

# **Manual para Generar Plantillas para la Campaña Dinámica BADC**

**Versión 1.0**

Traducción

**E69\_Rayonnant**

# Índice

1. Resumen de los Archivos de Plantilla

2. Descripción de Plantillas de Lugares

3. Creando Plantillas

3.1 Primera Etapa

3.2 Segunda Etapa

3.3 Tercera Etapa

4. Aeronaves

5. Imágenes

6. Configuración Final

7. Comentarios

# 1. Resumen de los Archivos de Plantilla

Las Plantillas en badc no son un archivo único, son varios, y son hechos de diferente manera

**geo\_obj.data** Contiene la información más importante, tiene 4 grupos principales de datos:

1. La localización de los aeródromos, su daño, sus lugares de aparcamiento, bando que lo controla, nombre código etc.
2. Coordenadas de sector, tipo( atacable o no), TTL, bando que lo controla etc.
3. Nombre de la ciudad, coordenadas, daño, radio de suministro, zonas blanco<sup>1</sup>, bando que lo controla, etc.
4. Una lista de la relación de ciudades y sectores, para saber que sector tiene una ciudad y no será atacable hasta que la ciudad tenga un 50% de daño.

**city.mis:** Este es un archivo de misión creado usando el FMB (Full Mision Builder). Es el archivo que más consume tiempo. Aquí determinas cuales ciudades son parte de la campaña, y haces las zonas blanco<sup>1</sup>, cada una de estas zonas es un lugar atacable; este archivo indica que tipos de objetos serán colocados y donde. NO tiene los objetos. Los indicadores son una combinación de puntos de ruta de aeronaves de diferentes altitudes para representar los objetos (se explicara después). Luego el generador colocara los objetos "bajo" las líneas de los puntos de ruta.

**tanks-wp.mis:** Esta es la plantilla de ubicación de los tanques, cada sector de 10x10 km necesita tener 2 o más "puntos". Luego el generador usara 2 de esos puntos para colocar los campamentos y un grupo de 9 tanques. Como veras 2 puntos como mínimo son necesarios, más es mucho mejor, pues así el generador tendrá más de una combinación de puntos para elegir, haciendo más impredecible donde estarán los campamentos y los tanques en movimiento. Esos "puntos" son tanques estáticos. Realmente no importa que tipo de objeto sean, el generador buscara las coordenadas de los objetos (los puntos). En sectores donde es imposible colocar dos puntos de tanques, se volverán inatacables.

**red\_obj.mis:** Una misión que contiene un Li-2 por aeródromo, nada más.

**blue\_obj.mis:** Igual, pero con JU-52, y números de objeto que comienzan con 10000<sup>2</sup>.

**frontline:** Este archivo contiene el estado de la línea de frente, tiene una bandera por cuadro, excepto por los sectores inatacables.

**aircraft.data:** Este archivo tiene la información de aviones disponibles, tipo, tareas, airclass, etc.

**imap.data:** Este archivo es una imagen html, imagen para ayudar a la selección de objetivo de a la hora de la creación de una misión. Contiene los nombres de los sectores (y adicionalmente campos y nombre de ciudades) para ser mostrados de forma emergente cuando se coloque el ratón sobre la imagen de frente.

Para crear una plantilla para un mapa, todos los archivos mencionados arriba deben ser escritos, cada archivo tiene condiciones particulares que cumplir. Los archivos de plantilla en una primera impresión parecen muy complejos, pero no lo son.

*1.Spot Zones en el original. NdT.*

*2. and objects numbers starts with 10000 en el original NdT*

## 2. Descripción de Plantillas de Lugares

Proveo en esta guía una explicación archivo por archivo y algunas herramientas para la semi-automatización de la creación de plantillas; pero antes de empezar a mirar dentro de cada archivo, prefiero explicar la diferencia entre "plantillas de objetos estáticos" y "plantillas de lugares"

**Plantillas de objetos estáticos** Son archivos creados con el FMB, tienen todos los objetos del mapa. Generalmente contienen miles de carros, aviones, AAA, trenes, etc todos dispersos en un mapa particular. Luego cuando ataques un lugar, el generador de misiones busca un arreglo de objetos cerca del lugar que atacas, los "copia" y "pega" en la misión que está creando.

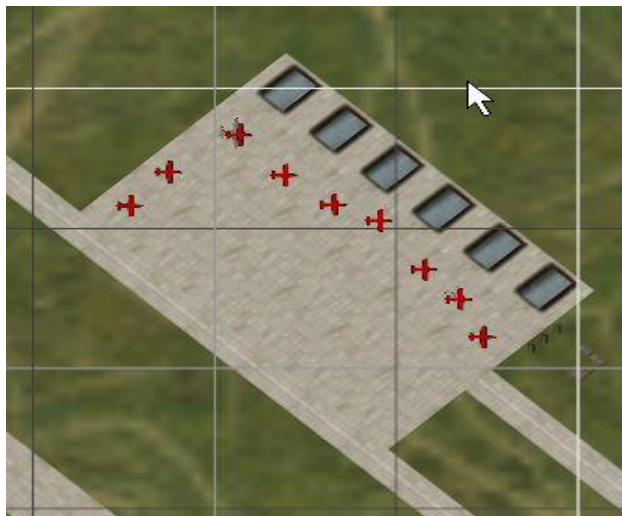
Este sistema tiene ventajas y desventajas, son fáciles de construir, pueden tomar varias horas para construirlas, pero después son fáciles de usar para el programa generador con el "copia y pega". La desventaja es que no puedes cambiar los tipos de objetos, necesitas una plantilla roja y una azul, los objetos están todos en el mismo sitio y luego de varios ataques al mismo sitio la gente puede saber donde está la AAA y donde los "premios gordos" (mejores sitios para bombardear).

Badc usa 2 archivos de plantillas de objetos estáticos: red\_obj.mis y blue\_obj.mis. Sólo contienen aviones estáticos colocados en las bases, 1 por base y usado solamente para hacer que las bases muestren correctamente el color en los mapas (sin bases blancas).

**Plantillas de lugares** Estas plantillas **no** tienen objetos que después estarán en las misiones. Tienen los "lugares" donde queremos tener objetos ubicados. Luego el generador insertará los objetos en el lugar correcto en la misión que estamos generando. Eso es todo

Las plantillas de lugares no son difíciles de hacer, sólo necesitan un poco de convenciones (reglas) sobre como esos lugares son marcados, para que luego el generador entienda donde y como deben insertarse los objetos. Badc usa plantillas de "lugares" para aeródromos, puntos de ruta de tanques y objetivos de ciudades. Vamos a ver un ejemplo general, luego veremos las reglas detalladas de cada plantilla.

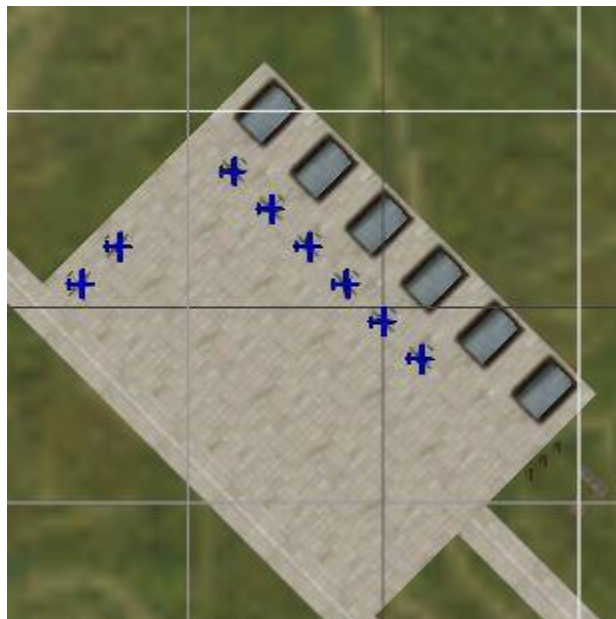
Digamos que estamos creando una **plantilla estática** para un aeródromo, colocamos varios aviones y podemos ver esto:



Después de que esos aviones son seleccionados (tipo) y ubicados tenemos que rotarlos uno por uno y luego alinearlos. Ahora vamos a usar el mismo aeródromo, y usar la idea de "plantilla de lugar". Queremos tener algunos aviones alienados en 2 líneas, como se muestra en la siguiente imagen:

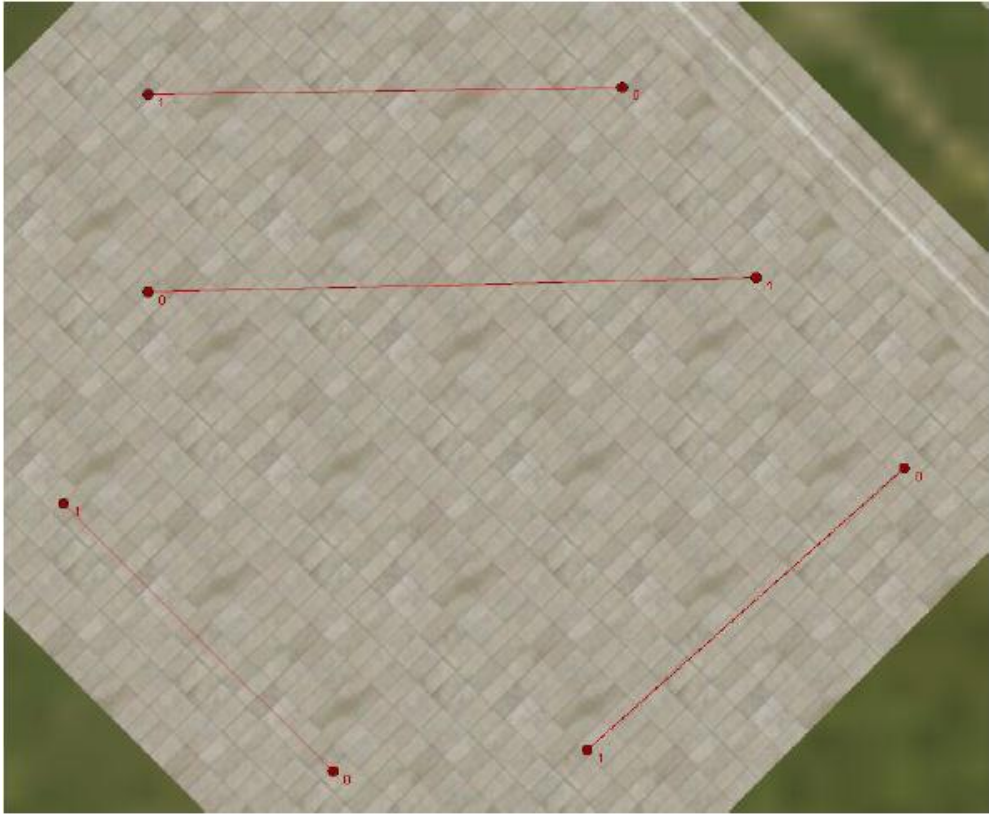


Así que luego el generador insertara aviones como en la siguiente imagen (están correctamente alineados y rotados, quizá no sea visible por el tamaño de la imagen).



Y bien, **como hacemos esas líneas** para que después el generador coloque los objetos? usamos una aeronave (cualquiera volable) y asignamos unos puntos de ruta, de esta forma "dibujamos" líneas. Tal vez no puedas entender esto la primera vez que lo leas, pero es muy fácil, colocando los puntos de ruta y "dibujando" una línea con un avión hará dos cosas: decirnos donde están las coordenadas y mostrarnos donde se colocarán los objetos.

Vamos a ver un ejemplo funcionando, la siguiente imagen muestra varios puntos de ruta donde quiero colocara algunos aviones. Usó una aeronave B-17 para "dibujar" esos puntos de ruta:

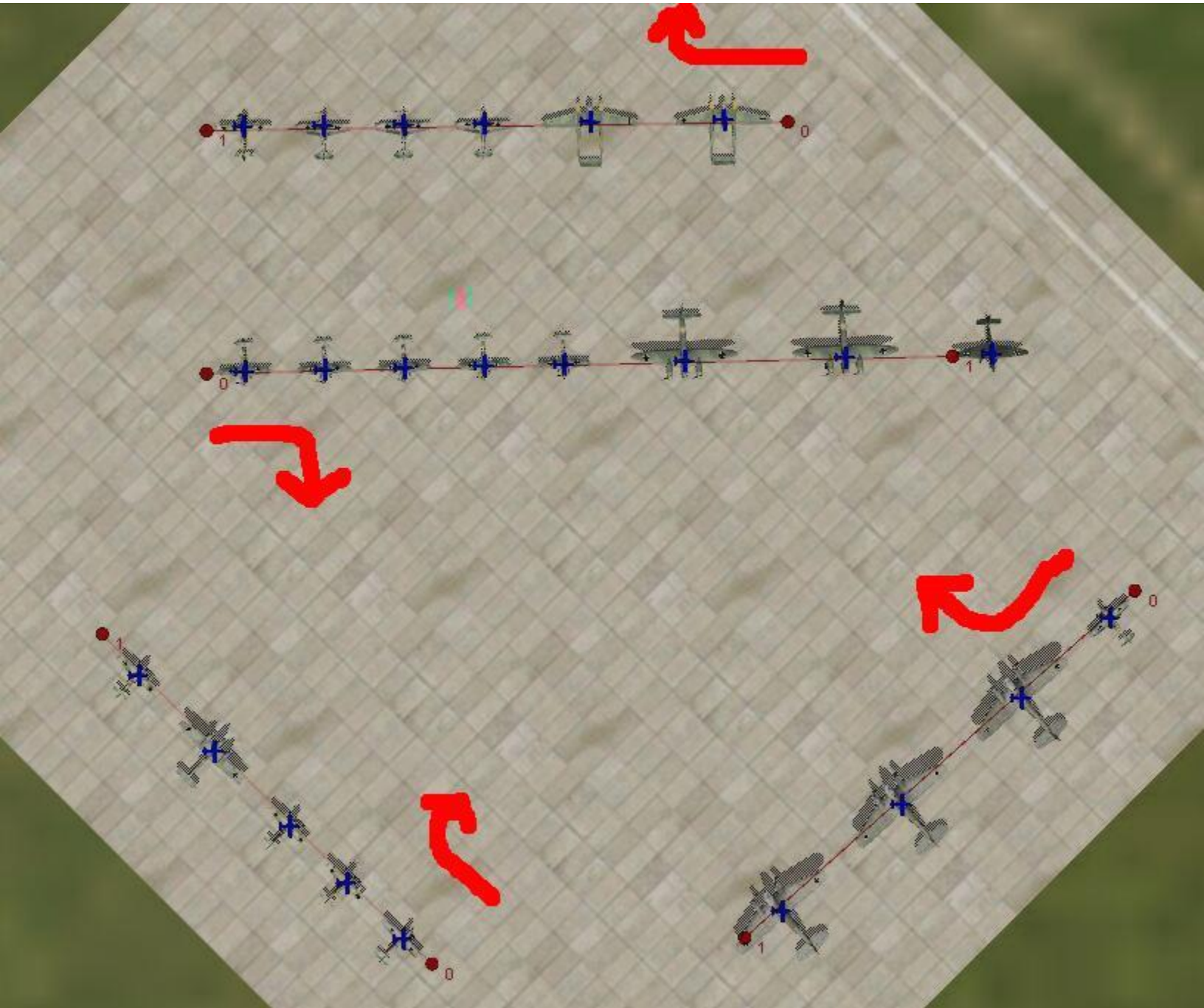


La siguiente imagen muestra como se colocan las aeronaves:



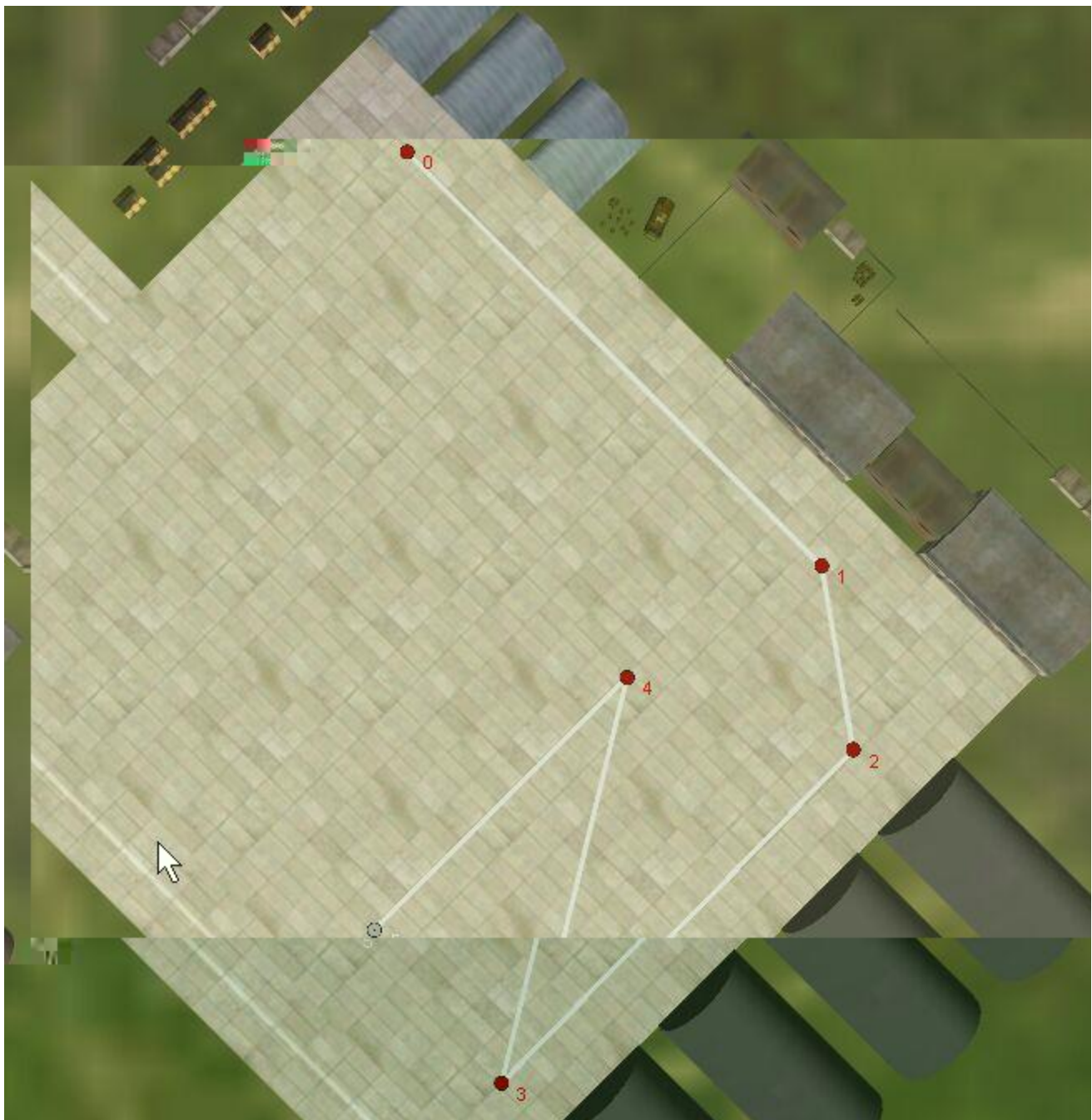


Compara las últimas dos imágenes, mira la **dirección** de los puntos de ruta y como los aviones son ubicados, notarás que los aviones siempre son colocados con la nariz apuntando a la derecha de las líneas. Cuando digo "derecha" quiero decir: imagina que estás de pie en el punto de ruta 0 y miras en dirección al punto de ruta 1, los aviones estarán mirando a la "derecha" de esa dirección. Para entender fácilmente combine las dos imágenes en una sola y dibuje una gran flecha roja de la dirección "derecha". Cuando haces plantillas no observarás los aviones, y tienes que crear esta imagen en tú mente.



De esta manera funcionan las **plantillas de lugares**. Sólo usas una misión para ubicar vuelos, nunca colocas los objetos. Confía en que luego el generador usara ese archivo con los puntos de ruta del vuelo para leer las coordenadas y colocar los objetos que desees. Puedes usar cualquier avión para "dibujar" lugares, no importa el bando o el tipo de avión, el generador colocara aviones estáticos rojos si luego la base es soviética, o azules si la base es alemana. Asegúrate de entender todo hasta este punto antes de leer los consejos que siguen.

Para evitar usar muchos aviones volables y para acelerar la creación de la plantilla, usamos el **mismo** vuelo para hacer varios lugares. Cómo? bien, el generador buscara en **pares** de puntos de ruta porque sabemos que una línea está definida por 2 puntos. Así que cuando crees tus lugares puedes seguir añadiendo puntos usando el mismo vuelo como se muestra en la imagen siguiente.



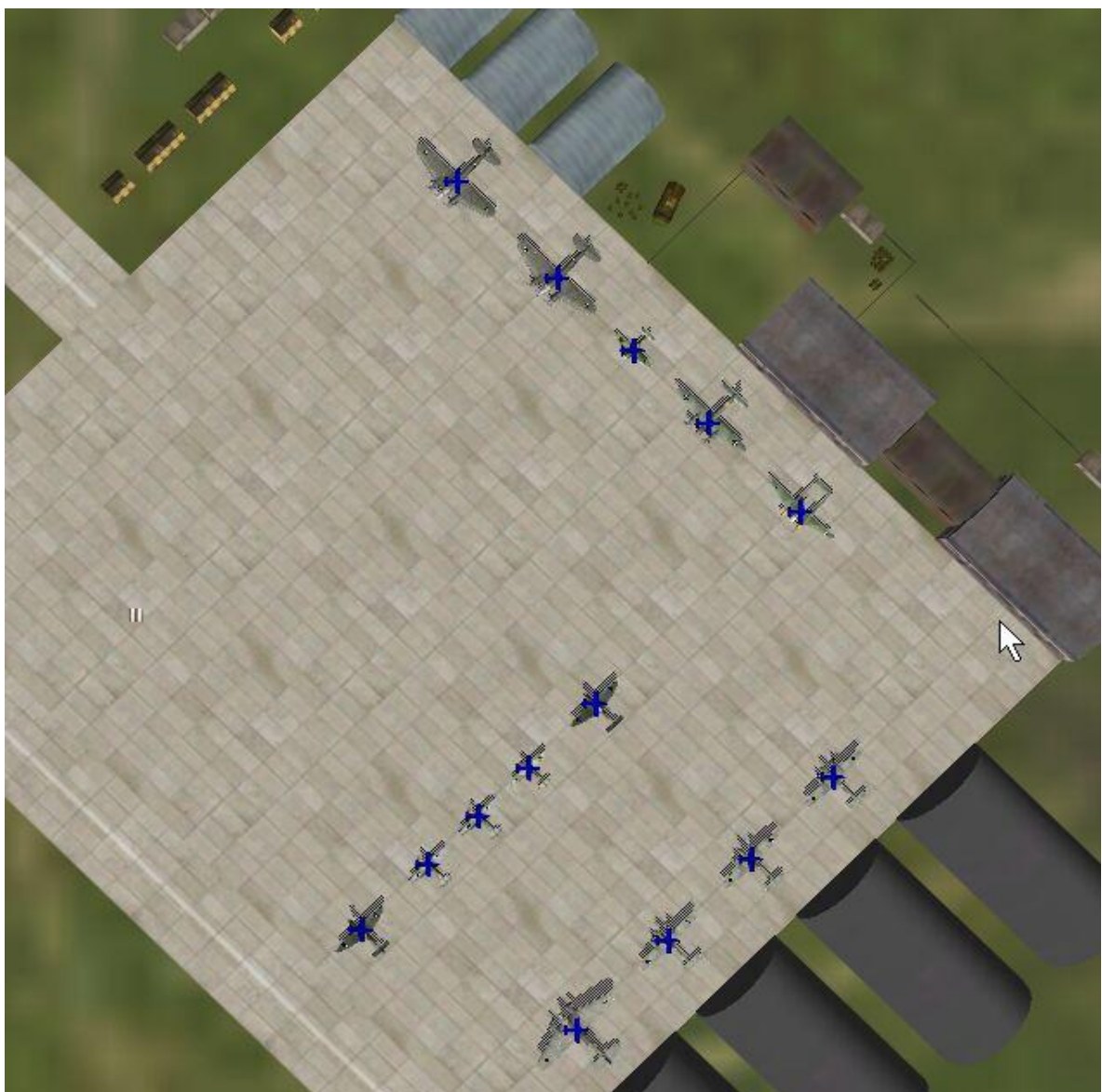


Esos puntos de ruta indican 3 lugares: ubica aviones de PR0 a PR1, ubica aviones de PR2 a PR3, ubica aviones de PR4 a PR5. Cuando "dibujas" esos puntos de ruta tienes que mantener pares de puntos de ruta. Me digo a mi mismo cuando hago estas plantillas:

colocando PR0 pienso: aviones **de aquí**  
colocando PR1 pienso: **hasta aquí**  
colocando PR2 pienso: aviones **de aquí**  
colocando PR3 pienso: **hasta aquí**  
colocando PR4 pienso: aviones **de aquí**  
colocando PR5 pienso: **hasta aquí**

Como puedes ver, eres libre de seleccionar el punto de partida entre lugares. En este ejemplo, las líneas entre PR1 y PR2 no definen un lugar, es sólo una reubicación de la línea y tienes que ignorarla. De igual forma para las líneas entre PR3 y PR4.

Realmente es muy sencillo después de que lo intentas varias veces. La siguiente imagen muestra como los aviones son ubicados usando los lugares mostrados en la imagen anterior:



Y la siguiente imagen es la combinación de ambas, mostrando lo que tienes que "imaginar" al momento de "dibujar" las plantillas. Recuerda, es **MUY** importante usar pares de puntos de ruta, porque definen líneas. Sólo recuerda usar la nemotecnia "**de aquí hasta aquí**".

### 3. Creando Plantillas

Crear una plantilla para un mapa significa que tienes que crear varios archivos. Puedes hacerlos a mano cada uno, pero esto generalmente introduce varios errores, debido a que muchas coordenadas deben ser escritas y un error en una coordenada producirá misiones incorrectas. Así que yo proveo algunas herramientas para crear plantillas de una forma semiautomática.

Creamos plantillas en varias **etapas** ( o pasos), porque algunas etapas necesitan datos de etapas previas. Por ejemplo, el archivo geo\_obj.data contiene cuatro (4) grupos principales de información. Uno de esos grupos son las definiciones de sector, y para escribir las definiciones de sector necesitas tener los puntos de ruta de los tanques, y los puntos de ruta de los tanques se harán usando una plantilla creada después de terminar la "etapa 1", donde definimos aeródromos y lugares de aparcamiento. No te preocupes ahora por esto, lo único importante ahora es saber que hay varias "etapas".

Esta plantilla de creación semi-automática usa scripts de perl, así que necesitas tener perl instalado. Si no tienes perls, no será posible hacer las plantillas como describo en este manual.

El proceso de creación de plantillas requiere que trabajes con el FMB, y tiempo para ejecutar algunos scripts, así que para hacer las cosas fáciles, crea una carpeta en la carpeta missions del FB. Generalmente llamé a la carpeta con el nombre del mapa en el que estoy construyendo las plantillas, por ejemplo para las plantillas de normandy1 tengo la carpeta "**D:\Ubi Soft\IL-2 Sturmovik Forgotten Battles\Missions\normandy1**" donde coloco todos los scripts de perl y mantengo los archivos de las plantillas. Una vez están terminadas las renombro, agregando un código de 2 o 3 letras antes de cada archivo para facilitar el reconocimiento.

Copia todos los scripts que vienen con este manual en la ubicación que quieras.

#### 3.1 Primera Etapa

Vamos a empezar a construir plantillas para el mapa Normandy1. Como se describió antes colocare todos los scripts en una nueva carpeta llamada normandy. Entonces iniciare el IL2FB, y cargare el mapa "normandy1". Lo primero que hay que hacer es colocar un "airfield campfire" en la parte más arriba al extremo derecho, mejor si es un poco **afuera** del último sector letra/número (si puedes, si no colócalo tan lejos como puedas en la parte alta derecha del mapa). Esto será usado por los scripts para determinar los límites del mapa. Guarda el archivo como "**initial.mis**". Este archivo ahora proveerá información importante: los límites del mapa y el nombre del mapa.



La imagen de arriba muestra la localización del "airfield campfire" para el mapa normandy1. Este mapa tiene unas coordenadas en la esquina superior derecha 0-10, así que el fuego de campamento está ubicado en el sector P-11. Por supuesto, las coordenadas P-11 no son mostradas en el FMB. Este cuadrado en la esquina superior derecha tendrá diferentes coordenadas en diferentes mapas.

Este nuevo archivo initial.mis contendrá datos de aeródromos, además de otra información. Para cada aeródromo en el mapa que queramos usar para la campaña, necesitamos recolectar algo de información: las coordenadas de su localización, los dos rumbos de despegue y aterrizaje, la orientación de la pista, los "lugares" donde nos gustaría colocar aviones aparcados y el lugar para un avión estático que será usado para identificar el bando del aeródromo.

Toda esta información es recolectada agregando 3 objetos a cada aeródromo:

1. El primer objeto será un JU-52 estático, colócalo donde quieras.
2. El segundo objeto es un vuelo (cualquier avión, yo uso B17) con 2 puntos de ruta, el 0 es un punto de **despegue (takeoff)** y el 1 es un punto de **aterrizaje (landing)**. Este vuelo proveerá información sobre coordenadas para sitios de despegue y aterrizaje, y además proveerá "una ubicación central" de un aeródromo, y rumbos de pista.
3. El último objeto es un vuelo, **uno diferente** (cualquiera, pero yo uso un B17 de nuevo) que contendrá la información de sitios de aparcamiento para las aeronaves, como se describe en "colocando plantillas". Puedes agregar hasta 9 lugares, eso es hasta 18 puntos de ruta. Recuerda la orientación a la "derecha" y la nemotecnia "de aquí a aquí". Cuando tengas todos estos 3 objetos colocados, selecciona otro aeródromo y repite el proceso agregando un nuevo JU-52, un B17 con 2 puntos de ruta (uno de despegue y otro de aterrizaje), y otro B17 para "dibujar" los lugares de aparcamiento. La siguiente imagen muestra una captura con esos 3 objetos en un aeródromo.



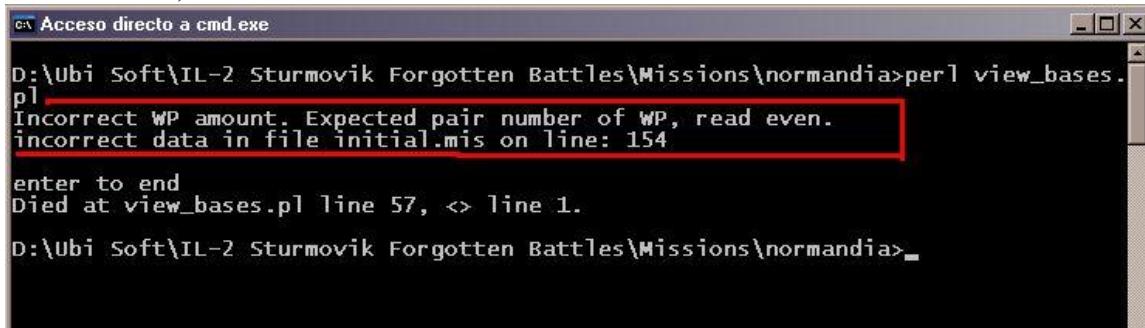
De vez en cuando te gustaría revisar como están colocados, si la orientación es correcta, si tienes pares de puntos de ruta correctos para los lugares de aparcamiento, y así sucesivamente. Para ver una previsualización de tu trabajo, ejecuta el script "view\_bases.pl". Este script leerá tu archivo "initial.mis" y creará un nuevo archivo llamado "preview\_bases.mis". Carga este archivo en el FMB y revisa como están colocados los aviones. Este script mostrara un error si tienes un número impar de puntos de ruta. La previsualización sólo mostrará aviones alemanes, como si la base estuviera bajo control alemán, luego el generador colocara aviones alemanes o soviéticos.

Recuerda seguir trabajando en "initials.mis". Es un **error común cargar la previsualización y entonces seguir agregando Ju-52, sitios de aparcamiento y puntos de ruta de aterrizaje y despegue en la misión de previsualización**. La misión de previsualización es sólo eso, una misión para ver como el generador leerá los puntos de ruta. El archivo importante es "initial.mis", después de que inspecciones todo recuerda cargar de nuevo la misión "initial.mis" para seguir trabajando en ella.

La siguiente imagen es como vera una base en "preview\_base.mis". Observe que el JU-52 estático no se mostrará. "preview\_base.mis" sólo mostrará los "lugares de aparcamiento" de los aviones



Así es como se ve un mensaje de error cuando tratas de hacer una previsualización de un número impar de puntos de ruta. Cuando esto sucede tienes que cargar "initial.mis" en el FMB y buscar donde está el problema. Una recomendación es hacer una previsualización cada vez que terminas una base. Así si hay un problema con pares de puntos de ruta el problema siempre será en la base en la que estabas trabajando. Adicionalmente, el error mostrara en que número de línea es la que está haciendo puntos de ruta impares, así que si quieres puedes borrar manualmente esa línea, editando "initial.mis" con un editor de texto (No es recomendado, mejor mirar los puntos de ruta en el FMB).



```
Acceso directo a cmd.exe
D:\Ubi Soft\IL-2 Sturmovik Forgotten Battles\Missions\normandia>perl view_bases.pl
Incorrect WP amount. Expected pair number of WP, read even.
incorrect data in file initial.mis on line: 154
enter to end
Died at view_bases.pl line 57, <> line 1.
D:\Ubi Soft\IL-2 Sturmovik Forgotten Battles\Missions\normandia>
```

Esto es todo lo que necesitamos para terminar la primera etapa, un "initial.mis" con un fuego de campamento y 3 objetos en cada base. Una vez te acostumbras a hacer estas bases, cada base no tomará más de un par de minutos. La forma en que yo trabajo es primero añadiendo el fuego de campamento, entonces agrego el JU-52 a todas las bases que quiero usar, así son un aviso visual para encontrar las bases. Luego agrego todos los puntos de ruta de despegue y aterrizaje usando B17 para cada base. De esta forma, cuando miró el mapa, tengo que ver "algo azul" (Los Ju-52) y "algo rojo" (los puntos de ruta de los B17). Finalmente agrego los lugares de aparcamiento para cada base, haciendo previsualizaciones de vez en cuando.

Si miras dentro del "initial.mis" que proveo como ejemplo veras los marcadores de línea de frente, no los necesitas aún, es algo que agregaremos en etapas posteriores y se explicara entonces.

Ahora estamos listos para ejecutar el script 1\_stage.pl (ejecútalo en una ventana cmd: perl -w 1\_stage.pl) este creara 5 archivos:

1. **blue\_obj\_mis:** Un JU-52 por base, en los lugares indicados por el JU\_52. Este archivo está listo para usar.
2. **red\_obj.mis:** Un Li-2 por base, en los lugares indicados por el JU\_52. Este archivo está listo para usar.
3. **tank\_wp.mis:** Una plantilla inicial para tanques, trabajaremos en esta en la próxima etapa.
4. **city.mis:** Una plantilla vacía donde agregaremos las ciudades para atacar y objetivos en la siguiente etapa.
5. **geo\_obj.data:** Es un contenido parcial de este archivo, que ahora contiene todas las descripciones de los aeródromos.

Si te da curiosidad puedes mirar dentro del archivo geo\_obj.data recién creado, ahí podrás ver varias líneas con la información de cada aeródromo. Por ejemplo estas son las líneas que representan el aeródromo mostrado en las imágenes de arriba. Esa información es: información general del aeródromo (primera línea), coordenadas de rumbo (AF06:Hx), seguido de varias líneas con información de las coordenadas de lugares de aparcamiento y ángulos para la colocación de aviones (AF06:Px)

```
AF06,aerodromo--D03,38788.61,29109.76,2,-D,03,2,0:2
AF06:H1,38048.61,28764.76,
```

```

AF06:H2,39530.43,29455.75,
AF06:P1,38451.23,29622.01,38526.13,29846.98,378,
AF06:P2,38509.59,29613.09,38578.11,29816.26,378,
AF06:P3,39102.59,29655.61,39159.29,29544.57,152,
AF06:P4,39185.28,29124.04,38982.10,29024.82,243,
AF06:P5,38386.74,29213.82,38443.44,29383.92,378,
AF06:P6,38625.36,29535.12,38665.52,29639.07,381,
#--- End AF06 (aerodromo--D03)

```

No te preocupes si no entiendes esta información, no necesitas hacerlo para seguir trabajando en las plantillas.

## 3.2 Segunda Etapa

Esta etapa es la que toma más tiempo, porque tenemos que trabajar en los puntos de ruta de los tanques y la plantilla de la ciudad. La dividire en 2 partes "A y B" para cada una de ellas.

### A- Plantilla De Puntos De Ruta De Tanques

La plantilla de tanques mantiene "puntos" donde el generador colocara después un campo de defensa (algunos objetos estáticos, AAA, y tanques estáticos) o un grupo de tanques en movimiento (el grupo atacante). Cada sector necesita al menos 2 "punto" para que así sea posible colocar ambos grupos, uno en cada punto. Más lugares de puntos son bienvenidos para que el generador luego pueda escoger diferentes combinaciones de ellos, haciendo las misiones menos repetitivas. La cantidad de "puntos" recomendada para cada sector es hasta 5, y no menor a 3 para hacer algunas variaciones en la selección.

Hay algunos sectores donde los tanques no se pueden mover. Por ejemplo el mar, lagos, sectores cubiertos de bosque, etc. En estos sectores colocaremos 0 puntos, es decir, no habra puntos de salida, y los sectores se volverán inatacables.

Ve al FMB y carga la misión "tank\_wp.mis" creada en la primera etapa. Veras algo como esto (sólo se muestra una pequeña parte)



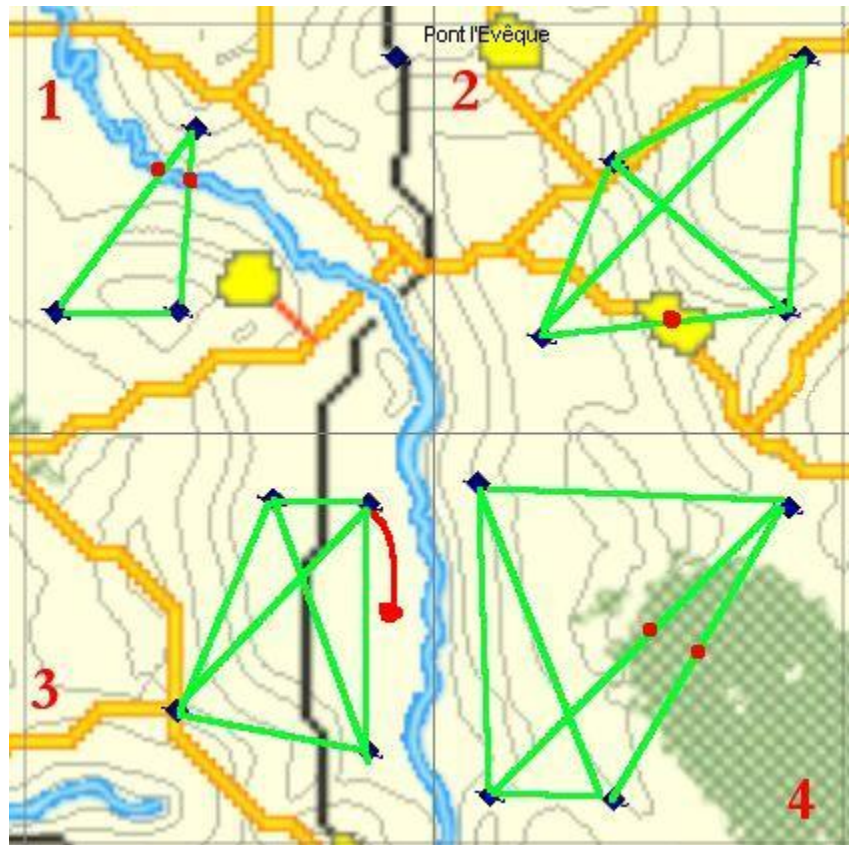
Puedes ver en **todos** los sectores 5 tanques, uno por cada esquina y uno más en el centro del sector. Este archivo es creado en esta forma para editarlo rápidamente. Es más fácil borrar y mover objetos que ya estén colocados en una misión. Cada uno de esos tanques estáticos que ves son "puntos". Como en los lugares de parqueo de usamos vuelos para luego colocar objetos, aquí usamos un tanque estático para indicar las coordenadas donde un campo, o un grupo de tanques en movimiento será colocado. En esa imagen nombre los sectores con números del 1 al 4, sólo para una fácil referencia en este manual.

La imagen previa muestra 4 sectores. La primero que hay que hacer es borrar todos los 5 puntos de los sectores inatacables (mar, lagos y sectores de sólo bosque). Luego, tienes que reubicar los puntos basado en unas simples reglas. Los puntos ( en realidad tanques en esta plantilla) no deben estar muy cerca, todos ellos deben estar dentro del sector, y ningún río ,laguna, ciudad o bosque debe interceptar su camino. Si no puedes colocar 5 de ellos, intenta borrar algunos de ellos para encontrar fácilmente lugares adecuados. La siguiente imagen muestra como edite esos puntos, el sector llamado 1 es inatacable (mar) y borre los 5 puntos. El sector llamado 2 tiene 4 puntos, el sector llamado 3 tiene 5 puntos y el sector llamado 4 tiene sólo 3 puntos:

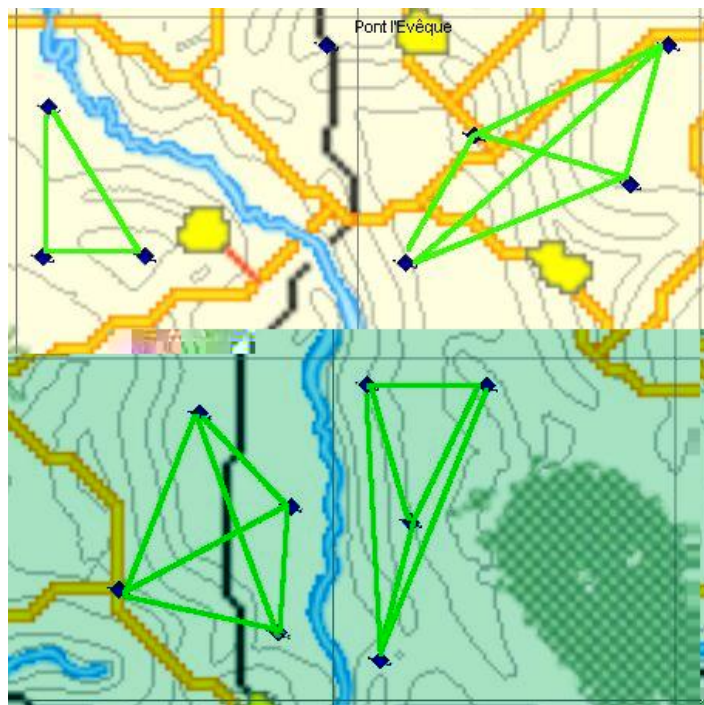


Como puedes ver, todos los puntos cumplen los requerimientos. Ahora te mostrare una imagen con una ubicación incorrecta de los puntos, agregare algunas líneas para conectarlos, y colocare un punto rojo sobre líneas verdes para mostrar un problema. Esas líneas tienen que ser hechas en tú mente cuando estés moviendo los puntos.





El sector llamado 1 tiene un punto en el otro lado del río. El sector llamado 2 tiene dos puntos incorrectamente ubicados porque pasan por una ciudad. El Sector llamado 3 está correcto, pero dos puntos están muy cerca, así que es recomendable mover uno un poco más lejos, como muestra la línea roja. El sector 4 tiene dos vías cruzando el bosque. En la siguiente imagen muestro como son corregidos estos puntos:

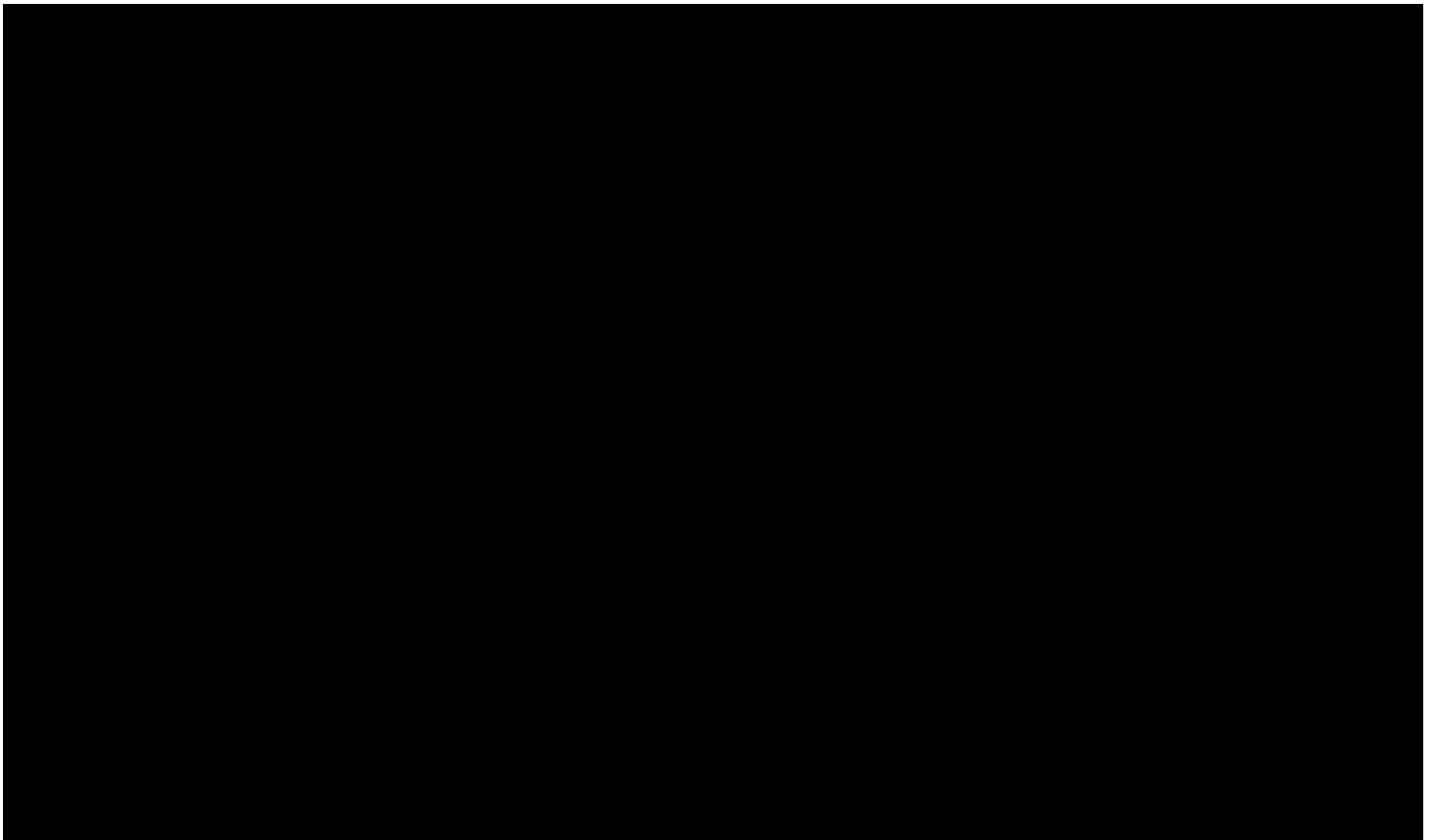


Tienes que revisar sector por sector la ubicación de los puntos, asegúrate que los sectores tienen al menos 2 puntos de ruta, pero es recomendado un mínimo de 3. Además asegúrate de no colocar puntos fuera del sector. Para un trabajo más cómodo, acércate y aléjate con el zoom para que las líneas de cuadrícula sólo muestren cuadros de 10x10, también es recomendable para áreas con colinas y montañas mirar exactamente el lugar donde colocas un punto, para que evites colocarlos en precipicios en las colinas.

Esta plantilla toma algún tiempo, cuando te acostumbras a trabajar con ella, puedes hacer varios sectores en un minuto. Lo más importante es hacer inatacables los sectores que no pueden contener puntos de tanque. Los otros sectores pueden ser editados o corregidos después (problemas con colinas, o un punto al otro lado de un río etc).

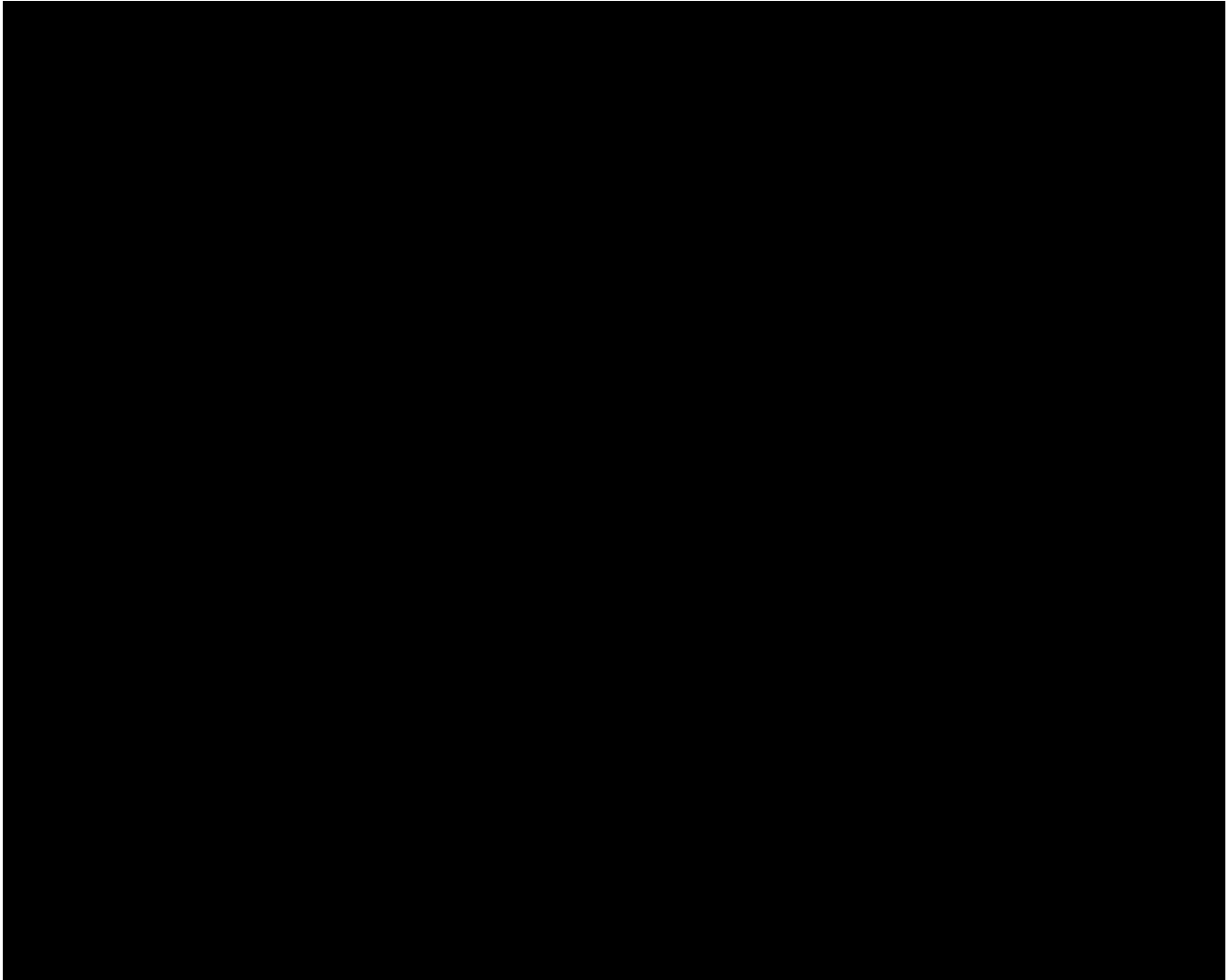
## **B- Plantilla de Ciudad**

Ahora empezamos con la plantilla de la ciudad. Esta plantilla contendrá las ciudades que queremos incluir como objetivos; sólo una ciudad por sector. Las ciudades no pueden estar más cerca de 10 km entre ellas. Carga city.mis en el FMB, lo primero a hacer es colocar un "globo de observación" 90m sobre cada ciudad que queramos incluir. De esta forma revisamos fácilmente que no tenemos 2 ciudades muy cerca. La siguiente imagen muestra como luce la plantilla después de añadir varias ciudades, las líneas son dibujadas sólo para mostrar la distancia que has revisado.

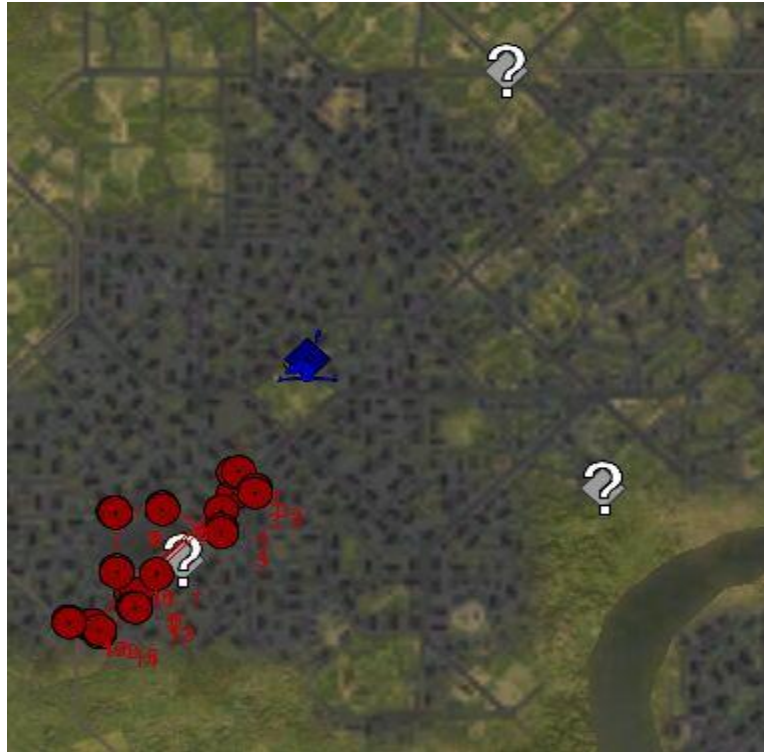


Después de que hayas terminado, tienes que acercarte con el zoom a cada ciudad y añadir varios "fuegos de campamento". Cada "fuego de campamento" será una "zona atacable", agrega al menos uno y máximo 9 zonas para cada ciudad. Todos los lugares marcados con fuego (zonas) tienen que estar cerca de la ciudad, nunca los coloques a una distancia mayor a 5 km. Además, cada zona de objetivo necesita tener 1 km o más de separación con cualquier otra zona, así que deja espacio para colocar objetos y no solaparlos.

La siguiente imagen muestra una ciudad con 3 zonas (3 fuegos), todos ellos dentro del límite de 5 km, y todos ellos separados al menos 1 km entre ellos.



Coloca más zonas en ciudades más importantes (5 o 6 está bien) y menos en pueblos pequeños (2 o 3). Una vez termines de colocar los marcadores de zonas de fuego, tienes que colocar objetos en cada uno de ellos. La siguiente imagen muestra una zona con objetos dejando las otras aún vacías.



La forma de colocar objetos es muy similar al método descrito como "colocar plantillas". De hecho, es el mismo con unas pocas opciones más. Tienes que dibujar un par de puntos de ruta, esos puntos de ruta indican la línea donde los objetos serán colocados. Pero ahora tenemos una característica extra, usando la **altitud** del punto de ruta podemos indicar diferentes tipos de objetos. Piensa en los puntos de ruta como un lápiz, y la altitud es el tipo de lápiz. Así que con diferentes altitudes podemos "dibujar" diferentes objetos.

- Puntos de ruta a 500m son vehículos paralelos a la línea de dibujo.
- Puntos de ruta a 1000m son vehículos rotados a la "derecha" de la línea de dibujo.
- Puntos de ruta a 1500m son trenes paralelos a la línea de dibujo, la locomotora está ubicada en el primer pr, el último vagón en el siguiente PR.
- Puntos de ruta a 2000m son AAA, sólo una AAA por par de puntos de ruta es usada.

De nuevo, no importa que avión uses para dibujar los lugares. Pero ahora es muy importante usar la **altitud** correcta para representar los objetos que quieres colocar. Generalmente uso un avión alemán (PR azul) para "dibujar" los lugares de la AAA, y uso un vuelo rojo para diferentes objetivos. Por ejemplo, con un B17 dibujo todos los vehículos rotados a la derecha. Si hay una vía de tren, agrego un nuevo vuelo de B17, asigno el PR a 1500m y agrego todos los trenes. Finalmente agrego un vuelo alemán (cualquiera, generalmente uso un JU-87) e inserto varios lugares de AAA.



La siguiente imagen muestra como luce un objetivo.



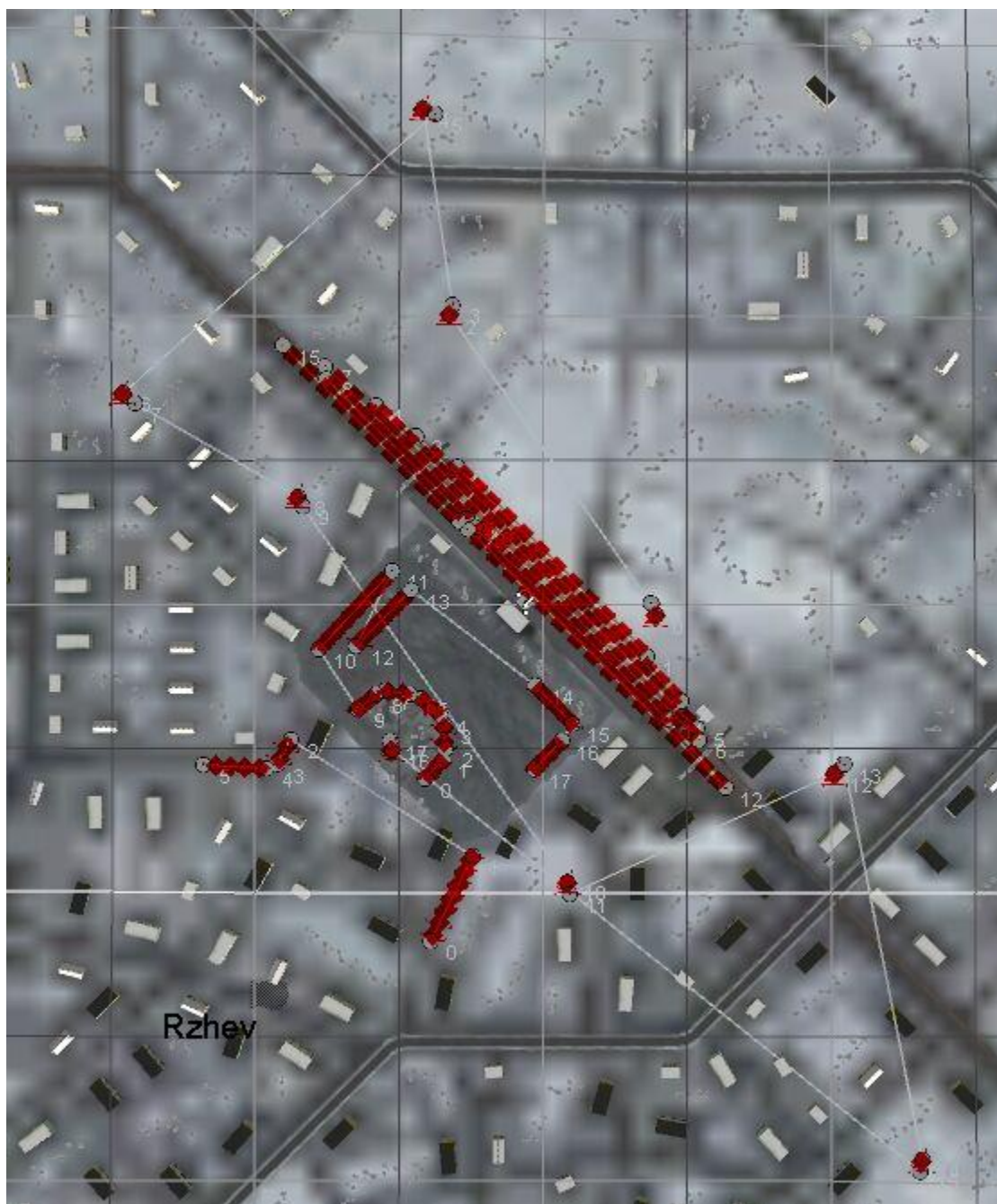
Como antes, tienes un script para previsualizar como serán leídos los objetivos por el generador. Este script es llamado "view\_city.pl" y leerá city.mis (el archivo en el que estás trabajando) y creara un nuevo archivo llamado "preview\_city.mis". Después de agregar una o más zonas, ejecuta el script y entonces carga la misión previsualizada en el FMB. Revisa como se colocan los objetivos, si todo lo que planeaste añadir está en el lugar correcto, etc. Luego regresa y carga de nuevo "city.mis". Como antes, no cometas el error de seguir agregando zonas en la misión previsualizada. Este script también te dirá si hay un problema con puntos de ruta impares.

La siguiente imagen muestra la previsualización de la zona mostrada antes como ejemplo.



Y esta imagen siguiente muestra la combinación de las 2 imágenes previas, allí puedes ver los objetos colocados sobre las líneas que estas dibujando. Esto es lo que tienes que tratar de imaginar cuando agregues los puntos. Observa que todos los objetivos en la previsualización son objetos rojos, pero luego, el generador coloca correctamente el bando basado en quien domina la ciudad.





Bien, ahora estamos listos para ejecutar la segunda etapa del script. Este script leerá ambas plantillas (Plantilla De Puntos De Ruta De Tanques y Plantilla de Ciudad) y **agregara información al archivo geo\_obj.data**. La información agregada es: información de sector, líneas como:

```
SEC-E08,sector--E08,45000,75000,30,1:0
SEC-E09,sector--E09,45000,85000,30,1:3
etc...
```

Información de ciudad, líneas como:

```
# Name F08
SUC02,SUM-city-F08,53945.30,73549.92,-,-,-,-:-:0
CT02,city-F08,53945.30,73549.92,25,tip0,TOTAL_ZONES_HERE,0,25:0
CT02Z0:54427.62,74109.38,1
CT02Z1:55014.97,73130.47,1
CT02Z2:54036.05,72934.69,1
```

Relaciones de ciudad y sector, líneas como:

```
poblado,city-H07,sector--H07
poblado,city-F08,sector--F08
poblado,city-I04,sector--I04
```

### 3.3 Tercera Etapa

Esta última etapa es la personalización de tus plantillas. El archivo `geo_obj.data` está casi completo pero necesita personalización. Edita el archivo con un editor de texto, versa que no hay nombres para las ciudades, y eso es porque tienen que se escritas manualmente. Para cada ciudad-XXX encuentra el nombre de la ciudad en el FMB y reemplázalo. Recomiendo usar buscar y reemplazar global de esta forma, así evitas equivocarte escribiendo nombres. Realmente no importa si el nombre de la ciudad es incorrecto, pero debe estar incorrecto en todos los lugares.

Por ejemplo, la ciudad en el mapa `normandy1`, en el sector F08 es **Bayeux** así que hacemos estos cambios en `geo_obj.data`:

Líneas iniciales afectadas:

```
# Name F08
SUC02,SUM-city-F08,53945.30,73549.92,-
```



```

.
.
.
poblado,Bayeux,sector--F08
.
.

```

Observa que cambie el texto "TOTAL\_ZONES\_HERE" por la cantidad de zonas, en este caso 3. Adicionalmente me gustaría asignar a Bayeux con un radio máximo de suministro de 40 kilómetros en vez del 25 km por defecto. Así que cambio ambos "valores de 25" a 40. Esos valores son el máximo radio de suministro y el actual valor de ratio de suministro (porque el daños es 0 ambos son iguales).

Una vez termine de añadir todos los nombres y asigne correctamente la cantidad de zonas y el radio inicial de suministro, tiene que definir cual será la línea inicial de frente de campaña y hacer que todos los objetos en el archivo goe\_obj.data estén .

Para lograr esto, cargue el "initial.mis" y coloque algunos marcadores de frente azules/rojos. Guarde el archivo y finalmente ejecute el script 3\_stage.pl , este script leerá la línea de frente que acaba de colocar en initial.mis y creara una plantilla frontline.mis. También sincronizara el archivo geo\_obj.data. Ahora casi hemos terminado.

## 4. Aeronaves

El archivo aircraft.data tiene la definición para cada aeronave en la que estés interesado para colocar en la campaña. Además este archivo contiene información de nombres de escuadrón y aviones y vehículos estáticos. Las líneas que comienzan con un numeral (#) son ignoradas.

No hay una manera automática de crear este archivo, así que abre un Nuevo archivo en un editor de texto y escríbelo siguiendo las próximas indicaciones y luego guárdalos como aircrafts.data.

Las siguientes líneas indican que los escuadrones de caza rusos pueden ser 176IAP o 3IAK. La selección será aleatoria. La labor de jabo tendrá al escuadrón 11IAP y transportes (suministro) y bombarderos (suministro) serán el escuadrón 34BAP. Los cazas alemanes serán de los escuadrones I\_JG3,II\_JG3,I\_JG27, los bombarderos usaran los escuadrones Stab\_KG27a,I\_KG27a,II\_KG27a,III\_KG53a,y así sucesivamente. Otras líneas como esta describen diferentes escuadrones para otros países/combinaciones de tareas. Cuando hay más de una definición, se escoge aleatoriamente. Al menos un escuadrón debe estar presente para una definición país+tarea existente.

```

##--red sqds
rusfig=176IAP,3IAK;
rusjab=11IAP;
rusbom=34BAP;
rustrp=34BAP;

```

```
gerfig=I_JG3,II_JG3,I_JG27;
gerjab=I_LG2;
gerbom=Stab_KG27a,I_KG27a,II_KG27a,III_KG53a;
gertrp=KGrzbV102a;
```

```
romfig=Eskadrila54;
romjab=Eskadrila54;
rombom=Eskadrila54;
```

Los nombres de escuadrón no pueden ser los que tú quieras. **Tienen que ser exactamente los mismos usados por el FB.** Si planeas añadir nombres de escuadrones, asegúrate que están disponibles en el FB. Luego, este archivo contiene descripciones del avión y son valores separados por comas, dos puntos la tarea y la cantidad de aviones. Los campos están en orden:

```
army, country+task, name of plane, class air, loadout, fuel, crusier alt, crusier
speed : task, number,
```

Army: 1 para aliados 2 para el eje. El país+tarea (romfig, gerjab, rusfig, hunfig, itajab, etc.) es una combinación de: 3 letras por país + 3 letras por rol. Luego es usado para obtener un nombre de escuadrón como III\_JG52 o 84IAP etc. Nota, actualmente solo 3 naciones son soportadas ( Rusia, Alemania y Rumania) (rus,ger and rom) Esto tiene que ser explicado en badc\_gen\_1.pl para otras naciones. Otros archivos se explican solos.

Las tareas son:

BA : **Bomber Attack.** Grupo de bombarderos atacando ciudades o aeródromos.

SUM: **SU**ply Mission. Grupo de aviones de transporte, haciendo una misión de suministro

BD: **Bomber Defend.** Grupo de bombarderos de defensa, atacando los tanques enemigos entrantes.

EBA: **Escort BA** bombers. Escolta bombarderos de ataque.

ESU: **Escort SU**ply transports. Escolta transportes de suministro.

EBD: **Escort BD** bombers. Escolta Bombarderos de defensa.

ET: **Escort Tanks.** Cuando estás atacando sector aviones realizando cobertura aérea e interceptando los **BD enemigos.**

INT: **INT**ercept. En general, interceptar transportes de suministros transportes y BA enemigos.

El número es la cantidad de aviones de ese tipo y es igual para cada entrada (no obligatorio). Aquí está como luce una entrada para un P-40 cazabombardero:

(NOTA: LA TAREA DE INTERCEPCIÓN ES UNA SOLA "I" no "INT" como se describe arriba)

```
# P40-E
1,rusjab,P-40E,1,air.P_40E,500lb,100,3000,400:BD,6,
1,rusfig,P-40E,1,air.P_40E,500lb,100,3000,400:ET,6,
1,rusfig,P-40E,1,air.P_40E,default,100,3000,400:EBA,6,
1,rusfig,P-40E,1,air.P_40E,default,100,3000,400:EBD,6,
1,rusfig,P-40E,1,air.P_40E,default,100,3000,400:ER,6,
1,rusfig,P-40E,1,air.P_40E,default,100,3000,400:I,6,
```

Para obtener los nombres classair, abre el FMB, carga cualquier mapa pequeño, por ejemplo online1. Entonces inserta el avión del que quieres obtener el nombre airclass. Digamos FW190-A8, así que inserta un vuelo de ese avión con armamento por defecto, entonces inserta otro FW190-A8 y asigna a este una

carga de bomba que quisieras usar, por ejemplo SC500; guarda la misión con cualquier nombre, ejemplo arinames.mis (puedes insertar más vuelos de otras airclass si lo necesitas). Abre la misión con un editor de texto, veras algo como esto:

```
[MAIN]
  MAP Net1Summer/load.ini
  TIME 12.0
  CloudType 0
  CloudHeight 1000.0
  army 1
  playerNum 0
[Wing]
  g0101
  g0100
[g0101]
  Planes 1
  Skill 1
  Class air.FW_190A8
  Fuel 100
  weapons default
[g0101_Way]
  NORMFLY 15421.69 25085.94 500.00 300.00
[g0100]
  Planes 1
  Skill 1
  Class air.FW_190A8
  Fuel 100
  weapons 1sc500
[g0100_Way]
  NORMFLY 15421.69 23440.96 500.00 300.00
[NStationary]
[Buildings]
[Bridge]
[House]
```

Ahí puede ver que el class air es: **air.FW\_190A8**

La armamento por defecto es **default**

La carga de bombas es: **1sc500**

Bien, vamos a construir esta entrada para el FW190-A8 usando armamento por defecto, y una bomba SC500 para los bombarderos de defensa (BD), y asignar 45/300 aviones. Primero escribe una sola línea, para la tarea de intercepción:

```
2,gerfig,FW-190A8,1,air.FW_190A8,default,100,3000,400:I,45,
```

Estos valores son:

2 = bando del eje

gerfig = 3 letras país + 3 letras código

FW-190A8 = es el nombre del avión a ser mostrado en los briefings, reportes y estadísticas

1 = porque es volable por humanos

air.FW\_190A8 = es la class air que encontraste como se explica arriba

default = es el armamento

100 = es la carga de combustible

3000 = es la altitud de crucero

400 = la velocidad de crucero

: es un separador

I = tarea de interceptación (Observa que es una sola letra, no 3 letras "INT")

45 es el número de la cantidad de aviones

, coma final

Cuando tienes esa línea, copia y pega varias líneas (4 más) para obtener esto:

```
2,gerfig,FW-190A8,1,air.FW_190A8,default,100,3000,400:I,45,
2,gerfig,FW-190A8,1,air.FW_190A8,default,100,3000,400:I,45,
2,gerfig,FW-190A8,1,air.FW_190A8,default,100,3000,400:I,45,
2,gerfig,FW-190A8,1,air.FW_190A8,default,100,3000,400:I,45,
2,gerfig,FW-190A8,1,air.FW_190A8,default,100,3000,400:I,45,
```

Finalmente cámbialas para tener las tareas EBA, EBD, ET y BD, asigna una carga de 1SC500 para las tareas ET y BD. Reduce la altitud de crucero a 1000m y asigna un código gerjab para el bombardeo defensivo (BD): yo generalmente agrego una línea comentada (#) con el nombre y otra información al comienzo:

```
# FW-190A8 45/300 planes as Pirx table for map bla-bla at year xxxx...
2,gerfig,FW-190A8,1,air.FW_190A8,default,100,3000,400:I,45,
2,gerfig,FW-190A8,1,air.FW_190A8,default,100,3000,400:EBA,45,
2,gerfig,FW-190A8,1,air.FW_190A8,default,100,3000,400:EBD,45,
2,gerfig,FW-190A8,1,air.FW_190A8,1sc500,100,3000,400:ET,45,
2,gerjab,FW-190A8,1,air.FW_190A8,1sc500,100,1000,400:BD,45,
```

Si investigas otros armamentos, sabrás que los cañones mk108 en el FW-190A8 se llaman "r22mk108" así que tal vez te gustaría agregar esos para la tarea de interceptación. Sólo cambia default por r22mk108 en la línea de tarea, para que se vea así:

```
2,gerfig,FW-190A8,1,air.FW_190A8,r22mk108,100,3000,400:I,45,
```

Repite este método para otros aviones. Yo generalmente hago una misión (airnames.mis) donde inserto todos los aviones que necesitare con varios armamentos. Así después abro la misión para leer todos los datos del air.class y los armamentos que necesito. En tú caso, tal vez necesites P51, P38, P47, 109-G6, B25 etc etc. colócalos en tú misión airnames.mis usando el FMB.

Luego en este archivo tenemos la distribución de los aviones y vehículos estáticos. Las líneas de aviones estáticos son así:

```
ST100,1,I153,vehicles.planes.Plane$I_153_M62,15:150
ST101,1,I16,vehicles.planes.Plane$I_16TYPE24,15:350
ST102,1,IL2,vehicles.planes.Plane$IL_2_1940Late,20:400
ST103,1,LaGG3,vehicles.planes.Plane$LAGG_3SERIES4,15:500
```

Y los campos son:

STx+ 2 números: x=1 para aliados y x=2 para el aviones estáticos del eje

Bando

Nombre externo (para reportes)

El nombre del objeto en FB

La envergadura aproximada del objeto

: número.

Los números son valores acumulativos, y el último avión estático debe tener el número 1000, tienen que estar en orden, por ejemplo arriba; tenemos 150 I-153 por 1000 aviones, entonces tenemos 200 (350-150) I-16type24, luego tenemos 50 (400-350) Il2 1940, después 100 (500-400) LaGG3 y así sucesivamente. Observa que mezclados con los aviones estáticos tenemos algunos camiones de combustible, estarán colocados como los aviones, simulando la carga de combustible. Finalmente la definición de vehículos estáticos, es casi la misma que la de los aviones estáticos, usando valores acumulativos de frecuencia hasta 1000<sup>3</sup>. Aquí algunos ejemplos para vehículos del bando azul:

```
SV201,2,VW82,vehicles.stationary.Stationary$VW82,15:80
SV202,2,VW82t,vehicles.stationary.Stationary$VW82t,15:160
SV203,2,OpelKadett,vehicles.stationary.Stationary$OpelKadett,15:260
SV204,2,OpelBlitz36S,vehicles.stationary.Stationary$OpelBlitz36S,15:340
SV205,2,OpelBlitz6700A,vehicles.stationary.Stationary$OpelBlitz6700A,15:440
SV206,2,Kettenkrad,vehicles.stationary.Stationary$Kettenkrad,15:500
```

Y los campos son:

SVx+ 2 números: x=1 para aliados y x=2 para aviones estáticos del eje

Bando

Nombre externo (para reportes)

El nombre del objeto en FB

La envergadura aproximada del objeto

: número.

## 5. Imágenes

La imagen a ser usada es BM 24bits/píxeles imagen que debes hacer con el programa de tú preferencia. Toma varias capturas de pantalla al mapa en el FMB. Luego edita y redimensiona para obtener un ancho de 900 píxeles. Puedes usar otros tamaños, pero no es recomendado un ancho mayor a 900 debido al desempeño y la carga del servidor. Más pequeño puede ser difícil ver la información. Siempre usa un ancho divisible en 4, porque los scripts no manejan relleno. Agrega a la imagen los nombres de las ciudades. Vamos a llamarla imp.data. GIMP tiene un buen editor de imágenes mapa.

Cuando construyas la imagen, cuenta cuantos píxeles por sector tienes. Luego necesitas que esa información se escriba en config.pl (explicado después). Por ejemplo, si tú imagen tiene 900 píxeles de ancho y el mapa tiene 15 sectores, la cuenta de píxeles por sector es 60. Tal vez uno menos porque los mapas tienen sectores parciales al final, donde están las coordenadas.

3. (using accumulative values frequency up to 1000). En el original NdT



La imagen del mapa es solo un archivo HTML, señalando a la imagen (no está, señala a la predeterminada front.jpg) y luego la definición del mapa así:

```
<br>
<p> Usa el mapa para encontrar el nombre del área de ataque</p>
<IMG SRC="/images/front.jpg" WIDTH=xxx HEIGHT=xxx BORDER=0 alt="" USEMAP="#map">
<MAP NAME="map">
<AREA SHAPE="RECT" COORDS="0,31,46,77" HREF="/no_click.html" alt="sector--A16">
.
. (otras líneas imap)
.
</MAP>
```

Nota que tienes que asignar ancho y alto, usualmente la herramienta imagemap hace eso.

## 6. Configuración Final

Una vez que tienes todos los archivos, es tiempo de renombrar archivos, para que se vuelvan únicos. Usando una combinación de 2 o 3 letras estaría bien. Así que renombramos para normandy1 usando:

```
initial.mis      -> NM1_initial.mis  (Esto ya no se usa, pero guárdalo por si acaso)
geo_obj.data     -> NM1_geo_obj.data
red_obj.mis      -> NM1_red_obj.mis
blue_obj.mis     -> NM1_blue_obj.mis
tank_wp.mis      -> NM1_tank_wp.mis
city.mis         -> NM1_city.mis
frontline.mis    -> NM1_frontline.mis
aircrafts.data   -> NM1_aircrafts.data
NM1.BMP          -> NM1_000.BMP  (000 es solo para decir esta es la imagen original
del mapa).
imap.data        -> NM1_imap.data
```

Ahora tienes que agregar esta información del mapa al archivo config.pl. Primero define un nombre de mapa (\$MAP\_NAME\_LONG), un texto corto con alguna información clara del mapa. Después de esto varias líneas tienen que ser incluidas dentro de un condicional if:

- 1) La información de la imagen del mapa
- 2) Los tipos de tanques a ser usados como defensa y ataque para ambos bandos, y la presencia de AAA.
- 3) El nombre de archivo para este mapa.
- 4) La lista de aviones a ser usada en este mapa. (para la forma impresa takeslot.pl<sup>4</sup>)El MAP\_NAME\_LOAD (en nuestro ejemplo es normandy/load2.ini) puede ser extraído mirando en cualquier plantilla recién creada. Abrir con un editor de texto y encontrar la línea que comienza con "MAP" entonces copiar la información que sigue:

```
$MAP_NAME_LONG="Normandy. June 1944";

if ( $MAP_NAME_LONG eq "Normandy. June 1944") {

    # 1 . Información de la imagen del mapa
    $ANCHO=900;          # NM1_000.bmp ancho de la imagen en pixels
    $ALTO=600;           # NM1_000.bmp alto de la imagen en pixels
```

4. (for takeslot.pl form printing) en el original NdT

```

$H_BLOCK_PIX=60; # tamaño horizontal de sector en pixeles
$V_BLOCK_PIX=60; # tamaño vertical de sector en pixeles
$LETRAS=16;      # letras de sector del mapa, 1 en exceso (La última de
Normandy1 es 0, eso es 15)
$NUMEROS=11;     # número de sectores en el mapa, 1 in exceso (En Normandy1 el
último número es 10)

# 2. Tipos de tanques y colocación de aaa
$ALLIED_TANKS_ATTK="Armor.3-BT7";
$AXIS_TANKS_ATTK="Armor.3-PzIIIIG";
$ALLIED_TANKS_DEF="vehicles.artillery.Artillery\$BT7"; # notice "escaped $"
$AXIS_TANKS_DEF="vehicles.artillery.Artillery\$PzIIIIG"; # notice "escaped $"
$AAA_IN_CHAMPS=0;      # colocar o no colocar AAA en campos de defensa5
$LATE_AAA_IN_CHAMPS=0; #colocar o no colocra AAA avanzada como nimrod y M166

# 3. Nombres de archivo
$GEOGRAFIC_COORDINATES="NM1_geo_obj.data";
$FRONT_LINE="NM1_frontline.mis";
$RED_OBJ_FILE="NM1_red_obj.mis";
$BLUE_OBJ_FILE="NM1_blue_obj.mis";
$CITY_PLACES="NM1_city.mis";
$TANKS_WP="NM1_tank_wp.mis";
$FRONT_IMAGE="NM1_000.bmp";
$IMAP_DATA="NM1_imap.data";
$MAP_NAME_LOAD="Normandy/load2.ini";
$FLIGHTS_DEF="MM1_aircrafts.data";

# 4 . Datos para takeslot.pl los nombres de aviones deben ser exactamente los
mismos usados en aircrafts.data
@VVS_SUM_PLANES=("Li-2","TB3-4M-34R");
@VVS_BA_PLANES=("IL2 series1","IL2 series2","SB-2M100A","SB2-M103","SU-2","TB3-
4M-34R");
@VVS_AI_PLANES=("Li-2","IL-4-DB3B","PE-2 1940","R-10","SB-2M100A","SB2-
M103","SU-2");
@LW_SUM_PLANES=("HE-111H2","JU-52");
@LW_BA_PLANES=("BF-110C4B","HE-111H2","JU-87B2","JU-88A4");
@LW_AI_PLANES=("JU-52","BF-110C4B","JU-88A4");
}

```

Los valores en VVS\_SUM\_PLANES, VVS\_BA\_PLANES, VVS\_AI\_PLANES son las listas de los aviones de suministro permitidos, los aviones de ataque y bombardeo y IA. Igualmente para los campos LW\_\* con aviones del eje. Todos estos valores son usados en take\_slot.pl para hacer la forma de selección de avión a la creación de la misión<sup>7</sup>. Los nombres de los aviones necesitan ser **exactamente** los mismos a los que se escribieron en el archivo aircrafts.data. Los que imprimí como ejemplo no son para Normandía, solo son un ejemplo.

Cuando estableces los tipos de tanques de ATAQUE, asegúrate de incluir en una variable llamada (TANK\_REGEX) una cadena para identificarla. Por ejemplo si agregas tanques KV1, tienes que hacer esta línea:

```

$TANK_REGEX="Pz|T34|ValentineII|BT7|M4A2|T70"; # cadena de identificación del tanque
usa | como separador

```

5. *place or not place AAA on field champs* en original NdT

6. *(place or not place advanced AAA, like nimrod and M16)* en original NdT

7. *All this values are used on take\_slot.pl to make the plane selection form at creation of missison.* en el original NdT

Ahora se ve así:

```
$TANK_REGEX="Pz|T34|ValentineII|BT7|M4A2|T70|KV1"; # cadena de identificación del tanque usa | como separador
```

Advierte que los nombres de los tanques de defensa, los estáticos, tienen un \$, agregando un backslash antes de \$. Por ejemplo, el nombre del objeto del FB para un BT7 estático es:

**vehicles.artillery.Artillery\$BT7** y tienes que agregar el backslash antes del símbolo \$, así:  
**vehicles.artillery.Artillery\BT7.**

## 7. Comentarios

Seguramente las plantillas tardan mucho en hacerse, y varios archivos deben ser editados, creados y cambiados. Después de que todos los archivos estén listos, tienes que probar si están correctos, por ejemplo, son usados todos los aviones, están presentes todas las ciudades y objetivos de suministro de ciudades, la imagen de frente e imagen de suministro están correctamente creadas etc.

Publicare más plantillas, las que usare en bellum: Kursk, Smolensk y Balaton. No hare más plantillas que las mencionadas anteriormente. Todas las preguntas son bienvenidas y seguramente ayudare a cualquiera con la construcción de plantillas. Al menos a quienes prueben que han gastado algo de tiempo para entender este manual ( además de mi mal ingles ) :).

Usa el foro de Bellum para preguntas (<http://il2fb-bellum.com.ar/foro>) y ayuda con problemas con las plantillas que estés haciendo (Si es posible proporciona los archivos en los que estas trabajando).

**Nota:** Se que pequeños mapas del FB pueden funcionar con el badc, pero realmente no sé si el generador/parser puede manejar grandes mapas, aquellos que tienes 2 letras por sector (Crimea, Finlandia, etc).

Eso es todo. :)