Как забабахать SP миссию в редакторе ARMA.

На примере миссии «Первая кровь».

Автор – sansey.

Под редакцией: SniperGRU, Wass.

ОТК 2009.

Введение.

Итак, вы поиграли в миссии и кампании, официальные и не совсем и где-то в глубине души у вас зародилось желание перейти на следующий уровень «козлищще – поклонничества» – попробовать свои силы в редакторе. Знаю, знаю, как это бывает – хочется сразу кампанию миссий на пятьдесят, не меньше, и чтобы за наших, и чтобы всех победили. Только одна беда – о чем конкретно делать мега-кампанию пока неясно, но это неважно, главное начать, тем более сначала надо разобраться в триггерах, вэйпоинтах и вообще в запредельном – в скриптах, а дальше, как кривая «офпшная» вывезет.

В этом пособии я не собираюсь подробно рассказывать ни про триггеры, ни про скрипты – об этом много и хорошо писали до меня люди, гораздо лучше разбирающиеся в предмете (см. справочник Ринзы, туториал по скриптописанию от Санька и т.д.). Я собираюсь рассказать, как лично сам делаю миссии, попутно предоставив начинающим мапмейкерам (или картоделам) некий «шаблон», по которому они тут же могли бы перейти к творчеству. Ибо, по моему убеждению, освоить редактор и научиться делать миссии – немного разные вещи, которые многие очень часто путают. Естественно, для создания миссий нужна минимальная база знаний о редакторе и встроенном скриптовом языке. Но не более. Остальное понадобится потом, а когда понадобится, то изучить получится гораздо проще и быстрей.

Хочу оговориться, что не претендую на истину в последней инстанции – все ниже изложенное, это мой личный опыт и, если можно так выразиться, мапмейкерский стиль, приобретенный во время создания и участия в создании миссий и кампаний в ОФП и Арме, за то время как я нахожусь в ОТК и до того. В этом руководстве представлены некоторые типовые способы решения вопросов, возникающих при создании обычной незатейливой миссии. Конечно, всегда можно сделать лучше, или просто по-другому. Использовать эти способы или нет – решать вам.

1. Делаем миссию.

Забудем пока о проекте кампании, которая должна порвать «комьюнити» и показать всем, как надо. Будем делать самостоятельную незамысловатую «сингловую» миссию. О кампании будет особый разговор в соответствующей главе. Ниже приведена последовательность нехитрых действий для создания миссии. То есть:

1) Создаем миссию.
2) Создаем вступительный и заключительный ролики
3) Пишем брифинги, делаем овервью, разбираемся с озвучкой и прочее.

Всего и делов то ☺. Осталось начать да кончить.

Почему же последовательность именно такая? Почему сначала, скажем, не сделать вступительный ролик? Потому, что мы создаем сингловую (миссию для одного игрока) миссию – то ради чего все делается. Сама миссия – то самое «мясо», большая и главная часть всей работы, то во что игроки будут играть. Остальное – это красивая обертка, которая для миссии, безусловно, неоспоримый плюс, но сама по себе несет небольшую смысловую и «геймплейную» нагрузку. Кроме того, нужно оставлять себе «пути к отступлению» – по ходу работы над миссией что-то может поменяться, и, если изменения будут расходиться со сделанным ранее роликом, то уже придется как-то выкручиваться по факту или отправлять что-то в корзину. Брифинг, овервью и озвучка делаются последними. С озвучкой все понятно – в процессе создания и тестирования текст может меняться, так зачем наговаривать по нескольку раз то, что после изменений в миссии пойдет в утиль. С брифингом и овервью все чуть-чуть хитрее. Естественно, когда определен сюжет миссии, уже должны быть какие-то наметки, но по тем же причинам, что с роликами и озвучкой финальная версия делается в самом конце. Еще пару слов о тестировании. Разговор о нем пойдет в отдельной главе, но, рекомендую, чтобы миссия тестировалась по завершению каждого из трех этапов кем-то кроме картодела. Итак, с порядком действий определились, теперь разберемся, о чем же будет наша миссия.

В архиве с миссией лежит папка bolvanka. В ней есть полезные заготовки для будущей миссии:

1) init.sqs – наиглавнейший и незаменимейший скрипт. Вся его прелесть в том, что это зарезервированное движком игры имя скрипта и, поэтому, он запускается при старте миссии. Здесь удобно задавать группам позывные, указывать разного рода переменные и т.д. и т.п. Вообще все то, что можно прописывать в строке инициализации юнитов, но гораздо проще собрать в один файл (и гораздо легче искать ошибки в одном месте, а не где попадя тому кто идет следом – тестирует).

2) *description.ext* – файл, в котором описываются «идентити» для персонажей, все сторонние звуки и музыка, добавляются стволы и патроны в «оружейку», чтобы их можно было выбрать в брифинге перед миссией. В прилагаемом файле описаны все наиболее часто используемые классы, так что его удобно использовать в качестве болванки для своей миссии. Подробное описание «дескрипшина» можно посмотреть, например, все в том же руководстве Ринзы.

Примечание: в данном дескрипшене при описание классов использована функция define, позволяющая резко сократить количество писанины (привет от СИ++).

3) *stringtable.csv* – файл, в котором описывается весь текст в миссии. То есть абсолютно весь текст. Дурной тон, если у вас в миссии будет текст, не описанный в стрингтейбле (и если Вы мечтаете о переводе на другие возможные языки…). Кроме того, текст, собранный в одном файле, удобно редактировать – не надо разыскивать какой-нибудь потерявшийся в триггерах хинт.

4) *camer.sqs* – заготовка для ролика, неважно интро это, оутро или небольшой ролик, где-то в середине миссии.

5) *overview.html* – заготовка для оверьвьюшки (нараспев: просто добавь джипееееег…)

6) *briefing.html* – соответственно заготовка для брифинга.

1.1. О чем миссия? Сюжет.

«Дык», о чем? Глядя на поделки энтузиастов глаза разбегаются от всевозможных сюжетных решений. Но есть один небольшой секрет – если сюжеты немного упростить, то их легко структурировать и привести к нескольким шаблонам, из которых затем «выращивать» свою собственную миссию. Если вспомнить официальную кампанию ОФП «Cold war crysis 1985» там было четыре главных героя – пехотинец, танкист, летчик и диверсант. К «пехотно-диверсантным» вариантам отнесем все возможные миссии за партизан, гражданских и т.д. – и вот они первые грубые лекала для будущего шедевра.

Дальше больше – можно вывести основные шаблоны миссий для этих четырех героев и собрать их в небольшую таблицу:

<table>
<tr><th>Пехотинец</th><th>Диверсант</th><th>Танкист</th><th>Летчик</th></tr>
<tr><td>Зачистка: захват какой-либо позиции.</td><td>Диверсия: подрыв танков, складов и т.д.</td><td rowspan="5">Танковая миссия ☺ (см. пехотинца)</td><td>Атака (конвоев, баз и т.д.).</td></tr>
<tr><td>Оборона: удержание какой-либо позиции.</td><td rowspan="4">Убийство: охота за vip, снайперская миссия.</td><td rowspan="4">Перевозка десанта</td></tr>
<tr><td>Засада.</td></tr>
<tr><td>«Дорожная миссия»</td></tr>
<tr><td>Патрульная миссия</td></tr>
</table>

Теперь по порядку о каждом из шаблонов.

Зачистка и оборона – классика жанра, тут и пояснять вряд ли что-либо надо. Тем не менее, хотя это и самые банальные из возможных сюжетов они всегда оставались и останутся самыми востребованными. Ведь что такое война – сплошная череда наступления и обороны. При выборе такого сюжета главное как-то создать игроку мотивацию для отстрела врагов и не давать почувствовать себя одиноким «рэмбо» на поле боя, остальное за вас сделает игра.

Засада. В, основном, среди картоделов практикуются засады на конвои. Причем, если игрок, грубо говоря, лежит в кустах с пулеметом – это один тип миссии, а если он же едет в обреченном конвое – совершенно другой. Однако, с точки зрения создания миссии, засада делается по одному и тому же принципу. И, конечно, не стоит зацикливаться только на конвоях. Например, засада в лесу против пехоты. В редакторе – то же самое, по геймплею совершенно другое.

«Дорожная миссия». Классический пример – миссия «После Монтиньяка» из того же «Колд вара». Герою необходимо выбраться из окружения, добраться до точки эвакуации. По сути «дорожная миссия» - незамысловатая история о путешествии из точки А в Б. Тем не менее, многие считают тот же «После Монтиньяка» одной из самых атмосферных миссий в кампании.

Патрульная миссия. Название говорит само за себя – патрулируем, участвуем в стычках с противником. Пример такой миссии можно найти все в том же «Колд варе»

Миссии за диверсанта с точки зрения редактора довольно простые. Здесь не нужно создавать видимость боя, ведь диверсанты работают малыми группами и их главное оружие – скрытность. Необходимо всего лишь создать вражеские патрули, некую видимость шевеления на вражеских базах и самое главное – реакцию на действия игрока (кстати говоря, момент про реакцию на игрока технически очень схож с реализацией засады). Несмотря на, казалось бы, простоту таких миссий, среди них встречаются и такие шедевры как небезызвестный «Красный код» от МОНАХа.

На заметку: всё зависит от глубины проникновения в сюжет, иначе степени его детальной проработки.

Танковые миссии. За шаблоны к ним сгодятся пехотные. В основном же больше всего распространенны миссии типа «Зачистка» или лучше сказать «Атака» - прорыв танками

обороны противника, захват контрольных точек. Ну, так, собственно, для этого танки и предназначены. Но бывает по всякому, все зависит только от фантазии картодела.

Летные миссии. Чаще всего в итоге сводятся к «слетать туда и всех убить», но также как и с танками бывает по всякому.

Итак – вот они, основы. Выбрать вариант технической реализации миссии уже гораздо проще. На меня, например, при создании миссии «смотрел» пехотный вариант «Зачистки». Как уже упоминал выше, «Зачистка» и «Оборона» – самая соль ОФП, те миссии, за которые мы его любим, и которые дают почувствовать себя винтиком огромной боевой системы. Остров для миссии, конечно, будет наша уже до тошноты любимая Сахрань. Южная Сахрань. Остров поменьше – тормозов поменьше.

Когда выбран шаблон, самое время включать творческие участки мозга и выбрать, что же будем оборонять от кого, зачем и почему. На этом этапе начинает закладываться так называемая «атмосферность» миссии. То насколько вы заставите игрока поверить в вашу историю при помощи нескольких строчек в брифинге и «пары» реплик персонажей.

Также неплохо на этом этапе определиться, будет игрок командовать своим отрядом, или находится в подчинении. Если игрок командует, то это несколько упрощает задачу – пропадает необходимость сражаться с возможной тупостью бота начальника. Мой главный герой – рядовой. Всю миссию он следует приказам начальства. Однако, начальство может погибнуть, поэтому в миссию заложена возможность и такого финала.

1.2. Как воплощать сюжет? Структура миссии.

Более-менее определившись с сюжетом, время подумать о том, как воплощать его в жизнь. Стоит отметить, что в данном случае слово «сюжет» я использую условно. Конечно же, это еще далеко не сюжет, скорее пока только идея, грубая болванка, из которой предстоит выточить небольшую историю и, хотя бы, десять пятнадцать минут геймплея. Тут в игру вступает структура миссии, которая помогает не «высасывать из пальца» подробности сюжета, но «выяснять» их.

Структура миссии незамысловата – в миссии есть начало, середина и конец ☺.

Начало миссии – некое вступление, знакомство с персонажами и постановка для этих персонажей целей. В определенных условиях начала может и не быть – например, можно ограничиться описанием в брифинге и сразу окунуть игрока в гущу событий, но так или иначе в роли начала тогда выступит середина миссии.

Середина миссии – основная и самая главная часть. Та часть, где проходят бои, и сотни ботов кладут свои жизни на алтарь развлечения игрока. Та часть, на которую уходит большинство вэйпоинтов, триггеров и скриптов. Ну и, конечно, та часть, где находят больше всего глюков.

Концовка миссии. Само собой, что миссию некрасиво будет просто взять и завершить посередине. Конец должен быть понятным и логичным. Сюда относятся скрипты и триггеры, отвечающие за выполнение заданий и прочие вещи, необходимые для удачного по исполнению финала миссии.

Теперь разберемся на примере «Первой крови», что там начало, а что конец.

Начало. Конвой морпехов следует до блокпоста перед базой, там они спешиваются и выдвигаются на позиции для атаки.

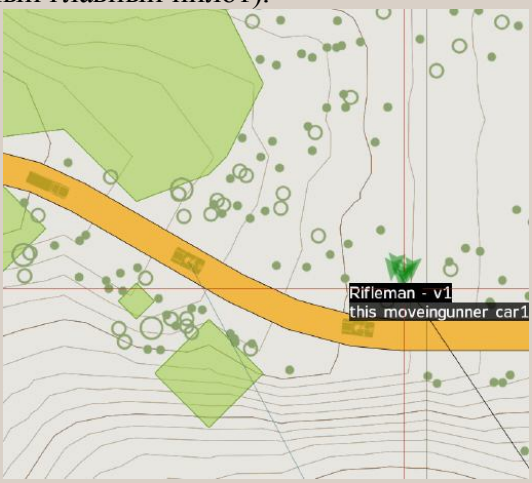
Середина миссии. Собственно, атака на базу. Бой.

Концовка. Контратака сахранцев, и один из двух финалов – либо игрок остался один из отряда, либо нет.

1.3. Делаем начало.

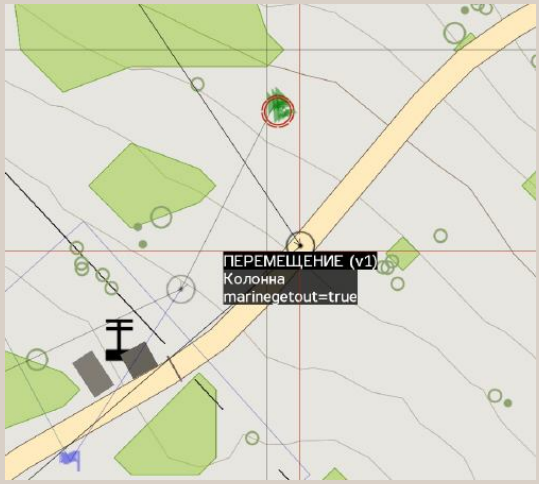
Важно! Если нет необходимости, всем объектам в миссии нужно задавать положение «нет», вместо стандартного положения «в строю». Это влияет на производительность миссии.

Итак, первым делом требуется организовать колонну из двух хаммеров и двух грузовиков. Первое правило колонны в Арме – в колонне должен находиться транспорт с одинаковой скоростью движения. То есть либо гусеничный, либо колесный. Бывают исключения, но такие случаи нужно рассматривать отдельно и большую и более-менее адекватно движущуюся колонну из разношерстного сброда сделать довольно трудно. Второе правило – в колонне должна быть твердая вертикаль власти. То есть все водители колонны должны находиться в одной группе, где командир в самой первой машине и дальше по убыванию старшинства. Следует помнить, что во всех наземных транспортных средствах самый главный командир, потом стрелок и самый младший водитель (в самолетах и вертолетах самый главный пилот).



Поэтому моя водительская группа из шести человек (имена v1 – v6) распределена следующим образом: командир v1- стрелок первого хаммера, v2 его водитель, v3 стрелок второго хаммера и т. д. Во избежание путаницы всем водителям в группе заданны звания, для того чтобы быть наверняка уверенным в их правильном месте в строю. То есть v1 – майор и дальше по убыванию до v6 рядового. Имя этой группе задано в init.sqs в пятой строчке – ham (подразумевается хаммер ☺, удобно давать переменным «говорящие» хотя бы лично для вас имена, самое главное избегать зарезервированных переменных, таких, как, например, fire). Дальше, для группы ham создается вэйпоинт (главное не забыть для колонны поставить построение «колонна») до места назначения. Всё, колонна готова и теперь уверенно доберется до своей точки назначения, сшибив по дороге всего пару дорожных знаков.

В поле «по активации» вэйпоинта группы ham написано marinegetout=true



marinegetout – это глобальная (доступная другим скриптам, триггерам и вейпоинтам) переменная, которую я ввел в игру, просто прописав этот набор букв в вэйпоинте. Для удобства, можно все глобальные переменные описать в init.sqs, но прелесть скриптового языка Армы в том, что переменные не надо нигде описывать заранее. Они создаются сразу же, как только с ними начинаешь производить какие-либо операции, как в данном случае переменной присвоено значение true. Что все это значит? Это значит, что как только колонна доберется до места, переменная marinegetout послужит сигналом, для высадки морпехов, но об этом чуть позже.

На заметку: следует понимать, что глобальным переменным кроме значений true/false (верно/ложно) можно присваивать численные значения.

Три группы морпехов рассадить по машинам при помощи всем известных скриптовых команд, прописанных в инициализации командиров, не составит никакого труда даже новичку. Команда игрока посажена в два хаммера, имя группы задано в поле инициализации командира – plg=group this. Две группы, которые сидят в грузовиках – mor1g и mor2g.

Теперь посмотрим на самый главный скрипт – init.sqs.

Первые четыре строчки задают группам радиопозывные, чтобы в миссии сражались не альфы и бравы, а шакал 1 и шакал 2. То есть в stringtable.csv в соответствующем месте пишется свое название группы, а потом это название уже в init.sqs присваивается командиру требуемой группы.

Далее, с 10 по 16 строчки – затемнение экрана, выключение звука и рации, пауза в 10 секунд, выход из затемнения и включение звука и рации. Это пауза в начале миссии, чтобы игрок не заметил, как колонна вначале раскачивается и трогается с места.

Дальше идет небольшой диалог в машине. Особенность его реализации состоит в том, что юниты только открывают рот, а звук проигрывается командой playsound:

playsound "mi1"

of say "car1"

(это связанно с тем, что обычный разговор в машине ни черта не слышно, естественно такая реализация подходит только для SP, так как playsound слышно в любой точке острова).

Хороший тон в миссии – перед любой репликой (а особенно перед радиорепликой) принудительно ставить нормальное время – setAccTime 1.0

И вот, после диалога, мы, наконец, добрались до магической 34ой строчки:

@marinegetout

Которая означает паузу в скрипте, пока не будет выполнено условие *marinegetout.* То есть скрипт приостанавливается, пока колонна не доберется до своего «вэйпа», и это условие не выполнится. А как только оно выполнится – разгрузка:

{unassignvehicle _x} foreach units plg

То есть вот так, при помощи одной переменной и пары строчек в скрипте стала ненужной возня с несколькими вэйпоинтами разгрузки и их синхронизацией, для того чтобы высадить морпехов из машин.

Небольшое отступление про вэйпоинты.

Особенности вэйпоинтов в Арме таковы, что когда их хотя бы десяток на миссию, то боты начинают жутко тупить при переходе от одного вэйпоинта к другому, иногда бывает они останавливаются секунд на 30-40 прежде чем перейти к следующей маршрутной точке. Кроме того, обилие вэйпоинтов сказывается на производительности миссии. Спасает положение динамическое создание и/или телепортация уже существующих вэйпоинтов (СМ. ПРИМЕР НИЖЕ). При таком способе снижается нагрузка на миссию, кроме того, боты без малейшего промедления реагируют на новую маршрутную точку.

Итак, выгружены все морпехи, кроме экипажей хаммеров. Водители грузовиков, тоже выгружены и присоединены к группе игрока:

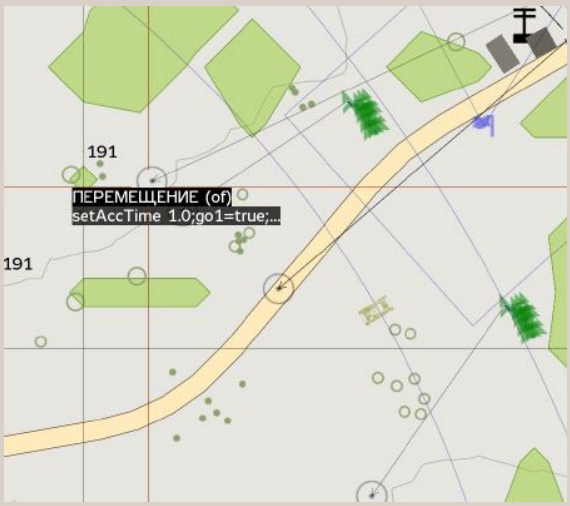
[v5,v6] join plg

unassignvehicle v5

unassignvehicle v6

Освободившиеся от оков транспорта, группы начинают движение каждая к своему, заранее поставленному, вэйпоинту. Чтобы избежать несчастных случаев в этой толчее, пришлось добавить группе игрока лишний вэйпоинт, синхронизированный с триггером. Вся эта канитель, только для того, чтобы группа игрока подождала у обочины, пока хаммеры проедут за ограду, но это – особенность данной миссии, при другом рельефе местности можно было бы этот огород не городить.

Теперь взглянем на последние вэйпоинты всех четырех групп.



В поле «по активации» у всех четырех что-то вроде:

setAccTime 1.0;go1=true;leader plg sideradio "rad1"

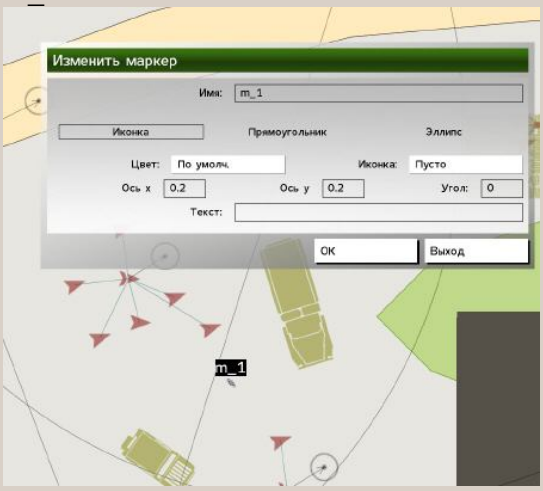
То есть, установка нормального времени, радио-доклад командира группы и присвоение соответствующей переменной верного (*true)* значения.

Теперь смотрим init.sqs строчки 48-51 – в скрипте пауза, до тех пор, пока все 4 группы не займут свои позиции и не доложат по рации. Хочу обратить внимание, что построение, установленное группам «колонна в шахматном порядке» не случайно – так они быстрее протискиваются через узкий проход в ограде. Теперь, когда все готовы к атаке, в миссии установлено автосохранение (строка 54). На этом *начало* миссии завершено, переходим непосредственно к бою.

1.4. Делаем середину.

Первое, что нужно сделать – послать морпехов в атаку на вражеские позиции. Для этого прекрасно подойдут динамические вэйпоинты. Смотрим init.sqs строчки с 66 по 74:

plg addWaypoint [getmarkerpos "m_1",0] – группе добавляется вэйпоинт, координатами служит невидимый маркер m_1



(очень удобно для таких целей использовать невидимые маркеры – они не загромождают миссию, как, например, геймлоджики)

[plg,3] setWaypointType "SAD" – задается тип вэйпоинта, в данном случае «найти и уничтожить». Цифра 3 – порядковый номер вэйпа, нумерация начинается с 1, то есть если посмотреть внимательней, у этой группы в миссии уже было два вэйпоинта, а создается третий.

[plg, 3] setWaypointFormation "LINE" – задается построение «строй» – такое построение гораздо более удобно для атаки чем «колонна в шахматном порядке».

Ну и немного погодя, когда завяжется бой, в строчках 93-95 задается боевое поведение

[plg, 3] setWaypointBehaviour "COMBAT"

Тогда же стартует группа хаммеров, если хаммеры послать одновременно с пехотой, то они, естественно, пехоту обгонят и будут сразу сожжены пулеметным огнем.

На этом управление союзническими силами в миссии завершено. Дальше до самого конца миссии руководить ими будет только искусственный интеллект.

Теперь посмотрим на обороняющихся:

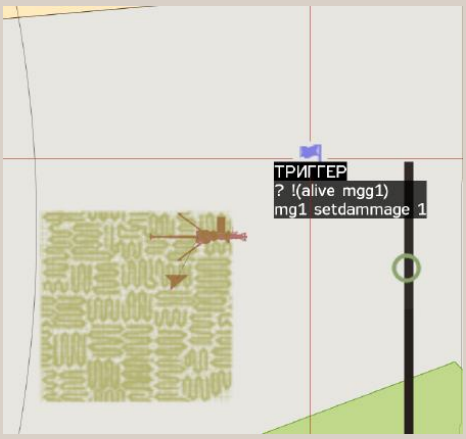
Имеются несколько групп с вэйпоинтами «защищать», управление ими ограничивается тем, что, в определенный момент, солдатам, у которых было поведение «безопасно», задается боевое поведение:

{_x setbehaviour "aware"} foreach units pr2g

Плюс самый дальний отряд pr5g, находящийся в резерве, в разгар боя бросается в атаку при помощи телепортации его вэйпоинта:

[pr5g,1] setwppos getmarkerpos "m"

Осталось только разобраться с пулеметами. В Арме существует неприятная особенность – даже когда стрелок станкового пулемета убит, боты зачастую продолжают стрелять по этому пулемету, до его полного уничтожения с пулеметом около ограды эта проблема решается одним триггером:



То есть ДШК задается имя mg1, в инициализации ему прописывается mgg1=gunner this (тем самым стрелку задается имя mgg1), ну а теперь триггер по условию смерти стрелка «убивает» и сам пулемет.

С пулеметом находящимся на крыше все организовано немного хитрее.

Как только начинается бой из init.sqs 91 строчка запускается скрипт [] exec "mg.sqs".

Этот скрипт заставляет солдат на крыше по очереди занимать место за пулеметом. В принципе, это можно было решить и проще – просто объединить несколько солдат в группу и одного из них посадить за пулемет, тогда командир сам назначал бы нового стрелка на место убитого, но в данной миссии сделано вот так.

Несколько слов о дополнительной озвучке. В миссии присутствуют скрипты отвечающие за крики раненых:

gkrik.sqs для сахранцев,

akrik.sqs для морпехов.

Принцип реализации такой – каждому «кричащему» солдату прикрепляется так называемый обработчик событий, «эвентхэндлер» на событие «хит», то есть ранение:

{_x addEventHandler ["hit",{_this exec "akrik.sqs"}]} foreach units plg

Теперь, как только солдата ранят, запускается соответствующий скрипт, который случайно выбирает один из восьми доступных криков.

Подробнее об обработчиках событий: http://redhammer.h14.ru/EH.rar

Кроме того, в миссии есть функции, отвечающие за боевые выкрики:

gor.sqf для сахранцев,

aor.sqf для морпехов.

В качестве исходных данных для них используется имя «орущей» группы, например:

[pr5g] execVM "gor.sqf";

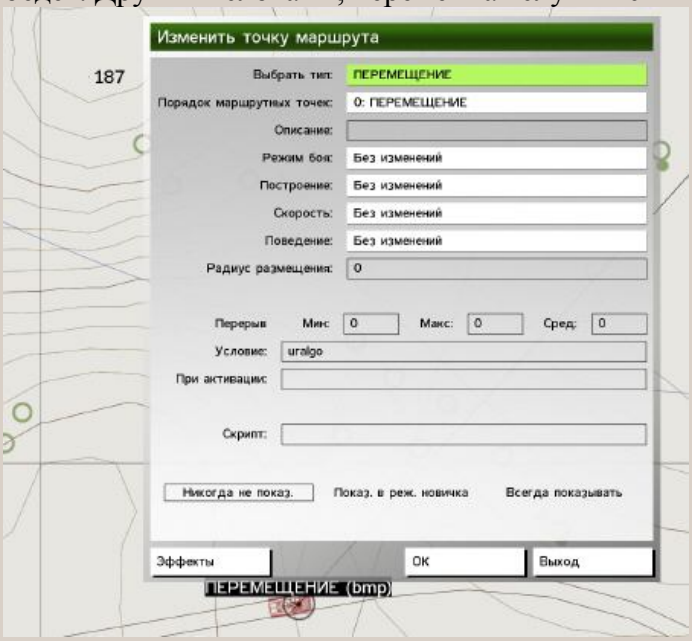
1.5. Делаем концовку.

Итак, конец миссии – контратака врагов, и выбор одного из двух финалов.

Контратака состоит из БМП с десантом и двух УРАЛов с пехотой.

Все перемещения БМП можно было сделать при помощи динамических вэйпоинтов или вообще только скриптом. Но, так как миссия не особенно загружена, зачем усложнять себе жизнь? К концу миссии уже видно, сколько вэйпоинтов всего, поэтому в данной ситуации не жалко трех вэйпоинтов на БМП плюс один вэйпоинт для десанта.

1ый вэйпоинт БМП – в условии стоит «uralgo», то есть пока эта переменная не станет true дальше БМП не поедет. Другими словами, переменная служит сигналом к контратаке.



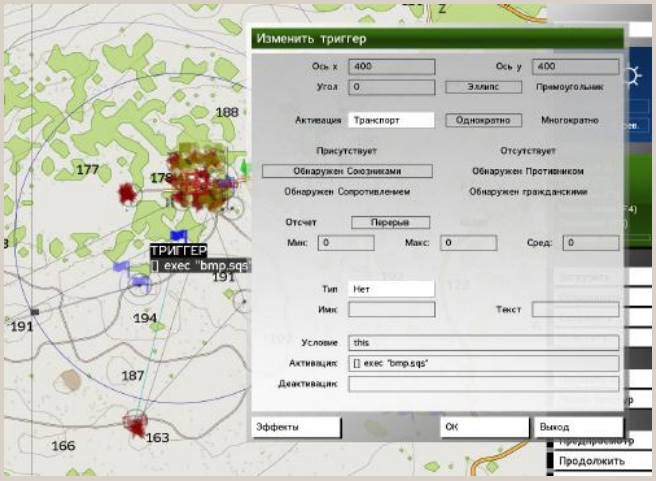
2ой вэйпоинт – выгрузка десанта, в поле «по активации»:

{unassignvehicle _x} foreach units bmpg;

То есть, выгружается группа десанта, у которой было задано имя bmpg

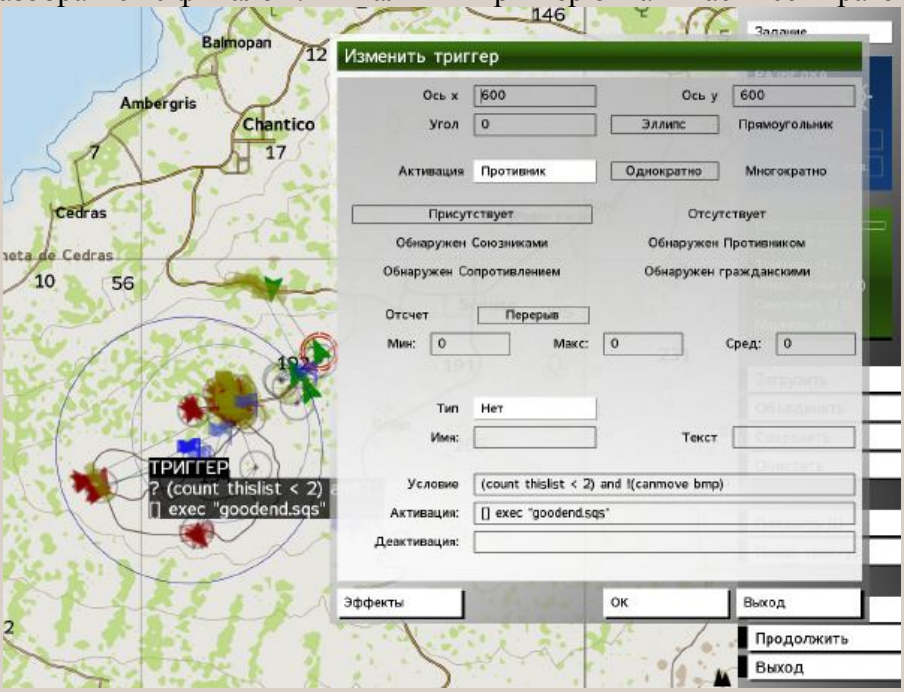
И 3ий вэйпоинт – «найти и уничтожить»

Кроме того, к БМП присоединен триггер, как только БМП попадает в его зону действия, запускается скрипт bmp.sqs, в котором командир группы игрока по радиосвязи предупреждает об опасности:



Колонна УРАЛов организована подобным образом.

Осталось разобраться с финалом. Финальный триггер охватывает всех врагов в миссии:



В активации стоит «Противник». Условие «присутствует» или «отсутствует» в данном случае неважно. Для того, чтобы избежать «синдрома последнего солдата» в строке условие написано следующее:

(count thislist < 2) and !(canmove bmp)

Это значит, что как только противников станет меньше двух, (то есть останется один враг) и БМП будет обездвижено – будет запущен скрипт goodend.sqs.

Для проверки уничтожения техники рекомендую использовать именно условие *canmove*, так как технику необязательно могут уничтожить, но как только техника обездвижена, ее покидает экипаж, и она может считаться нейтрализованной. Но, конечно, если задача миссии именно уничтожить технику – то условие должно быть соответствующим.

В скрипте goodend.sqs идет проверка, остался ли игрок один в группе и в зависимости от этого активируется соответствующий финальный триггер.

2. Делаем ролики.

Несколько слов о роликах. Большой разницы в том, интро это, оутро, или зарисовка в середине миссии, для создания ролика нет. Но, ***если ролик делается в миссии, всегда в самом начале задавайте нормальное время командой setAccTime 1.0***

Посмотрим на заготовку для ролика camer.sqs в папке bolvanka.

Начало ролика.

titleCut ["","Black faded"] – затемнение в начале ролика.

0 fadesound 0 – выключение звука

showcinemaborder false – отключение черных кинематографических полос вверху и внизу экрана (если хотите).

_camera = "camera" camcreate [0,0,0]
_camera cameraeffect ["internal", "back"] – создание камеры.

_camera camPrepareFocus [-1,-1] – отключение автофокусировки камеры (чтобы не было замыленности изображения)

далее вставляем первое положение камеры, с которого начнется ролик, выжидаем необходимую паузу для того чтобы все прогрузилось и начинаем киносеанс:

titleCut ["","BLACK IN"]; titleFadeOut 1
1 fadesound 1

Конец ролика.

titleCut ["","BLACK OUT"]; titleFadeOut 2
2 fademusic 0
2 fadesound 0
~2.1
Затемнение, постепенное затухание музыки и звука, и необходимая для этого пауза.

_camera cameraeffect ["terminate", "Back"]
camdestroy _camera
Удаление камеры.

end=true – условие на срабатывание финального триггера (не забыть, в интро/оутро поставить финальный триггер с условием end).

Начинающие кинематографисты обычно совершают несколько типовых ошибок при работе с камерой.

Во-первых, очень любят перелетать камерой с места на место.

Камера – это всего лишь инструмент, а не сам ролик, не надо злоупотреблять перелетами и скачками планов. Обычно в ролике большинство планов или статические, или камера находится на месте и только поворачивается. Конечно, где необходимо можно и нужно делать панорамные пролеты, но повторюсь, не стоит этим злоупотреблять.

Во-вторых, во время диалогов режет глаза, когда камера попеременно поворачивается то к одному собеседнику, то к другому. Не надо недооценивать статические ракурсы! Для примера посмотрите на операторскую работу во время диалога в каком-нибудь кинофильме. Дурной тон если намертво прикреплять направление камеры к пешему юниту – при любом шевелении юнита камера будет дергаться, что плохо отразиться на впечатлении от ролика.

В роликах обычно самые продолжительные и распространенные статические сцены – диалоги. Чтобы не показывать в течение нескольких минут просто говорящие головы давным-давно придуман и успешно используется как в кино, так и на телевидении операторский прием с чередованием планов. Есть четыре плана - крупный, общий, отвлеченный и детальный. На примере выглядит так: стоит солдат около костра, и долго рассказывает другому солдату о недавнем бое. Сначала идет крупный план на его лицо, затем общий план – видно большую часть его тела, отвлеченный план – показывается его слушатель сидящий у костра, детальный план – показываются дрожащие пальцы рассказчика. Далее случайное чередование таких планов позволяет «оживить» статичную картинку подобного монолога. Следует понимать, что если с крупным и общим планом все ясно, то отвлеченный и детальный планы не обязательно должны быть одними и теме же. Если же говорить не о монологе, а о разговоре двух и более персонажей, то тут открыто обширное поле для фантазии и экспериментов оператора. Как правило, в роликах сначала берут план на одного, а потом на другого персонажа. Можно же, самое простое, при смене планов менять углы камеры, или задерживать крупный план на одном персонаже, чтобы отразить его реакцию на реплику другого, или акцентировать детальный план на жесте какого-либо собеседника.

Неотъемлемая часть роликов – анимации. Тут вроде все понятно – выбирается подходящая анимация и проигрывается в нужный момент. Вот только во многих роликах бывает, что анимации проигрываются в ненужный момент, или у бота во время разговора что-то переклинивает, и он начинает мотать головой в разные стороны. Чтобы такого избежать, достаточно использовать команду disableAI “anim” – бот превращается в замершую статую, и будет проигрывать только те анимации, которые от него потребуются.

3. Остальная мишура.

Для придания миссии окончательного, завершенного вида необходимо добавить брифинг, овервью и свои звуки.

Начнем с овервью. В папке bolvanka лежит образец овервью для квадратной картинки размером 512 на 512, это может быть картинка формата jpg или paa.

Если вы хотите использовать прямоугольную картинку, то следует изменить размеры в овервью с <img src="PicName.jpg" width="170" height="170"> на нечто вроде

<img src="PicName.jpg" width="155" height="77">

для того, чтобы отображение на мониторе приобрело более приемлемое соотношение вертикали и горизонтали.

В брифинге самое сложное, это сделать гиперссылки на маркеры. А делается это очень просто: текст, где текст – это текст гиперссылки, а markername – имя маркера в миссии. При написании брифинга полезным будет тэг
, обозначающий перенос на новую строку.

Несколько слов про стрингтейбл. Если вам необходимо в тексте написать кавычки то следует помнить, что кавычки в кавычках равным двойным кавычкам. Например, нужно задать название миссии Операция «Епономать». В стрингтэйбле это выглядит так:

STR_On_M, "Операция ""Епономать"""

Кроме того следует помнить о символе \n который служит для переноса текста на новую строку.

Дескрипшен. При работе с дескрипшеном самое главное это «правило холодильника», которое гласит – если открыл холодильник, так закрой его! То есть если где-либо есть открытая скобка, она обязательно должна быть закрыта.

Описание всех классов (музыка, звуки и т.д.) идет независимо друг от друга и их можно свободно переставлять местами. Основной принцип структуры такой – пишется название класса, потом скобка открывается пишется все что относится к классу и скобка закрывается. Например:

class CfgMusic {описание всех треков музыки}

Не стоит смущаться того, что для удобства редактирования классы не описываются в одну строчку и выглядит это обычно похожим на следующее:

class CfgMusic
{
описание
всех
треков
музыки
}

Отдельно хочется отметить классы Weapons и Magazines, описание которых есть в примере в папке bolvanka. При помощи этих классов добавляется оружие и боеприпасы в арсенал, из которого можно выбрать подходящее снаряжение в брифинге перед началом миссии.

4. Полезные ссылки.

1) Начальное руководство от Rinza http://flashpoint.ru/forum/showthread.php?t=12948

2) Учебник по скриптописанию от Sanek http://otk-studio.ru/forum/download/file.php?id=419

3) Справочник мапмейкера. АрмА. Версия-2.0 http://otk-studio.ru/forum/viewtopic.php?f=2&t=180

4) Обработчики событий http://redhammer.h14.ru/EH.rar

5) Список всех скриптовых команд http://community.bistudio.com/wiki/Category:Scripting_Topics

6) Тема «Всё что нужно для карто- и аддоностроения» на flashpoint.ru http://flashpoint.ru/forum/showthread.php?t=43200

7) И, конечно же, http://otk-studio.ru