

## Полный редактор миссий для игры «Ил-2 Штурмовик. Битва за Сталинград.»

(на примере написания одиночной миссии)



Вводная часть беззастенчиво позаимствована у камрада **No601\_Prangster**. Перевод вольный, но сути это не меняет; практическая часть основана на своём конкретном примере; имеется некоторое кол-во вставок из руководства по редактору для игры **«Rise of Flight»** (РоФ), т. к. он является «предком» рассматриваемого редактора.

# Оглавление

<b>Часть первая — файлы миссии.....</b>	<b>3</b>
1.1 Файлы.....	3
1.2 Открытие миссии в редакторе.....	3
<b>Часть вторая — основные сведения.....</b>	<b>4</b>
2.1 Как работает редактор.....	4
2.2 Общий обзор.....	4
2.3 Объекты.....	6
2.4 Команды.....	6
2.5 Группы.....	7
<b>Часть третья — практика.....</b>	<b>7</b>
3.1 Выбор нужной карты в редакторе.....	8
3.2 Основные настройки миссии.....	9
3.3 Оформление полётной карты.....	12
3.4 Декорации.....	15
3.5 Реализация сценария.....	18
3.6 Подготовка аэродрома.....	19
3.7 Настройка самолётов.....	22
3.8 Полёт по маршруту. Взлёт. Посадка.....	24
3.9 Наземная техника — движущаяся и статичная.....	29
3.10 «Закрытие» целей. Счётчики и субтитры.....	35
3.11 Боты. Атака наземных и воздушных целей.....	37
3.12 ПВО.....	51
3.13 Поезда. Артиллерия. Антураж.....	52
3.14 МСУ, не вошедшие в миссию.....	61
<b>Приложения.....</b>	<b>62</b>
1. Типы линий.....	62
2. Стрелки.....	65
3. Расшифровка меню назначения связей.....	66



## Часть первая — файлы миссии.

### 1.1 Файлы.

На текущий момент, любая миссия игры **Ил-2 БзС** — это совокупность следующих семи файлов: *Mission File.eng*, *Mission File.ger*, *Mission File.pol*, *Mission File.rus* — файлы локализации — содержат брифинг миссии, титры, имена иконок — всю текстовую информацию на русском, английском, немецком и польском языках, примерно соответствуют файлу *.properties* из игры **«Ил-2 Забытые Сражения»** («старичка Ил-2»).

*Mission File.list* — содержит ссылки на файлы локализации и на все сторонние файлы (например, медиа), если они есть.

*Mission File.Mission* — собственно, сама миссия — несжатый текстовый файл, примерно соответствует файлу *.mis* из «старичка».

*Mission File.msnbin* — бинарный файл миссии, ускоряющий процесс её загрузки в редакторе и время скачивания/загрузки на сервере.

Все файлы мультиплеерных миссий хранятся в **«...data\Multiplayer\Dogfight»**, а одиночных миссий — в **«...data\Missions»**.

### 1.2 Открытие миссии в редакторе

Если у вас имеется файл **.Mission** — вы можете открыть и редактировать его, просто указав его местоположение, если имеется только файл **.msnbin** — он не будет виден сразу, чтобы открыть его, необходимо начать набирать имя миссии в строке **File name** (см. **рис.1**), возможно, вам придётся набрать имя миссии полностью, включая расширение файла (**.msnbin**), а затем нажать клавишу **Enter**.

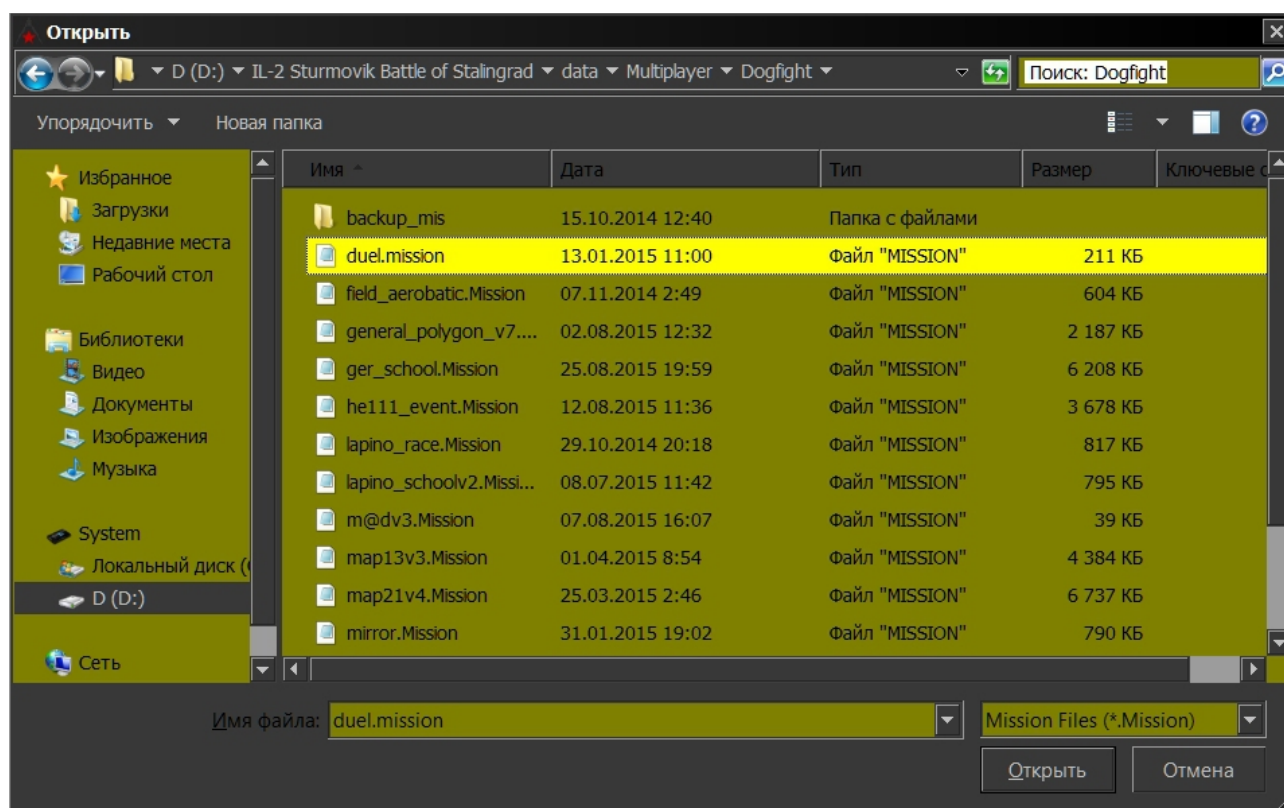


рис.1

## Часть вторая — основные сведения.

### 2.1 Как работает редактор

Полный редактор миссий (далее - редактор) позволяет создавать и редактировать мультиплеерные и одиночные миссии для игры «Ил-2 Штурмовик. Битва за Сталинград». Редактор является отдельной программой, но использует те же ресурсы, что и сама игра, поэтому вы не можете запускать редактор и игру одновременно. Редактор находится в папке с игрой по пути: «...bin\editor\STEditor.exe».

Миссии в редакторе содержат объекты и команды. К объектам относятся: здания, наземная техника, самолёты и эффекты (дымы, сирены и проч). Команды (или **MCU — mission control units**) управляют поведением объектов. Говоря упрощённо, создание миссии — есть расстановка объектов и соединение их с командами различными видами связей, о которых чуть ниже.

### 2.2 Общий обзор.

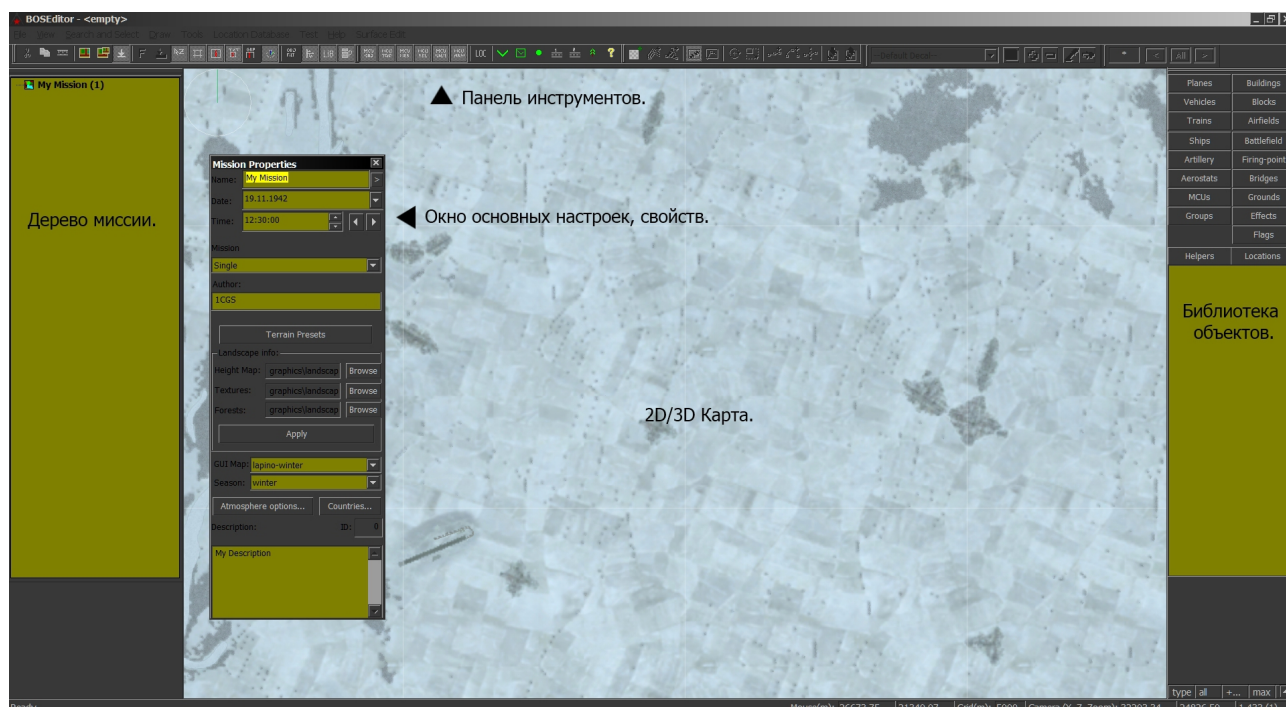


рис.2

**2D/3D карта** — собственно, область, в которой вы создаёте миссию — расставляете объекты, назначаете команды и т. д. (о нюансах загрузки в редактор конкретной, нужной вам карты будет сказано отдельно). Имеются следующие виды этой области, переключение между которыми осуществляется на панели инструментов (**Toolbar**):

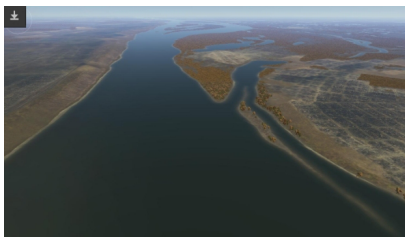
2D Топографическая карта:



2D Карта местности (или физическая карта)...



... с переключением на 3D Вид (свободная камера):



Также, на панели инструментов имеется кнопка, включающая мини-карту, для облегчения навигации:



Сама навигация осуществляется при помощи мыши:

В 2D режиме, при зажатой правой кнопке, вы перемещаетесь по карте, а колесом/средней кнопкой — изменяете масштаб.

В 3D режиме — зажав левую кнопку мыши, вы осматриваетесь вокруг. Зажав правую кнопку — перемещаетесь вперёд, назад, вправо и влево. А при помощи колеса/средней кнопки — перемещаетесь выше или ниже.

**Object Library** (Библиотека объектов) — здесь находятся все объекты и команды, которые можно использовать в миссии (сгруппированы по типам). Нажимая кнопки на этой панели, внизу вы будете видеть список объектов/команд соответствующего типа. Для добавления объекта/команды в миссию, выберите их в списке и расположите на карте левым щелчком мыши.

**Mission Tree** (Дерево миссии) — список всех объектов/команд, уже добавленных в миссию.

**Mission Properties** (Окно свойств) — плавающее диалоговое окно, отображающее основные свойства миссии или выбранного объекта. Оно может быть закрыто, для того чтобы открыть его снова, просто щёлкните правой кнопкой мыши на карте или по любому объекту и выберите **Properties** (Свойства).

Вверху этого окна, справа от поля **Name** (Имя), расположена кнопка «>», которая открывает окно **Advanced properties** (Дополнительные свойства или Расширенные настройки), это окно также может быть открыто двойным левым щелчком мыши по «живому» (см. ниже) объекту.

## 2.3 Объекты.

Все объекты в миссии делятся на «живые» и «неживые» (в дословном переводе используется понятие «сущность»). «Неживые» объекты не взаимодействуют друг с другом и ими нельзя управлять при помощи команд. Более того, наземная техника и самолёты просто не появятся в миссии, если не будут «оживлены». Отвечает за это кнопка **Create Linked Entity** в окне свойств объекта:

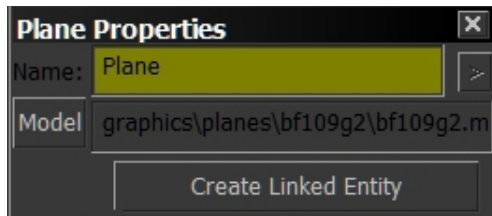


рис.3

По сути, «неживые» объекты в миссии — это декорации из вкладок **Buildings, Blocks** и **Bridges** библиотеки объектов — постройки, статичные самолёты, автомобили и т. п. Миссия может содержать множество таких объектов, т. к. они потребляют мало ресурсов компьютера и практически не влияют на производительность миссии.

**Внимание!** Все объекты, которые вы добавляете в миссию из библиотеки — изначально «неживые» и если вы хотите, чтобы они действовали в миссии, не забывайте нажимать кнопку **Create Linked Entity**.

## 2.4 Команды.

**Mission Control Units** или **MCU** — определяют поведение объектов в миссии — говорят им, что и когда надо делать. Для этого они должны быть присоединены к объектам посредством связей (**links**), которые бывают четырёх видов:



рис.4

**Target Link** — целевая связь (красная), включает («дёргает») **MCU**, к которому присоединена, например, таймер на **рис.4**, по истечении заданного времени, «дёргнет» команду активации и объект (в данном случае — самолёт) появится в миссии.

**Object Link** — объектная связь (зелёная), связывает команду с объектом, на **рис.4** эта связь обеспечивает активацию самолёта при срабатывании соответствующей команды.

**Event Link и Report Link** — «события» и «рапорты» — исходящие от объекта связи (фиолетовые/синие), срабатывающие при изменении состояния объекта. Например, если в качестве «события» выбрана смерть объекта, то фиолетовая связь на **рис.4** «дёргнет» таймер, к которому присоединена, когда самолёт будет уничтожен. О рапортах ещё будет сказано отдельно, при непосредственном описании миссии.



## 2.5 Группы.

Объекты и **MCU** могут быть объединены в группы. Для того, чтобы создать группу, выделите несколько объектов/**MCU** последовательно, с зажатой клавишей **Ctrl** (или просто обведите их рамкой, зажав левую кнопку мыши), затем щёлкните правой кнопкой мыши где угодно, но не снимая выделения, и нажмите **Group Objects** (сгруппировать объекты).

Группы могут быть самостоятельными элементами миссии (например, одна зенитка со своей логикой стрельбы), а могут быть связаны между собой любым типом связей, если связаны объекты/**MCU** в этих группах.

Группы можно копировать, вставлять, вырезать (**Ctrl+C**, **Ctrl+V**, **Ctrl+X**), а также сохранять в отдельную папку по вашему выбору (в контекстном меню: **Save Selection to File**) и импортировать их оттуда в любую миссию (меню **File ---> Import from file**). Если группа создана на одной карте, а импортируется на другую, она может оказаться далеко от рабочей области и даже за пределами карты, но её легко найти в дереве миссии и переместиться к ней, используя функцию «**Move Camera to Object**» в контекстном меню.

Группы — это удобно! Они позволяют избегать путаницы от множества объектов на карте и применять «домашние заготовки», что существенно сокращает время работы над миссией. Для редактирования объектов в группе, выберите её, щёлкните правой кнопкой мыши и нажмите **Set Group as Working**, для выхода из группы — нажмите **Reset Working Group**.

Группы могут включать в себя другие группы.

=====

## Часть третья — практика.

Здесь мы простимся с руководством уважаемого зарубежного товарища и далее будет описано создание одиночной миссии по образцу подходящей по времени и месту действия миссии из игры «**Ил-2 Штурмовик. Забытые Сражения**». Такая миссия легко находится в числе стандартных миссий для самолёта Ил-2 мод. 42:

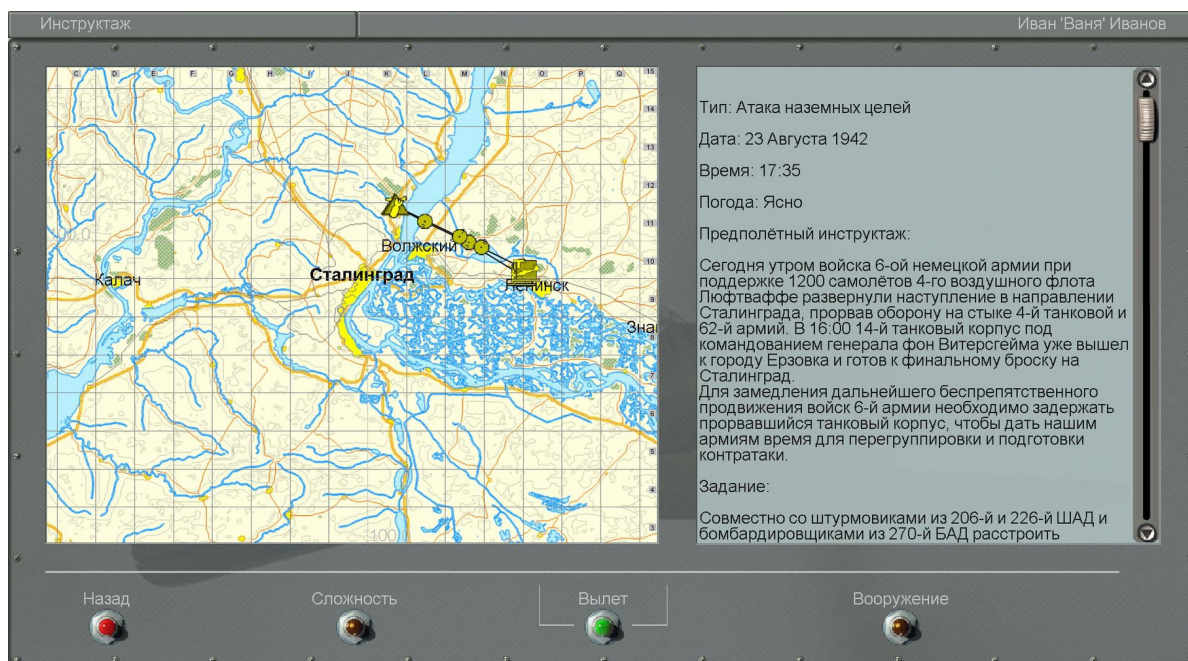


рис.5



Попробуем сделать максимально точную копию этой миссии, покажем, так сказать, «преемственность поколений» :)

### 3.1 Выбор нужной карты в редакторе.

Итак, открыв редактор, видим пустую карту «Лапино» (причём, скорее всего, зимнюю):

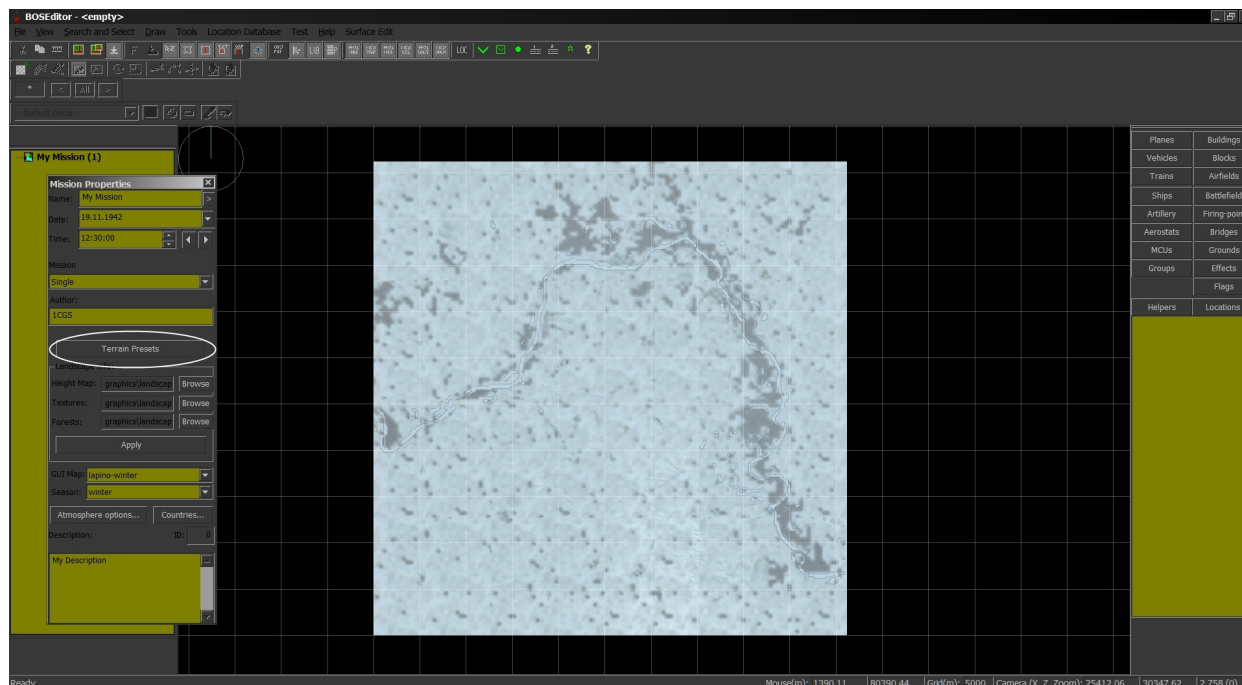


рис.6

Но нам нужна карта Сталинграда, поэтому, первым делом, меняем «Лапино» на «Сталинград» — основную карту игры. Для этого жмём кнопку **Terrain Presets** (см. **рис. 6**), в появившемся окне выбираем **stalingradsummer** (в миссии лето) и нажимаем **Apply** (применить):

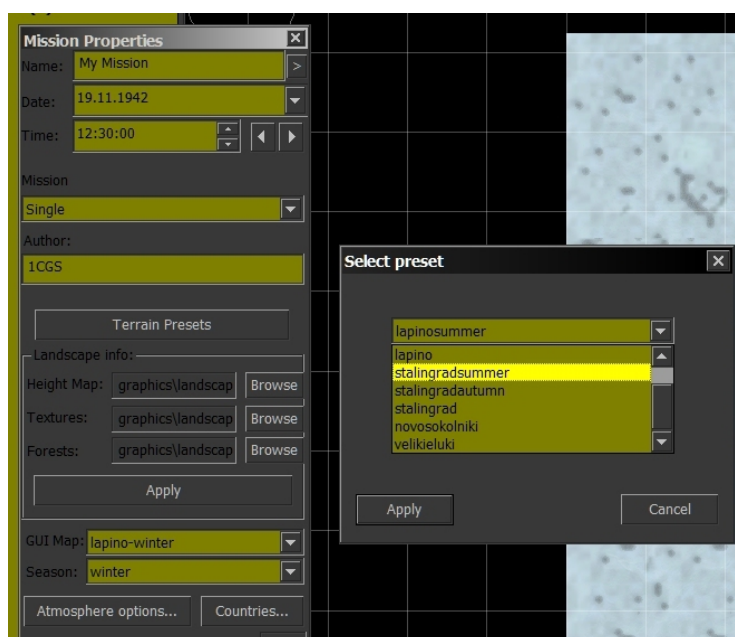


рис.7

Здесь редактор предложит сохранить будущую миссию — так и сделаем:

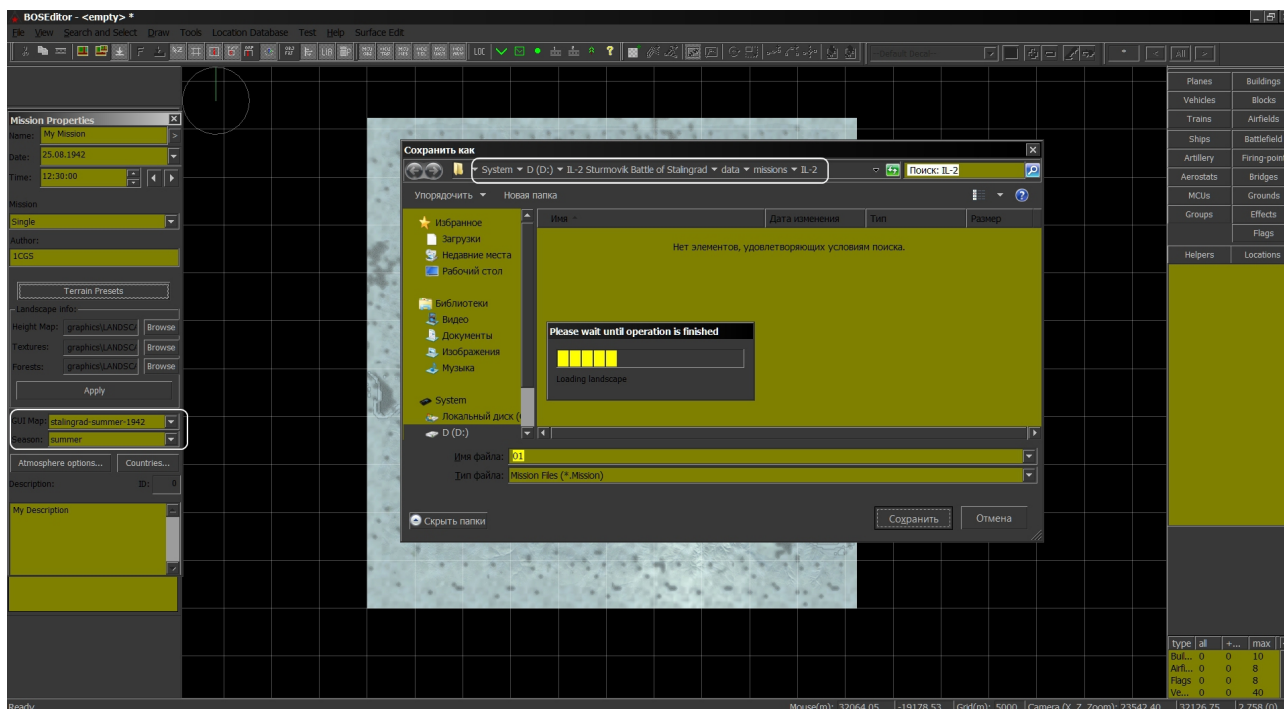


рис.8

Теперь миссия находится по пути «...data\Missions\IL2», называется «**01**» и найти её в игре можно будет в разделе «Сценарии». Папка «**IL2**» создана только для того, чтобы файлы этой миссии лежали отдельно от всех остальных. Можно было сохраняться попросту в папку «**Missions**».

Не забываем проверить, что **summer** (лето) выбрано в поле **Season** (время года), а в поле **GUI Map** стоит **stalingrad-summer-1942**, чтобы топографическая карта совпадала с физической картой местности.

### 3.2 Основные настройки миссии.

Обратимся снова к окну **Mission Properties**, время года уже задано, теперь надо написать название, выставить дату, время начала миссии, указать тип миссии и своё авторство (по желанию). Тип миссии выставляем **Single** для одиночной миссии и **Deathmatch** для мультиплеерной, прочие типы используются генератором кампании игры:

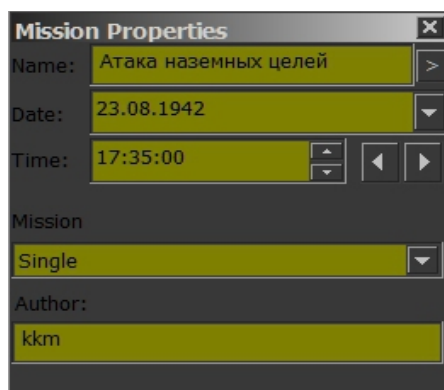


рис.9

Далее, определим погоду в миссии — настроить её позволяет кнопка **Atmosphere options**:

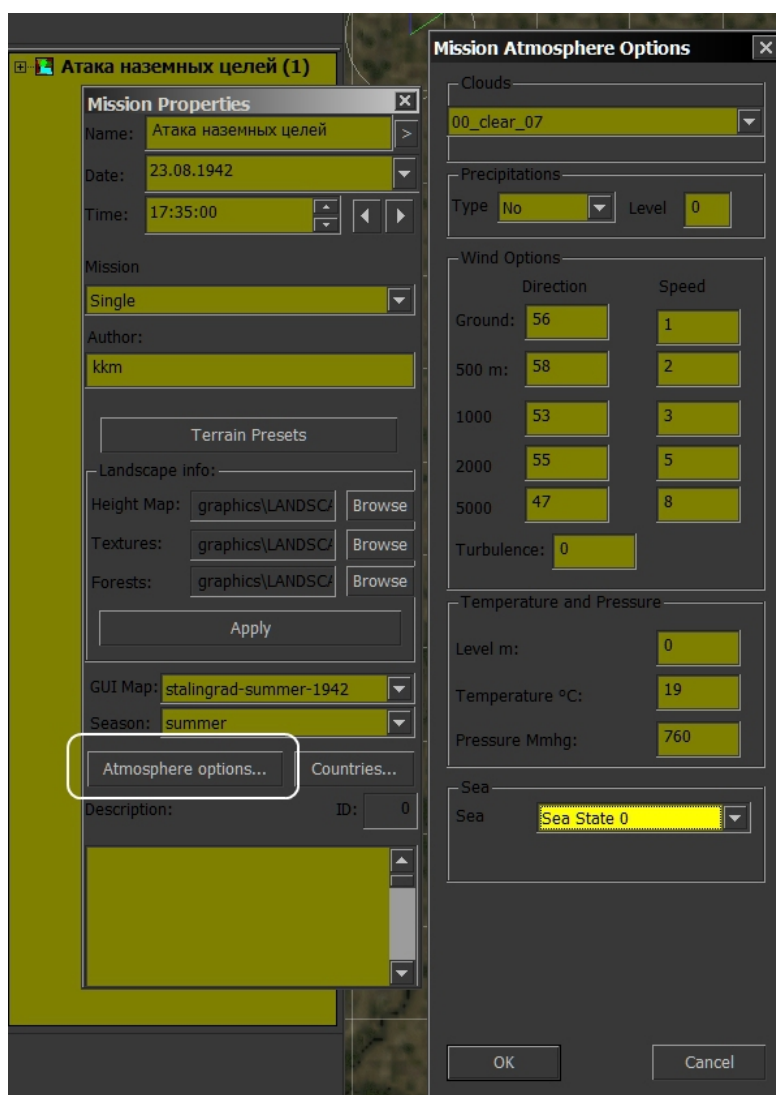


рис.10

В этом окне выбираем облака (**Clouds**), т. к. в миссии-образце ясно, то ставим любые по вкусу из группы **clear**. Вы можете видеть различия между имеющимися в редакторе **пресетами** (жёстко заданными конфигурациями) облаков, выбирая эти пресеты и просматривая их затем в 3D режиме карты (детально, параметры пресетов можно посмотреть в файле «...data\GUI\me\_weather\_presets.ini»).

Осадки (**precipitations**) нам не нужны, но необходимо заметить, что дождь или снег в миссии возможен только при выбранном пресете облаков из группы **overcast** (всё покрыто).

Ветер (**Wind options**): устанавливаем направление (**Direction**) и скорость (**Speed**) ветра. Здесь отметим, что в редакторе выставляется навигационный ветер, т. е. градусы говорят *куда* он дует. Подробнее об этом можно прочитать, например, [здесь](#).

Остаётся задать турбулентность (**Turbulence**), обычно 0-2 балла (выше — это уже серьёзная болтанка), температуру воздуха и давление. С погодой закончили.

Теперь напишем **брифинг**, точнее, скопируем его из миссии-образца. Вот как он выглядит в «старичке»:

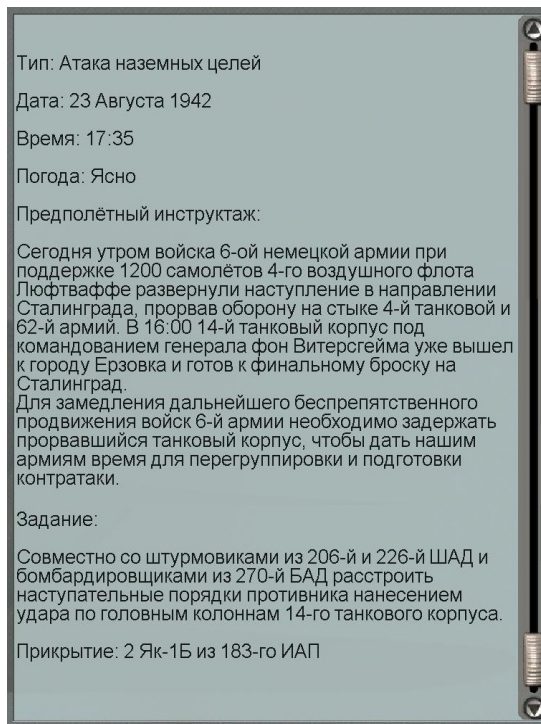


рис.11

Для того, чтобы получить нечто похожее, наберём или скопируем этот текст, используя любой текстовый редактор. Текст нужно отформатировать, соблюдая следующие правила:

Весь текст пишется в одну строчку, нестандартные символы (например, апострофы) не используются.

Для переноса текста на другую строку используется **<br>**. Если между абзацами текста необходимо сделать пустую строку – ставим **<br><br>**.

Для подчеркивания текста в начале слова используется **<u>**. Окончание подчеркивания – **</u>**, аналогично, для выделения части текста жирным шрифтом или курсивом используем **<b></b>** и **<i></i>** (см. для справки: [«Что такое теги?»](#)).

Текст нашего брифинга примет следующий вид:

**Дата**: 23 Августа 1942 г.  
**Время**: 17:35  
**Погода**: Ясно, ветер у земли 1 м/с, 56°  
Турбулентность отсутствует.  
Предполётный инструктаж:  
Сегодня утром войска 6-ой немецкой армии при поддержке 1200 самолётов 4-го воздушного флота Люфтваффе развернули наступление в направлении Сталинграда, прорвав оборону на стыке 4-й танковой и 62-й армий.  
В 16:00 14-й танковый корпус под командованием генерала фон Витерсгейма уже вышел к городу Ерзовка и готов к финальному броску на Сталинград.  
Для замедления дальнейшего беспрепятственного продвижения войск 6-й армии необходимо задержать прорвавшийся танковый корпус, чтобы дать нашим армиям время для перегруппировки и подготовки контратаки.  
Задание:  
Совместно со штурмовиками из 206-й и 226-й ШАД и бомбардировщиками из 270-й БАД расстроить наступательные порядки противника нанесением удара по головным колоннам 14-го танкового корпуса.  
Прикрытие: 2 Як-1Б из 183-го ИАП.



Скопируем этот текст в поле **Description**/Описание (см. **рис.10**) и в игре он будет выглядеть так:

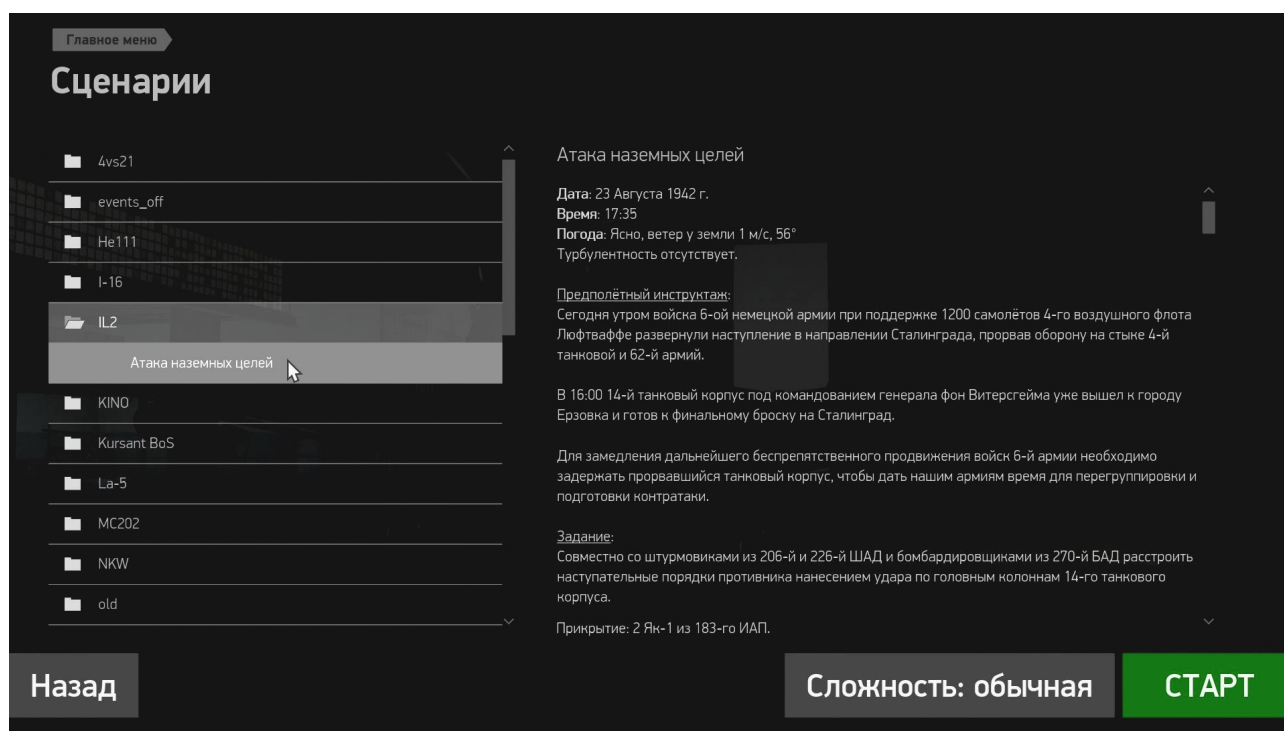


рис.12

### 3.3 Оформление полётной карты.

Обозначим на карте линию фронта и укажем маршрут полёта. Всё «рисование на карте» осуществляется иконками (**MCU Translator: Icon**), которые, будучи связаны между собой целевыми связями, образуют на карте линию заданного типа. На **рис.13** слева от самой иконки — окно базовых настроек, справа — расширенных.

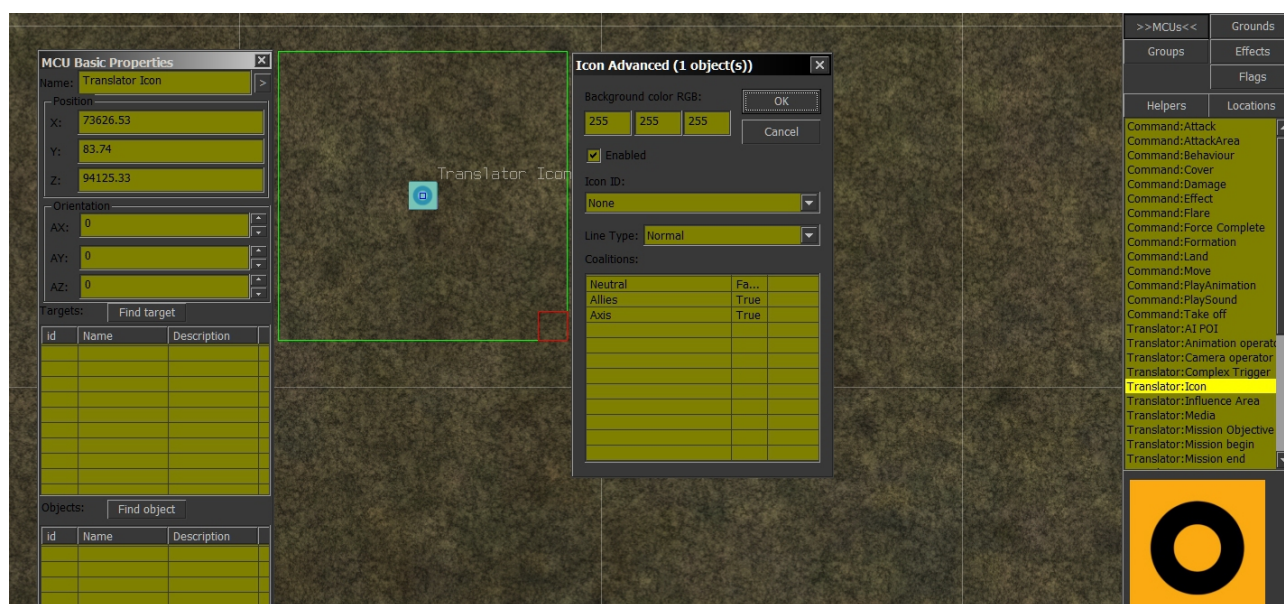


рис.13



Для того, чтобы нарисовать на карте нужную линию, достаточно настроить одну иконку, затем, не снимая с неё выделения, нажать клавишу **Shift** и левым щелчком мыши расставить остальные иконки — целевые связи между ними проставятся автоматически, а настройки этих иконок будут одинаковыми. Этим же способом расставляются маршрутные точки (**Trigger: Waypoint**) для наземной техники и самолётов.

Подробнее о настройках на **рис.13**: в окне базовых настроек (**MCU Basic Properties**) изменяем только самое верхнее поле — имя (**Name**). То, что написано там — отображается на карте, поэтому надо удалить дефолтную надпись «Translator Icon» (заменим её везде на порядковые номера маршрутных точек).

**Внимание!** Когда вы расставляете иконки в автоматическом режиме, имя, которое было задано для первой из них, не запоминается и все последующие будут иметь дефолтное имя: «Translator Icon» - удалять/менять его надо вручную.

Расширенные настройки (**Icon Advanced**) — тут интересно всё:

**Background color RGB** — цифры в этих трёх окошках определяют цвет линии, которая отображается на карте. Также, этим цветом будет написано то, что содержится в поле **Name** иконки, если эта иконка связана с другой целевой связью — надпись на отдельностоящей иконке всегда будет чёрного цвета.

**Enabled** — галка (чекбокс) вкл/выкл. Иконки интерактивны и можно добавлять их на карту или убирать их с неё прямо во время миссии (при помощи триггеров **Activate/Deactivate**). Это очень удобно, например, для отображения текущего состояния целей, но не стоит злоупотреблять этой функцией, т. к. одновременное изменение состояния большого числа иконок может приводить к вылетам игры.

**Icon ID** — поле, где задаётся вид самой иконки на карте, её значок. Таких значков много, чем они отличаются можно видеть прямо в редакторе, поэтому расписывать их все здесь не будем. Скажем только, что для того, чтобы нарисовать линию фронта (любую «просто линию») никакие значки на карте не нужны — в этом поле оставляем **None**.

**Line Type** — тип линии, т. е. то, как будут выглядеть целевые связи между иконками на карте. В редакторе вы не увидите различий — только обычную целевую связь, поэтому неплохо бы посмотреть все эти типы заранее (см. приложение 1. **Типы линий.**), чтобы понимать, что именно будет у вас нарисовано.

**Coalitions** — коалиции, настройка актуальна для сетевой игры, т. к. позволяет показывать нарисованное только той стороне, которой вы хотите его показать.

#### **Важно:**

Вне зависимости от заданного цвета линий сами значки иконок всегда будут только красные (RGB — 102.0.0) или синие (RGB — 18.89.117), если хотите, чтобы линия совпадала по цвету с иконкой, используйте эти настройки цвета или, для простоты, «автоцвет»: красный (RGB — 10.0.0) и синий (RGB — 0.0.10).

Исключения: иконки типа **Waypoint**, **Action Point**, **Take off** и **Land** всегда будут красными (RGB — 191.0.0), а иконка **Airfield** будет представлять собой чёрный стилизованный значок аэродрома.

Если в настройках типа линии выбрано «**Position Type 0**», то настройка цвета этой линии не важна — такая линия всегда будет двойной красно-синей и отлично подходит для изображения линии фронта.

Вернёмся к нашей миссии. Исходя из вышеизложенного, получаем такие настройки иконок для изображения линии фронта и маршрута:

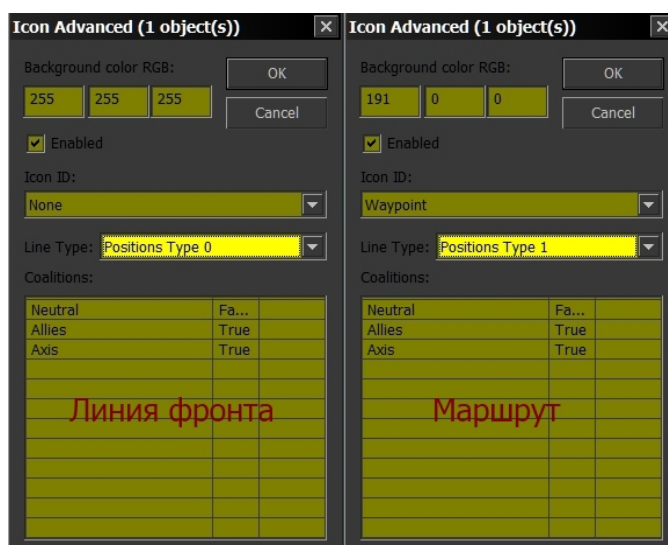


рис.14

Расставим эти иконки в соответствии с маршрутом и очертаниями линии фронта по состоянию на 23-е августа. Маршрутные точки привяжем к хорошо заметным наземным ориентирам (нас. пункт, аэродром), а линию фронта просто скопируем из миссии подходящего периода кампании игры, ибо можем вполне довериться историческим познаниям её (кампании) создателей.

В редакторе это будет выглядеть так:

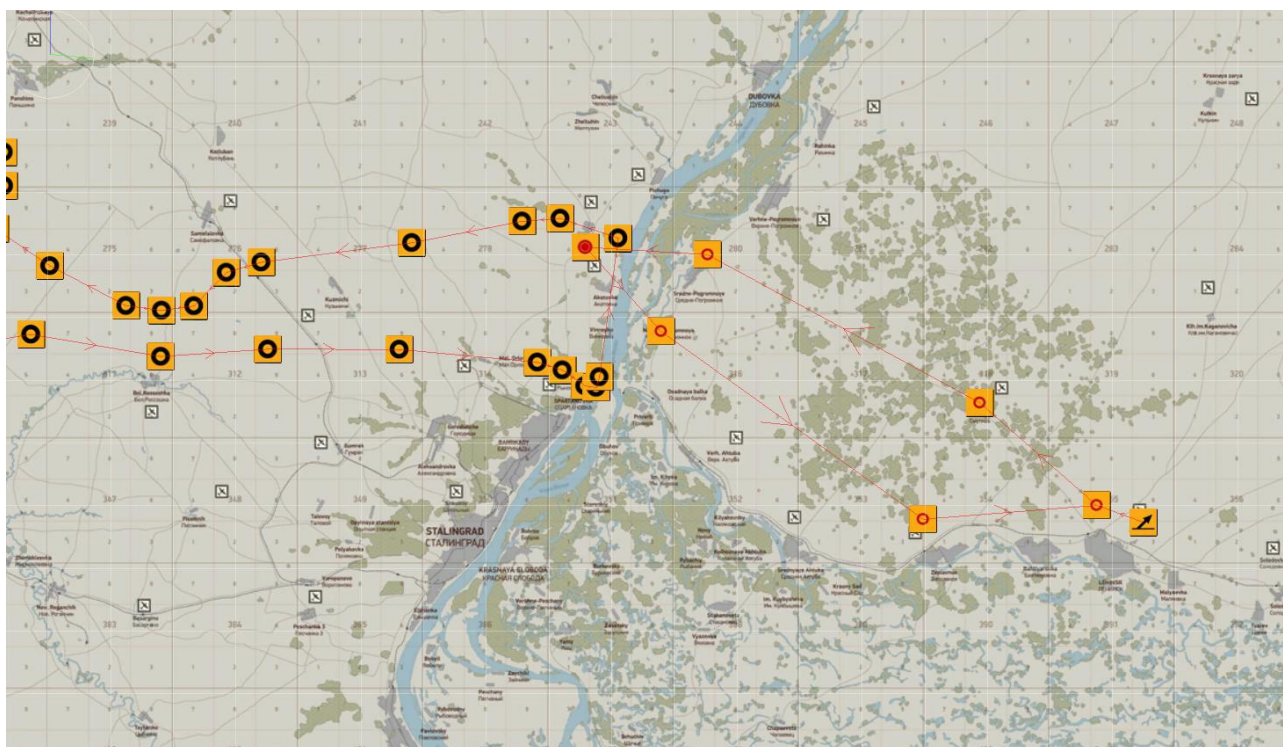


рис.15

А на карте в игре увидим следующее (цифры курсов и расстояний проставятся автоматически):

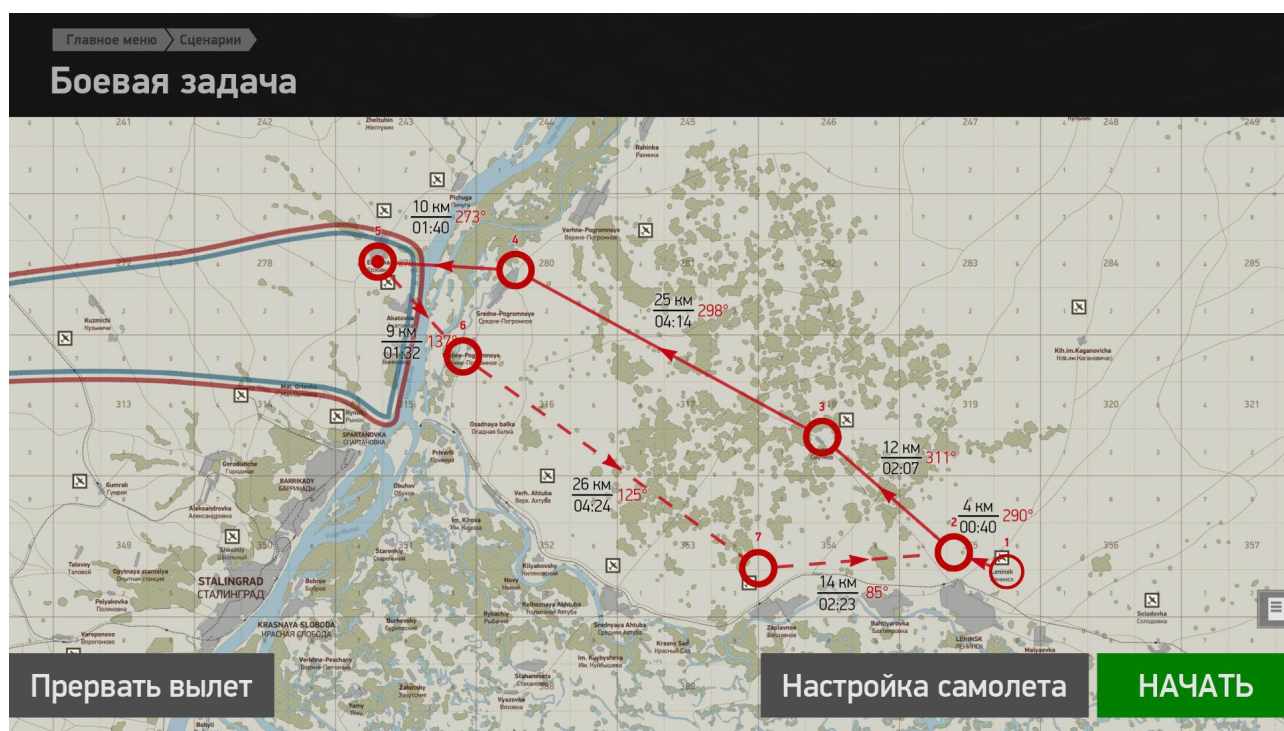


рис.16

Для того, чтобы маршрут отхода отображался пунктиром, ТИП ЛИНИИ в соответствующих иконках (**5**, **6** и **7**) изменён по сравнению с **рис.14** на «**Position Type 2**».

Тип иконки стартового аэродрома (**1**) изменён на «**Take off**», а иконки места выполнения задания (**5**) — на «**Action Point**».

На этом оформление закончено, разумеется, это лишь вариант, не посягающий на вашу свободу творчества в этом направлении (см. также приложение [2. Стрелки.](#)).

### 3.4 Декорации.

К декорациям относятся все «неживые» статичные объекты на карте — города, деревни, мосты, ж/д станции, а также обстановка аэродромов. Все эти объекты уже заботливо расставлены по своим местам картоделами и находятся по пути «...data\Template», откуда их и нужно импортировать на карту как группы. Все группы декораций имеют «говорящие» названия и ошибиться при выборе трудно.

Т.к. карта «Сталинград» достаточно велика, все объекты из дефолтных групп не потребуются — они только излишне «утяжелят» миссию. Определим «зону боевых действий» в миссии (она будет примерно соответствовать **рис.16**) и удалим всё, что находится за границами этой зоны. Воспользуемся группировкой объектов, копипастой и сочетанием клавиш «выделить всё» (**Ctrl+A**).

Это несложный процесс, но он требует аккуратности, т. к. можно случайно сдвинуть выделенные разрозненные объекты и тогда (особенно, если снять выделение) легче будет повторить все операции с этими объектами заново, чем поставить их на места по одному.

Декорации имеют несколько небезынтересных параметров:

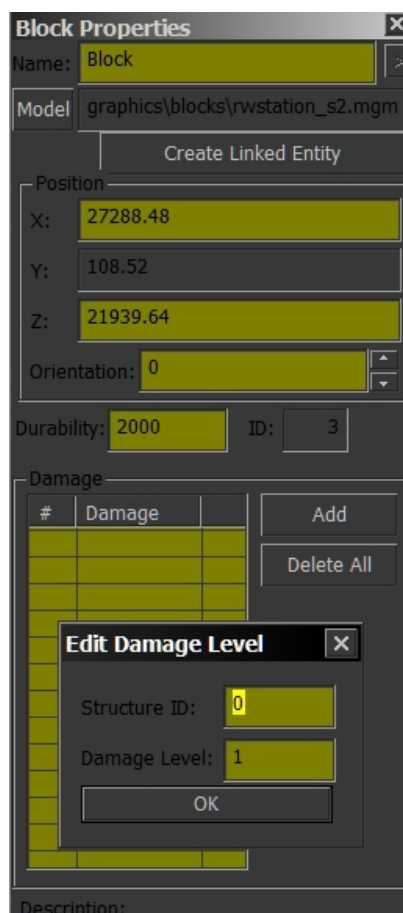


рис.17

Во-первых, их можно «оживлять», присваивать им принадлежность к той или иной стороне и делать целями для стороны противоположной (всё та же кнопка **Create Linked Entity** + выбор нации в окне доп. настроек).

В этом случае, важно значение в поле **Durability** (Прочность), от него напрямую зависит, насколько легко/трудно будет уничтожить объект. Тут широкое поле для экспериментов). Можно только сказать, что рекомендованное максимальное значение прочности для большого объекта (например, деревенского квартала) равно 5000, для бетонного ж/д моста — 7000-12000, а прочность «живой мелочи», типа статичных автомобилей рекомендуется делать минимальной, т. е. 500 единиц.

Во-вторых, декорации можно разрушать и, если постройки самого Сталинграда уже изначально представляют собой руины, то точно так же можно сделать руинами любую деревню, мост и т. п.

Для этого пользуемся кнопкой **Add** (Добавить) и в окошке **Edit Damage Level** задаём номер разрушаемой «секции» объекта (**Structure ID**) и уровень разрушения (**Damage Level**), ноль соответствует целой «секции», единица — полностью разрушенной.

Все разрушения отображаются прямо в редакторе.



Вот, например, как выглядит целая ж/д станция и она же, но полностью разрушенная (обратите внимание на поле **Damage** этой разрушенной станции):

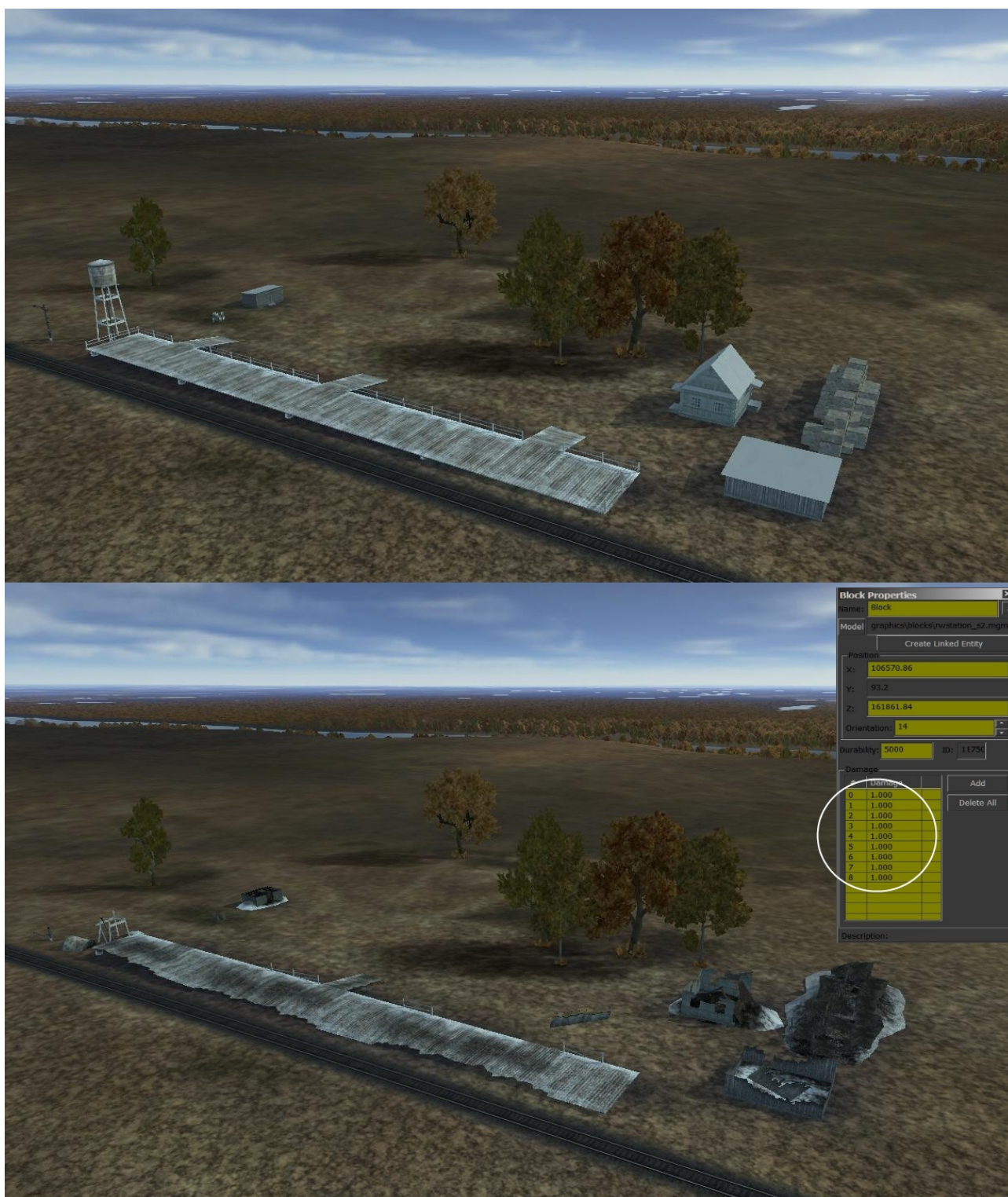


рис.18

Остаётся только добавить, что если вы наметили в качестве цели какой-то конкретный объект (ту же станцию или мост), то лучше вычленить его заранее из группы остальных декораций, ибо «живая цель» - товар штучный и работать с ним придётся особо.



### 3.5 Реализация сценария.

Изучив миссию-образец, выясним для себя, что именно в ней происходит: 4 самолёта Ил-2 (игрок — ведущий) под прикрытием 2 Як-1Б атакуют две танковые колонны по 4 машины в каждой, имея основной задачей полное уничтожение одной из этих колонн. Одновременно, в том же районе (около н.п. Ерзовка), ещё две пары Илов штурмуют статичные танки, а группа бомбардировщиков Пе-2 (3 штуки) бомбит этот район с горизонта.

ПВО наземных целей слабое, на перехват прилетают 4 «Фридриха» (Ф-2). Дополнительной задачей для советской авиации является уничтожение 75% статичных танков, а скрытой задачей — уничтожение близлежащего моста.

Взглянем на карту:



рис.19

Как видим, в отличие от «старичка Ил-2», никакого моста рядом с **Ерзовкой** нет (ближайший — почти за 10 км и он на своей территории), зато рядом есть аэродром. Не будем выяснять «какая карта географичнее», а применимся к местности и сделаем скрытой целью **склад ГСМ** на аэродроме **Акатовка** — это выглядит довольно логично: аэродром захвачен совсем недавно, самолёты эвакуированы, а топливный склад... ну, не оставлять же его врагу.

Теперь внесём коррективы в **плейнсет** (набор самолётов, использующийся в миссии) — Як-1Б станут просто Як-1, «Фридрихи» Ф-2 станут Ф-4 и можно начинать «расставлять фигуры», т. к. в остальном, больших расхождений с исходной миссией пока не предвидится.

Правда, на первый взгляд, в условиях **БзС**, миссия кажется сложной для выполнения — 4 «Фридриха» - это грозная сила, но будем пока придерживаться оригинала, насколько это возможно. Только тесты покажут, надо ли что-то менять.

### 3.6 Подготовка аэродрома.

«Домашний» аэродром — **Ленинск**, игрок и ведомые им боты базируются на нём:

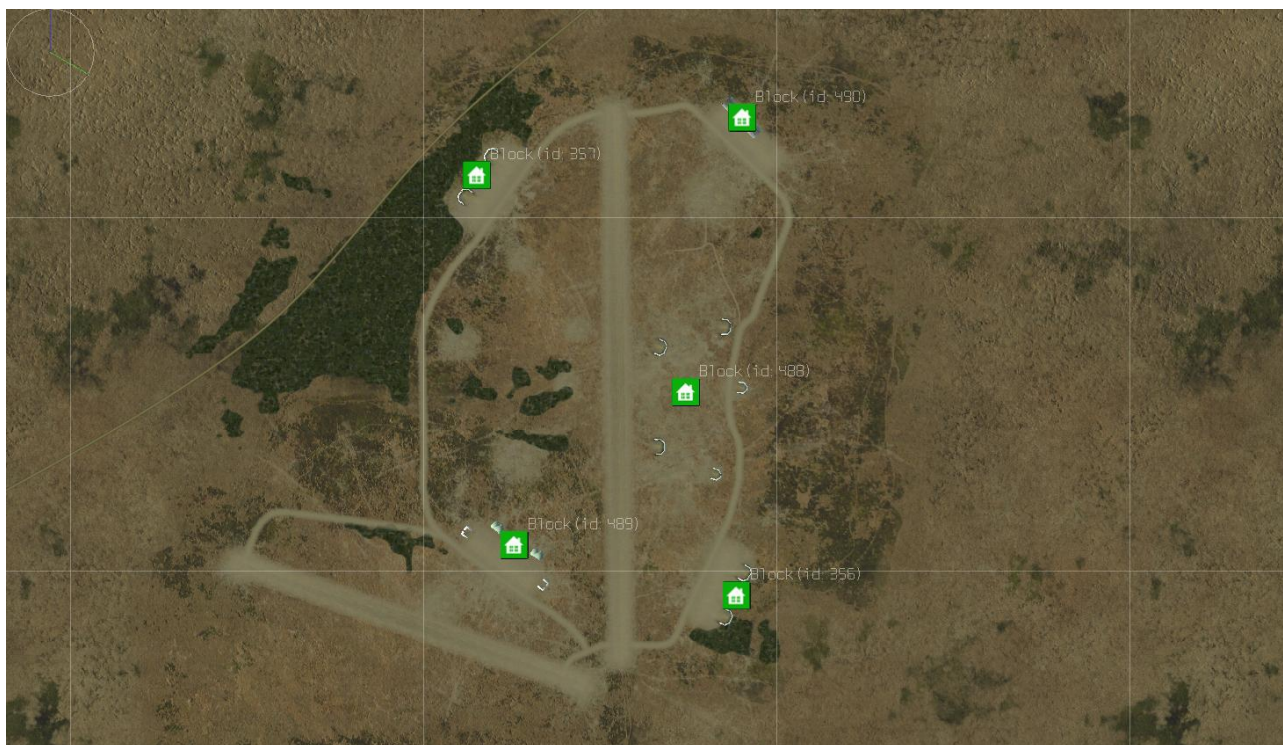


рис.20

Сейчас в миссии имеется только декорация аэродрома, для того, чтобы самолёты-боты и автопилот игрока знали как взлетать, садиться и рулить на данном аэродроме, а также для корректного радиообмена в игре необходим сам аэродром — объект из библиотеки (**Airfields** ---> **fakefield**). Расположим его в Ленинске, «оживим» и займёмся настройкой:

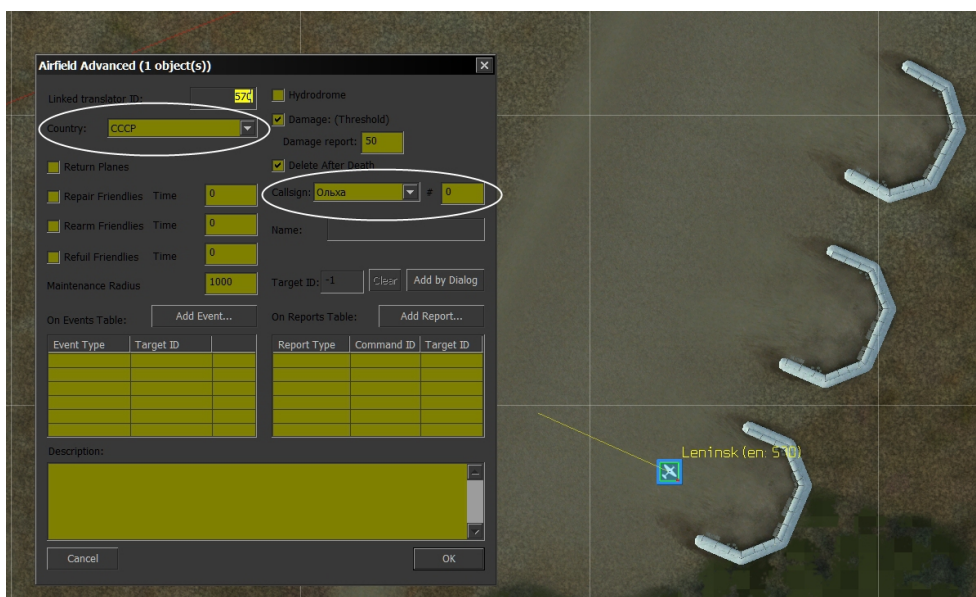
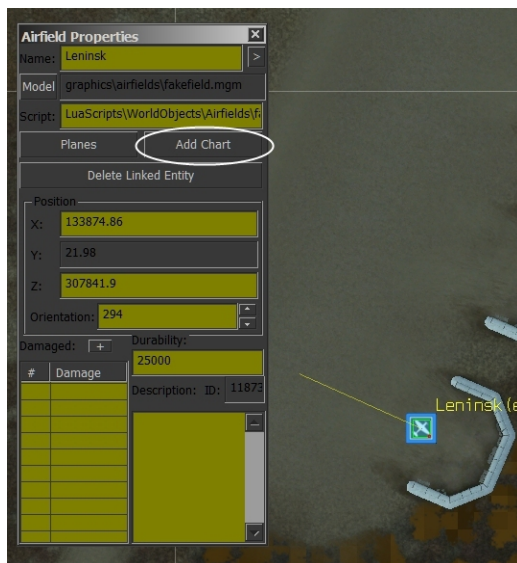


рис.21



На **рис.21** открыты сразу расширенные настройки аэродрома (двойной левый щелчок по «живому» объекту), т. к. здесь интересны только 2 поля — **Country** и **Callsign** (Страна и Позывной), выбрав СССР и любой, приглянувшийся позывной, перейдём к окну базовых настроек — на нём придётся задержаться подольше:



**рис.22**

Нам нужна кнопка **Add Chart** (Добавить схему), нажав её, увидим, что её название поменялось на **Edit Chart** (Редактировать схему) и появилось окно для этого редактирования:



**рис.23**

Оранжевая линия на рисунке и определяет схему движения на аэродроме: взлётно-посадочную полосу (**VPP**), маршруты руления (**TAXI**) и расположение стоянок (**PARK**). На **рис.23** представлена конфигурация этой схемы по умолчанию и наша задача - «подогнать» её к топографии аэродрома Ленинск.

Делается это просто: в окне **Airfield Chart Edit** (Редактирование схемы аэродрома) имеется список всех маршрутных точек с их координатами. Выбрав левым щелчком точку прямо в этом списке, её можно перетаскивать при помощи мыши, наведя на эту точку курсор (в режиме редактирования он примет форму крестика). Можно редактировать и непосредственно выставляя нужные координаты точек (**Edit Taxi Point**), но лучше оставьте этот способ генератору игры — вручную быстрее и проще.

Таким образом, поставив точки **PARK** на стоянки, точки **VPP** — по оси нужной нам полосы и «обеспечив проезд» между ними точками **TAXI**, получим такую «карту дорог» для Ленинска:



рис.24

Стрелка между точками **VPP** показывает направление взлёта/посадки, цифрой **1** обозначена стартовая стоянка (она не важна, т. к. в миссии будет старт с полосы), а цифрой **2** — стоянка, куда боты (и автопилот) будут заруливать после посадки.

Кнопкой **Add Taxi Point After/Before** можно добавить точку рулёжки перед или за редактируемой точкой, если в процессе «прокладки дорог» не хватит этих точек по умолчанию, а кнопкой **Invert Graph** — развернуть всю схему движения на 180°.

Не забудем вписать в поле **Name** название аэродрома — оно будет отображаться в чате игры — и на этом настройка завершена.

#### Важное дополнение:

Для сетевой игры, самолёты на аэродром добавляются при редактировании списка, открывающегося по кнопке **Planes** (см. рис.22 или 24), а галка **Return Planes** (рис.21) приплюсовывает успешно севшие на аэродром самолёты к этому списку, если они были в нём изначально.

Галки ремонта, перевооружения и дозаправки (**Repair**, **Rearm** и **Refuel**) на данный момент не работают, гидродромов (**Hydrodrome**) тоже пока нет, а кнопки **Add Event.../Add Report...** служат для добавления событий/рапортов аэродрома из списка — по номеру (**ID**) или по имени целевого объекта в миссии, что зачастую гораздо удобнее, чем тянуться курсором и щёлкать мышью по карте (актуально для всех объектов в редакторе).

### 3.7 Настройка самолётов.

Начнём с самолёта игрока как главного героя миссии: выберем из библиотеки (вкладка **Planes**) самолёт Ил-2 мод. 42 и поставим на старт в начале полосы, т. к. миссия начинается именно на полосе. «Оживим» наш Ил, дадим ему имя **Player** и откроем настройки самолёта:

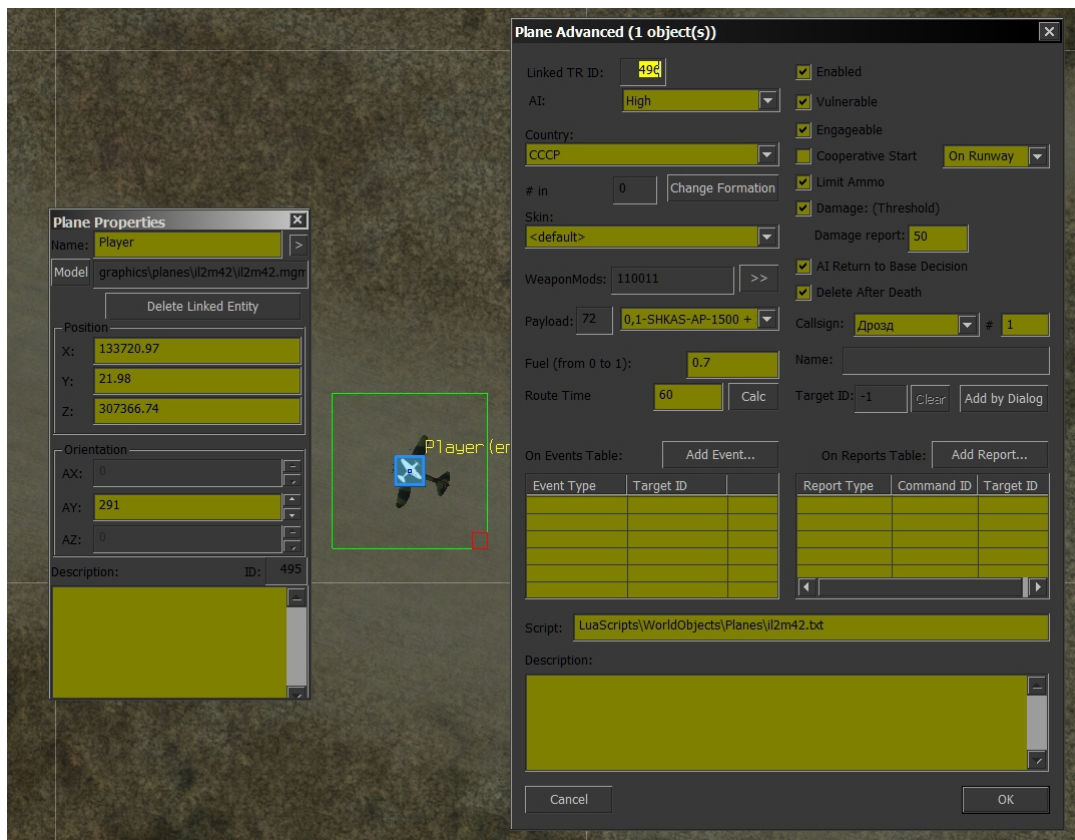


рис.25

В левом окне выставлено имя самолёта, кроме этого, обратите внимание на 2 поля: координата **Y** (**Position**) и координата **AY** (**Orientation**) — первая показывает высоту объекта над уровнем моря (на рис.25 самолёт стоит на полосе и 21,98 м — высота полосы над уровнем моря), вторая — собственно, ориентацию или курс. Высоту будем менять на нужную, при оформлении старта в воздухе, а второе поле в случае необходимости поможет сориентировать несколько объектов на карте абсолютно одинаково.

В окне расширенных настроек выставляем уровень бота (**AI**) — не стоит сразу ставить **Player**, т. к. проще настроить 1 самолёт, а остальные 3 скопипастить, но в оффлайн-миссии не может быть больше одного игрока и эта настройка сбросится при копировании, поэтому «припишем» самолёт игроку в последнюю очередь — главное, не забыть это сделать).

Далее, в левой части окна, выбираем страну, схему окраски (**Skin**), модификации вооружения (**WeaponMods**), конкретную боевую загрузку самолёта (**Payloads**), кол-во топлива (**Fuel**), а в правой части окна — в верхнем поле ставим тип старта «с полосы» (**On Runway**), в нижнем (**Callsign**) — позывной самолёта и номер этого позывного.

Галка/чекбокс **AI Return to Base Decision** (по умолчанию выключена) поставлена для того, чтобы повреждённый самолёт-бот выходил из боя и возвращался на ближайший дружественный аэродром.



Остальные настройки оставляем по умолчанию — перечислим их коротко:

Поле **# in Formation** – номер объекта в группе.

Кнопка **Change Formation** – изменение положения игрока (не **AI**) в группе.

Галка/чекбокс **Enabled** — если включено, объект появится при загрузке миссии сразу, если нет — появится позже, тогда, когда это будет нужно по сценарию.

Галка **Vulnerable** – уязвимость объекта от повреждений или нет.

Галка **Engageable** – будет объект атакован **AI** или нет.

Галка **Cooperative Start** – включить, если миссия сетевая.

Галка **Limit Ammo** – бесконечный боезапас или нет.

Поле **Damage – Threshold** или **Step**.

Здесь чуть подробнее:

**Threshold** (пороговое значение) – сколько осталось «жизни» у объекта в процентах.

**Step** (шаговое значение) – количество полученных повреждений объектом в процентах.

Поле **Damage report** – количество процентов. Цифра в этом поле определяет срабатывание «событийной» связи **OnDamaged Event** от объекта, если такая связь установлена (в случае **Threshold** связь сработает, когда кол-во «жизни» объекта сравнится с указанным в поле значением, а в случае **Step** связь будет срабатывать каждый раз, когда объект получит указанное кол-во повреждений в %).

Поле **Route Time** с кнопкой **Calc** - предоставляет возможность рассчитать время на маршруте при заданной вами скорости.

Почти все вышеперечисленные параметры имеются и в настройках наземной техники (танки и проч.) и означают в точности то же самое.

Вот и всё - «размножаем копированием» настроенный Ил, головному самолёту ставим уровень **AI Player**, целевой связью от каждого ведомого бота к самолёту игрока образуем из самолётов одну группу, по желанию придумываем имена ведомым и меняем им номера позывных на порядковые — штурмовики готовы к взлёту:



рис.26

### 3.8 Полёт по маршруту. Взлёт. Посадка.

Основные точки маршрута уже намечены иконками — «нарисуем маршрут», ориентируясь по ним.

Выделим самолёт игрока и с зажатым **Shift**’ом поставим первую маршрутную точку рядом с иконкой **№2** (см. **рис.16**). Маршрутная точка (или **Контрольная точка — КТ**) представляет собой триггер (**MCU Trigger: Waypoint**), который заставляет объект двигаться к себе (т. е. **КТ является также и командой** для связанного с ней объекта) и срабатывает по этому объекту. При срабатывании, этот триггер может «дёргнуть» как следующую **КТ**, так и любое другое событие в миссии. Настройки:

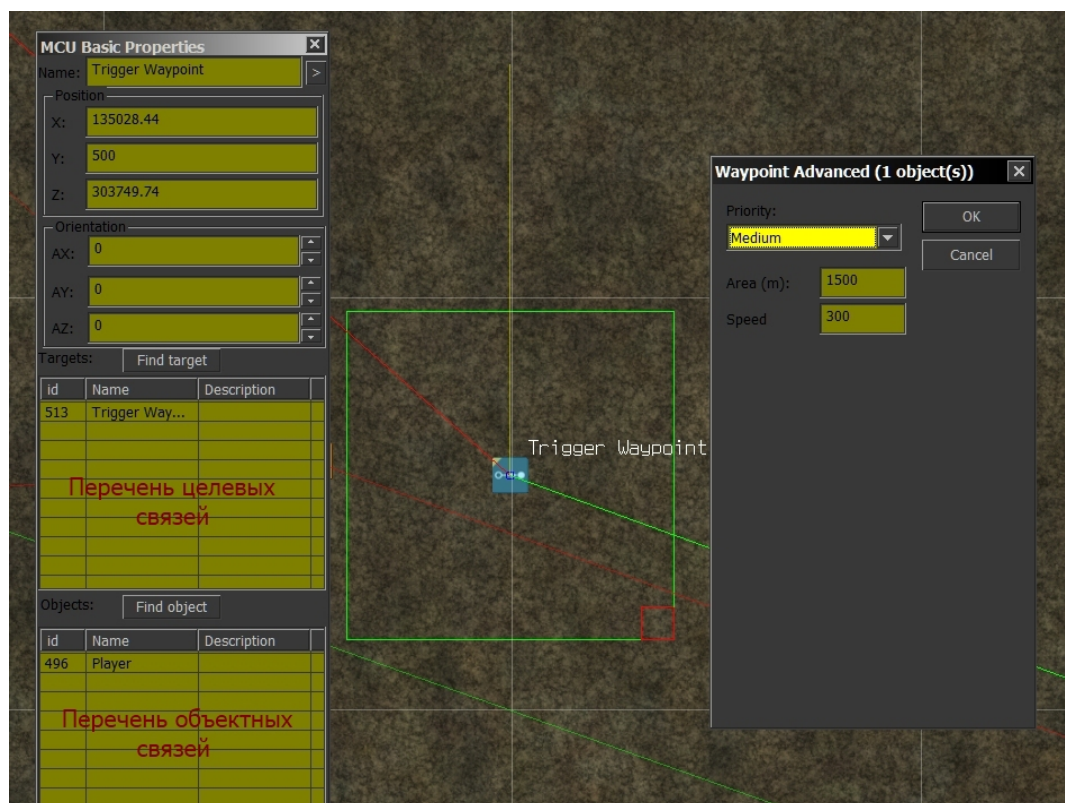


рис.27

В левом окне ставим высоту прохождения **КТ** (координата **Y=500** м), в правом — радиус срабатывания **КТ** =1,5 км и скорость, которую самолёт будет стремиться выдерживать на пути к точке =300 км/ч. Приоритет (**Priority**), с которым самолёт будет стремиться к точке не меняем — для подавляющего большинства случаев настройка этого параметра как **Medium** — правильный выбор. Цитата из мануала для редактора **РФ**:

**«Low** (низкий) – AI во время полета к точке с таким приоритетом, будет производить поиск любых доступных целей и по возможности будет атаковать их;

**Medium** (средний) – AI во время полета к точке с таким приоритетом будет отклоняться от маршрута или предпринимать какие-либо действия, если AI определит угрозу для себя (атакующий сзади самолет – AI будет обороняться);

**High** (высокий) – AI будет лететь к точке вне зависимости от исходящих угроз до тех пор, пока он не достигнет заданной точки (не выполнит команду).»

К этому можно добавить, что при полёте по маршруту с высоким приоритетом, боты-стрелки бомбардировщиков и штурмовиков не будут стрелять по атакующим их самолётам (да и ни по чему не будут), так что, если вам когда-нибудь понадобится мишень....

Настроив первую **КТ**, расставим от неё все остальные (с **Shift**'ом, как уже умеем) — настройки сохранятся, а нужные связи проставятся автоматически:



рис.28

Обратите внимание на **КТ (4)** в районе цели — её радиус изменён на 6 км(!), т. к. именно эта **КТ** будет тем триггером, который отдаст Илам команду атаковать танки и если оставить радиус прежним (1,5 км), команда придёт слишком поздно и самолётам придётся перестраиваться для атаки прямо над целью.

Также, на **рис.27**, посмотрите на поля, где перечислены все целевые/объектные связи, исходящие от представленной там **КТ**. Такие поля есть у всех **MCU** в редакторе и щелчком левой кнопки мыши по ним удобно назначать новые связи по имени/номеру целевого объекта миссии (у объектов редактора — техники, строений и проч. подобную функцию выполняют кнопки «**Add Event.../Add Report...**» в окне расширенных настроек, о чём уже упоминалось при описании подготовки аэродрома). Ненужные связи удаляются из этих списков клавишей **Delete**.

Раз уж речь зашла об удобствах, перечислим здесь некоторые, самые востребованные сочетания «**горячих клавиш**» редактора:

**F9** — переключение между 2D и 3D видом карты.

**Shift+F9** — переключение между топографической и «физической» картой.

**Shift+T** — целевая связь от выделенного объекта.

**Shift+O** — объектная связь от выделенного объекта.

**Shift+M** — переключение в «режим измерения» (**Measure Mode**) — рулетка или линейка (на панели инструментов также есть кнопка переключения в этот режим).

Ну и как раз перед началом описания **ЛОГИКИ МИССИИ**, будет уместно сказать, что все её (логики) примеры не являются истиной в последней инстанции, а просто показывают «как это может быть сделано». Разумеется, многое может быть сделано иначе, а кое-что вероятно и проще, и остроумнее, чем в приведённых примерах.



Тем временем, маршрут штурмовиков проложен, займёмся взлётом и посадкой:

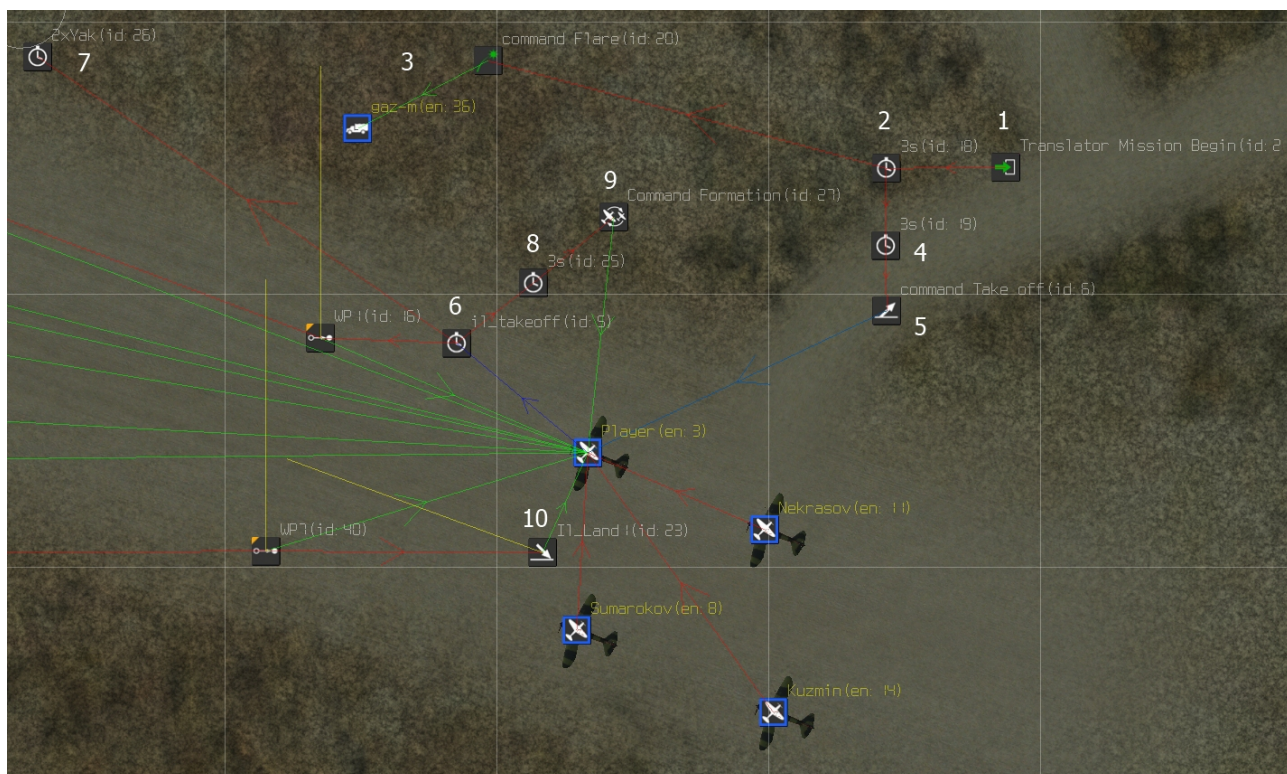


рис.29

На этом рисунке, первая и последняя **КТ** (**WP1** и **WP7** на рис.) подвинуты так, чтобы они «попали в кадр», в миссии эти точки, конечно, стоят на своих местах — за три с лишним км. от старта. Все связи и их направления на **рис.29** хорошо видны по цвету и стрелкам, за одним исключением, о котором будет сказано особо.

Начнём по порядку:

**1. Translator: Mission Begin** – запускает миссию, с него начинаются все события в ней — хотя бы один такой транслятор необходим для старта миссии, хотя, как правило, их гораздо больше — все они срабатывают в начале миссии и могут запускать разные логические цепочки в ней.

**2.** Таймер на 3 секунды (**Trigger: Timer**) – стандартная задержка начала миссии, необходимая для подгрузки всех объектов миссии на карту.

**3.** «Зелёная ракета на старт» - сразу после старта миссии легковушка (**gaz-m**), стоящая рядом с местом старта, запустит зелёную сигнальную ракету — так может делать любая наземная техника по команде **Command: Flare** (цвет сигнала можно выбрать в настройках этой команды).

**4.** Таймер 3 секунды — небольшая задержка между самым стартом миссии и отдачей команды на взлёт. Попросту — чтобы аэродром не оглушал сразу, после того как игрок снимет игру с начальной паузы, своим «Дрозд, взлетайте!» (вещь ни разу не обязательная).

**5.** Собственно, команда на взлёт (**Command: Take off**) — присоединена к самолёту игрока объектной связью — этого не видно на **рис.29**, т. к. поверх «зелёной стрелки» объектной связи наложилась «синяя стрелка» рапорта о взлёте.



Остановимся на этом подробнее:

Выделив любой объект в редакторе, правой кнопкой мыши можно вызвать контекстное меню, где, при наведении курсора на строчку **Selected Object Menu**, раскроется подменю, в котором можно назначить объекту любой тип связи, переключиться на 3D вид объекта (**Look from This Object**) или удалить все исходящие от объекта связи (**Delete all output links**) – это наиболее часто используемое меню/подменю (см. приложение 3. Расшифровка меню назначения связей.):



рис.30

Для того, чтобы команда взлёта отработала корректно и самолёт «знал, что делать» после того как взлетит, выберем в этом подменю «рапорт о взлёте» (**OnTookOff Report**).

Рапорт (любой), в отличие от целевых и объектных связей, и от «сообщений о событии» (**Event links**), ставится в два приёма — сначала к команде, о выполнении которой сообщает объект (в данном случае — **Command: Take off**), а затем — к **MCU**, который определит дальнейшее поведение объекта (здесь — таймер **il\_takeoff; 6**).

Вы не ошибётесь в последовательности, если при назначении рапорта посмотрите на курсор: сначала он изменит форму и над ним появится надпись «**CMD**», а после выбора команды надпись над курсором изменится на «**Target**».

Вернёмся к рис.29:

**6.** Таймер **il\_takeoff** (0 секунд) – получив рапорт о взлёте, «дёргает» первую **КТ** маршрута, которая «говорит самолёту», что после взлёта неплохо бы полететь дальше и показывает куда именно. Одновременно с этим, **il\_takeoff** запускает ещё 2 таймера — **7** и **8** (собственно, таймер **il\_takeoff** и нужен был для такого «ветвления», если бы, кроме взлёта, нас ничего не интересовало в миссии, рапорт был бы поставлен от команды взлёта, прямо на первую **КТ** и Илы прекрасно пошли бы по маршруту).

7. Таймер «**2xYak**» - будет сообщать «крыше» штурмовиков о том, что Илы взлетели и надо их прикрывать. Понадобится, когда займёмся расстановкой Яков.

8. Таймер, который через 3 секунды после взлёта, «дёрнет» команду (9) на построение самолётов. Нужен для того, чтобы «разнести во времени» включение первой **КТ** и команду построения, т. к. нельзя отдавать объекту несколько команд одновременно — это **важное правило**, его несоблюдение приводит к неправильной работе логики миссии.

9. **Command: Formation** – команда самолётам на построение, объектной связью соединена с головным самолётом группы (самолётом игрока) — вообще, когда имеем дело с любой техникой, объединённой в группу, все команды отдаются только головной машине (лидеру группы), остальные члены группы будут тоже выполнять их.

В настройках этой команды можно выбрать разные виды построений и для самолётов, и для наземной техники, а также — указать, насколько плотным должен быть этот строй. Цитата из руководства по редактору **РоФ**:

«**Plane: None** – самолеты: колонна.

**Plane: V-Form** – самолеты: V образное построение, с ведущим самолетом в вершине.

**Plane: Left Edge Form** – самолеты: левый пеленг.

**Plane: Right Edge Form** – самолеты: правый пеленг.

**Vehicle: On Road Column** – движение техники по дорожной сети, в колонне.

**Vehicle: Off Road Column** – движение техники в колонне, по прямой к точке, не учитывая рисунок дороги.

**Vehicle: Off Road User Formation** – движение техники в строю, заданном пользователем в редакторе (то есть так, как пользователь их расставил в редакторе), по прямой к точке.

**Vehicle: Forward** – движение передним ходом.

**Vehicle: Backward** – движение задним ходом (только для гусеничной техники).

**Vehicle: Stop** – остановка на маршруте.

**Vehicle: Panic Stop** – остановка с рассыпанием техники по местности.

**Vehicle: Continue Moving** – продолжение движения после остановки.

**Vehicle: Set Direction and Stop** – остановка с поворотом по направлению, для этого нужно задать команде **YOri**.

Список **Formation Density** – указывает дистанцию между объектами в построении от растянутой (**Loose**) до сжатой (**Dense**).»

В этой миссии Илы полетят к цели в плотном левом пеленге:

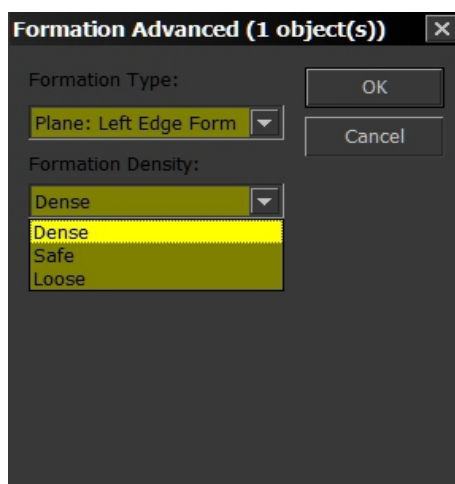


рис.31

**10. Command: Land** — команда посадки, запускает её последняя контрольная точка (**WP7**), эта команда должна находиться на поверхности земли в районе посадочной полосы. Если есть сомнения, что команда (или какой-нибудь другой объект в миссии) «стоят на земле», или просто для определения высоты поверхности в любом месте карты, воспользуйтесь кнопкой на панели инструментов **Set on Ground** («установить на землю»):



Вот всё, что надо сделать, чтобы самолёты основной группы (группы игрока) в миссии взлетали, шли по намеченному маршруту и возвращались на аэродром Ленинск.

Теперь надо подумать о том, что же они будут атаковать — нарисуем им цель.

### 3.9 Наземная техника — движущаяся и статичная.

«...14-й танковый корпус уже вышел к городу Ерзовка и готов к финальному броску на Сталинград». В редакторе это будет выглядеть следующим образом:



**рис.32**

Вся техника на рисунке уже «оживлена» и сделана «немецкой», поэтому иконки объектов в редакторе красные (для нейтральных или «советских» объектов они будут синими как, например, на **рис.30**).

Расставив технику в соответствии с миссией-образцом, надо спросить себя: какой примерно рисунок боя хотелось бы получить? От ответа на этот вопрос будут зависеть настройки многих объектов и **MCU** в миссии.

Общий замысел следующий: вся техника изначально в миссии неактивна и появляется на своих местах только при подлёте к ней игрока; **колонна 1**, поедет по дороге на юг, по направлению к Сталинграду, а **колонна 2** проедет немного по той же дороге, а затем свернёт с неё и направится к Сталинграду «огородами». Основной задачей будет уничтожение любых четырёх танков из состава этих колонн.



Дополнительную задачу (уничтожение 75% статичной техники) с некоторой вероятностью могут выполнить Пешки и вторая группа Илов, в случае, если игрок эту статику атаковать не будет совсем. Намёком на скрытую задачу послужат несколько грузовиков у топливного склада, которые будут начинать движение, только если игрок окажется от них совсем близко (естественно, что в одиночной миссии почти вся логика строится «по игроку», исходя из его предполагаемых действий).

Теперь организуем движение: у всей техники в настройках надо снять галочку **Enabled**, страна (Германия) уже выбрана, в остальном — настройки по умолчанию. Танки в колоннах привязываем целевыми связями к головным машинам.

### Колонна 1



рис.33

1. Нулевой таймер «**start\_t1**» - сюда приходит сигнал, по которому колонна появляется в миссии и начинает движение по дороге.

2. **Trigger: Activate** — активатор, соединён объектной связью с головным танком колонны, отвечает за появление колонны в миссии.

3. Таймер 2 секунды — через 2 секунды после активации танков, запускает их движение, «дёргая» первую **КТ(4)**, одновременно запускает секундный таймер (5).

4. Первая **КТ** колонны — радиус 10 м, скорость 10 км/ч. Скорость подобрана при тестировании, учитывая рельеф местности и возможности ботов. Танки едут очень медленно, но так и надо.

5. Таймер 1 секунда — задержка между командой на движение и командой формации (6), которую он и включает.

6. **Command: Formation** — команда танкам ехать колонной по дороге с минимальными интервалами (**Vehicle: On Road Column, Dense**).



## Колонна 2

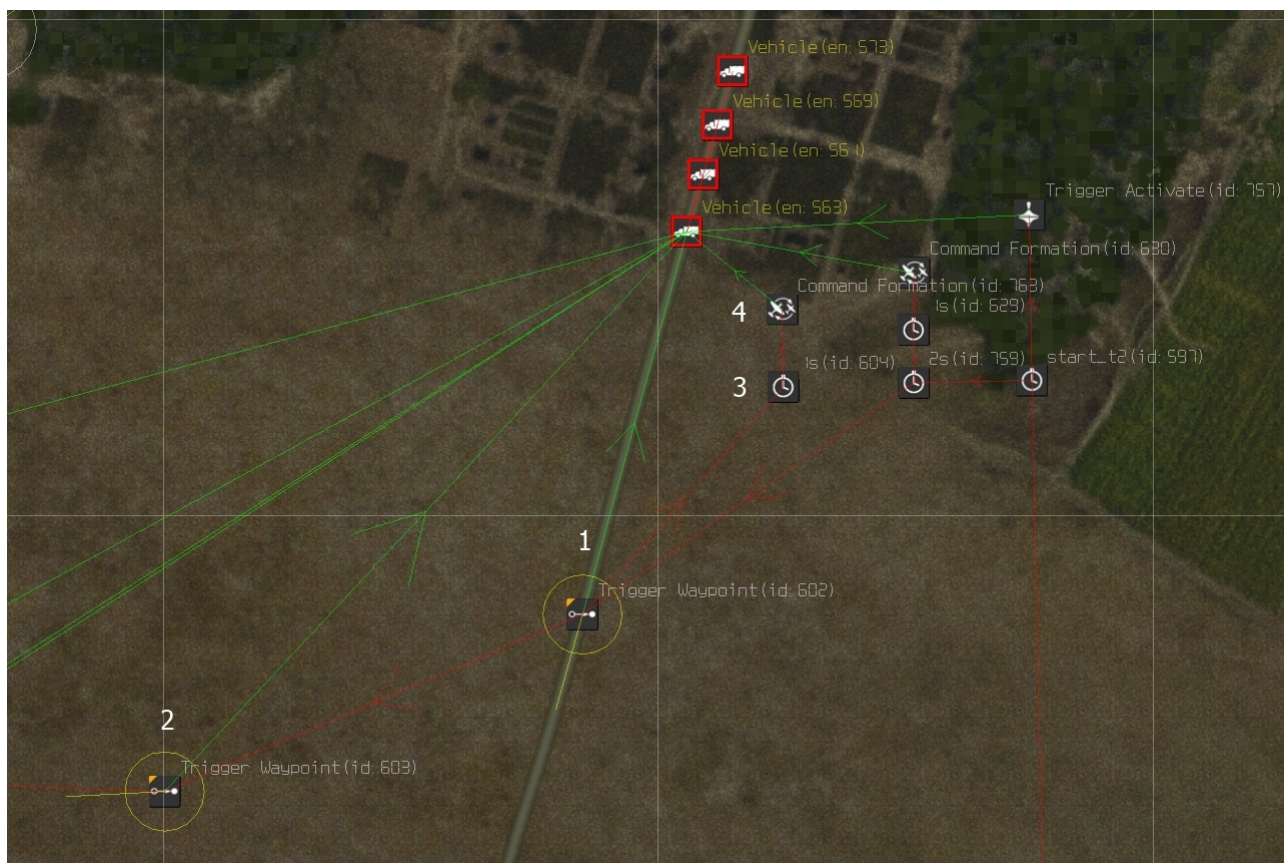


рис.34

Активация и начало движения — как у **колонны 1**. Отличие состоит в том, что когда головной танк колонны достигнет первой **КТ (1)**, эта **КТ** как триггер «дёрнет» не только следующую **КТ (2)**, но и секундный таймер (3), который запустит команду формации (4) с настройкой **Vehicle: Off Road Column**. Таким образом, колонна сначала получит команду следовать ко 2-ой **КТ (2)**, а через 1 секунду — команду сменить формацию для движения по бездорожью.

Подобным образом можно заставить технику при необходимости вернуться на дорогу, остановиться и затем продолжить движение и т. д. и т. п.

Заметим, что **колонне 2**, в силу рельефа местности у Ерзовки, вскоре после старта придётся въезжать на довольно существенный холм, поэтому, в настройках соответствующих **КТ** скорость оставлена по умолчанию (100 км/ч), но это не значит, что танки понесутся вдруг с головокружительной быстротой — они просто будут «выжимать максимум», на который способны в данной местности.

Как видим, движущаяся техника — это довольно просто технически, особенно, движение по дорогам, ибо все наземные юниты «знают» карту дорог в игре. Им надо только «подсказывать» куда свернуть на перекрёстках, чтобы они ехали по задуманному вами маршруту, так что число контрольных точек для них обычно очень невелико. Ниже приведён пример такой «подсказки», вместе с указанием на довольно распространённую ошибку в этом случае.



рис.35

Маршруты грузовиков у склада ГСМ:

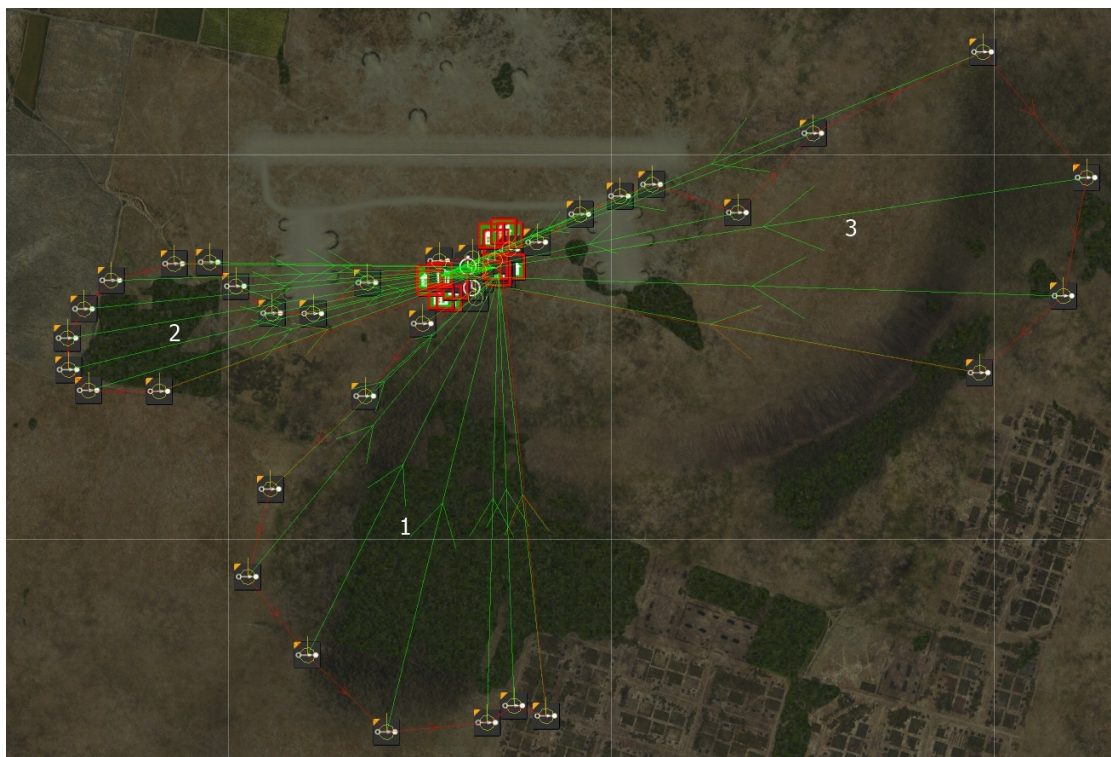


рис.36

Первая машинка «попытается спрятаться в деревне», вторая - «скрыться за лесом», а третья - «съехать в низину и затаиться»). Активация происходит по тому же сигналу, что и у танков основной/дополнительной цели, а вот начало движения рассмотрим подробно.

Контрольных точек на бездорожье приходится ставить прилично, поэтому, чтобы не путаться в многочисленных связях, на рисунке ниже показана логика для одного автомобиля и у него оставлены только две **КТ** — первая и последняя.



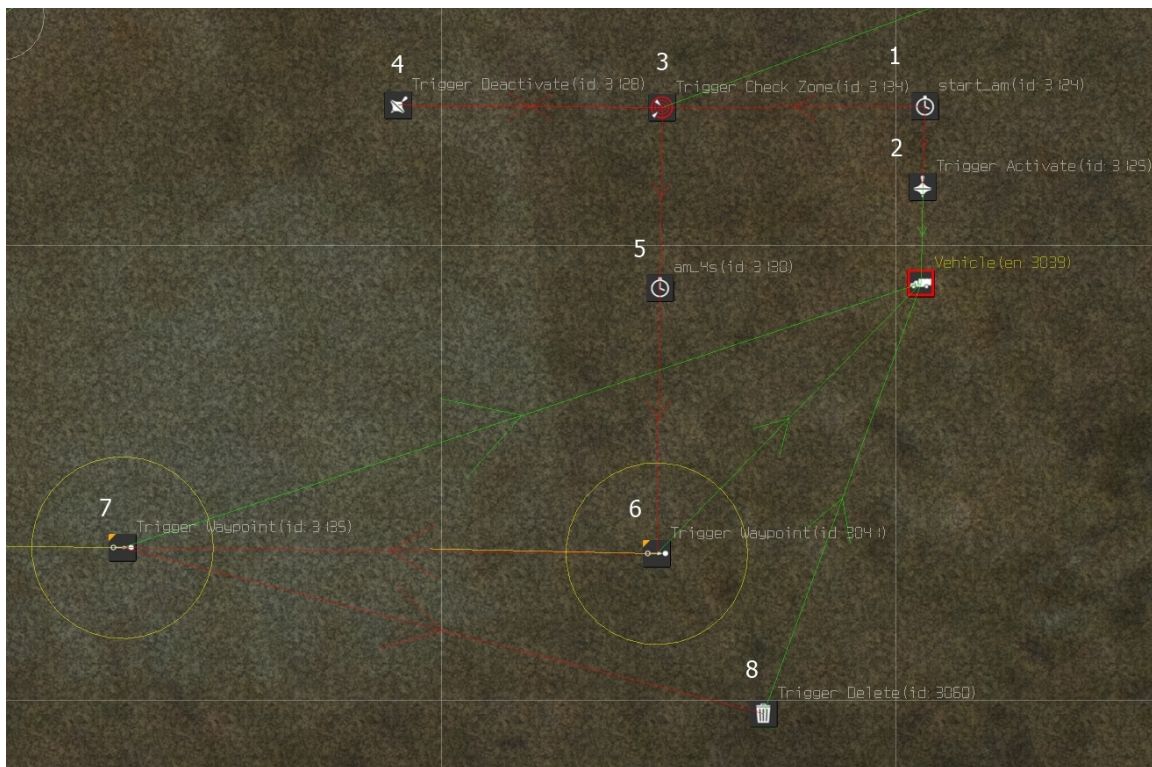


рис.37

1. Таймер **start\_am** — получает сигнал при подлёте «группы игрока» к цели, отдаёт сигнал активатору автомобиля, одновременно, включает **Trigger: Check Zone (3)**.

2. **Trigger: Activate** — активирует автомобиль.

3. **Trigger: Check Zone** («чекзона») — проверка нахождения объекта внутри или вне заданного в триггере радиуса. Настройки, в нашем случае, следующие:

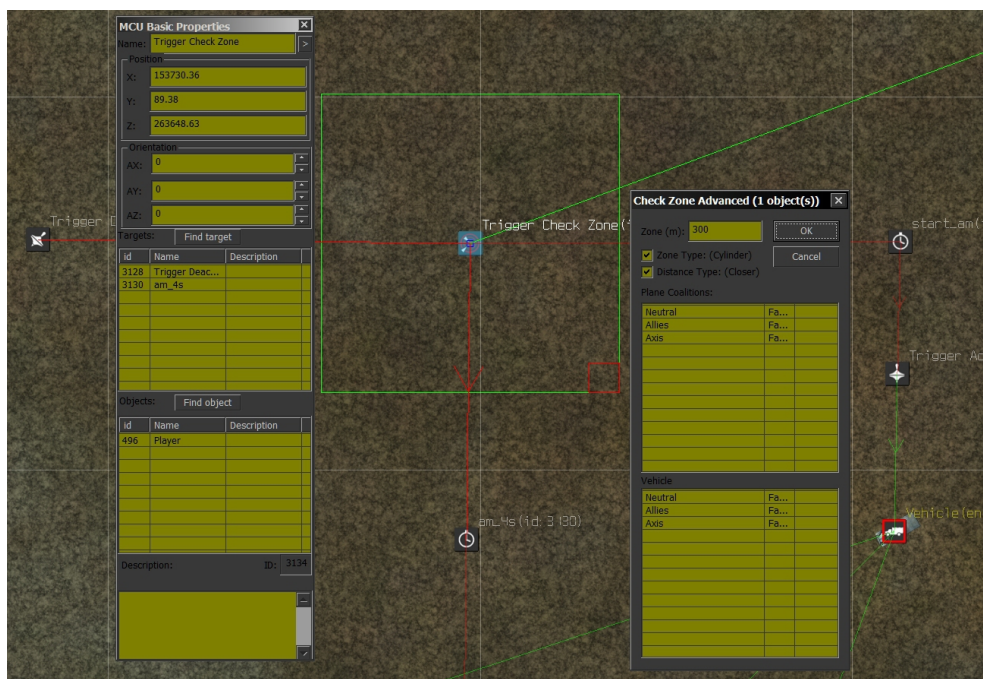


рис.38

Триггер привязан к самолёту игрока объектной связью и срабатывает только по этому самолёту, когда тот войдёт в радиус его действия (800 м). При срабатывании, «чекзона» запускает таймер (5), который «дёрнет» первую **КТ** автомобиля (6) и деактивирует саму себя при помощи деактиватора (4) — это сделано потому что «чекзона» реагирует каждый раз, когда целевой объект входит в её радиус, а здесь нужно единичное срабатывание.

7. Последняя точка маршрута грузовика — она «дёрнет» **Trigger: Delete (8)**, который удалит грузовик из миссии, если случится так, что игрок «вспугнёт машинку», но не уничтожит её (если грузовикам вообще не суждено будет поехать — их «поделит на ноль» одна из контрольных точек на маршруте отхода игрока при срабатывании, она же, кстати, даст команду удалить и уцелевшие танки).

Может возникнуть вопрос: «Зачем вообще что-то удалять?»

Скажем сразу, что это необязательно, если в сетевой игре подобное «обрубание хвостов» помогает разгрузить карту, что никогда нелишне, то в данной одиночной миссии (в которой «живых» юнитов вообще не очень много) триггеры **Delete** поставлены из следующих соображений: весь этот «отработанный материал» — машинки, вставшие в последней точке маршрута, не до конца уничтоженные танки и проч., игрок видеть не должен, а на настойках сложности, отличных от «экспертных», это позволяет сделать кнопка F5.

И ещё немного о настройках триггера **Check Zone** на **рис.38**:

**Zone Type** (тип зоны) — по умолчанию цилиндр, второй вариант — сфера. Если нужно поставить сферическую «чекзону», не забывайте, что центр сферы будет находиться на высоте, указанной в окне базовых настроек триггера (координата **Y**).

**Distance Type (Closer/Further)** — дистанция (ближе/дальше); в случае **Closer** чекзона работает, когда целевой объект приблизится к центру зоны на заданный радиус, в случае **Further** — когда отдалится от этого центра.

**Важно:** если для одного объекта проверяются оба условия (ближе/дальше), то сначала должен включаться триггер с настройкой **Closer**, а уже потом, триггер с настройкой **Further** и радиусы зон, заданные в этих триггерах должны слегка различаться (см. **рис.65** в [3.13 Поезда. Артиллерия. Антураж.](#)).

Поля **Plane/Vehicle (Coalitions)** позволяют настроить триггер не на конкретный объект, а просто на технику определённой стороны. Настройка особенно актуальна для сетевых миссий, где нельзя «привязаться» к самолётам, управляемым игроками. В нашем случае стоит объектная связь и эти поля остаются по умолчанию.

Все движущиеся объекты в составе целей расставлены, отметим разве, что таймеры, запускающие движение грузовиков (5 на **рис.37**), сделаны разными (0, 2 и 3 секунды), чтобы машинки не начинали ехать «все вдруг», ведь они стартуют на глазах игрока.

Так же, разнесена во времени на несколько секунд и активация групп техники. И опять же — не то чтобы это было необходимо, было бы слишком поспешным утверждение, что это поможет «избежать возможного фриза» или что-то подобное, но как-то на автомате в подобных ситуациях ставятся разные таймеры.



Со статичной техникой всё вообще элементарно, надо только обеспечить каждую группу активатором (т. к. юниты не связаны целевыми связями, активируется каждый из них) и предусмотреть удаление (**Trigger: Delete** присоединяется к каждому юниту даже, если они объединены в группу):

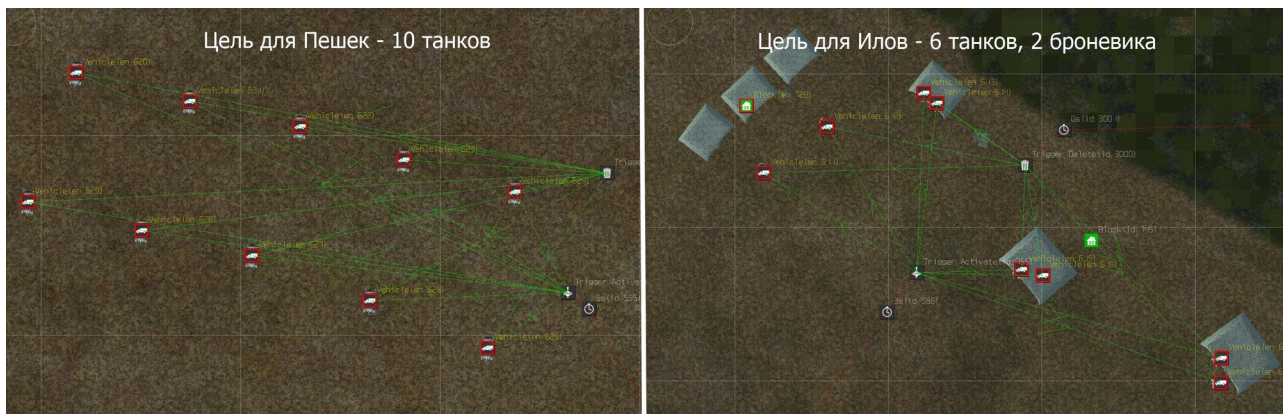


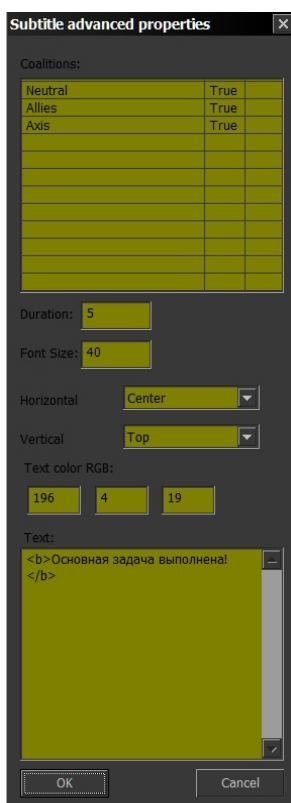
рис.39

### 3.10 «Заккрытие» целей. Счётчики и субтитры.

Следуя за оригиналом, мы выделили в миссии 3 задачи — основную, дополнительную и скрытую. Сообщить игроку, что он выполнил ту или иную задачу, нам помогут два **MCU – Translator: Subtitle** (титры) и **Trigger: Counter** (счётчик).

Благородная простота исходной миссии диктует только 4 возможных текстовых сообщения по ходу игры: «Основная/Дополнительная/Скрытая задача выполнена» и «Миссия выполнена». Так и сделаем. Добавим к этому только сообщение о критических повреждениях, т. к. игра о них промолчит.

Титры в игре настраиваются так:



**Coalitions** – возможность показать надпись только одной стороне, актуально для сетевой игры. Также, можно показывать сторонам разный текст одновременно (требуются свой транслятор для каждой стороны).

**Duration** – время в секундах, которое текст будет на экране.

**Font Size** – размер шрифта.

**Horizontal/Vertical** – регулировка расположения текста на экране.

Цвет текста в **RGB** и, ниже — поле для самого текста.

Настройки для данной миссии — на рисунке; остальные надписи не будут выделены тегами **<b></b>**, а цвет сообщения «Миссия выполнена» будет зелёным (RGB 0,225,0).

Три условия, по которым должны появляться титры, уже известны: смерть четырёх танков в колоннах, смерть 75% статичной техники и критические повреждения самолёта игрока.

Условия выполнения скрытой задачи будут максимально простыми — уничтожение хотя бы одного объекта из тех, что там есть, включая грузовики (прочность/ **Durability** у всех строений «скрытой цели» тоже невелика — 500-1500 единиц).

Выполнение всей миссии будет «засчитываться» при успешной посадке в Ленинске, при условии, что в миссии была хотя бы попытка атаки целей. Позже, это будет описано подробно.

Рассмотрим теперь **Trigger: Counter** — именно он будет считать уничтоженную технику и запускать титры.

**Trigger: Counter** — это простой счётчик, любая входящая связь, сработав, увеличивает его на единицу, когда число входящих сообщений сравнится с цифрой, заданной в счётчике, он выдаст исходящий сигнал.

Если в настройках поставить галку **Reset After Operation**, счётчик обнулится по достижении заданного значения и будет готов к повторному использованию.

При деактивации триггером **Deactivate**, уже посчитанный результат не сбрасывается и если счётчик активировать снова, он продолжит счёт с той цифры, на которой был остановлен.

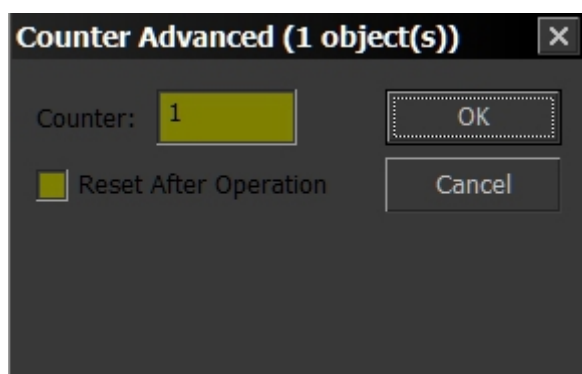


рис.40

Для того чтобы уведомления о выполненных задачах приходили как надо, нам понадобится один счётчик «на 4» (колонны), один счётчик «на 12» (75% от 18-и единиц статики) и счётчик «на 1» для склада ГСМ. В качестве входящих сообщений используем «событийную связь» **OnKilled Event** (где назначается связь было показано на **рис.30**).

Правда, есть одно но — по неизвестной науке причине, техника в игре иногда может выдавать более одного сообщения о своей смерти, поэтому, для корректного подсчёта «вражеских потерь» придётся применить



Выглядят они так:



рис.41

Т.е., **OnKilled Event** от каждого танка к счётчику «на 1» и уже от него — целевая связь к счётчику общей суммы, который запустит титры. Итого,  $18+8=26$  дополнительных триггеров, т. к. в случае с ГСМ, нет нужды городить огород — там титры должны приходить после первого же «сообщения о смерти» и эти сообщения можно отправлять сразу в общий счётчик «на 1». (Снизу на рисунке виден «пучок» целевых связей, приходящих в общий счётчик от второй группы техники).

На этом с целью всё: что должно поехать — поедет, уничтоженное посчитается и если игрок будет молодцом, игра скажет ему об этом. Начнём расставлять остальные самолёты в миссии.

### 3.11 Боты. Атака наземных и воздушных целей.

Возвращаемся к вопросу про «рисунок боя» - что делает наземка уже известно, теперь надо понять, как будут вести себя самолёты, ведь, помимо «группы игрока», в миссии ещё 13 замечательных ботов.

Изначально представлялась такая картина: 4 бота-Ила второй группы подходят к цели чуть раньше «группы игрока» и начинают атаку прямо на глазах у игрока (заодно, для тех, кто летает без маркеров будет дополнительная наводка на расположение танков).

Мессеры-перехватчики прилетают через небольшой промежуток времени, примерно достаточный для того, чтобы игрок сделал один заход на цель без помех (ПВО не учитываем).

Пешки подходят позже всех, когда уже завяжется бой, они вообще — этакий «глас с небес»: откуда-то сверху прокричали: «сброс по команде», сбросились и ушли.







рис.43

1. **Translator: Mission Begin** – начало миссии.

2. Таймер 3 секунды — стандартная задержка после старта, запускает движение Яков к первой **КТ** (3) и секундный таймер (4), который «дёргает» команду формации (5) — правый пеленг, средняя (**safe**) дистанция.

Крыша полетела «нарезать круги» над Ленинском, следующие **MCU** вступают в действие после взлёта штурмовиков:

6. Таймер **2xYak** — мы уже видели его на **рис.29**, когда оформляли взлёт. Получив сигнал о взлёте Илов, этот таймер прежде всего деактивирует все КТ Яков деактиватором (7), т. к. круговой маршрут закончен и надо исключить возможность срабатывания этих **КТ** в дальнейшем. Затем, через таймер секундной задержки (8), истребителям даётся команда прикрытия — **MCU Command: Cover** (9). Эта команда объектной связью соединяется с головным Яком, а целевой — с головным Илом. Настройки по умолчанию (галка **Cover Group** означает, что «сданы под охрану» все 4 штурмовика целевой группы):

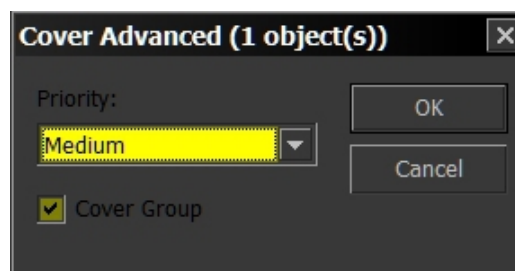


рис.44



Теперь, до возвращения этих самолётов с задания, умываем руки. Задача прикрытия поставлена и остальное — на совести **AI** (Искусственного Интеллекта, ИИ). К слову, об особенностях его в игре «**Ил-2 БзС**» можно узнать из [этой статьи](#).

Но вернёмся к **рис.43** — там ещё остался «набор **MCU**», который начнёт работу, когда игрок пойдёт на посадку — этот сигнал придёт к таймеру (**10**) от последней **КТ** Илов.

Таймер **10** запускает команду «отмены текущей задачи» — **MCU Command: Force Complete (11)** — она скажет Якам: «свободны!» и поблагодарит в случае успешной «прикрышки». Далее, через обязательную задержку в неск. секунд (таймер **12**), истребителям будет дана команда посадки (**13**).

Настройки **Force Complete** по умолчанию (не меняем):

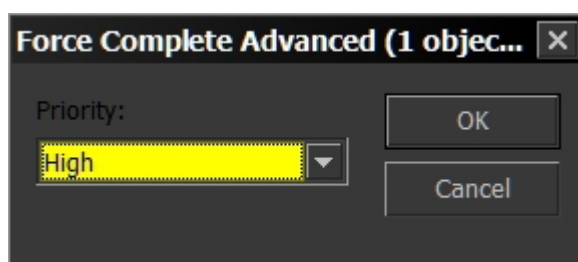


рис.45

«Крыша» готова, ненадолго вернёмся к «группе игрока» — пока она просто идёт по маршруту. Заставим её атаковать танковую **колонну 1**.

Схематично (всё не относящееся к атаке колонны удалено, а нужные **MCU**/объекты «придвинуты в кадр») делаем так:

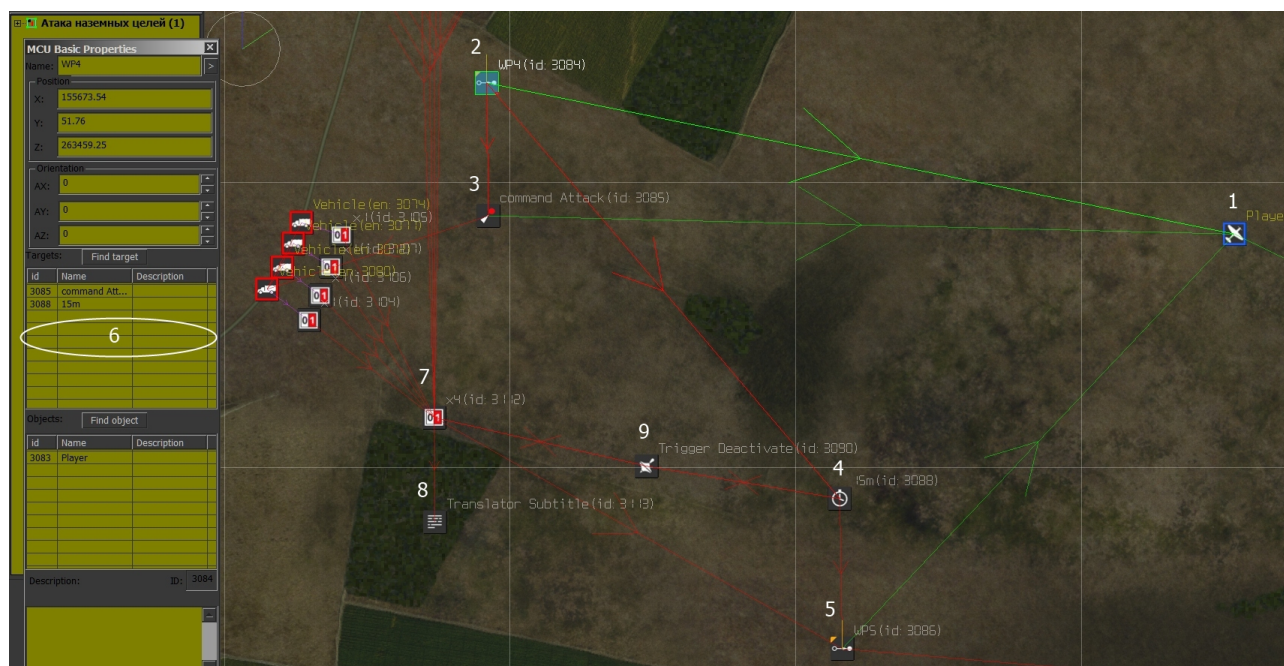


рис.46



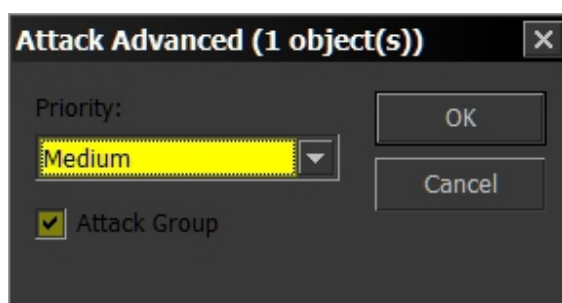
На рисунке, самолёт игрока (1) «дёргает» четвёртую **КТ** (2 на **рис.46**) своего маршрута (напомним, что радиус этой **КТ** =6 км), которая запускает команду атаки – **MCU Command: Attack** (3) и таймер (4) на 15 минут.

Зачем этот таймер? - дело в том, что на время атаки цели, надо прервать полёт Илов по маршруту, а для этого целевая связь между их четвёртой и пятой **КТ** (2 и 5 на **рис.46**) должна быть удалена из поля **Targets** (6) в окне базовых настроек 4-ой **КТ** (клавишей **Delete**).

Теперь у Илов есть 15 минут на выполнение основной задачи, после чего таймер (4) «дёрнет» **КТ** (5) и они вернуться на маршрут (полетят домой); но, скорее всего, 4 целевых танка умрут гораздо раньше, поэтому **КТ** (5) включается также счётчиком (7), который был поставлен, чтобы запустить титры «Основная задача выполнена» (8).

Другими словами, штурмовики полетят домой, либо через 15 минут бесплодных атак (что маловероятно), либо, выполнив основную задачу. Это — взаимоисключающие события и деактиватор (9), поставленный между счётчиком (7) и таймером (4) и связанный с ними целевыми связями по принципу «туда-обратно», гарантирует, что только одно из них сможет «дёрнуть» **КТ** отхода штурмовиков (5).

**MCU Command: Attack** объектной связью присоединена к лидеру атакующей группы (самолёту игрока), а целевой связью — к головному танку нужной колонны. Команда имеет следующие настройки по умолчанию, менять которые незачем:



**рис.47**

Переходим ко второй группе штурмовиков. Они, так же как и Пешки, и Мессера, неактивны в начале миссии и появляются уже по ходу её, в нужное нам время. Для этого используем уже знакомую «чекзону» (**MCU Trigger: Check Zone**), которая активирует их при подлёте игрока.

Прежде чем начать детально описывать как всё это работает, стоит заметить, что изначально, все появляющиеся в воздухе самолёты расставлены на карте «на глаз», также «на глаз» заданы радиусы «чекзонам» и время основных таймеров. Окончательный вид они принимают только после нескольких тестовых прогонов миссии (в данном случае, пришлось чуть подвинуть одну «чекзону» и 2-3 раза поменять время в одном из таймеров).

На рисунке ниже, все нужные **MCU** придвинуты «в кадр», лишнее удалено — так будем делать и впредь в подобных случаях и, пожалуй, без дальнейших напоминаний.

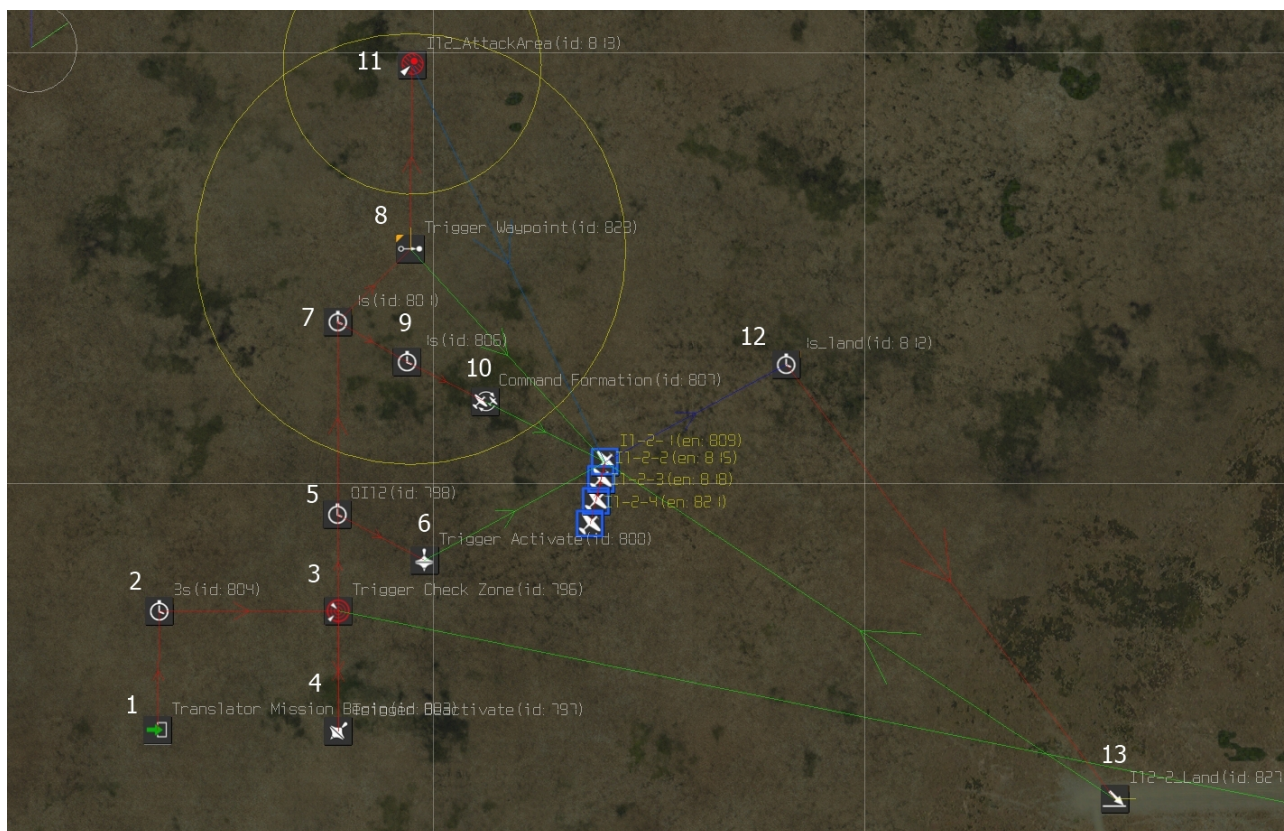


рис.48

**1 и 2.** Старт+3 секунды — как обычно.

**3.** «*Чекзона*» - объектная связь к самолёту игрока, радиус 15 км, запускает нулевой таймер (**5**), деактивирует сама себя деактиватором (**4**) — всё это мы уже делали.

**5.** Таймер, запускающий тоже уже знакомую процедуру: активация Илов (**6**), секунда задержки (**7**), запуск первой **КТ** (**8**), секунда задержки (**9**), команда формации (**10**).

**11.** **MCU Command: Attack Area** – команда «атаковать область», тут подробно:

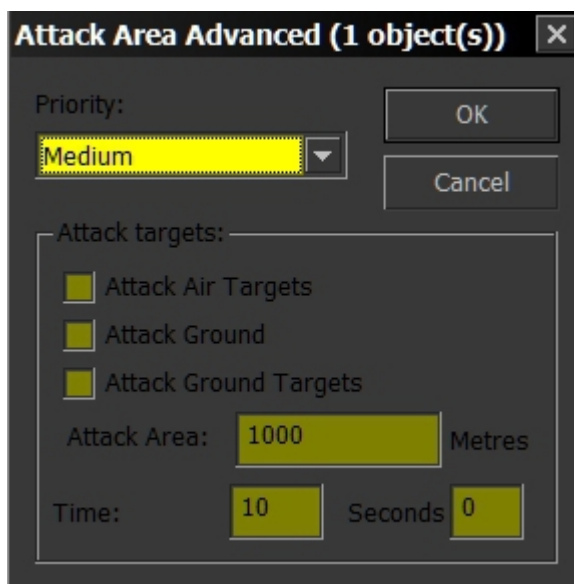


рис.49

Команда атаки области заставляет любую технику (и самолёты, и наземку) атаковать в заданном радиусе (поле **Attack Area**), в течение указанного времени (поле **Time**) нужные цели: воздушные (**Air Targets**), наземные (**Ground Targets**) или просто стрелять/бомбить по площади (**Ground**). Важным параметром для этой команды является *высота* (координата **Y** в окне базовых настроек, см. **рис.50**); например, при бомбометании боты сначала будут занимать указанную высоту, а только потом строить заход на цель и это надо учитывать.

Для наших штурмовиков настройки будут такие:



**рис.50**

Время действия команды, её расположение и высота подобраны опытным путём, радиуса в 200 м на такую цель более чем достаточно, а галка **Attack Ground** включена потому, что в этом случае Илы атакуют «с ходу», а не делают сначала долгий заход (можно было, конечно, повозиться с высотой и расположением **КТ**, которая «дёргает» команду атаки (**8** на **рис.48**), но так было проще, а нужный результат не изменился — иногда Илы уничтожают цель полностью, а иногда — нет).

**Attack Area** объектной связью присоединена к головному Илу, но на **рис.48** этого *не видно* и цвет связи, скорее, синий — подобное уже встречалось при оформлении взлёта «группы игрока» и здесь та же причина — рапорт.

**OnAreaAttacked Report** — от команды атаки (**11**) к таймеру (**12**), который «дёргает» команду посадки (**13**) — таким образом, когда истекнут 8 минут, указанные в настройках **Attack Area**, группа сообщит, что выполнила эту команду и через секунду получит команду на посадку. Илы начнут отход «по нулям».



Ещё раз — основное меню для назначения «событий» и рапортов:

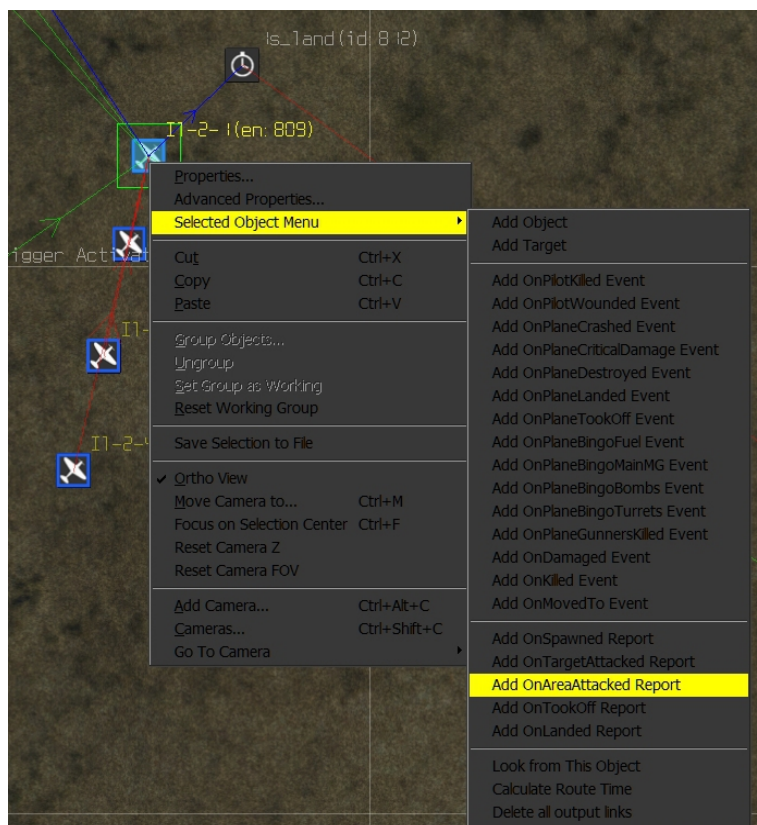


рис.51

Вооружение Илов — 2 бомбы по 250 кг + пушки ВЯ, без задней турели, в настройках ставим старт в воздухе и не забываем снять галочку **Enabled**.

Для посадки оборудован аэродром **Верхняя Ахтуба** уже описанным способом редактирования схемы движения на аэродроме (см. **рис. 23** и **24**). Так в миссии появляется ещё один «живой» аэродром (забегая вперёд — не последний) и переговоры садящихся самолётов второй группы с ним будут развлекать игрока по дороге домой.

(Обратите внимание: этот аэродром будет ближайшим советским аэродромом от цели и все подбитые боты, выйдя из боя, потянут именно к нему.)

А теперь — бомбардировщики: появляются они в миссии аналогично предыдущей группе. «Чекзона», активация, первая **КТ**, формация — всё то же, кроме таймера, выделенного кружком на **рис.52** (четыре минуты). Задержка нужна, чтобы Пешки подошли к цели последними, как запланировано (это можно было сделать также настройкой чекзоны или изменением начального положения самолётов в миссии).

Основные отличия от Илов здесь — настройки **Attack Area** и то, что Пешки не возвращаются на аэродром (условимся считать, что он далеко), а пропадают из миссии на обратном пути от цели, когда расстояние между ними и самолётом игрока превысит 25 км.

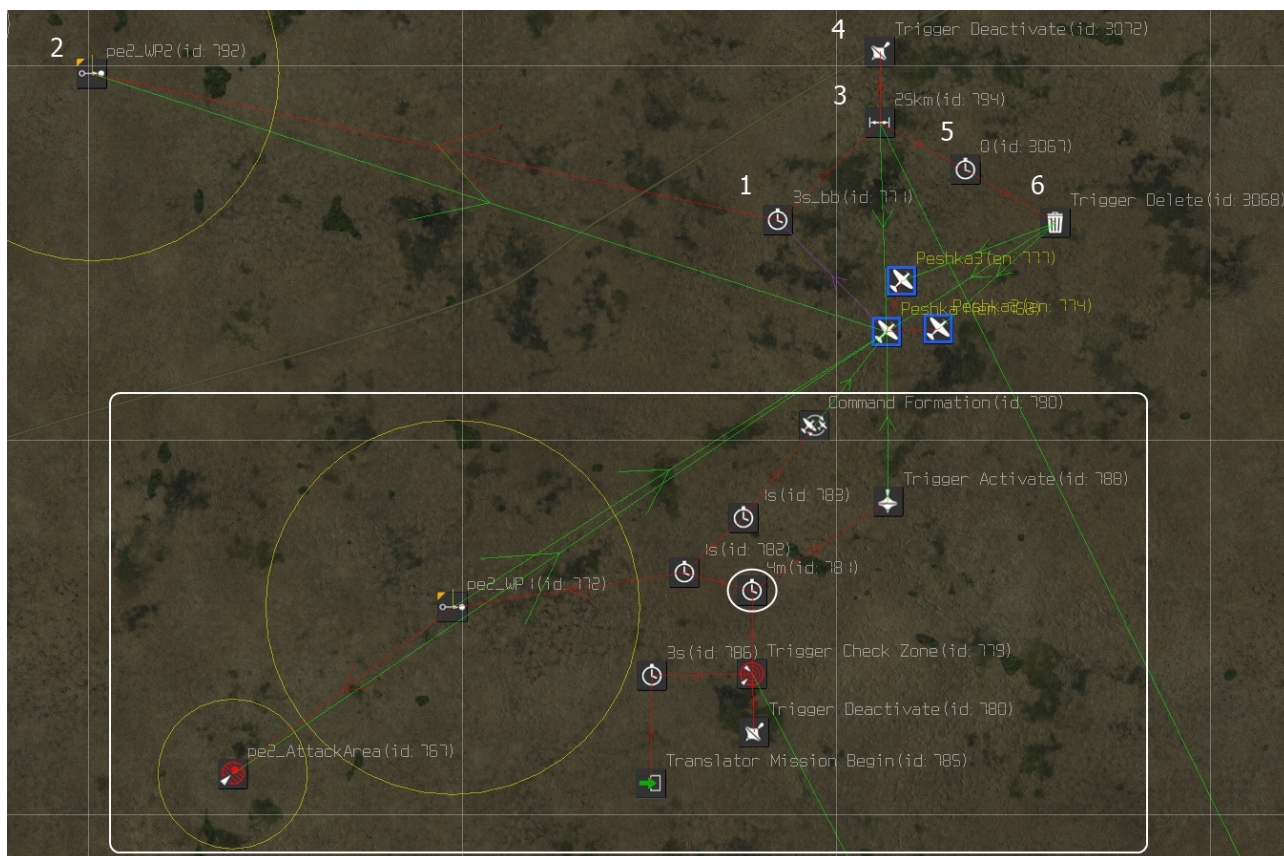


рис.52

Рамкой обведена часть **MCU**, повторяющая логику предыдущей группы.

После сброса бомб головной самолёт посылает сообщение об этом таймеру **3s\_bb** (1), при помощи «событийной» связи — в известном меню (см. **рис.51**) или в настройках самолёта — пункт **Add OnPlaneBingoBombs Event**, таймер (1) «дёргает» **КТ** (2) — она поставлена довольно далеко к северу от цели. Пешки разворачиваются и «пафосно улетают в закат»; также таймер (1) включает **MCU Trigger: Proximity** (3).

Триггер **Proximity** предназначен для динамического измерения расстояния между объектами, хотя бы один из которых присоединён к нему объектной связью, и срабатывает, когда эти объекты приблизятся друг к другу или отдалятся друг от друга на заданную дистанцию.

Здесь триггер настроен на срабатывание в случае, когда расстояние между самолётом игрока и головной Пешкой превысит 25 км (см. **рис.53**). Сработав, он деактивирует сам себя деактиватором (4) и, через нулевой таймер (5), «дёрнет» **Trigger: Delete** (6), который и удалит Пешки из миссии (можно было и просто «выключить» их тем же деактиватором 4, но пусть останется так).

В настройках команды **Attack Area** (см. **рис.54**) ставим высоту 3000 м, все галки, кроме **Attack Ground**, снимаем и Пешки бомбят по площади с горизонта; расположение команды подобрано так, что умирает обычно 4 целевых танка и, помня о том, что Илы могут уничтожить до восьми единиц наземки, получаем запланированное «рэндомное» выполнение ботами дополнительной задачи в миссии (100% её выполнение без учёта действий самолётов и ПВО противника). Не забываем, что команда атаки прерывает полёт Пешек по маршруту и целевая связь между их контрольными точками должна быть удалена.



## Настройки **Proximity**:

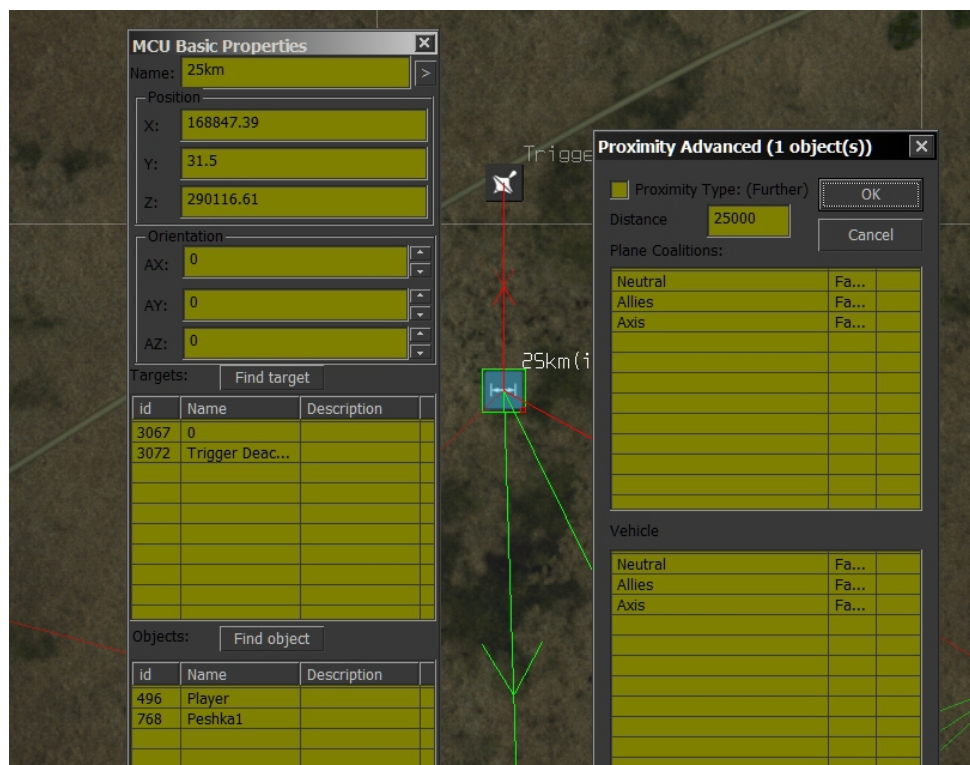


рис.53

## Настройки **Attack Area**:



рис.54

**Замечание:** если хотите, чтобы Пешки атаковали с пикирования, пользуйтесь командой **Attack** (высота полёта должна быть не менее 2500 м), а не **Attack Area** — к Штукам тоже относится.



Все советские самолёты в миссии расставлены по своим местам, настроены и нацелены. Настало время усложнить им жизнь, т. к. без противодействия с основной и дополнительной задачами справится даже автопилот.

Мессеры: сначала был сделан просто «патруль противника», т. е. группа из четырёх 109-х, подлетающая к цели где-то через минуту после самолёта игрока, и, имевшая задачу «атаки зоны» (**Command: Attack Area** радиусом 10 км), но по результатам тестовых вылетов было принято решение слегка «добавить драйва»:

Итак: перехватчики разделены на две пары — 1-я подходит на высоте 1500 м через минуту после начала атаки всё с той же целью «атаки зоны» и боты в ней — низкоуровневые; 2-я пара (боты-ветераны) подходит на 2-3 минуты позже 1-ой, на высоте 1000 м и имеет конкретную задачу — атаковать группу игрока (**Command: Attack**).

Появление обеих пар Мессеров сделано одинаково, рассмотрим детально только одну из них — первую:

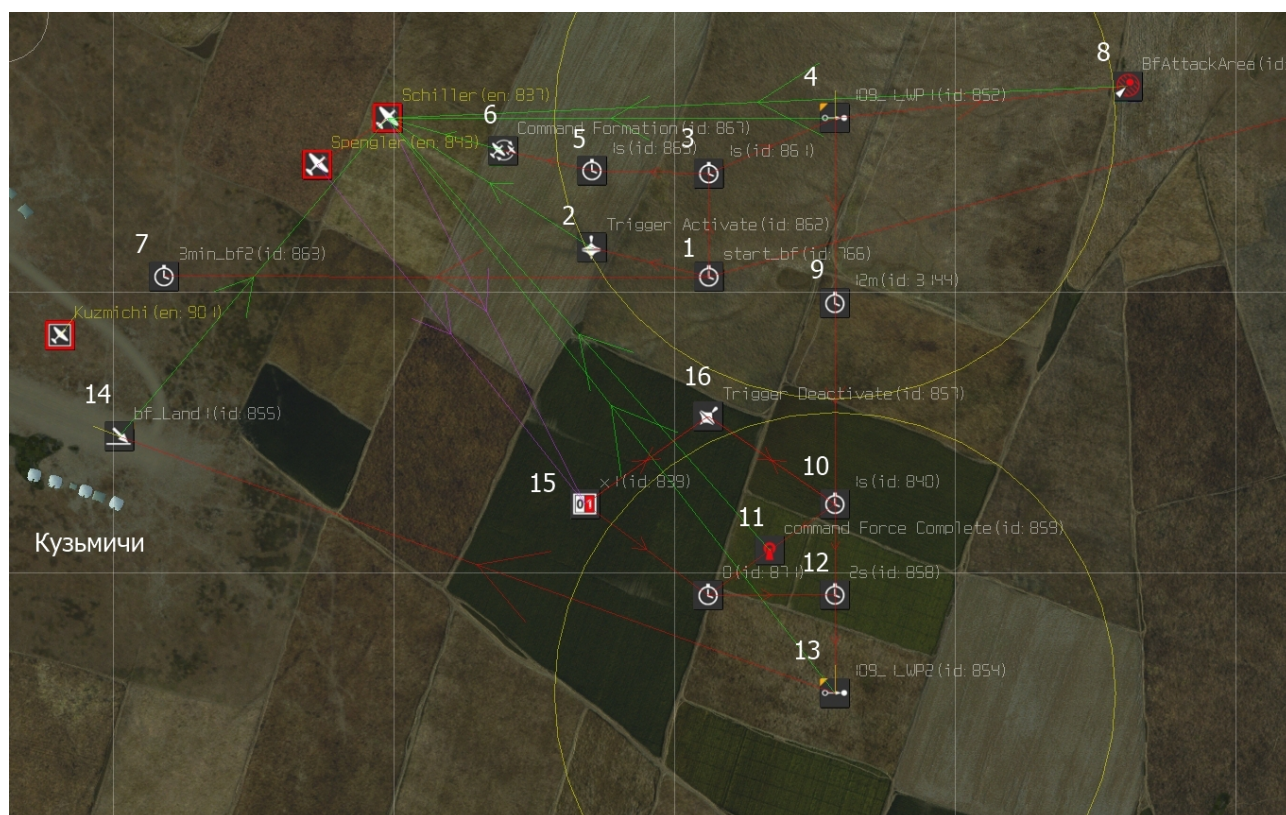


рис.55

В нулевой таймер **start\_bf** (1) приходит сигнал (от той **КТ** группы игрока, которая также отдаёт команду атаки Илам — см. **рис.46**) и запускается знакомая процедура: активация (2) — секунда (3) — полёт к первой КТ (4) — секунда (5) — формация (6). Вместе с этим включается таймер (7), который через 3 минуты даст старт второй паре истребителей.

Первая пара, достигнув **КТ** (4), получит команду атаки области (8) со следующими настройками:



рис.56

Далее по **рис.55: КТ (4)** запустит таймер (**9**) на 12 минут, который, через секундную задержку (**10**), отдаст команду отмены действия (**11**) и ещё через 2 секунды (**12**) «дёрнет» **КТ (13)** и Мессеры полетят домой. Достигнув **КТ (13)**, истребители пойдут на посадку на аэродроме **Кузьмичи**, за это отвечает команда **Land (14)**.

В миссии предусмотрен и другой вариант выхода пары 109-х из боя: если один самолёт получает критические повреждения, срабатывает та же «цепочка возврата» - единственный счётчик (**15**) включает команду **Force Complete (11)** и, через 2 секунды (**12**) - «дёргает» **КТ(14)**, т. е. точку отхода.

Варианты взаимоисключающие, т. к. **КТ (14)** не должна запускаться дважды, и деактиватор (**16**) позволит сработать только первому по времени из них.

Работу счётчика (**15**) обеспечивают «событийные» связи от каждого самолёта **OnPlaneCriticalDamage Event** (меню на **рис.51** для справки).

И последнее: контрольная точка (**14**) имеет высокий (**High**) приоритет выполнения как команда, т. е. Мессеры полетят в Кузьмичи «со всех ног», ни на что не обращая внимания (настройка аэродрома по примеру **рис.23, 24**).

Вторая пара перехватчиков сделана точно так же, а отличается от первой командой атаки (**Command: Attack** – целевая связь к самолёту игрока, объектная — к лидеру пары), высотой появления (1000 м), уровнем самих ботов (ветеран) и таймером, аналогичным таймеру **9** на **рис.55** — второй паре на атаку даётся только 10 минут.

(И ещё раз, важно: нет целевых связей между **КТ** истребителей, иначе никаких атак не случится — Мессеры пойдут по маршруту и сядут в Кузьмичах почти сразу.)

Настройки первой пары Мессеров на рисунке ниже:



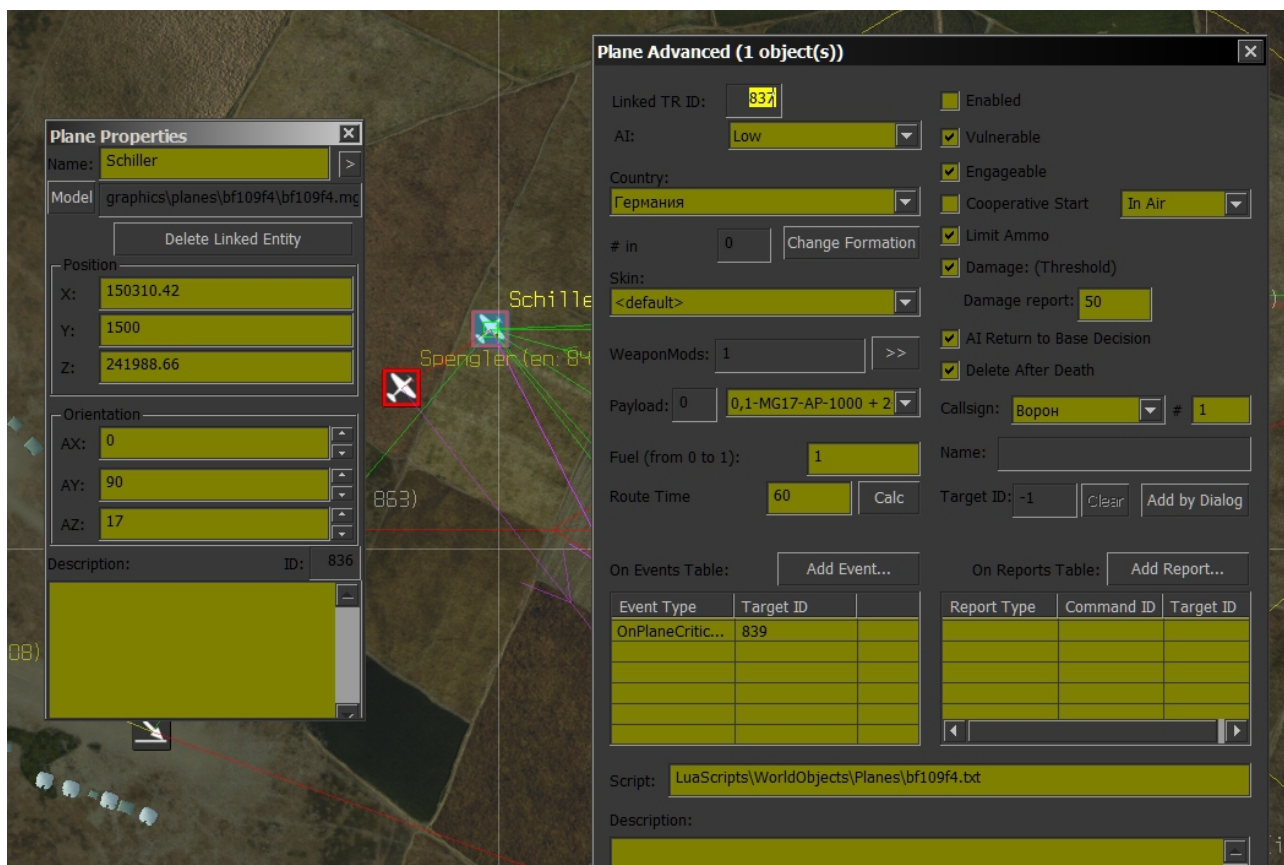


рис.57

Осталось ещё показать, как будет засчитываться выполнение миссии (всей миссии, а не только заданий в ней!) игроку.



рис.58



Основное условие — посадка на аэродроме Ленинск, дополнительное — хотя бы попытаться пострелять по целевой наземке. Да, игрок может «развернуться и улететь» от цели — миссия будет выполнена, плохо, но выполнена. Задачи миссии выполнены не будут (если только боты не постараются с дополнительной задачей).

На **рис.58** на врезке показано, что от каждого танка в колоннах, поставлена «событийная» связь к единичному счётчику (1) — **Add OnDamaged Event** (сообщение о повреждениях). Сработав, этот счётчик запустит чекзону (2), привязанную к самолёту игрока и расположенную в Ленинске (радиус зоны — 2 км), а также отдаст один сигнал счётчику (3).

Счётчик (3), собственно, и запускает титры «Миссия выполнена» (4), а для его срабатывания необходимо 3 события:

1. Повреждение одного целевого танка — сообщает счётчик 1.

2. Успешная посадка — сообщает самолёт игрока, посредством «событийной» связи **OnPlaneLanded Event**.

3. Посадка именно в Ленинске — сообщает чекзона (2); она для этого и ставилась: чтобы при посадке вне двухкилометровой зоны «домашнего» аэродрома миссия не засчитывалась (кстати, вполне уместно будет поставить такую же чекзону и в Верхней Ахтубе, ботам же можно «тянуть на ближний» и нет причин запрещать это делать игроку. Например, чекзону в Ахтубе может включать связь **OnPlaneDamaged Event** от самолёта игрока, чтобы миссию можно было корректно завершить там только на повреждённом самолёте).

Остальные **MCU** на **рис.58** нужны для случаев безусловного провала миссии — гибели пилота или критических повреждений самолёта игрока.

Критические повреждения: от самолёта игрока к таймеру (5) — сообщение **OnPlaneCriticalDamage Event**, таймер запустит титры (6) - «Критические повреждения — миссия провалена».

Смерть пилота: от самолёта игрока к таймеру (7) — сообщение **OnPilotKilled Event**, можно поставить титры, аналогичные предыдущим, а можно и не ставить — игра сама выдаст в чат: «такой-то убит».

Все эти варианты завершения миссии — успех и две неудачи — взаимоисключающие, чтобы сообщение «Миссия выполнена» не пришло после смерти пилота или после сообщения о критических повреждениях и наоборот — деактиватор (8) «пропустит» только то событие, которое произойдёт первым. Это «состоявшееся событие» запустит таймер (9) и миссия завершится через 30 секунд. Завершением миссии «заведует» **MCU Translator: Mission end (10)**.

**Замечание:** аварийная посадка тоже отнесена к «неудаче» миссии и соответствующая «событийная связь» (**OnPlaneCrashed Event**) поставлена к таймеру (5), так что, не ломайте шасси приземляясь. Впрочем, в этом случае смело можно записывать и «успех», т. к. пилот-то остаётся жив, но пусть нахождение способа как это сделать будет чем-то вроде «домашнего задания» для неравнодушных.

Вот и всё о «зачёте результатов» — танки целевых колонн может повредить только сам игрок или ведомые им боты, а эти боты не пойдут в атаку, если игрок не долетит до **КТ** над целью и т. д. Все оговоренные ранее условия выполняются.

Теперь в миссии готовы все самолёты и целевая наземка — общий вид в редакторе:

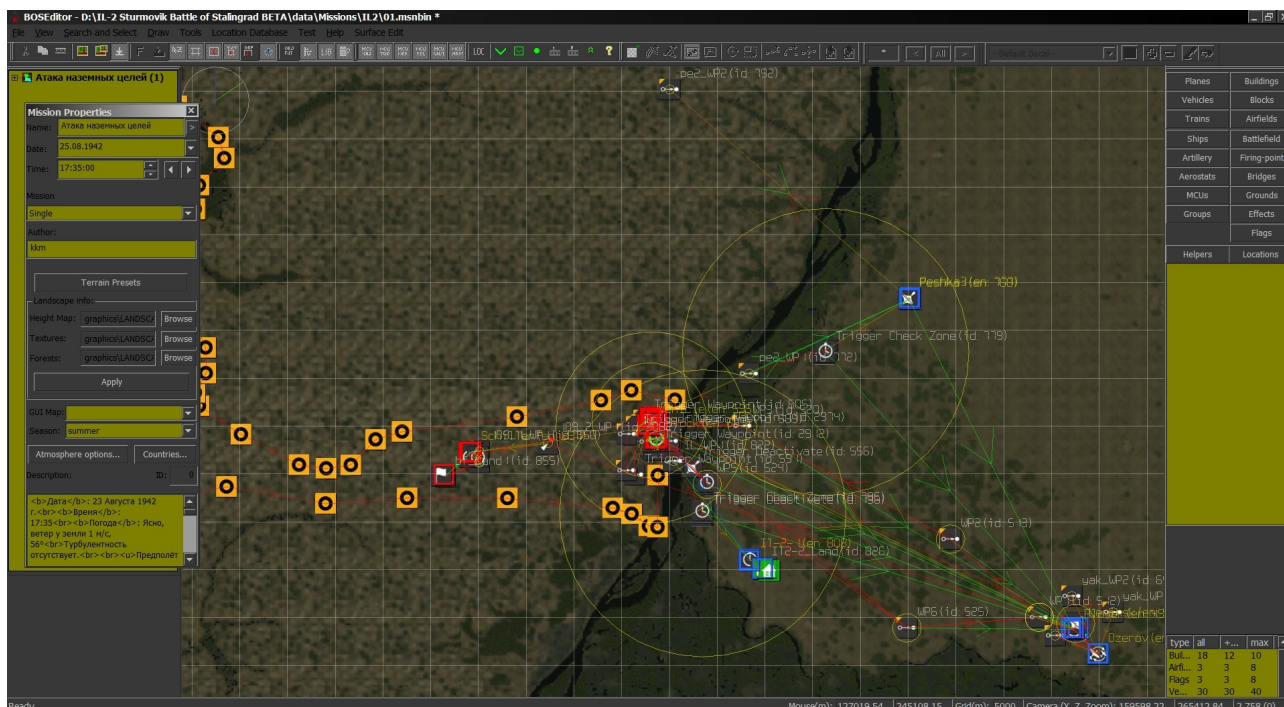


рис.59

Как выясняется, «всё не так уж сумрачно вблизи»©. Сейчас надо расставить зенитки, наладить некую «аэродромную жизнь» и подумать о декорациях и эффектах.

### 3.12 ПВО.

В исходной миссии, ПВО в районе цели минимальное, не будем усердствовать и тут: два мобильных **Flak 38** (по одному у каждой группы статичной техники) и два зенитных **MG** в Акатовке у топливного склада — этого достаточно (уровень **AI** у всех самый низкий). В самом деле, немцы только что прорвались и ни о каких стационарных зенитках речь не идёт.

Задействуем уже имеющиеся **MCU** активации/удаления (1), «пристегнём» к ним флаки и пулемёты, снабдим их командой атаки зоны (3) и включим эту команду через секунду-другую после появления юнита (2):

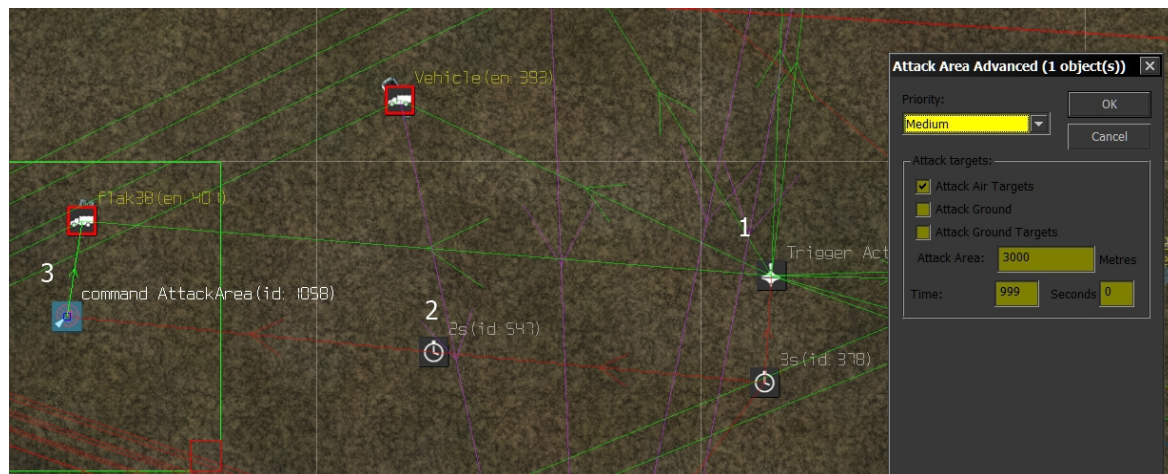


рис.60

На аэродромах всё совсем просто — там зенитки стоят с начала и до конца миссии и нужно только обеспечить их командой стрельбы (**Mission Begin** ---> **Timer** (3 секунды) ---> **Attack Area** и объектная связь к орудию).

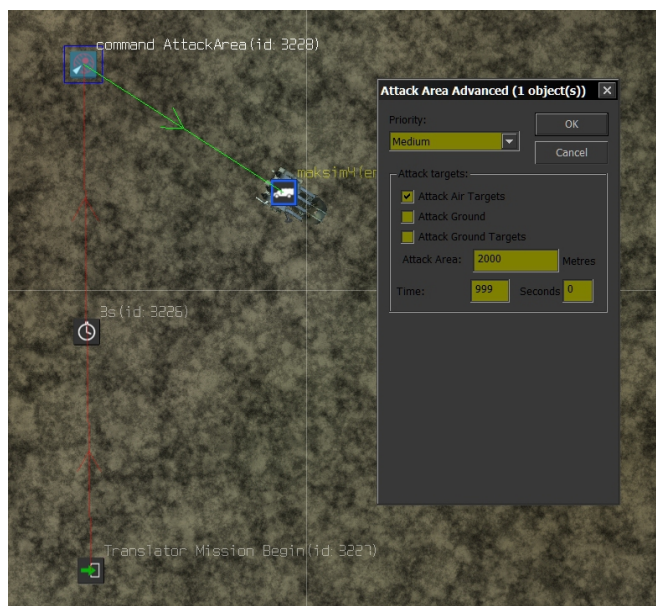


рис.61

Расстановка, состав и количество, естественно, произвольные — руководствуемся тем, что в Ленинске и Кузьмичах ПВО будет для декорации, т. к. враги туда просто не долетят, а вот в Ахтубу, отходящие от цели штурмовики, вполне могут «притащить хвост» и есть смысл поставить заслон посерьёзнее, и поднять уровень ботов-зенитчиков.

### 3.13 Поезда. Артиллерия. Антураж.

Все «сюжетообразующие» объекты в миссии уже расставлены и работают, оставшиеся будут служить для того, чтобы мир в игре не выглядел пустым, т. е. создавать так называемую «атмосферность»™. Конечно, у каждого могут быть об этой атмосферности свои представления, поэтому, коротко перечислим, что показалось уместным сделать в этой конкретной миссии, останавливаясь лишь тогда, когда попадутся ещё не описанные технические моменты.

Миссия штурмовая, высота полёта в ней невелика, думается, кое-что из описанного ниже не останется незамеченным даже теми, кто летает на экспертных настройках сложности.

Т.к. контрольные точки маршрута игрока привязаны к нас. пунктам, а рядом с двумя из этих точек есть и аэродромы, то первым делом, приходит в голову «добавить жизни» именно там.

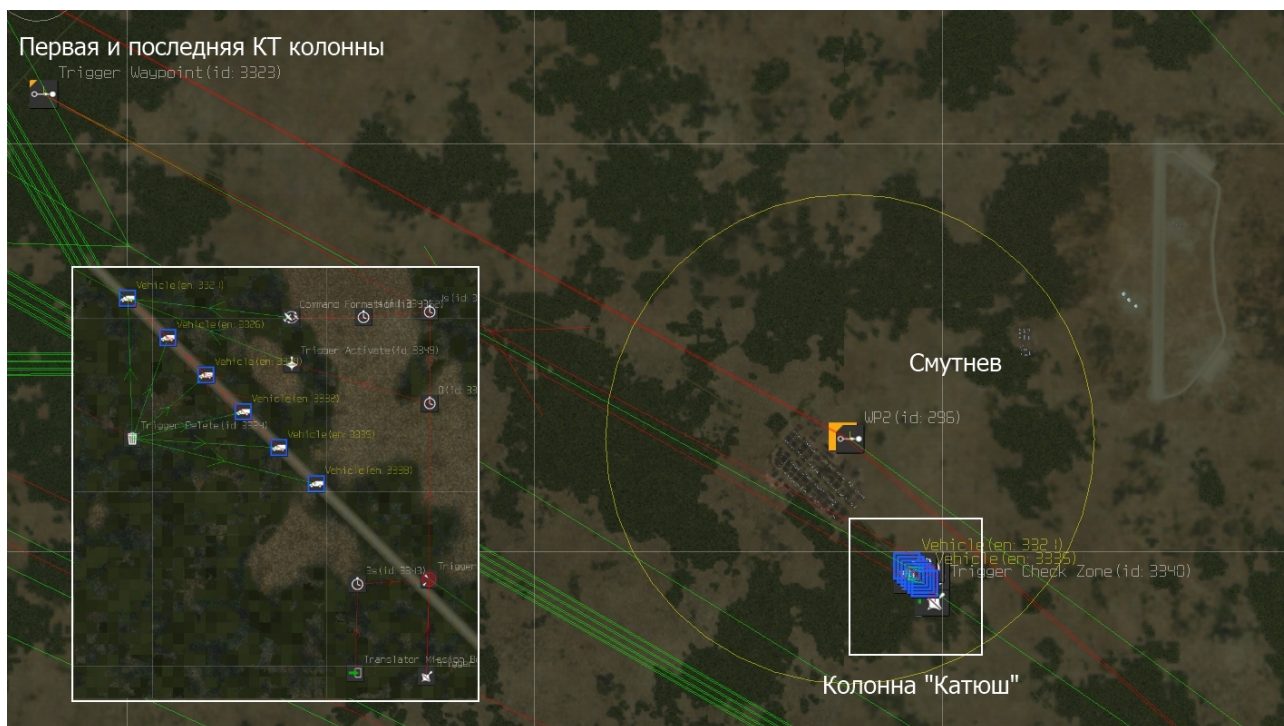
**Район цели:** первым делом, «сломать» саму **Ерзовку** — наступление танкового корпуса и нарядные домики плохо сочетаются друг с другом. Дымы, воронки — ещё остановимся на них подробнее.

Ну и конечно, 3 действующих аэродрома: **Ленинск**, **Верхняя Ахтуба** и **Кузьмичи**. На всех них стоят зенитки, все можно «просматривать по F5», так что, какое-то дополнительное оформление просто необходимо и Ленинску безусловно — особое внимание.

Пойдём по маршруту:

**Смутнев** (посёлок и рядом аэродром) — колонна «Катюш» едет в сторону фронта, при подлёте игрока должна быть совсем рядом с посёлком или ехать прямо по нему:

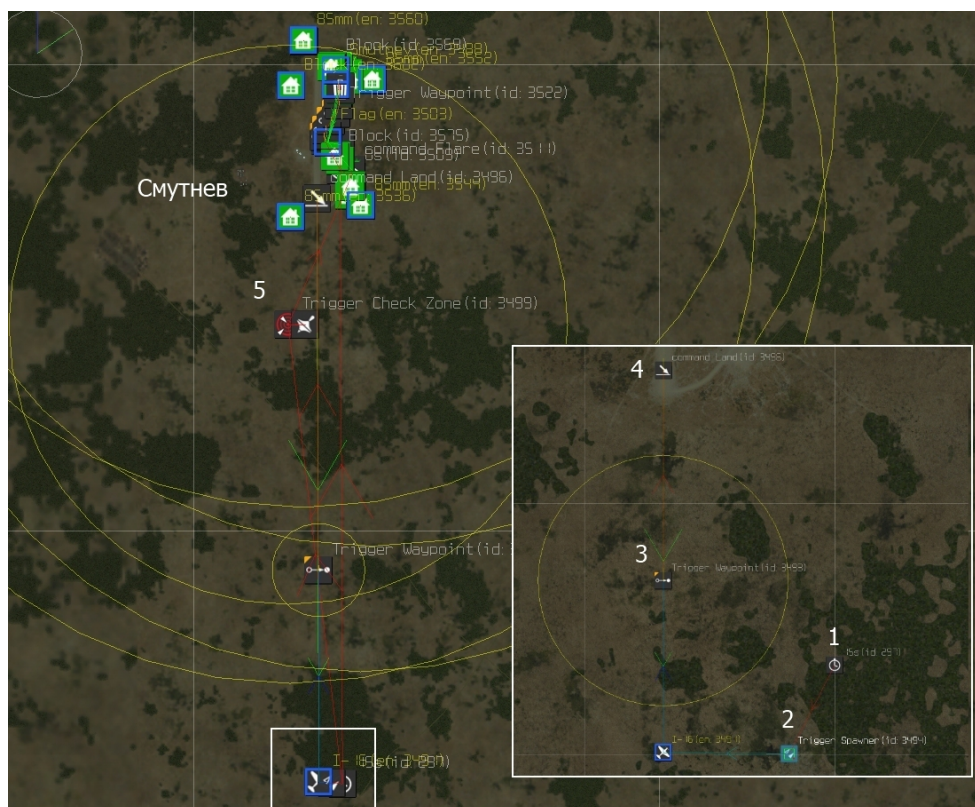




**рис.62**

Ничего нового — активация «чекзоной» (радиус 9 км), запуск движения, формация и удаление из миссии по достижении последней **КТ**.

А на аэродром в это время садится одиночный «Ишак» — **И-16**:



**рис.63**

Пришлось «оживить» ещё один аэродром «сверхплана», Ишак сел бы и без этого, но не было бы его переговоров с аэродромом.

На **рис.63**, на врезке под цифрой **2** изображён новый для нас **MCU: Trigger Spawner** — этот триггер «рождает» объекты в миссии (в данном примере — самолёт И-16). В отличие от активатора, **Spawner** может создать сколько угодно копий своего целевого объекта, т. е. если, после того как Ишак приземлится и пропадёт из миссии, «дёрнуть» **Spawner** ещё раз — точно такой же Ишак появится в том же месте карты и повторит весь «жизненный путь» первого — пойдёт на посадку в Смутневе.

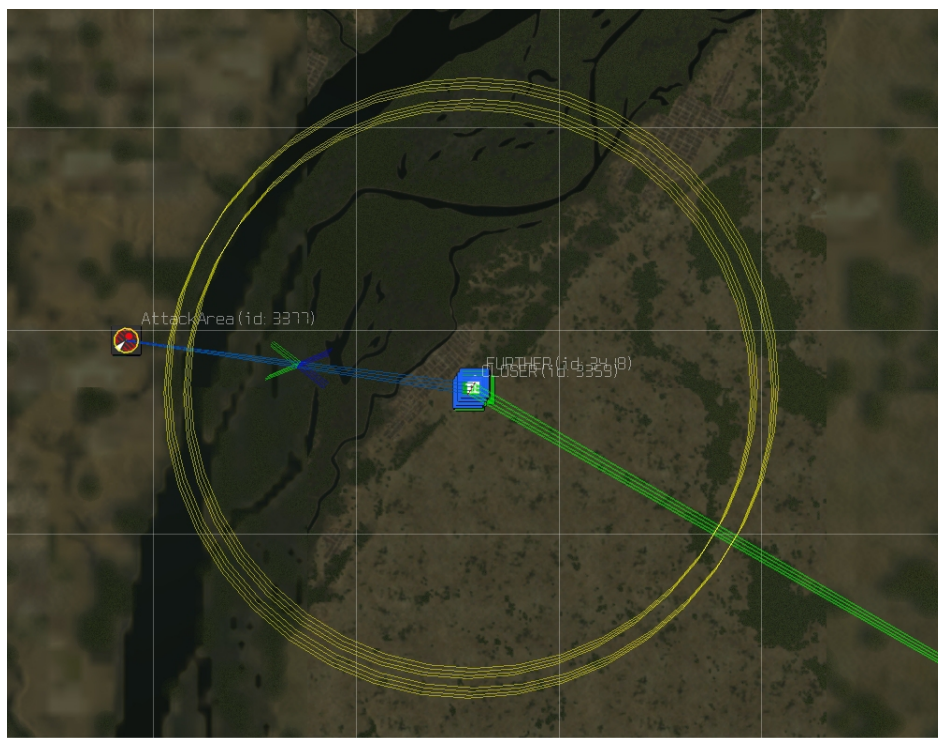
Главное ограничение при использовании **Spawner'a** состоит в том, что с его помощью можно «рождать» только одиночные объекты; вы можете присоединить к нему группу техники и эта техника появится в миссии, но группой не будет; в колонне автомашин, например, по маршруту поедет только головной автомобиль, остальные останутся без команды.

Так же, этим триггером нельзя «рождать» игрока, а объект, к которому от триггера поставлена целевая связь, не будет присутствовать в миссии до своего «рождения», причём, галку **Enabled** в настройках такого объекта убирать не надо (в отличие, опять же, от процедуры активации).

В настройках всех самолётов и наземки есть специальный рапорт для этого триггера: **OnSpawned Report** — с его помощью объект может «сказать сам себе», что делать сразу после «рождения». На **рис.63** такой рапорт поставлен от самого триггера (**2**) к **КТ** (**3**), которая запускает команду посадки (**4**); стартовый сигнал приходит в таймер (**1**) от первой **КТ** маршрута игрока.

Таймер (**1**) также активирует легковушку на аэродроме, а «чекзона» (**5**) заставляет её дать сигнальную ракету и немного проехать по аэродрому при подлёте игрока на 3 км.

**Средне-Погромное** (большой нас. пункт) — поблизости батарея дальнобойных орудий, стреляет в район цели в Ерзовке — общий вид:



**рис.64**



Здесь за основу взята готовая группа гаубиц со всей логикой, извлечённая из *генераторной миссии игры*. Необходимо сказать, что изучение таких миссий вообще очень помогает разобраться в том, что и как работает в редакторе (и в игре) — многие велосипеды стоят там готовые, их не надо изобретать.

**Важно:** последняя запущенная вами оффлайн-миссия, будь то быстрый вылет или миссия в кампании, сохраняется в «...data\Missions» как «\_gen.Mission», её можно открыть в редакторе, пересохранить в любое место как одиночную (*single*) и т. д.

Гаубицы на **рис.64** активируются при подлёте игрока на 6 км и удаляются из игры, когда игрок отлетит от них на 6,5 км, логика рассчитана на многократное срабатывание и для одного орудия выглядит так:

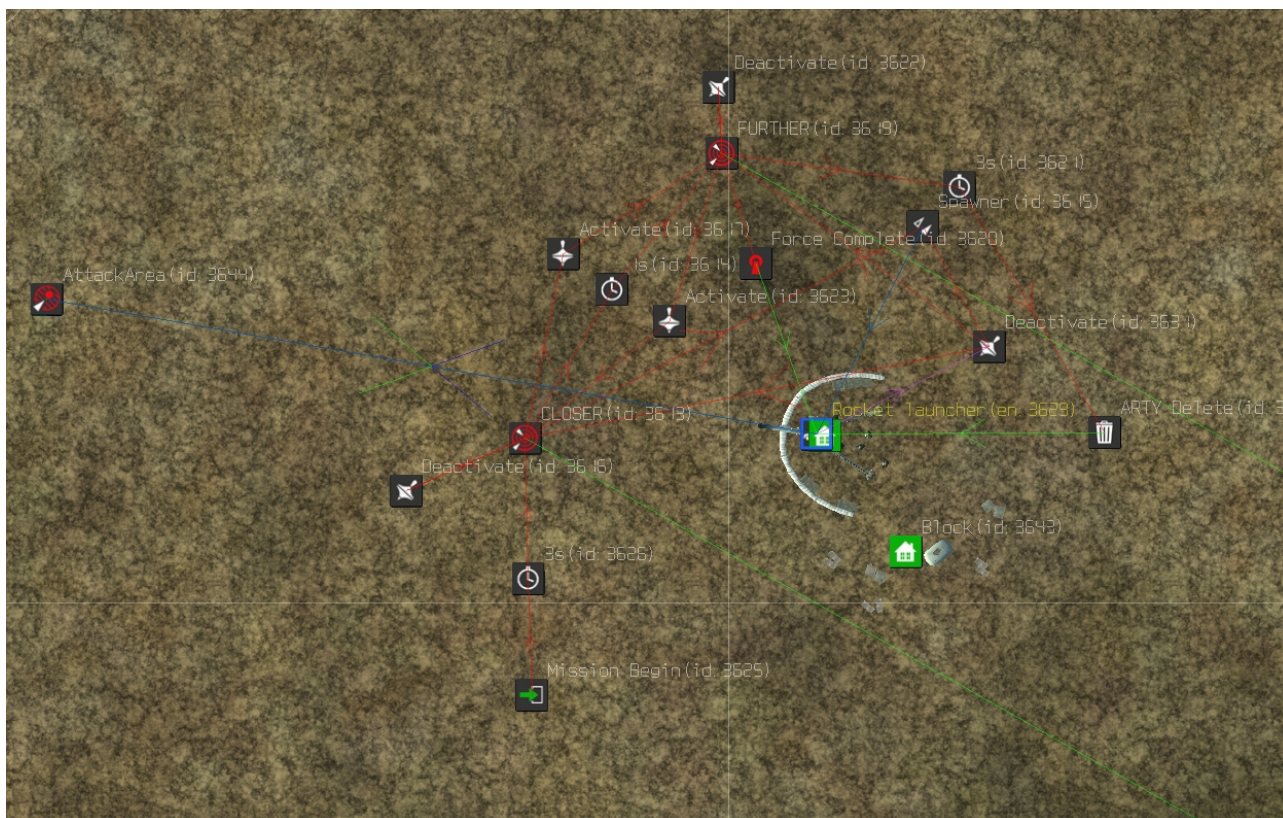


рис.65

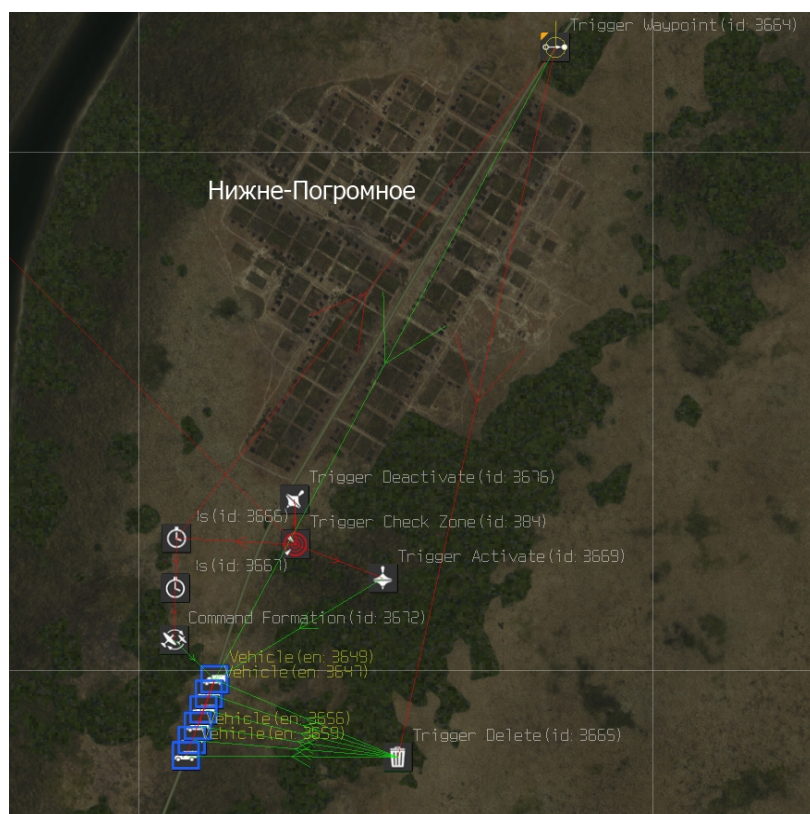
Не станем подробно разбирать эту группу, все **MCU**, входящие в неё, уже были описаны выше, заметим только, что **рис.65** является хорошей иллюстрацией работы триггеров **Check Zone** в случае двойной проверки (ближе/дальше) для одного объекта — сначала включается чекзона «ближе» (**closer**) меньшего радиуса (6 км), а затем, на выход самолёта из зоны, включается чекзона «дальше» (**further**) большего радиуса (6,5 км).

На **рис.64** хорошо видно множество зелёных объектных связей, идущих от чекзон группы к самолёту игрока. Казалось бы, проще настроить эти чекзоны на «любой советский самолёт», но т. к. триггеры сработают при каждом входе/выходе из зоны каждого объекта, а предугадать все возможные положения самолётов во время игры несколько затруднительно, лучше поставить эти объектные связи, во избежание накладок в работе всей группы **MCU**, тем более, что сделать это очень легко, воспользовавшись назначением связей по имени объекта в окне базовых настроек триггера **Check Zone** (о этом уже говорилось ранее, см. пояснения к **рис.27**).

Команду стрелять после появления в миссии, пушки получают с помощью недавно рассмотренного **OnSpawned Report**.



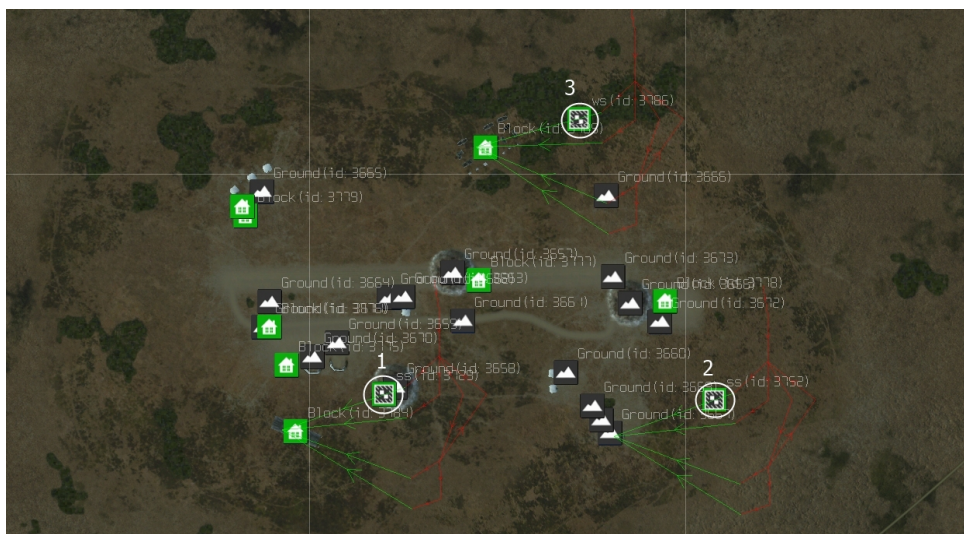
**Нижне-Погромное** — колонна тяжёлых танков (**КВ-1**) едет через деревню:



**рис.66**

Всё знакомо: активация по чекзоне и т. д., чекзону включает **КТ** маршрута игрока над целью, т. е. достаточно игроку долететь до Ерзовки и на обратном пути танки будут ждать его появления в пяти км от них. Маршрут танков совсем короткий, потому что **КВ-1** — самый медленный из движущихся наземных юнитов в игре.

**Аэродром**, севернее нас. пункта **Заплавное** — недавно подвергся налёту вражеских бомбардировщиков. Хорошо заметны воронки, разбитые самолёты; некоторые строения ещё дымятся. В редакторе:



**рис.67**

И в игре:



рис.67

Чтобы сделать подобную декорацию, воспользуйтесь объектами из вкладок **Grounds** и **Effects** библиотеки объектов, их не так много по умолчанию — пять видов дымов, сирена и четыре вида воронок:

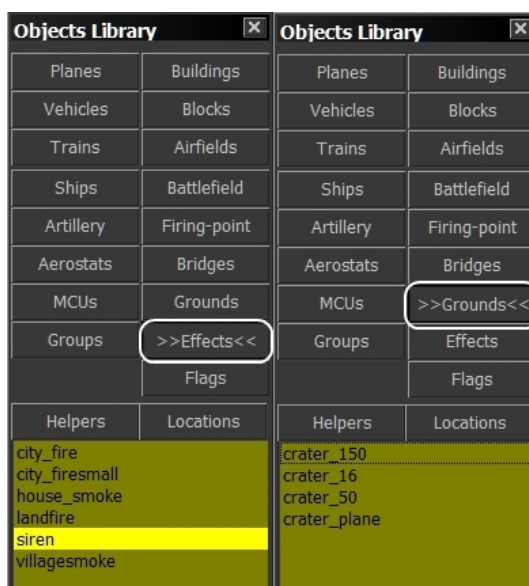


рис.68

Цифрами **1**, **2** и **3** на **рис.66** отмечены группы **MCU**, которые обеспечивают дымы на **рис.67**. Это, опять-таки, готовые группы из кампании игры, их можно встретить во многих «прифронтовых» деревнях и в самом Сталинграде, в тех её (кампании) главах, где бои идут в непосредственной близости от города.

### Несколько слов об эффектах в игре:

Для работы они должны быть «оживлены».

В отличие от построек и наземной техники, высота (координата **Y**) у них может меняться вручную и (особенно при импорте групп с эффектами на карту) за этим нужно следить, чтобы дымы в игре не «повисли в воздухе» — здесь пригодится уже упоминавшаяся кнопка **Set on Ground** на панели инструментов.

Для дымов важна ориентация по направлению (координата **AY**), она должна соответствовать направлению ветра, выставленному в миссии, чтобы дымы «стелились по ветру» (малозаметный нюанс, неактуальный для «посадочного костра»/**landfire** и «дыма из трубы»/**house\_smoke**, т. к. эти дымы всегда направлены вертикально вверх).

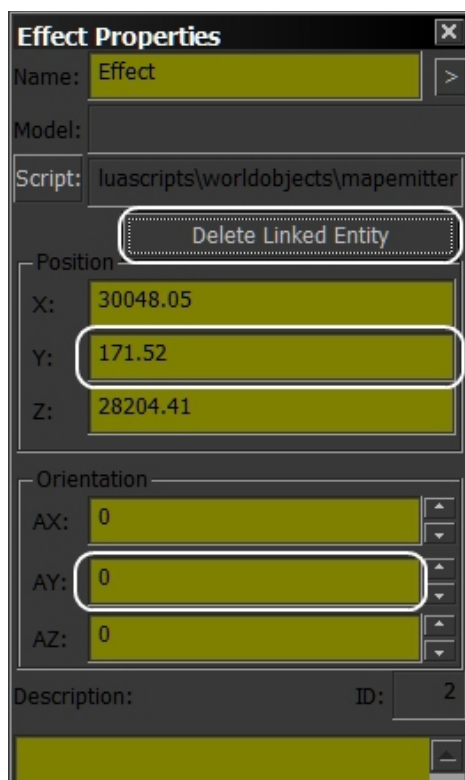


рис.69

Запускает/останавливает и дымы, и сирену команда **MCU Command: Effect** с настройками, соответственно, **start** или **stop**.

Эффекты типа: **city\_fire**, **city\_firesmall** и **villagesmoke**, имеют ограниченное время отображения и их надо «зацикливать» для получения «постоянного дыма» — таймеры во взятых из игры готовых группах лучше всего помогут разобраться с этим.

Всё вышесказанное *не относится* к воронкам (**Grounds**), за исключением замечания про высоту — её надо контролировать кнопкой **Set on Ground**.

Впрочем, в редакторе это отлично видно — подвигайте любую воронку по карте, рассматривая её в приближении, и всё станет ясно.



Применим эти же эффекты для окончательного оформления цели около **Ерзовки**, саму деревню «разрушим» способом, описанным в главе [3.4 Декорации](#). Думается, что и аэродромы в Акатовке и Ерзовке, тоже не должны поражать «утренней свежестью»:



рис.70

## Поезда.

Последняя часть маршрута игрока перед посадкой в Ленинске проходит рядом с железной дорогой и было бы по меньшей мере странно, упустить такую возможность и не запустить по этой дороге паровоз.

Во вкладке **Trains** библиотеки объектов находятся локомотивы и все доступные виды вагонов в игре, но для того, чтобы запустить состав, не надо «цеплять вагоны по одному» — это делается в настройках самого локомотива:

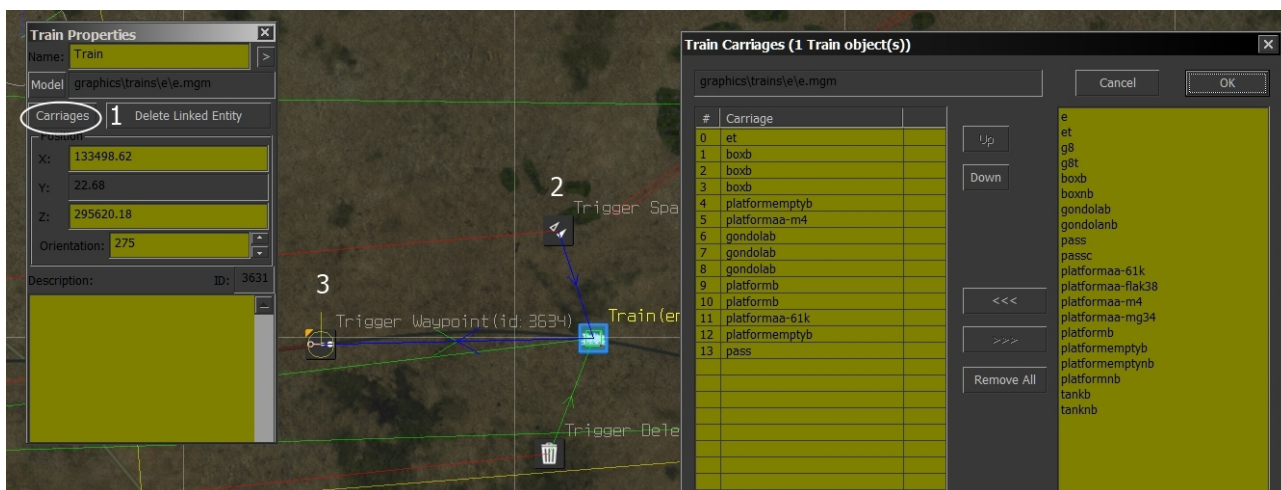


рис.71

Итак, поставим локомотив на рельсы, «оживим» его и нажмём на кнопку **Carriages** в окне базовых настроек (1 на **рис.71**). Появится окно «**Train Carriages**», т. е. «Состав поезда». В правом поле этого окна — список всех доступных вагонов, в левом поле — список вагонов, уже прицепленных к локомотиву.

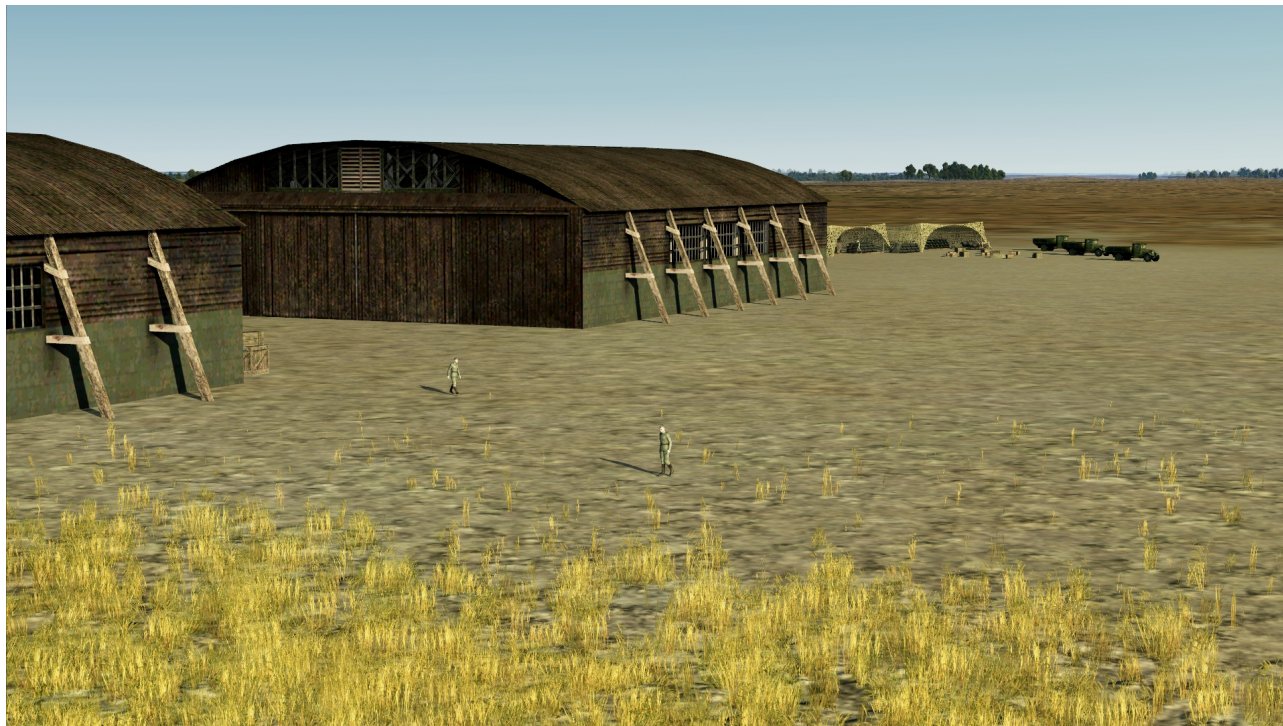
Кнопками «<<<» и «>>>» добавляем/убираем вагоны, кнопками «**Up**» и «**Down**» меняем их положение в составе и, таким образом, формируем эшелон.

Поезд на **рис.71** появляется в миссии с помощью **spawner'a** (2), сигнал к которому приходит от очередной чекзоны (её, в свою очередь запускает **КТ** маршрута игрока — см. пояснения к **рис.66**), и начинает движение, послав рапорт **OnSpawned Report** своей первой **КТ** (3).

Необходимо заметить, что для движения поездов в игре, достаточно «дёрнуть» первую **КТ** их маршрута — специальной команды «ехать по дороге» им не требуется, но будьте внимательны, если маршрут поезда проложен через разъезд/станцию с несколькими ж/д путями — не все из них могут быть «рабочими» и лучше проверять как поезд проезжает в подобных местах.

И, наконец, аэродром **Ленинск** — технически, ничего нового там нет: декорации из вкладки «**Blocks**», автомобиль, поставленный с таким расчётом, чтобы попадать в поле зрения игрока на старте, колонна на ближайшей дороге и несколько грузовиков,двигающихся по принципу «от склада к складу», которые активируются перед посадкой игрока.

Единственное, о чём ещё не было упомянуто — это простейший способ получить «людей» на аэродроме — **ботов-солдат**, которые, расхаживая между строениями, дополняют собой картину «аэродромной жизни»; для этого надо всего лишь «оживить» несколько декораций и присвоить им нужную «национальность» (после этого можно нажать на **Delete Liked Entity**, чтобы во всех смыслах «не умножать сущности» — боты-солдаты останутся):



**рис.72**



### 3.14 MCU, не вошедшие в миссию.

Миссия готова. При её создании не были использованы некоторые **MCU** — они просто не понадобились. Однако, представляется нужным написать о них, хотя бы и с минимальными пояснениями:

**1. Command: Behaviour** — команда, динамически изменяющая, некоторые параметры объектов в миссии. Как то: уязвимость/неуязвимость объекта, конечный/бесконечный боезапас, уровень **AI** ботов и нац. принадлежность юнита. Особый интерес представляет возможность изменения плейнсета на аэродромах в онлайн-игре (галка **Float Param** команды и настройка **Set Index** самолётов на аэродроме). Не работает с декорациями.

**2. Command: Damage** — возможность повреждать/ремонттировать объекты с настройкой от небольших повреждений до полного уничтожения и от лёгкого ремонта до полного восстановления.

**3. Command: Move** — не используется.

**4. Command: PlayAnimation** и **Command: PlaySound** — команды проиграть анимацию или звук для **MCU Translator: Animation operator**, работают только в связке в этом транслятором.

**5. Translator: AI POI** — самолёты на автопилоте и боты, не имеющие команд в миссии, будут лететь к центру зоны, указанной этим транслятором, а настройки транслятора определяют поведение таких «потеряшек». Оффлайн, насколько известно, не используется. Работоспособность в текущей версии игры не проверялась.

**6. Translator: Camera operator** — возможность расставлять камеры в миссии со множеством гибких настроек, с помощью **MCU Trigger: CameraPoint** эти камеры могут двигаться по заданным траекториям (**CameraPoint**, по сути, контрольные точки для камер, настраиваются как обычные **KT**, также могут расставляться автоматически, с зажатым **Shift**'ом).

**7. Translator: Complex Trigger** — триггер, использующийся в онлайн-миссиях, в основном как «продвинутый вариант чекзоны», срабатывает по изменению состояния указанной техники в зоне, имеет много настроек, не все из которых работают на данный момент.

**8. Translator: Influence Area** — возможность «размечать территорию» (определять её нац. принадлежность) в миссии. Используется онлайн, если сервер запущен с «экспертным» пресетом сложности (и только с ним), этот транслятор «посчитает вам плен» или посадку «у своих».

**9. Translator: Media** — медиа-вставка в миссию: изображение, видео, флешка. Весьма полезная вещь, о нюансах использования можно написать отдельное руководство.

**10. Translator: Mission Objective** — определяет выполнение/невыполнение задач в миссии. Нужен для получения слепополётной статистики, которая в одиночных миссиях не работает. Используется в кампании игры и отчасти онлайн (при помощи стороннего софта).

**11. Translator: Next Mission** — позволяет запустить на сервере не следующую или случайную миссию из ротации, а именно ту, на которую ссылается этот транслятор.

**12. Translator: Server Input** — предоставляет возможность «дёргать» события в миссии извне.

**13. Trigger: Date time** — возможность «дёрнуть» события в миссии по отметке времени внутри этой миссии, а также, привязать события к «восходу» или «закату» в миссии. Время «восходов» и «закатов» по дням, можно узнать из файла **Daytime.cfg**, который находится в папке **Data** игры.



## Приложения.

### 1. Типы линий.

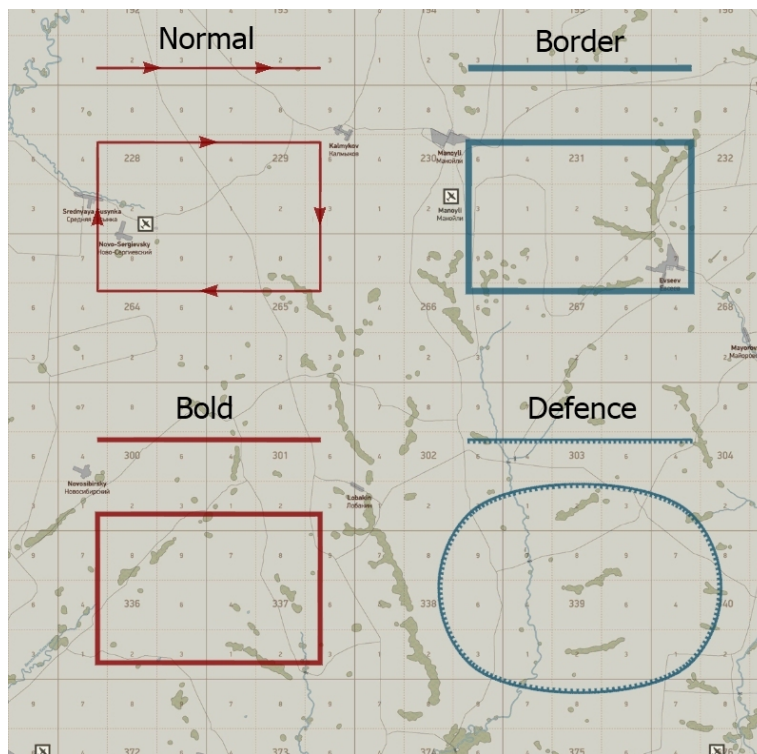


рис.73

Линии типа «Зона» (**Zone**):

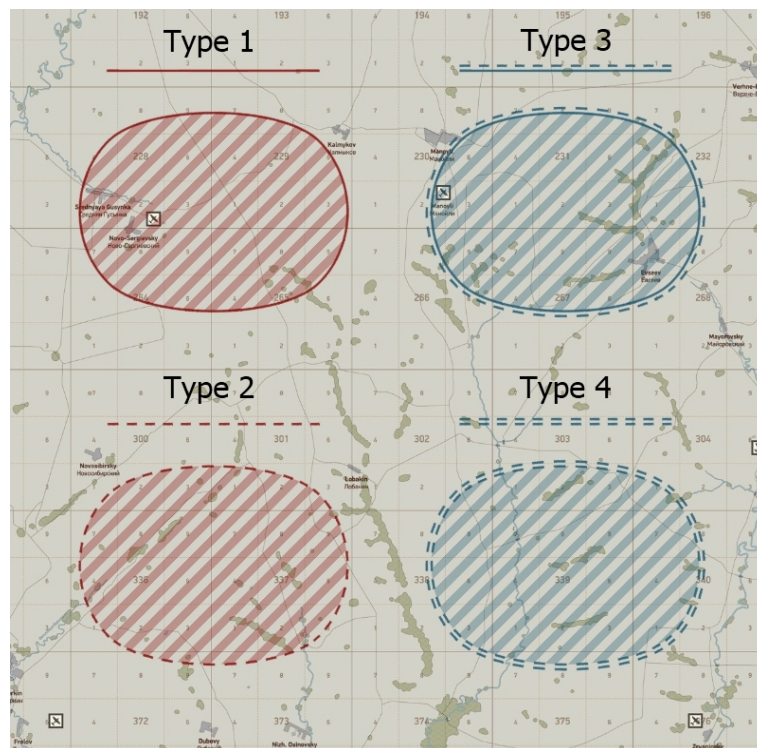


рис.74

Линии типа «Сектор» (**Sector**):

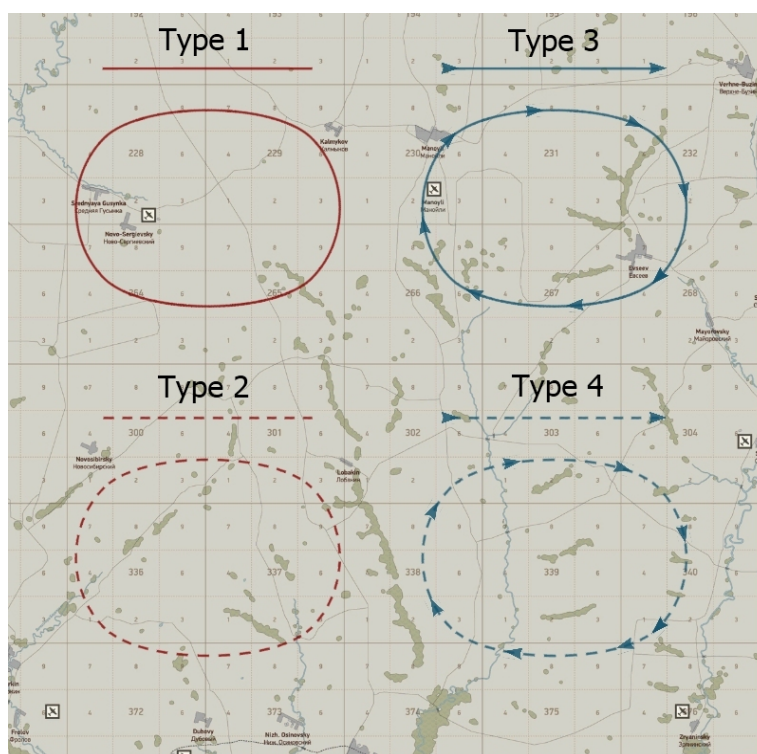


рис.75

Линии типа «Позиция» (**Positions**):

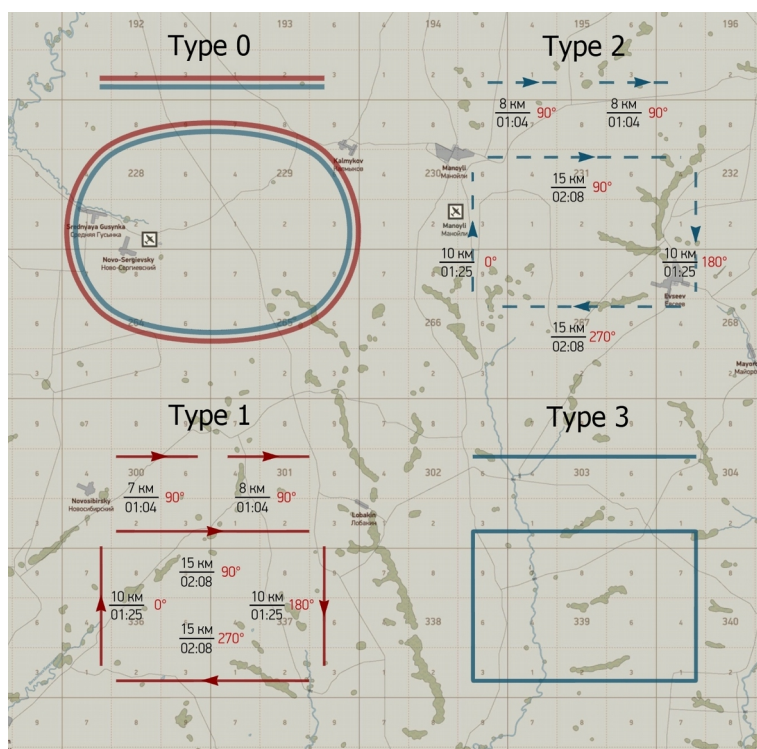


рис.76

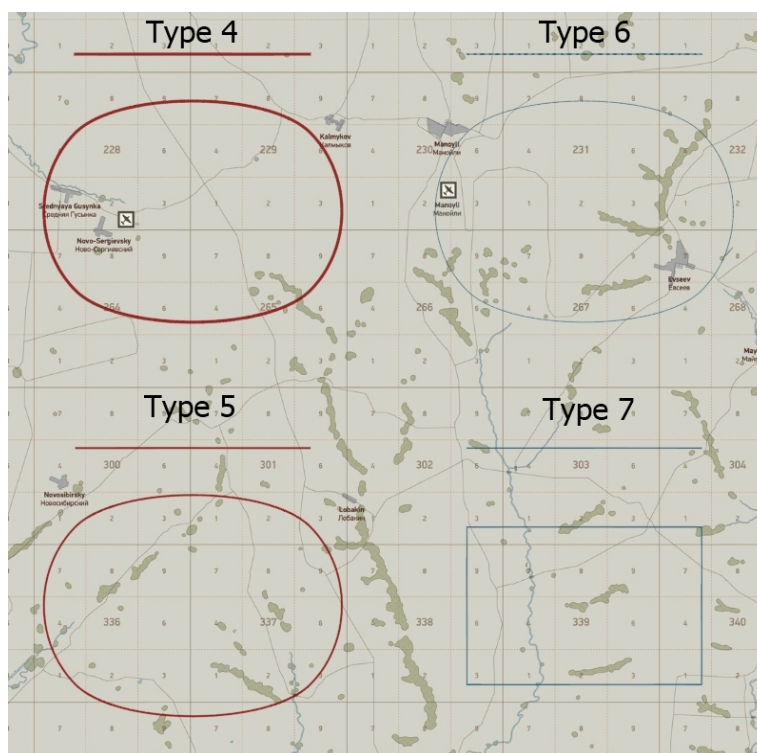


рис.77

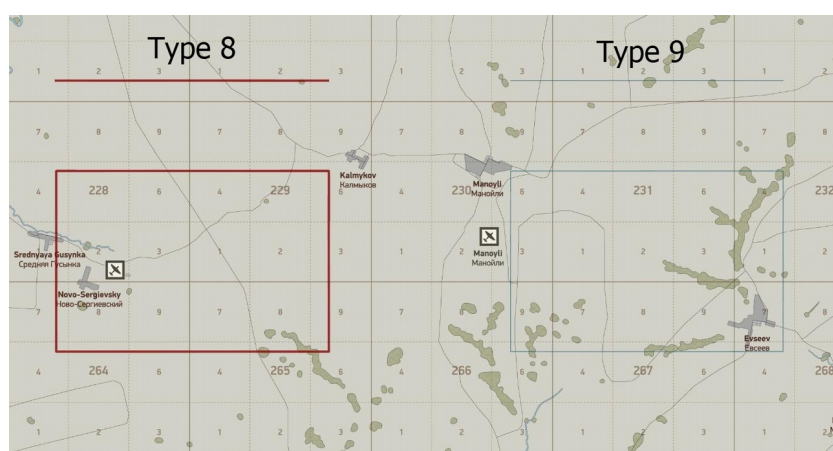


рис.78

Все линии изображены «автоцветом»: красные — **RGB 10, 0, 0** и синие — **RGB 0, 0, 10** (помним, что эта настройка не важна для линии **Positions Type 0**).

При расстановке иконок учитывайте, что направление целевых связей между ними определяет положение стрелок на линиях, «зубцов» для линии типа «оборона» (**Defence**) и сплошной/пунктирной составляющей для «зоны 3-го типа» (**Zone Type 3**).



## 2. Стрелки.

Простейший пример стрелок, которые можно нарисовать линией типа **Attack**:

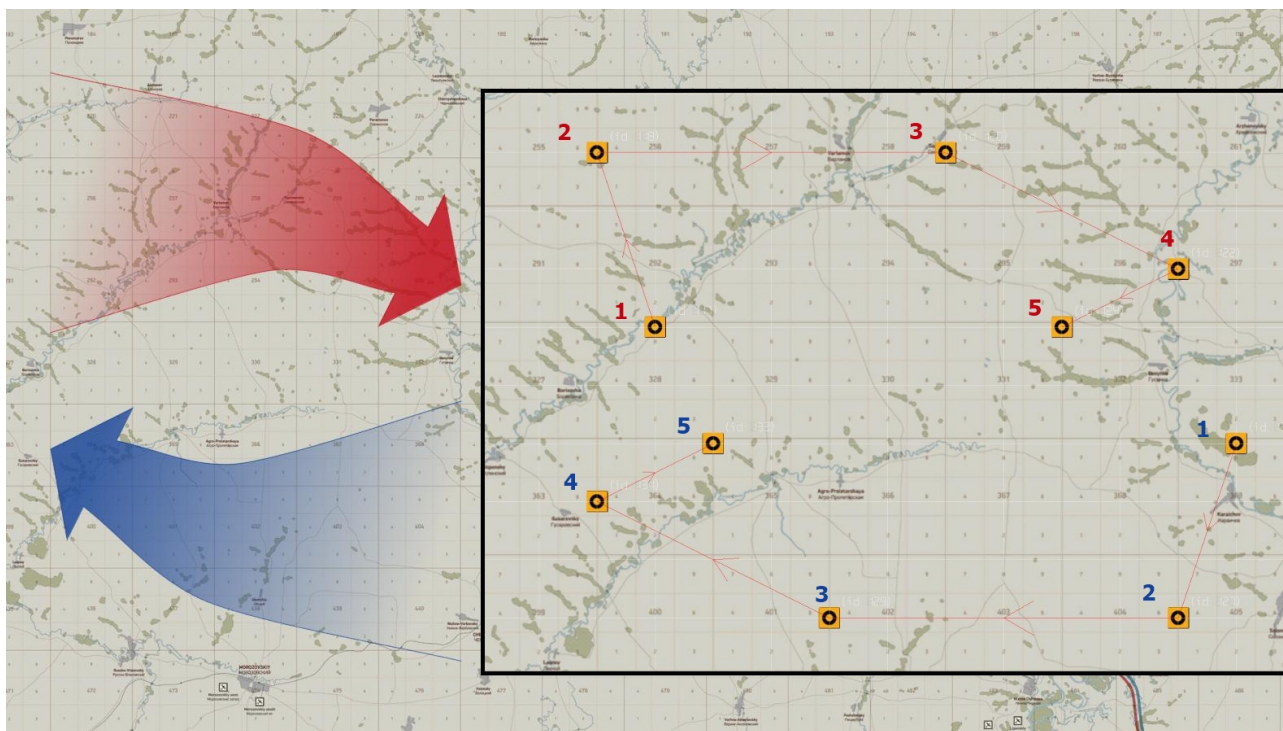


рис.79

А вот, что получится, если к красной стрелке добавить всего лишь одну целевую связь, а к синей — одну дополнительную иконку:

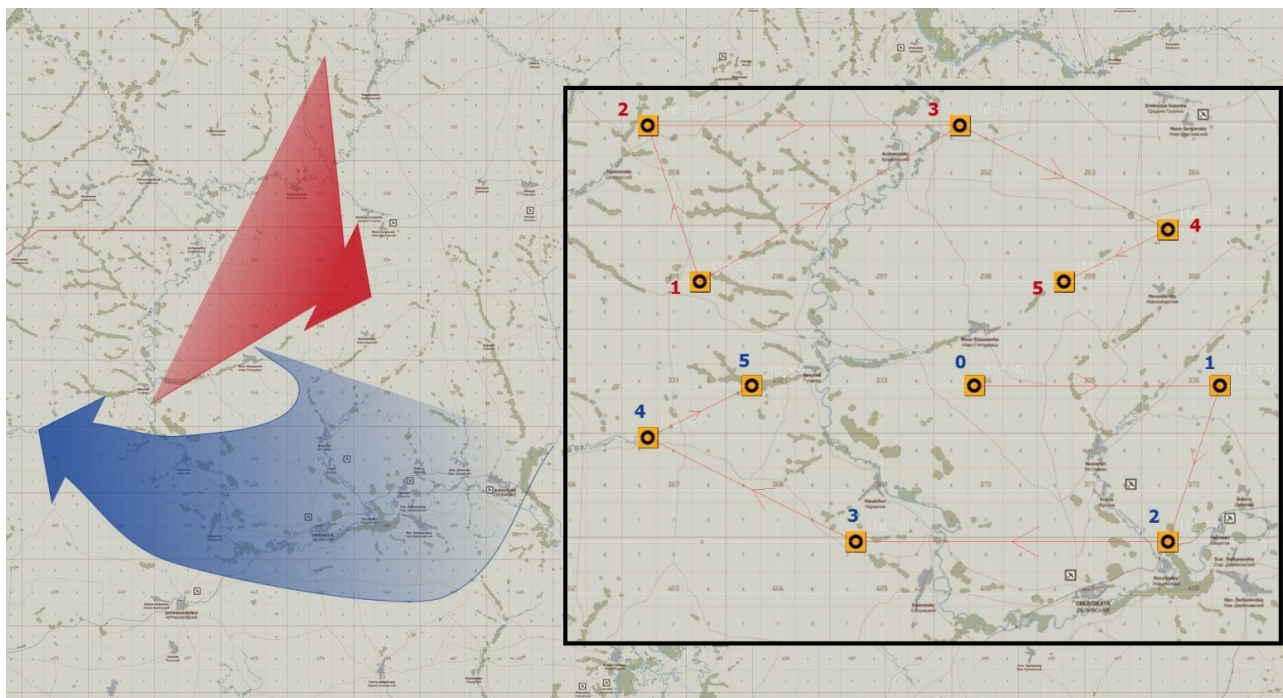


рис.78

Так что, экспериментировать с этой линией можно и нужно.

### 3. Расшифровка меню назначения связей.

Add Object	Назначить объектную связь
Add Target	Назначить целевую связь
Add OnPilotKilled Event	Пилот убит
Add OnPilotWounded Event	Пилот ранен
Add OnPlaneCrashed Event	Самолёт разбит (авария)
Add OnPlaneCriticalDamage Event	Критические повреждения
Add OnPlaneDestroyed Event	Самолёт уничтожен
Add OnPlaneLanded Event	Посадка
Add OnPlaneTookOff Event	Взлёт
Add OnPlaneBingoFuel Event	Кончилось топливо
Add OnPlaneBingoMainMG Event	Кончился бк
Add OnPlaneBingoBombs Event	Сброшены все бомбы
Add OnPlaneBingoTurrets Event	Кончился бк турелей
Add OnPlaneGunnersKilled Event	Убиты бортстрелки
Add OnDamaged Event	Объект повреждён
Add OnKilled Event	Объект убит
Add OnMovedTo Event	Сообщение о движении
Add OnSpawned Report	Рапорт о "рождении"
Add OnTargetAttacked Report	Выполнена команда атаки цели
Add OnAreaAttacked Report	Выполнена команда атаки зоны
Add OnTookOff Report	Рапорт о взлёте
Add OnLanded Report	Рапорт о посадке
Look from This Object	3D вид объекта
Calculate Route Time	Время на маршруте
Delete all output links	Удалить все исходящие связи

рис.79

Некоторые пояснения (основано на тестах и наблюдениях, официальных подтверждений не имеется, достоверность не гарантирована):

«**OnMovedTo Event**» — установлено, что связь срабатывает, когда объект проходит контрольную точку, прочие свойства (если есть) неизвестны.

«**OnPlaneBingoMainMG Event**» — установлено, что связь срабатывает, когда закончится весь бк — и пушки, и пулемёты.

«**OnPlaneBingoFuel Event**» — остаток топлива, при котором работает связь, неизвестен.

Различия между «**OnPlaneCrashed**», «**OnPlaneDestroyed**» и «**OnKilled**» сообщениями видятся такими: в первом случае — аварийная посадка с не критическими повреждениями (например, подлом стойки шасси), во втором — разрушение планёра (таран, земля) и в третьем — собственно, «килл», «фраг» или «сбитый», т. е. пропал маркер самолёта и самолёт ушёл кому-то в статистику.

Составители: **ККМ** полка **-DED-**

Вычитка и замечания: **Vachik**

Заглавная картинка: **МК.Mr.X**

2015 г.

