selected location is indicated in a little black box in the upper right of the screen, and has nothing to do with where you are actually placing the vehicles on the map. To change the selected location, use the .field command (e.g. ".field 8" would make F8 the selected location). If the "**Field**" toggle button is selected, the vehicles in the routine will always be the same color as the selected location. If the location changes colors while the AI routine is running, the AI routine will instantly restart with the vehicles having a new color to match that of the location. If the "**Nme Field**" toggle button is selected, the vehicles in the routine will always be a <u>different</u> color than the selected location. If the location changes colors, the AI routine will instantly restart with the vehicles being a new color which is different than that of the location.

C. Setting a Start or Restart Delay

The **Start Time** and **Rstrt Time** input boxes allow you to set a delay to the initial spawning and/or respawning of the vehicles in the routine. The units for these variables are inconveniently in thousands of seconds (e.g. entering "1" equals 1,000 seconds), and HAVE to be integers (no decimals). This means that initially you can only input durations in 1,000 second intervals (that's 16 minutes and 40 seconds). However, you can change the Start or Restart times to values less than 1,000 seconds by editing the .mbl file directly. If you open a .mbl file you will see that the 6th and 7th lines are "STARTTIME" and "RESTARTTIME", respectively. The number following each of these is the delay time in seconds.

D. Placing the Starting Position

Each AI routine is composed of a Start position and at least one destination position, which is called a waypoint (WP). Typically an AI routine has a series of waypoints, to which the vehicles travel in a sequential order. The Start position is always the first thing that you place. This is done by left-clicking anywhere on the map at a point where you would like the vehicles to start. For the example in this guide, the Start position has been placed just south of F8 (see Figure 2). Notice that when you first left-click on the map, a green diamond appears with a yellow square around it. The diamond always represents the Start position and it is green in Figure 2 because the country color of the vehicles is set to green. Around this green diamond is a yellow square which indicates that the position is the currently selected one that is being edited. By placing the Start position, a "waypoint control frame" appears in the control menu on the left of the screen. Inside of the waypoint control frame is an input box called "Altitude". This value sets the starting altitude for all of the vehicles in the AI routine. For ships and ground vehicles, this value should be set at 0 feet, or whatever default value the Mission Editor automatically enters. For aircraft, make sure that the starting altitude is above the ground level. "Altitude" is the only variable available for the Starting position.

Remember that waypoints are "destination points" on the map to which vehicles attempt to fly, drive, or sail. Consequently, the Start position is not considered a true waypoint. All the

Start position does is designate the starting X (East/West), Y (North/South), and Z (altitude) coordinates at which the vehicles will start their route to Waypoint 1.

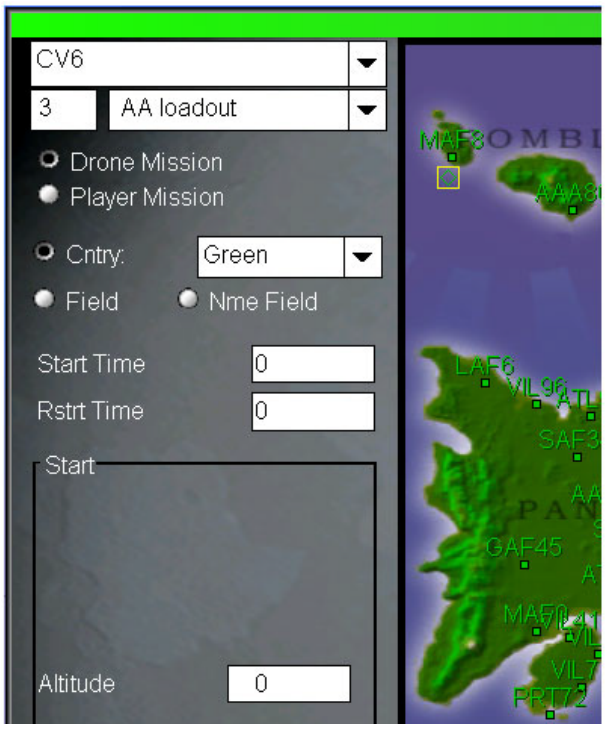


Figure 2 – Placing the Starting Position: A Start position, represented by the green diamond, has been placed just south of the location MAF8. The diamond is green to indicate the country with which the vehicles are associated. The yellow box around the Start position indicates that this position is the currently active one.

E. Placing Waypoint 1

Now that you have a Start position placed, it is time to create the first waypoint, which is called Waypoint 1 (WP1). To place WP1, left-click on the map wherever you would like the vehicles' first destination to be. Notice than when you do this, a red cross with a "1" in the center appears on the map. A white line traces the path between the Start position and WP1 (see Figure 3). You will notice that the yellow square that was previously around the Start position (the diamond) is now around WP1 to indicate that it is the currently selected waypoint. The vehicles in the routine will always start off at the Start position facing in the direction of WP1.

As soon as you place WP1, the menu to the left of the screen changes, revealing more controls within the Waypoint control frame. The input boxes "**MPH**" and "**Turn Dist**" become available as well as list of radio buttons used to set the waypoint type. Which radio buttons become visible depends on the vehicle type (i.e. ship, aircraft or ground vehicle) that you selected. Aircraft have NAV; Fld Ops; STRIKE; CAP; and INTRCPT/ESCRT. Ground vehicles have NAV; HOLD; and STRIKE. Ships only have NAV. Although some of the waypoint types are interesting, probably 99% of the time you will only need to use the NAV (navigation) waypoint type. The NAV waypoint type tells the vehicles to simply move to the next waypoint. The other waypoint types will be discussed later in this guide.

The white line traced between the Start position and WP1 represent the **waypoint segment** for WP1. All of the settings for WP1 apply to this segment. It is important to remember that a waypoint segment always *ends* with the waypoint with which the segment is associated. The main settings for a waypoint segment are "Altitude", "MPH" and "Turn Dist".

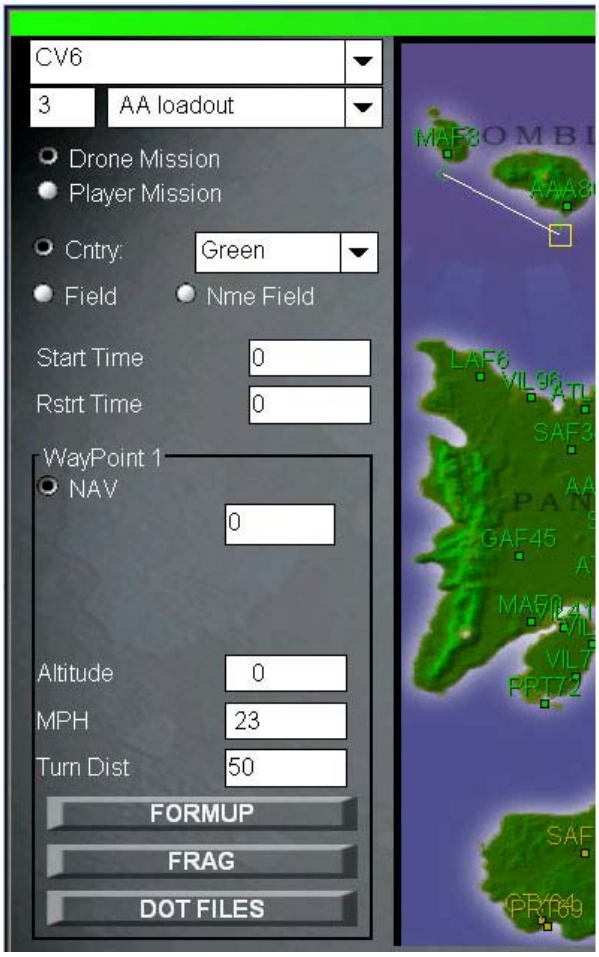


Figure 3 – Placing Waypoint 1: Placing Waypoint 1 makes visible the "MPH" and "Turn Dist" input boxes, the Waypoint Type radio buttons, as well as the "Formup", Frag", and "Dot Files" buttons.

E.1. Altitude

When the vehicles in the AI routine are ships or ground vehicles, any value entered in the "Altitude" input box is ignored when the routine is run. Instead, ships and ground vehicles will always be at sea or ground level. However, if the vehicles in the AI routine are aircraft, it is important to enter a correct "Altitude" value. Aircraft in an AI routine will start out at the altitude specified by the Start position but then immediately begin changing their altitude as they proceed toward WP1, trying to reach the altitude specified by WP1. There is a limit to the rate at which AI aircraft can change their altitude. If the difference between the altitudes of the Starting position and Waypoint 1 is small, and the waypoint segment for WP1 is long, then the aircraft will quickly reach the altitude specified for WP1 and stay at it for the remainder of their trip to WP1. However, if the difference in altitudes is large and the waypoint segment is small, the aircraft may not be able to reach the altitude specified by WP1 by the time they reach it. Also, setting a waypoint's "Altitude" to a value too close to ground level will cause the aircraft in the routine to do unpredictable things. Usually they will simply fly as close to the ground that the AI deem safe (~500 feet above ground).

E.2. MPH

The input box labeled "MPH" controls the speed at which the vehicles in the AI routine will move along the waypoint segment to the waypoint. As the label suggests, this value is in miles per hour. There are a few things to keep in mind when setting the speed for the vehicles. First, each vehicle has an upper limit of how fast it can go. You can set the speed

for a destroyer to 300 mph, but your ship will only travel at 40 mph (the top speed of all ships). Second, if you set the speed to the maximum possible for the vehicle type, some of the vehicles may have a difficult time maintaining formation, especially during turns or when the formation changes. And third, ground vehicles may not be able to attain the speed specified for a waypoint if the ground vehicles are traveling up inclines in the terrain.

E.3. Turn Dist

The input box labeled "Turn Dist" specifies how close (in feet) each vehicle in the AI routine must be to the waypoint in order to have successfully "reached" that waypoint. You can imagine the turn distance as the radius of a circle centered on the waypoint. If the vehicles in a routine cross into this imaginary circle, then they have successfully reached the waypoint. A smaller turn distance value means that the vehicles need to get closer to the waypoint's position in order to trigger the end of that waypoint segment and the beginning of the next. **The red cross at each waypoint is a graphical representation of its turn distance**, with each arm of the cross being equal to the length of the turn distance. The turn distance should probably be left at its default value unless you are specifically troubleshooting an issue. If the "Turn Dist" value is set too low (especially for ships and aircraft) it can result in vehicles being unable to reach the waypoint because their turning radius is larger than the waypoint's turn distance. This results in the vehicles endlessly circling the waypoint, unable to turn sharp enough to ever reach the waypoint's turn distance (as illustrated in Figure 4).

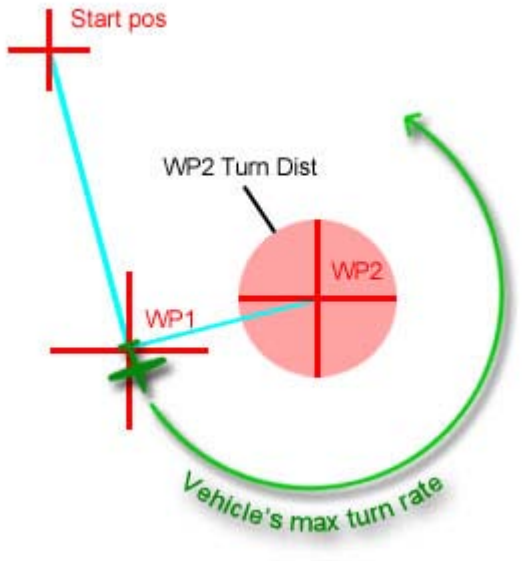


Figure 4 – Red Crosses Represent Waypoint Turn Distances: Problems can arise if waypoints are too close together and/or a waypoint turn distance is too small to allow a vehicle to turn tight enough to reach it.

Below the "Turn Dist" input box are three buttons: FORMUP, FRAG, and DOT FILES. These buttons open up different pop-up windows which allow you to control certain parameters of the vehicles as they travel along the waypoint segment of the currently selected waypoint. The parameters which these pop-up windows control are: vehicle formation, targets, and dotfile triggers, respectively.

E.4. Setting the Vehicle Formation for a Waypoint Segment

Clicking on the FORMUP button reveals a pop-up window like the one shown in Figure 5. As can be seen in the figure, there are 11 different formations which you can select for the vehicles in the routine. You can adjust the distance between the vehicles within the

formation by changing the “Form Len” (which stands of “formation length”) input box from “Default” to a numeric value, which is the distance in feet.

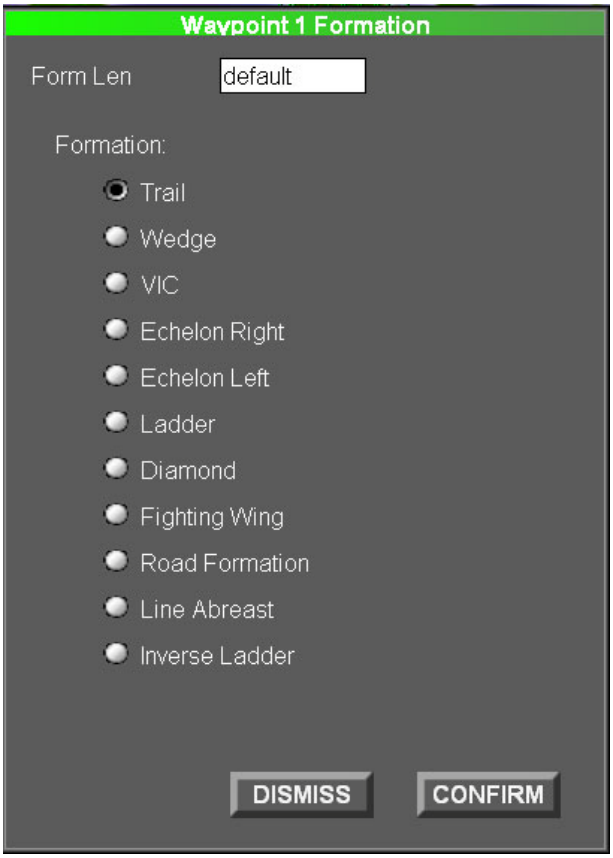


Figure 5 – The Formation Pop-up Window: Clicking on the “Formation” button brings up the Formation pop-up window which has 11 different vehicle formations to choose from. At the top of the window is the “Form Len” input box for specifying the distance (in feet) between vehicles within the formation. The “default” value is 50 feet for aircraft and ground vehicles and 1,500 feet for ships.

The pattern and spacing for all of the formations are shown in Appendix A. The figures in Appendix A show each of the formations facing the top of the page. Each is colored-coded to avoid confusion about where one formation ends and another begins. Each of the small colored squares which represent vehicles is numbered. These numbers indicate the position within the formation of each vehicle in the routine. For instance, if you only have five vehicles in your routine, then only the squares 1-5 will be included in the formation. The formations are shown on a grid of gray and white squares. With the exception or the “Fighting Wing” formation, each gray and white square represents the formation length (“Form len”) value entered in the popup window. The “default” formation length is 50 feet for aircraft and ground vehicle formations, and 1,500 feet for ship formations. Most formations are straight-forward, with only a few exceptions. When the vehicles are aircraft, the Ladder and Inverse Ladder formations stack the vehicles at slightly different altitudes. In the Ladder formation, vehicles #4 and #8 are at a higher altitude (equal to half the formation length) than vehicles #3 and #7, which are higher than #2 and #6, which are higher than #1 and #5. The Inverse Ladder formation has the vehicles stacked in the opposite order, with vehicles #1 and #5 being the highest and #4 and #8 being the lowest. If ships or ground vehicles are set to the ladder or inverse ladder formations, the vehicles are all placed at the same altitude. While both the Trail and Road formations have the same single-file “footprint”, these formations are different because of the way the vehicles follow the lead vehicle. When in the Road formation, each vehicle turns to the next waypoint only after it itself has reached the waypoint. In contrast, when in the Trail formation, all the vehicles turn at the same time, following the lead vehicle once the lead vehicle has reached the waypoint.

E.5. Setting the Targets for a Waypoint Segment

Clicking on the FRAG button opens the Targets pop-up window for the currently selected waypoint segment. This window allows you to control which types of targets are valid for the vehicles to attack while travelling along the waypoint segment. Which options and controls are available in the Target pop-up window will depend on both the vehicle type (aircraft, ships, or ground vehicles) and the waypoint type. As can be seen in Figure 6, you can assign the vehicles to attack enemy vehicles, ground structures, and/or fixed ack and artillery positions by clicking on the appropriate toggle button. Note that "fixed" ack refers to ack positions which are part of the terrain itself (and are therefore considered ground objects) and do include ack which are part of AI routines (which are technically considered "AI vehicles").

Clicking the "No Engage" toggle button causes the vehicles to **not** make offensive or defensive maneuvers if enemy vehicles come within range. With the "No Engage" toggle button selected, the vehicles will still fire at all the enemy target types that you have selected for them, but they will not alter their course on their way toward the active waypoint. Even vehicles that are stationary (because they are holding their position or going 0 mph) will begin taking evasive maneuvers in response to nearby enemy vehicles *unless* the "No Engage" toggle button is selected.

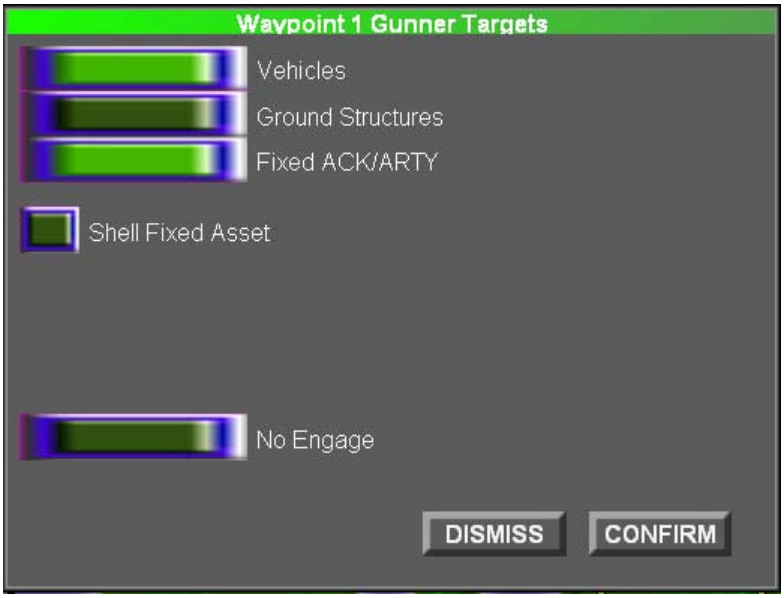


Figure 6 – The Targets Pop-up Menu: This figure shows the buttons and controls available if the vehicles in the routine are either ships or ground vehicles. If the vehicles are aircraft and the waypoint type is NAV, there will only be a toggle button labeled "Aircraft".

Instead of specifying that the vehicles should attack general enemy target types, you can instead click on "Shell Fixed Asset", which specifies that the vehicles should attack one specific ground object in the terrain. Clicking on the "Shell Fixed Asset" check box makes all the toggle buttons disappear, while making visible a target input box, a black target pull-down menu, and an input box for something called the CEP. If "Shelled Fixed Asset" is selected, the vehicles in the routine will only attack the ground object specified in the target input box, leaving all other targets (enemy vehicles and structures) alone, as they travel along the active waypoint segment. The target pull-down menu is populated with all of the ground object labels of the location nearest the active waypoint. You can scroll through the list to find the object you want. The input box labeled CEP stands for "circular error

probable". This is the radius (in feet) of a circle centered on the target, into which half the ordinance will fall. The purpose of the CEP input box is to simulate the variability normally associated with bombs dropped from an aircraft, by adding a degree of randomness to where the bombs fall. The smaller the CEP value, the better "aim" your vehicles will have, and vice versa. For convenience, when "Shelled Fixed Asset" is selected, all ground labels are temporarily made visible (even if they were initially toggled off).

E.6. Setting the Dotfile (or Dot Command) Triggers for a Waypoint Segment

Clicking on the DOT FILE button brings up the Dot Files pop-up window (see Figure 7). You can use this window to specify for a dot file (.dtf file) or a dot command to be run as the vehicles in the routine move along the waypoint segment.

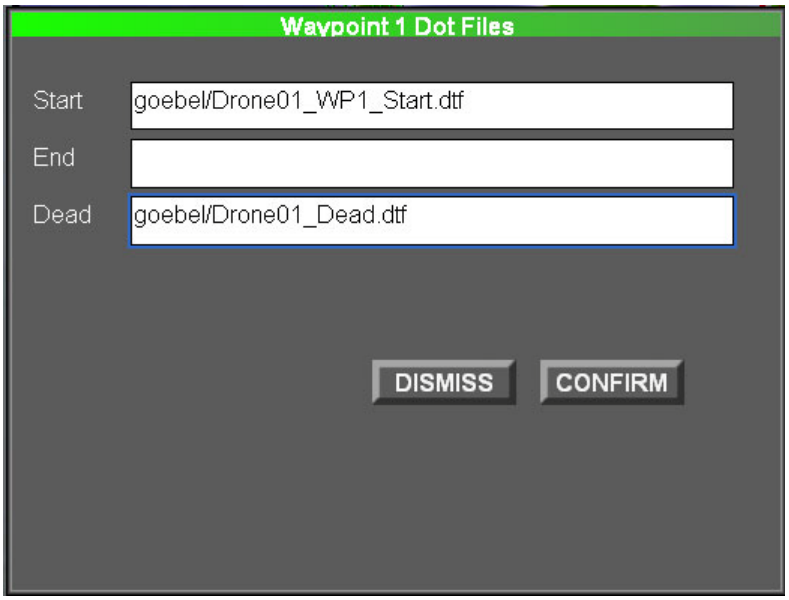


Figure 7 – Dot Files Pop-up Window: This pop-up window is used to specify dot files (.dtf) or dot commands to be run when triggers are activated along the waypoint segment.

For each waypoint segment there are three built-in triggers: START, END, and DEAD. By default, these triggers are not assigned any values. By using the Dot File pop-up window you are able to assign a single dot file or dot command to each of the trigger conditions.

START – Triggered as soon as the vehicles start the waypoint segment.
END – Triggered as soon as the vehicles end the segment (by reaching the waypoint).
DEAD – Triggered only if all of the vehicles are destroyed within the waypoint segment

A dot file or dot command can be assigned to any of the waypoint segment triggers. If the name of a dot file is specified for a trigger, then that file is called and its contents are run when the trigger is activated. Similarly, if a dot command is specified, then that dot command is run when the trigger is activated. If a dot file (or dot command) is assigned to the Start trigger, then that file is called as soon as the vehicles start that waypoint segment. If a dot file (or dot command) is assigned to the END trigger, then the dot file is called as soon as the vehicles reach the end of the waypoint segment. The DEAD trigger is perhaps the most useful. If the last remaining vehicle in a routine is destroyed while within the waypoint segment, then any dot file (or dot command) assigned to the DEAD trigger is called. Remember that the default for all routines is to simply restart the entire routine once

all its vehicles have been destroyed. The DEAD trigger can be used to stop a routine from restarting if it is destroyed. This is by far the most common use of the dot file triggers. Figure 8 shows the locations of triggers along the overall route of an AI routine.

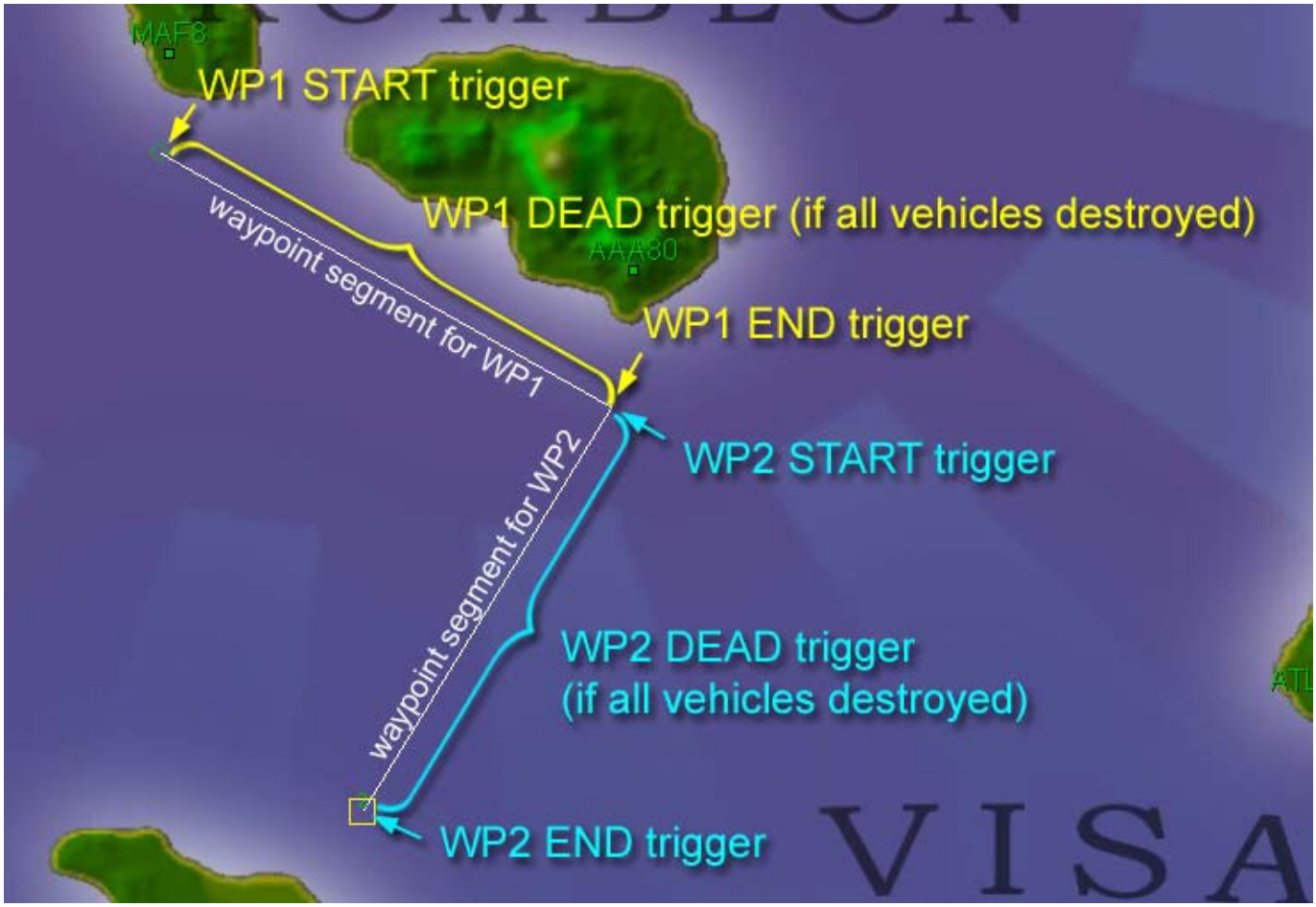


Figure 8 – Locations of Waypoint Triggers: Each waypoint segment has three trigger types. The START and END triggers are always activated at the start and end of the waypoint segment, respectively. The DEAD trigger for a waypoint segment is only activated if the last remaining vehicle is destroyed within the waypoint segment.

As an example, let's use the CV routine we've been referring to throughout this guide. Let's assume that we've named our AI routine "Drone01.mbl". A simple dot file named "Drone01_Dead.dtf" could be created using a text editor (e.g. Notepad) and placed in a folder called "goebel" within the main WarBirds folder. You can name your dot file anything you want, but it is helpful if the name reflects the purpose of the file. The contents of this dot file would simply be something like that shown in Figure 9.

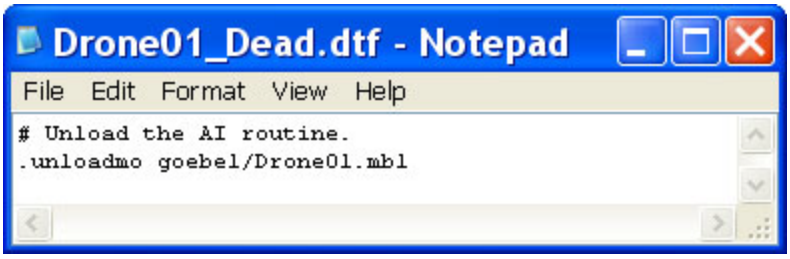


Figure 9 – Example of a Basic Dot File for the DEAD Trigger: The ".unloadmo" command is used to unload the routine "Drone01.mbl".

By specifying this dot file (and its path) for the DEAD triggers for each waypoint in the routine, if all the vehicles are destroyed anywhere along the routine's overall route, the file "Drone01_Dead.dtf" will be called, unloading the routine so it will not restart. Keep in mind that if you forget to assign the dot file to the DEAD trigger of one of the waypoints, the file would not be called (and the routine would not be unloaded) if the last vehicle is destroyed within that particular waypoint segment.

Instead of creating the dot file "Drone01_Dead.dtf" and specifying its name and path in the DEAD input box, we could instead achieve the same effect by simply using the ".unload" dot command directly in the DEAD trigger input box. As can be seen in Figure 10, the dot command entered is the same one we used in the sample dot file.

Figure 10 – A Dot Command Entered into the DEAD Trigger Input Box: Instead of providing a waypoint trigger input box with the name and path of a dot file, a dot command can be entered.

Entering a dot command directly into a waypoint trigger input box is a much easier and cleaner way of assigning an action to a waypoint trigger than is entering the name and path of a dot file. By using a dot command directly you not only don't have to spend any effort creating a dot file, but (more importantly) you never have to deal with the hassle of making sure that the dot file is located in the correct folder so that its path matches that specified in the AI routine. Not having to manage separate dot files with your AI routine is the big advantage of using dot commands directly in waypoint trigger input boxes.

The downside to assigning a dot command directly instead of a dot file is that, for each trigger only a single dot command can be assigned. If you would like two or more dot commands to be run when a trigger is activated, then you will have to use a dot file to run them. For instance, suppose that with our CV AI routine example, when the destruction of the last vehicle occurs, we not only want the routine to be unloaded, but we also want: an announcement to go out to both sides; F8 to turn red; and an AI routine of destroyers to be loaded. No single dot command can accomplish all of these tasks, so instead we would need to use a dot file which contained them all. Figure 11 shows a more complex version of the example dot file "Drone01-Dead.dtf".

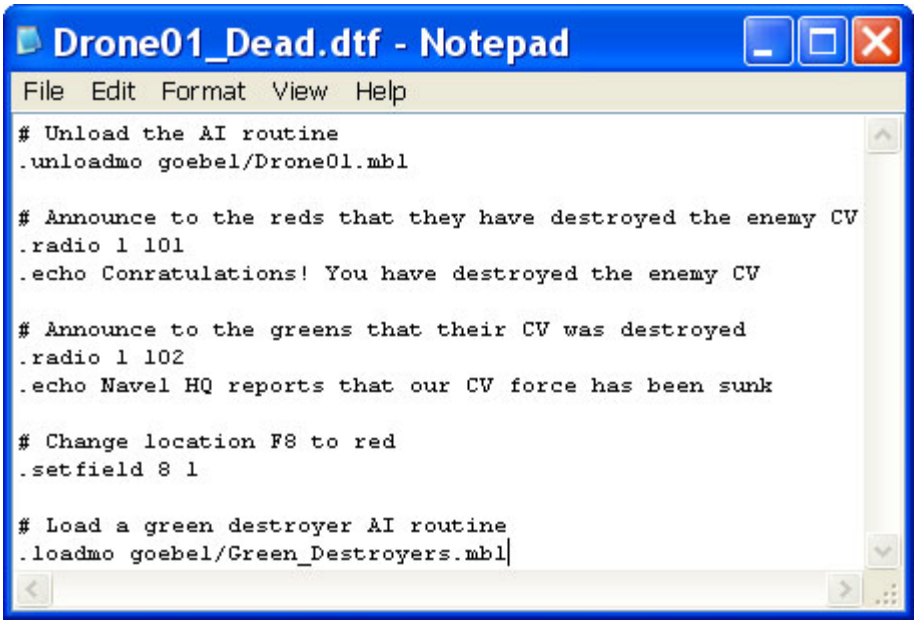


Figure 11– Example of a More Complex Dot File for the DEAD Trigger: In addition to unloading the AI routine with the ".unloadmo" dot command, other dot commands are used to cause many effects when the waypoint trigger is activated.

Waypoint triggers allow for many creative and interesting things to occur while an AI routine is running because any and all dot commands can be employed by the dot files called by an AI routine. For example, a convoy of transport ships could (using the WP1 START trigger) announce to its side that it is leaving port F20 for F5. Once the convoy arrives at F5 it could announce its arrival, enable Spitfires at F5, and load an AI routine of tanks that spawn at F5 (all using the WP1 END trigger). As the convoy passes close to F14, it could call for friendly air support if F14 was in enemy hands. As the last ship of the convoy is sunk, the convoy could (using the DEAD trigger for that waypoint segment) radio an SOS call with a last know position of being between F14 and F17.

Note: Waypoint trigger input boxes can be left empty but once you have entered text into one WarBirds won't allow that input box to be empty again. Any changes you make to text you had previously entered into a trigger input box will be saved so long as you don't erase all of the text, leaving the input box blank. The work-around is, instead of erasing all of the unwanted text previously entered in a trigger input box, simply replace it with a space (" ").

F. Saving the Routine

It's now time to save your routine; not necessarily because it is finished, but because **saving the routine enables more options** to become available (see Figure 12). In the lower right of the screen, click on the SAVE button. This will open the Save pop-up window, prompting you for a file name for the routine. There are two misleading aspects to this window. First, while the top input box is labeled "Enter File Name:" you may enter a file path as well. The default path is the WarBirds folder, so if you simply enter "TestCV1", then a file will be created in your WarBirds folder named "TestCV1.mbl". However, if you had previously created a folder called "MyAI", when prompted to for a file name for a new AI routine, you could type "MyAI/TestCV1", and the file "TestCV1.mbl" would be created in your folder called "MyAI". Be warned though, if the folder "MyAI" doesn't exist, then WarBirds will not save

your AI routine and won't warn you that it couldn't save it. The second misleading aspect to this window is that while it looks like the Description input box does something, it actually doesn't. Whatever you type in the Description input box will not be saved.

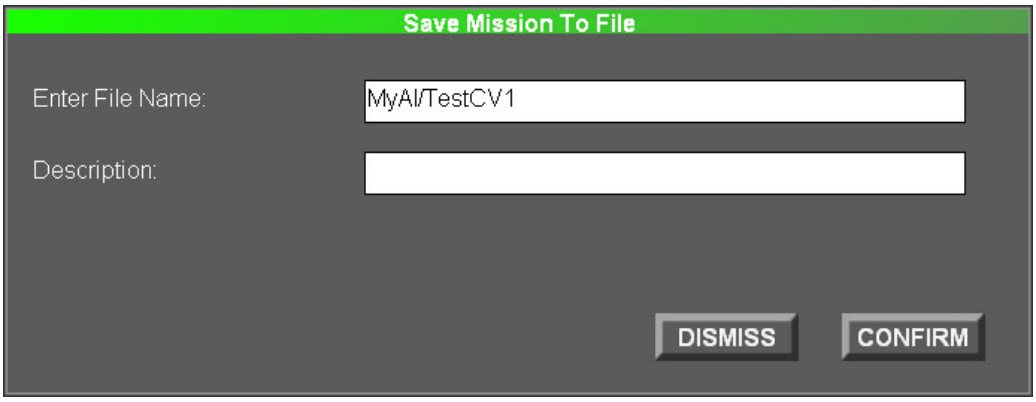


Figure 12 – The Save Popup Window: This popup window is used to save your AI routine. If the ".mbl" file extension is not added to the file name (as is the case in this figure), WarBirds will automatically add it for you. If no folder pathway is specified, the AI file will be saved into the main WarBirds folder. If a folder is specified (e.g. the folder "MyAI" is in this figure), the AI file will be saved into that folder *if the folder exists*, otherwise the file will not be saved.

Once your file is successfully saved, some new options and controls become available. In the lower-left the "Loaded routines" pull-down menu appears with a list of all of the AI routines loaded in the Mission Editor. Along the bottom of the screen, the BUILD, FLY, VIEW, and UNLOAD buttons appear. Finally, the "Detail" pull-down menu appears near the upper-left of the screen (under the vehicle count input box). You will also notice that right below the "Detail" pull-down menu, the toggles for selecting a drone or offline mission will have disappeared and replaced with the name of the mission type. Each of the buttons and controls which become available by saving a routine is described below.

Tip: To create a second AI routine that is similar to an existing one, simply load the existing AI routine as a template, click the save button, and give it a new name. This creates a new copy *of the original "template" AI routine that can be edited to your liking.*

F.1. The "Loaded routines" Pull-down Menu

The "Loaded routines" pull-down menu is a list of all of the AI routines that are currently loaded into the Mission Editor. Each time you save a new AI routine or load an existing routine, the routine's name is added to the "Loaded routines" pull-down menu. Selecting a routine from the list makes that particular routine active. You may only edit the active AI routine. If a routine is loaded but not active only its waypoint segments will appear on the map (in the color which matches its vehicle association).

Caution: If you start creating a new routine but haven't yet saved it, using the "Loaded routine" pull-down menu to make another routine the currently active one will cause your unsaved routine to disappear.

F.2. The BUILD Button

The BUILD button is for saving your AI routine as an offline mission. At this time, not much is currently known about this function.

F.3. The FLY Button

The FLY button is used to view your AI routine from the cockpit of a different vehicle. Clicking this button causes you to spawn in whatever vehicle you were last flying (or driving) at the position of your active location. Note that where you spawn has nothing to do with where your AI vehicles are. While in the Mission Editor, you don't have access to the normal controls to select your vehicle, starting altitude, and location. However, you can set each of these within the Mission Editor by using dot commands. You can change the vehicle in which you spawn by using the **.plane <vehicle ID>** command (e.g. ".plane 31" sets your vehicle to a P-51D). See the Appendix B for a list of vehicle IDs. You can change the altitude at which you spawn by using the **.startingalt <alt>** command (e.g. ".startingalt 1000" makes you spawn at 1000 feet). You can change your active field (indicated in the upper-right of the Mission Editor screen) by using the **.field <field #>** command (e.g. ".field 8" makes you spawn at F8).

F.4. The VIEW Button

The VIEW button is incredibly useful. Clicking this button both loads all of the AI routines listed in the "Loaded routines" pull-down menu (lower-left of the screen) and puts you into chase-view mode looking at the lead vehicle of the first AI routine listed in the "Loaded routines" pull-down menu. This function allows you to instantly check see how your vehicles behave when the AI routine is loaded and run. From this vantage point you can see if the speed, formation, starting position, heading, vehicle type, etc. are correct for your routine. While you are in chase-view mode looking at your vehicles, you can pan the view around using the numeric keypad; zoom in and out using the "[" and "]" keys; cycle between each of the vehicles by using Shift-9; or go to regular CMEYE mode by using Shift-8. To quit viewing your AI routine, type ".e" into the text buffer.

F.5. The UNLOAD Button

The UNLOAD button is used to unload the currently active AI routine.

F.6. The "Detail" Pull-down Menu

The "Detail" pull-down menu allows you to individually customize each of the vehicles in the routine. Clicking on this pull-down menu shows a list of all of the drones (vehicles) in the routine. Selecting one of the drones from the list opens up a Drone Detail pop-up window (shown in Figure 13).

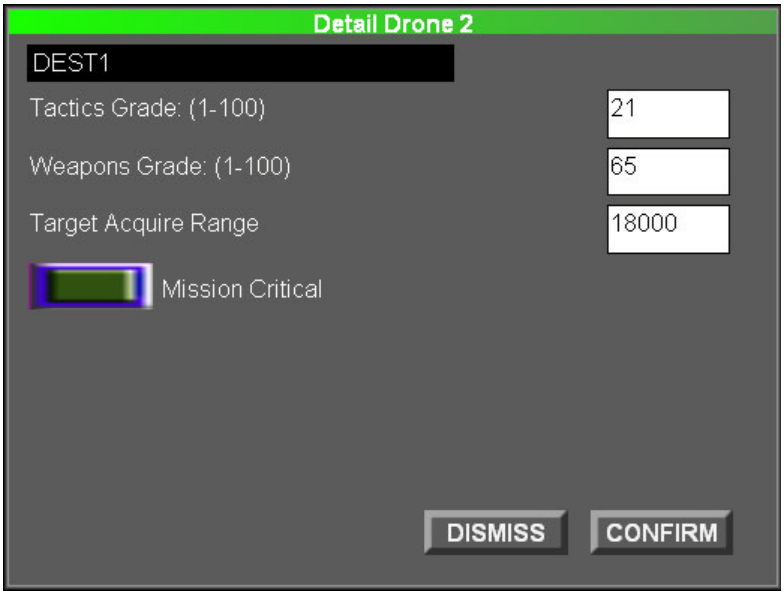


Figure 13 – The Drone Detail Pop-up Window: This window is used to set the parameters of individual vehicles in the routine. The pull-down menu in the upper-right of the window can be used to create a routine containing different vehicle types but only if the vehicles are not aircraft (in which case the pull-down menu doesn't even appear).

Note: if you change the number of vehicles in the routine, you must save the routine before the "Detail" pull-down menu is updated to reflect the new vehicle count.

Using the Drone Detail pop-up window, you can change how effectively the selected vehicle can maneuver to engage enemies (by adjusting the Tactics Grade) and how effectively it can fire its weapons (by adjusting the Weapons Grade). For both of these settings a value of 1 is the lowest and a value of 100 is the highest. You can also adjust the range (in feet) at which the vehicle will respond to enemy threats (by adjusting the Target Acquire Range). If an enemy target comes within the target acquire range, the vehicles in the routine will begin maneuvering (unless the vehicles are set to "No Engage") and shooting at the target (unless the vehicles are set to not attack that target type). The Drone Detail pop-up window also contains a toggle button labeled "Mission Critical". At the time of this writing, the purpose of this button is still unknown.

The final control in the Drone Detail pop-up window is the vehicle type pull-down menu. This pull-down menu only appears if the vehicles in the routine are ships or ground vehicles (i.e. not aircraft). Using this menu, you can create a formation made up of a mix of vehicles. For instance, you can set Drone 1 to be a US carrier ("CV6"), Drone 2 to be a light destroyer ("DEST1"), and Drone 3 to be a heavy destroyer ("DEST2"). Ship routines can only have a mix of ships, ground vehicle routines can only have a mix of ground vehicles, and aircraft routines can only have a formation of the all same aircraft type. **All of the parameters that are set in the Drone Detail pop-up window apply to the vehicles in the routine over their entire route and not just for a single waypoint segment**.

G. Repositioning Waypoints

Once you have placed waypoints, you will most likely want to reposition some of them. To do this, right-click (holding down the mouse button) on the waypoint you would like to move, then reposition it. If you have to move it a long way, then it may be easier if you switch from the bird's-eye view to the in-flight map and/or zoom out.

Because the Mission Editor does not display the grid over the map, it can be problematic if you want to place a start position or waypoint at a specific grid location (e.g. "6.7.2" or the

"4.5 cross"). The only solution to this problem is to manually edit the coordinates of the .mbl file itself. As an example, suppose that you wanted a fleet of transport ships to start at the "5.1 cross". The steps for doing this are:

1) Using the Mission Editor, create the AI routine of transport ships, placing the Start position in the general area of the map that you estimate the "5.1 cross" to be.
2) After saving the file, exit the Mission Editor and select Free Flight from the main WarBirds menu (making sure to choose the same terrain you used in the Mission Editor). Because Free Flight mode displays the grid over the in-flight map, it can be used to obtain the X and Y coordinates of any grid position.
3) Once in the terrain in Free Flight mode, move to the closest field to the "5.1 cross" and then enter cmeye-mode by typing "**.**cmeye" into the text buffer.
4) Using the in-flight map as a guide, fly to the "5.1 cross", stopping to hover there. While hovering over the "5.1 cross", activate the heads-up display (HUD) by pressing ctrl-H (if it's not already activated), and then write down both the X and Y coordinates of your position.
5) Exit out of WarBirds and open the .mbl file of the transport ships you had previously saved.
6) Once the file is open, find the line that starts with "**WAYPOINTSTART...**" and change the values of the X and Y coordinates in the line with the coordinates corresponding with the "5.1 cross" which you previously wrote down.
7) Save the .mbl file. The Start position will now be at the "5.1 cross".

The steps for having WP1 be at a specific grid position are the same as outline above except that instead you would edit the first line of the .mbl file that starts with "**WAYPOINTNEXT...**" If you wanted WP2 to be at a specific grid location you would edit the second line of the .mbl file that starts with "**WAYPOINTNEXT...**", and so on.

H. Changing the Selected Waypoint

Whichever waypoint on the map has the yellow square around it is the currently selected waypoint. Any changes made to the waypoint settings only apply to the currently selected waypoint. There are two ways to make changes the selected waypoint. First, you can left-click on any waypoint (or the Start position) to select it. Make sure you click close enough to the waypoint on the map to select it; otherwise you will instead create a new waypoint. If this happens, just delete it using the "Delete" key. The second way of changing the selected waypoint is by pressing the Tab key. The Tab key cycles through each of the waypoints sequentially, centering the selected one on the screen, and is thus an invaluable way to navigate around your AI routine's route. The Tab key is your friend!

I. Placing Additional Waypoints

Placing a second waypoint (WP2) is as easy as left-clicking the mouse somewhere on the map. All additional waypoints are added this way, and can be modified in the same manner that WP1 can be modified. When you left-click on the map, the new waypoint will be placed sequentially after whichever waypoint was previously selected. Each additional waypoint that you place will take on the same settings for "Altitude", "MPH", formation, and targets as the previously selected waypoint. New waypoints do not have to be added sequentially. For example, suppose you already have five waypoints. Selected WP1 and then left-clicking on the map will create a new waypoint that is positioned between WP1 and what *used to be* WP2. The new waypoint will be named WP2, and all of the subsequent waypoints will be renumbered accordingly. The only exception to this is the Start position. If the Start position

is selected and you left-click somewhere on the map, a new Start position is created, what used to be the Start position becomes WP1, and all of the remaining waypoints are renumbered accordingly.

J. Waypoint Types

Every waypoint is always assigned a waypoint type. There are several waypoint types to choose from, but which ones are available depends on which vehicle type (aircraft, ship, or ground vehicle) the routine consists of. The only two waypoint types available for routines containing ship or ground vehicle are NAV and HOLD. For aircraft routines the available waypoint types are NAV, Fld Ops, STRIKE, CAP, and INTERCEPT/ESCRT. Below are descriptions of each of the waypoint types.

NAV – The NAV ("navigation") waypoint type specifies that the vehicles are to move directly to the waypoint. However, the vehicles will take evasive maneuvers and engage targets if they are specified to in the Targets pop-up window (via the FRAG button). For routines in which the vehicles are aircraft, the only type of enemy target available to assign is “Aircraft”.

Fld Ops – The Fld Ops waypoint type has three subcategories (TAXI, TKOFF, and LAND). The TAXI subcategory makes the aircraft in the routine behave like ground vehicles. They will drive instead of fly to the waypoint. **Using the TAXI subcategory and giving the aircraft a speed of 0 mph is the only way to create parked aircraft.** Using any other waypoint type will cause the aircraft to spawn in the air going over 100 mph, even if you set their speed to 0 mph. The TKOFF subcategory starts with the aircraft at ground level and has them become airborne as soon as possible to fly to the waypoint. The LAND subcategory specifies for the aircraft to land, touching down at the waypoint. Selecting this subcategory will cause the waypoint to move to the runway of the nearest location. However, you will probably need to move it to suit your needs. It is difficult to get the aircraft to land correctly but it can be done. The location at which they are to land must be friendly or the aircraft will fly away. Also, the speed, altitude and approach vector must be within certain acceptable limits for the aircraft to land successfully.

STRIKE – Whereas routines consisting of ships or ground vehicles can be assigned different target types when using the NAV waypoint type, aircraft routines can only be assigned ground targets when using the STRIKE waypoint type. Selecting the STRIKE radio button automatically brings up the Target pop-up window (just as if the FRAG button was selected). Using the STRIKE waypoint type, aircraft can be assigned to attack enemy ground vehicles, ground structures, and/or fixed (i.e. terrain) ack and arty. Alternatively, the aircraft can be assigned to attack a specific fixed ground object by selecting “Strike Fixed Asset”. You may either select an object from the list or manually enter one not on the list. Finally, regardless of which target type you assign, you may toggle “No Strafe” to ensure that the aircraft only use bombs.

CAP – The CAP waypoint type is supposed to cause the aircraft to briefly patrol at the location waypoint before moving on to the next waypoint. However, in actual use, this waypoint type appears to work the same as the NAV waypoint type in every respect. Thus far I am unable to determine how CAP differs from NAV.

INTERCPT/ESCRT – The INTERCEPT/ESCRT waypoint type is used to assign aircraft in a routine to attack or escort vehicles in another routine. When this waypoint type is selected, a

pop-up window appears with a pull-down menu at the bottom. The pull-down menu will be populated with all other AI routines currently loaded in the Mission Editor (or it will be blank if no other routines are loaded). The AI routine you select to be intercepted/escorted will appear in the top input box. If the selected AI routine is the same country color, they will be escorted, but if they are an enemy country color they will be attacked.

HOLD – The HOLD waypoint type causes the vehicle formation to pause for a specified number of seconds upon reaching the waypoint before proceeding to the next waypoint (or restarting if the waypoint is the last one in the routine). The holding position happens at the *end* of the waypoint segment and not the beginning. This waypoint type is only available for ships and ground vehicles but not for aircraft, since they would fall from the sky if they paused at a waypoint.

IV. Special Routine Setups

Static Vehicles – It is common to have vehicles in a routine remain stationary. This is done to create a routine of parked ships, ground vehicles, or aircraft. It is also done when creating a routine of ack guns to supplement the "fixed" ack which come with the terrain. The most effective way of achieving stationary vehicles is to have a Start position and a waypoint with a distance between them which is much larger than the turn distance of WP1. The MPH settings for WP1 should be set to 0. Since the vehicles have a speed of 0 mph, they can never reach their destination. If the vehicles in the routine are aircraft, use the "Fld Ops/TAXI" waypoint type so that they behave as if they were ground vehicles. Next, you need to make sure that the vehicles don't take evasive maneuvers if enemy targets come within range. To do this, if the vehicles are ships or ground vehicles, make sure the "No Engage" toggle button is selected in the Target pop-up window (via the FRAG button). If the vehicles are aircraft, make sure that the "Aircraft" toggle button is NOT selected in the Target pop-up window. To be extra sure that the vehicles won't move in response to nearby enemy targets, you may wish to set each of the vehicles Target Acquire Range to 0 (via the "Details" pull-down menu).

Looping Paths – Sometimes you may want the vehicles in a routine to travel in a loop over the same path over and over, without any limits on the number of loops. To achieve this, you must keep in mind an important fact: once a routine reaches its last waypoint, the default action is for it to instantly re-spawn at the Start position. If you want the vehicles to look, from the players' perspective, like there is no interruption in the movement of the vehicles, it is best to have the last waypoint sit right on top of the Start position (see Figure 14A). This way the vehicles will re-spawn and appear at the same spot at which they had just disappeared because of reaching the final waypoint. Lining up the headings of the first and last segments of the circuit helps with the illusion. While the transitions between iterations of the routine can be made to look seamless, there are problems with this setup. If any vehicles were destroyed during the course of the first loop, those vehicles will be replaced when the routine restarts again. This not only can look odd, but it makes it impossible for damaged vehicles to retain their damage beyond a single loop of the routine. All damaged vehicles will be considered destroyed and credit will be given to the players who inflicted damage to them. If you are running an event that will last two hours, and a single loop through an AI routine's circuit is 30 minutes, it would be better just to build in four loops into the circuit so that the routine never needs to restart during the event (see Figure 14B).

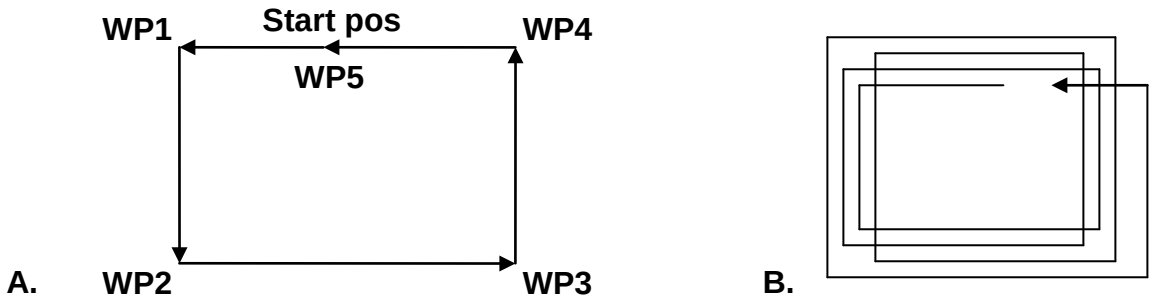


Figure 14 – Looping Vehicle Paths: A. If a routine has vehicles which must run the same circuit over and over, it may be best to create a path in which the final waypoint sits atop the Start position, with the heading of the first waypoint segment lining up with that of the last waypoint segment. B. Because destroyed vehicles will be replaced and damaged vehicles will be considered destroyed when a routine restarts, it may be better to instead create a routine with many circuits making up the overall route.

V. Running AI Routines

Once you have created an AI routine, you will most likely want to run it to see it in action. The procedure for running an AI routine is a little different depending on whether you are going to be running the routine offline or online. Running AI routines online is slightly more complicated than running them offline, and uses all of the same tools. Therefore, it is recommended that you first become familiar with the process of running routines offline before tackling the process of running them online.

A. Running AI Routines Offline

Once you have created and saved an AI routine, you can run it offline to see how it looks and behaves. The commands associated with managing AI routines are listed below. The text in brackets indicates a variable that you supply, and forward-slashes indicate an alternate form of the command.

.LOADMO/.LDMO <filepath> - Loads and begins running the .mbl file <filepath>
Example: ".ldmo MyAI/TestCV1.mbl" would load and begin running the AI routine called "TestCV1.mbl" located in the folder "MyAI"

.LOADMODIR/.LDMODIR <directory> - Loads and begins running all .mbl files in <directory>. If <directory> is left blank, all the .mbl files in the main WarBirds folder are loaded and run.
Example: ".ldmodir MyAI/BlitzAck" would load and begin running ALL of the AI routines (.mbl files) located within the folder "BlitzAck", which is located in the folder "MyAI"

.LISTMO - Lists information about all loaded AI routines; showing the names and status (RUNNING, WAIT, and OFFLINE) of all loaded routines, as well as the total number of drones (vehicles) loaded and running. If a routine's status is WAIT (due to a delay added to either the start or restart times) the number in parenthesis indicates the number of seconds left before the routine begins running.

.UNLOADMO <routine> - Unloads the AI routine <routine>.

.CLEARMO/.CLRMO - Clears all running and loaded AI routines.

.DIRMO <directory> - Lists all .mbl files in <directory>. If <directory> is left blank, all the .mbl files in the main WarBirds folder are listed.

The above commands are all that is necessary to control AI routines offline (controlling them online is a little more complicated and is discussed later on). The command **.LDMO** (or **.LOADMO**) is used to load and run an AI routine (.mbl file). When using this command, you must supply the routine's path (if any) and file name, including the .mbl file extension. The default path is the main WarBirds folder, so if your routine is in that folder, you only need to supply the routine's file name.

Tip: It is a good habit to load/run AI routines using the ".ldmo" command instead of the longer ".loadmo" command. This is because if you mistype ".loadmo" in any way, WarBirds interprets your command as ".load" which completely reloads the arena, resetting the fields, aircraft, and arena settings. During events, this can be a catastrophic mistake; one which is easily avoided by using the safer ".ldmo" command. For the same reason, "ldmodir" is preferable to "loadmodir".

An alternative way of loading routines is to use the **.LDMODIR** (or **.LOADMODIR**) command. This command tells WarBirds to load ALL of the .mbl files in the specified folder (directory). Using this command can save a lot of time and effort. The only danger is, if there are any stray .mbl files in the folder, they will be loaded and run as well.

To stop running and unload a single AI routine use the **.UNLOADMO** command. When using this command, you only supply the name of the routine and (unlike the **.LDMO** command) MUST NOT supply a file path or use the .mbl file extension or the command won't work. If you wish to stop running and unload ALL AI routines, then you should use the **.CLEARMO** command.

The **.DIRMO** command is a useful way to quickly see which .mbl files are in a particular folder. A list of all the .mbl files in a folder will scroll through the text buffer if you issue this command. If you happen to be running WarBirds in full-screen mode, this command can save you from having to quit out of the game to take a look inside some folders. However, its biggest value comes when working with AI routines online (discussed later).

In WarBirds, "loading" and "running" an AI routine are technically two different things. "Loading" an AI routine refers to when WarBirds reads the contents (i.e. instructions) of an .mbl file and loads them into the computer's memory. "Running" an AI routine refers to when WarBirds starts implementing those instructions. Why is there a distinction? WarBirds has maximum limit of 64 AI routines which can be loaded into memory at one time. However, the maximum limit of AI vehicles which can be actively running at any given time is 128. Since each routine can hold up to eight vehicles, it is possible to load far more vehicles than can be run at one time. Presumably, once the maximum of 128 vehicles running is reached, any additional vehicles which are part of a loaded routine are put into a queue and run only when spots become available. The commands **.LDMO** and **.LDMODIR** take care of both the loading and running of an AI routine all at once. Similarly, **.UNLOADMO** both stops and unloads the AI routine all at once.

It should be noted that there exist other dot commands for controlling AI routines, which I do not recommend using. These commands include: `.LOADMODIR_OFFLINE; .LOADMO_OFFLINE; .STARTMO; .ONLINEMO; .KILLMO;` and `.OFFLINEMO`. The reason these commands are not recommended is because they only perform one step of either the starting or stopping of a routine. The commands `.LOADMODIR_OFFLINE` and `.LOADMO_OFFLINE` only load routines without actually running them. The commands

`.STARTMO` and `.ONLINEMO` only run routines but only if they have been previously loaded. Finally, the commands `.KILLMO` and `.OFFLINEMO` only stop routines from running but do not unloaded them. It is far easier and cleaner to avoid these commands in favor is using the commands which either load/run or stop/unload routines all at once.

B. Running AI Routines Online

Running an AI routine online is a little more complicated than running one offline. First, it is important to understand that it is only possible to run a routine online using a special version of WarBirds called CMTools. WarBirds CMTools behaves exactly like the normal version of WarBirds, except that it allows for some special commands to be used online, including all of the commands needed for managing AI routines. If you have a version of WarBirds CMTools, simply run the program, log into an arena as usual, and then use the same commands listed in the previous section ("A. Running an AI Routine Offline") to manage your AI routines. However, most players do not have a copy of WarBirds CMTools. If your computer doesn't have a copy of CMTools, then the only way to run an AI routine online is to "commandeer" a computer that *is* running CMTools. Seriously. Luckily, iEN has computers running CMTools which are logged into some arenas for the sole purpose of having players commandeer them. These computers are known as "Generals" and have handles with all capital letters (e.g. PATTON, GOERNG, BOELKE, MNTGMY, etc.). To commandeer a General you need to have Campaign Manager (CM) permissions in the arena in which the General is running. If you don't have CM permissions or if there is no General in the arena then there is no way to run your AI routines online in the arena.

Assuming that you have CM permissions and there is a General logged into the arena, there are two steps to running AI routines online: 1) copying your AI routine files to the General and 2) telling the General to run the AI routines.

B.1. Sending Files to a General

There are two commands which can be used to send files to a General. These commands are:

`.SENDFILE <path>` `Send file <path> to whichever player radio channel-1 is tuned`
Example: ".sendfile MyAI/TestCV1.mbl" would copy the file "TestCV1.mbl" located in the folder "MyAI" to the computer of whichever player your radio channel-1 is tuned.

`.SENDDIR <directory>` `Send all files in <directory> across to whichever player radio channel-1 is tuned`
Example: ".senddir MyAI" would copy ALL the located in the folder "MyAI" to the computer of whichever player your radio channel-1 is tuned.

To use these commands to send files to a General you must first tune your radio channel-1 to the handle of the General (e.g. "**`.RADIO 1 PATTON`**"). Once you have tuned your radio channel-1 to the General, you can use **`.SENDFILE`** to send a single file or **`.SENDDIR`** to send all of the files within a folder to the General. It is important to remember that the General is simply a computer which is running CMTools and logged into the arena. Like your computer, the General has a main WarBirds folder. You use the **`.SENDFILE`** command with just a file name (without specifying a file path), if the file is to be copied from *your* main WarBirds folder to the *General's* main WarBirds folder. However, copying files into a General's main WarBirds folder is not recommended because if all CMs did this it would cause the General's WarBirds folder to become cluttered with everyone's files. A better practice is to keep all of your files within a folder (bearing your name) which resides within

the main WarBirds folder. Thus, every file which I send to a General is first placed within a folder called “goebel”, which is in my main WarBirds folder on my computer.

It is important to note that when using the **.SENDFILE** or **.SENDDIR** commands, whatever path that you specify in the command must match on both your computer and the receiving computer. The receiving computer will create any new folders it doesn't yet have, but only if it is running CMTools (like a General). Here is an example:

Let’s say that I have created two new files which I’ve named “XMasPartyRedCVs.mbl” and “BDBattleSetup.dtf”. I’ve placed both of these files in a new folder called “XMasParty”, which is itself within a folder called “goebel” in my main WarBirds folder (See Figure 15).

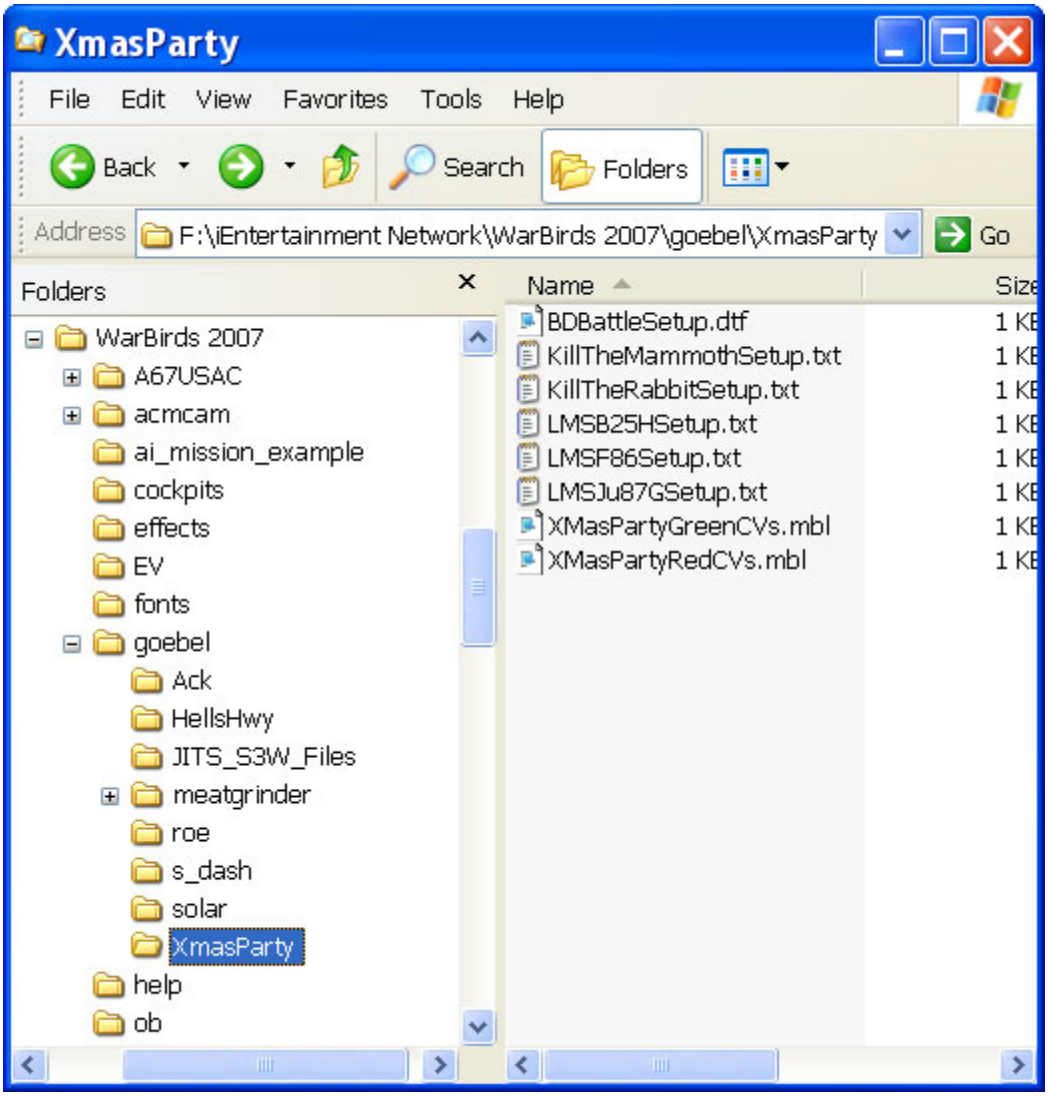


Figure 15 – Example File Path: This window shows the file path for files within the folder “XMasParty” within the folder “goebel”.

I then type:

```
.radio 1 PATTON
.sendfile goebel/XMasParty/XMasPartyRedCVs.mbl
```

My front end notes that PATTON already has a folder called goebel in his WarBirds folder (let’s say from a previous sendfile command) but doesn’t yet have a folder within it called

"XMasParty". Since PATTON is running CMTools, a new folder is created within "goebel" on PATTON's front end. Now that PATTON's file path matches that on my computer, my front end copies the file "XMasPartyRedCVs.mbl "from my computer to PATTON's. Had I used the "**`.senddir goebel/XMasParty`**" instead, then the folder "XMasParty would have been created with the folder "goebel" on PATTON's computer and then all of the contents of that folder would have been copied from my computer to PATTON's.

B.2. Commanding a General to Run Your Routines

Once your AI routine files are on a General, you can use your CM permissions to tell the General to run those routines. To activate your permissionsThe commands for managing your AI routines online via the General are exactly the same as those used to manage them offline (see "A. Running an AI Routine Offline"), with one exception. To let your front end know that the ".ldmo" command that you just typed in is meant for the General, you must have your radio channel-1 tuned to the General and **you must add a space before the command**. If your radio channel-1 is tuned to a General, anytime you add a space before a command, your front end will tell the General's front end to execute the command (see Figure 16).

A.

B.

Figure 16 – A Space Before A Dot Command: These two lines are not the same. (A) The dot command in the top text buffer would work fine offline, but not online. (B) Because the bottom text buffer includes a space before the dot command, it would direct the General (PATTON) to execute the ".ldmo" command.

All commands that you would normally use to manage your AI routines offline can be used to manage your routines online through a General by simply including a space before each command. For instance, the **`.LISTMO`** command preceded by a space will cause a list of all AI routines running from the General to be displayed in the text buffer. Adding a space before the **`.DIRMO`** command (e.g. " .dirmo goebel"; note the space before the ".") will list all of the mobile (.mbl) files in the specified directory on the General's computer. It should be noted that if an arena has two Generals logged into to it, you can run AI routines from one or both of the Generals. However, each General is independent of any others, so if you use issue a " **`.CLEARMO`**" command to PATTON, any AI routines running on MNTGMY will continue to run. Before using the **`.CLEARMO`** command online, be aware that other CMs may also be running AI routines in the arena on the same General. The **`.CLEARMO`** command will unload everyone's routines, not just yours. Before unloading any AI routines it is always safest to first send a " .listmo" command to the General to see what AI routines are currently being run.

There are a few other commands that are relevant to running AI routines online. These include:

.AILIST – Opens a window listing all active drone vehicles (including wingman drones) being run on the server *by any general*. This command is NOT issued through the General but by the player.

.CVENAB <1|0> <planeType> <country|-1> - Enables/disables **<planetype>** on CVs of **<country>**. Pass -1 in **<country>** to effect all CVs.
Example: ".cvenab 1 2 3" causes all loaded Gold CVs to the F6F5 enabled.

.PALIAS -1|<planetype> <alias> - Sets icon text for planetype <planetype> to <alias>. Pass -1 to clear all aliases. Can be used online if issued through an sdot command.
Example: ".sdot palias 79 POW truck" makes all trucks in the arena have the icon "POW truck".

As the command's description states, **.AILIST** is used to find out what AI drones (vehicles) are currently running in the arena on the server. Both player wingmen and .mbl file drones are displayed. If there are multiple Generals, all of the .mbl file drones from all of the Generals are displayed. This is different from the command which you would execute on the GeneralAnother helpful command when managing AI routines online is the **.DIRMO** command. This command displays all of the .mbl files within whatever folder you specify. Offline, this command isn't too useful since you can simply open any folder you want on your computer, but online it is extremely useful since it is the only way for you to find out the contents of a folder residing on a General.

The command **.CVENAB** is used to enable and disable vehicles on CVs. This command is executed on *your* front end (not a General's) and always has three parameters. The first parameter specifies whether a vehicle on the CVs should be disabled (0) or enabled (1). The second parameter specifies which vehicle (see Appendix A – Vehicle Codes) is to be disabled or enabled. Currently, entering "-1" for the vehicle ID does NOT specify "all vehicles" (although this may soon change), so each enabled vehicle which are not wanted on a CV must be disabled one at a time. The third parameter specifies to which country color (1 = Red; 2 = Green; 3 = Gold; 4 = Purple; and -1 = All colors) of CV the changes are applied. When a CV is loaded, it is assigned a number based on the order in which it was loaded. If there were 7 drones (vehicles) already loaded and running when a CV is loaded, the CV will be assigned the number "8" and will appear on the inflight map as "CV8".

To a limited degree it is possible to have CVs of the same country with different aircraft enabled. This is possible because the **.CVENAB** command only applies to CVs which are already loaded and running. For example, you could load and run an AI routine which contained a Red CV, and then enable SBDs on it (using ".cvenab 1 34 1"). Then you could load and run a second Red CV, and then enable F6F5s on both loaded CVs (using "cvenab 1 1 1"). The CV which was loaded first would have both SBDs and F6F5s enabled, while the CV which was loaded second would only have F6F5s enabled.

The command **.PALIAS** can be used to change how a vehicle's icon reads. To use this command online, it must be added to the arena settings as an sdot command. It would be most useful to employ the **.PALIAS** command when using a vehicle in an AI routine which is substituting for a vehicle not in the WarBirds stables. For instance you could issue the command ".sdot palias 57 IL-2" to make all Ju-87Gs (vehicle ID = 57) have icons which read "IL-2" in the game.

B.3. Sending Files from a General to Your Computer

Sometimes you may want to look inside a .dtf or .mbl file that is running on a General. To do this you will need to first copy the file onto your computer. The dot command for accomplishing this is the **`.sendfile`** command, but using it to send a file *to* your computer is a little more complicated than using this command to send files *from* your computer.

To send a file from a General to your computer, you must first know the path of the file. There is no way to determine the path of a file on a General, even if it is currently running, so you'll have to ask the creator of the file what its path is. Once you know the path of the file on the General, you will have to recreate that path on *your* computer. This is necessary because you aren't running CMTools so, unlike the General, your front end won't automatically create any new folders for you. For instance, if the file path on the General is "vslp/doa/zeppelin.mbl", then in your main WarBirds folder you'll need to create a folder named "vslp", and put within that a new folder named "doa". Once the paths match, you are ready to use the **`.sendfile`** command, but in a different way than you normally would. Remember, in general terms, if the **`.sendfile`** command is executed on Computer A, the file will be sent from Computer A to Computer B, which is designated by tuning the radio channel-1 of Computer A to the handle of whichever player owns Computer B. In this case, Computer A is the General and Computer B is your computer (normally this is the opposite). The steps for sending a file from a General to your computer are:

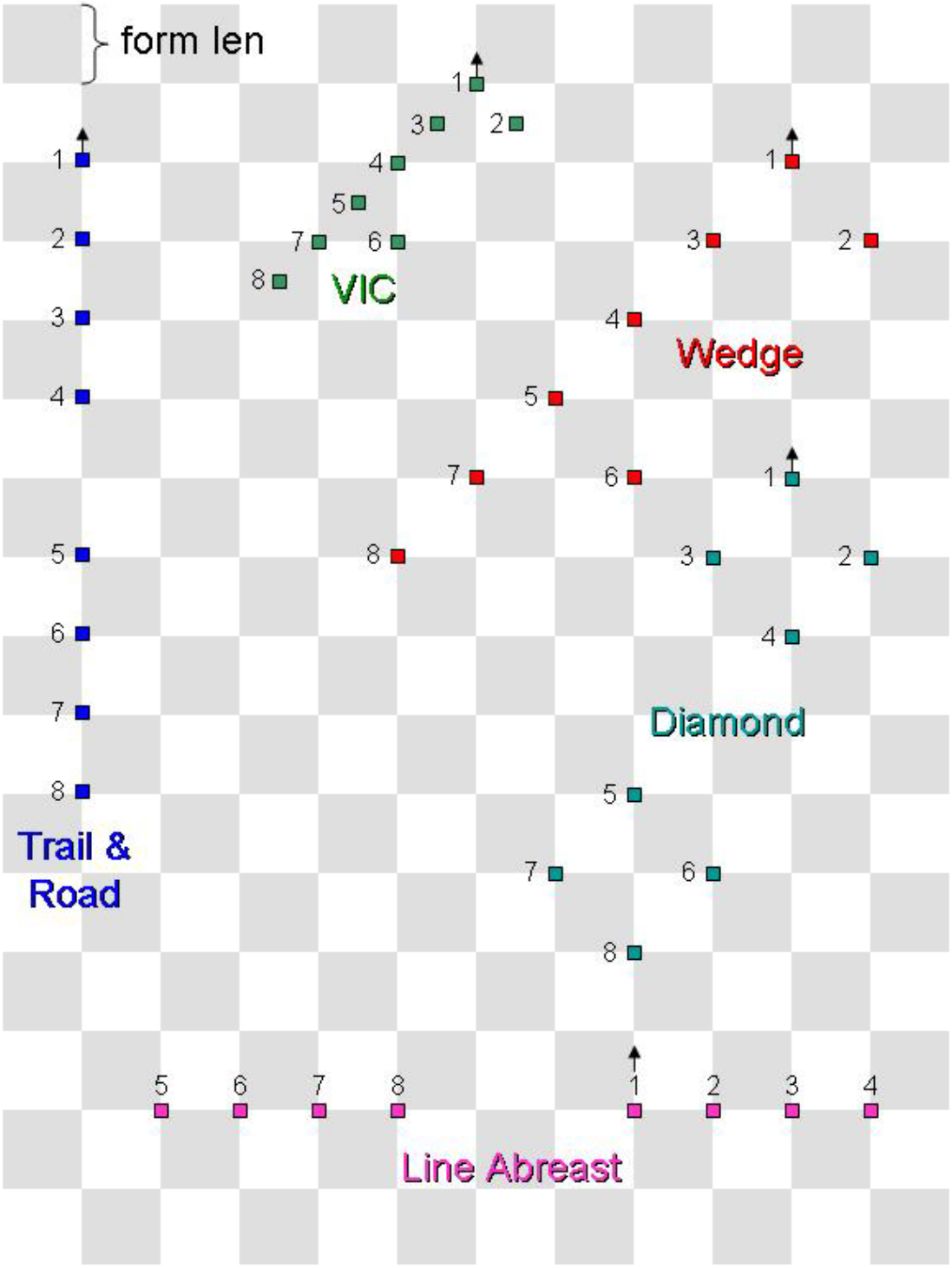
1) Tune your radio channel-1 to the general (e.g. "**`.radio 1 PATTON`**")
2) Tune the General's radio channel-1 to you (e.g. " **`.radio 1 goebel`**"; note the space before the "**.**")
3) Tell the General to send you the file (e.g. " **`.sendfile vslp/doa/zeppelin.mbl`**"; note the space before the "**.**")

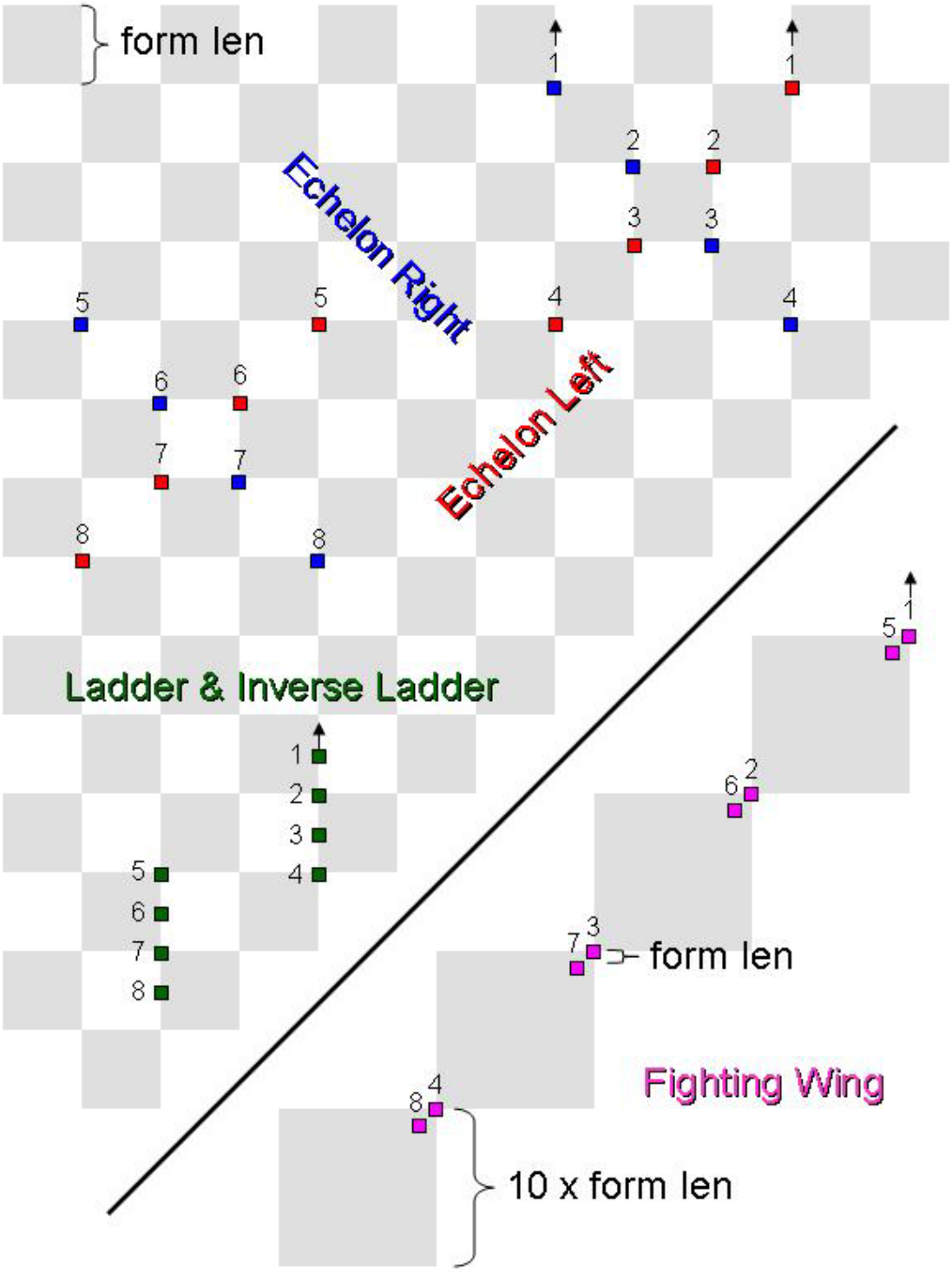
Remember, if the file's path on the General's computer does not have a matching path on your computer, the file will not be sent.

Notes:
- See if CEP actually works.
- See if a routine will restart and change colors if their associated location changes to ANY color.
- The ".unloadmo" command is used incorrectly in the section referring to DEAD triggers.
- Find out what "Mission Critical" does.
- Under Waypoint Types it says that ships and vehicle routines only have the NAV and HOLD waypoint types. However, in the section about placing WP1, it says that ships only have the NAV waypoint type. Which is it?

Appendix A - Formations: Patterns and Spacing

The available vehicle formations are shown below. Each one faces the top of the page and has been color-coded to more easily distinguish where one ends and another begins. Each square in the grid has a length equal to the formation length ("form len") value, except for the Fighting Wing formation in which the squares lengths are equal to ten times the formation length. The numbers (1-8) indicate where each of the vehicles is positioned within the formation.




form len
Echelon Right
Echelon Left
Ladder & Inverse Ladder
form len
Fighting Wing
10 x form len

Appendix B - Vehicle IDs

1	F6F5	55	G4M2	109	FREIGHT1	163	109G6AM
2	F4F4	56	KI61	110	SPIT5B1	164	109G10D
3	FM2	57	JU87G	111	BOAT1	165	109G14AM
4	F4U1D	58	B24D	112	EC03MK1	166	109G14AB
5	ZERO21	59	B24J	113	DO17Z	167	109G14AC
6	ZERO	60	F4F3	114	BF109G2E	168	109K4DB
7	ZERO52	61	CHUTE	115	IL2	169	109K4DC
8	KI43	62	PILOT	116	J2M2	170	109K14L
9	KI84	63	FOO1	117	J2M3	171	SpitM03F
10	BF109E	64	M3	118	KI611C	172	SpitM12F
11	BF109F	65	M4A1	119	FW190A2	173	SpitM20F
12	BF109G	66	M16	120	FW190F8	174	SpitM45B
13	109GR6	67	M5	121	P40N	175	SpitM45C
14	BF109K	68	MKIV	122	LANC1	176	SpitM45M
15	BF110C	69	MKIVD	123	LANC3	177	SpitM47H
16	BF110G	70	MKV	124	I16X	178	SpitM55M
17	FW1904	71	F86	125	KI44	179	T38
18	FW1908	72	A36	126	KI45	180	PRED
19	FW190D	73	KAGA	127	MIG15	181	EC6
20	HURRI1	74	DEST1	128	MIG15G	182	LZ30
21	HURRI2	75	MC202	129	P82	183	Nieuport-11
22	SPIT1	76	MC205	130	F80	184	Nieuport-17
23	SPIT5	77	DEST2	131	MGAAA	185	Nieuport-27
24	SPIT9	78	CV6	132	20MMAAA	186	GothaGIV
25	P38F	79	TRUCK2	133	40MMAAA	187	GothaGV
26	P38	80	P39Q	134	88MMAAA	188	TRENCHMG
27	P38L	81	P400	135	ECEC04	189	MGAAA
28	P39D	82	NFII	136	N1K-1J	190	HandlyPage_SUB
29	P40E	83	MC2022	137	C-47	191	M4A4
30	P47D	84	T34	138	LA5F	192	M35
31	P51	85	M4A3	139	LA5FN	193	SdKfz251
32	D3A	86	TRANS	140	LA7-2	194	MKIII
33	B5N	87	BF109F1	141	LA7-3	195	SGIII
34	SBD	88	G4M1	142	P-40n20	196	Tiger
35	JU88A	89	G4M3	143	P-40f	197	S35
36	B25H	90	SPAD7	144	De21	198	CharB1
37	B25J	91	CAMEL	145	Me163	199	T3476
38	B17	92	CL2	146	Ju188A	200	A67
39	TBF1	93	D5A	147	FordGPW	201	F6F-5-N
40	B17F	94	SPAD13	148	T6A	202	Bf-109F4-N
41	B25C	95	D7	149	BORG	203	FW-190D9-N
42	P40B	96	DR1	150	109E1A0	204	P-51D-N
43	P47C	97	F2B	151	109E3A1	205	B-17G-N
44	P51B	98	SE5	152	109E4AA	206	P-38L-N
45	SEAF2	99	M16X	153	109E4N	207	P-38J-N
46	SPIT14	100	13TK	154	109E7NZ	208	P-38F-N
47	F4U4	101	HE111H3	155	109F1N	209	F4U-1-X:1
48	ME262	102	KI27B	156	109F2NZ	210	F4U-1A-X:1
49	YAK3	103	TBD	157	109F4E	211	F4U-1D-X:1
50	YAK9D	104	F4U1	158	109F4EZ	212	F4U-4-X:1
51	JU87D	105	F4U1A	159	109G1AZ	213	SBD-5-X:1
52	MOSQ6	106	CORSAIR1	160	109G2A	214	TBD-X:1
53	MOSQ4	107	CORSAIR2	161	109G5ASZ	215	F4U-1D-X:2
54	JU52	108	CORSAIR4	162	109G6A		