Создание квестов и диалогов

[править]

Создание базового диалога

Итак, приступим к созданию диалога, в котором нам будет выдаваться новый квест. Участвующие лица: Сидорович (С) и Меченый (М)

М: Здравствуй, Сидорыч! Есть работенка?

С: О, неужели заглянул к нам на огонек, Меченый? Ладно... ближе к телу... эээ, к делу. У меня к тебе следующая просьба: сбегай к Волку, у него что-то намечалось.

М: Без проблем! Уже бегу.

Будем считать, определились с фразами базового диалога. Теперь необходимо добавить его в соответствующем виде, чтобы игре он был «понятен».

Создание «скелета» диалога.

Для того, чтобы добавить наш диалог, необходимо открыть для редактирования файл gamedata\config\gameplay\dialogs\_escape.xml

Для тех, кто знаком с языком XML структура данного файла будет ясна с первого взгляда. Но, поподробнее об этом.

Для того, чтобы добавить в игру диалог предусмотрены так называемые тэги. В данном случае, тэгами являются такие конструкции:

1. <game\_dialogs> и закрывающий к нему - </game\_dialogs>

2. <dialog id = ""> закрывающий - </dialog>

3. <phrase\_list> - </phrase\_list>

4. <phrase id = ""> - </phrase>

5.

<text> </text>

6. <next> </next>

В принципе, этих конструкций уже достаточно для того, чтобы создать диалог. Но, диалог, созданный на основе этих конструкций, не сможет стать основой для нового квеста. Для того, чтобы во время диалога вам выдавался квест, есть соответствующий тэг. Но об этом – чуть позже.

Поясню, что обозначает каждый тэг в отдельности.

1. Определяет содержимое файла (в данном случае, говорит о том, что файл содержит диалоги);

2. Определяет новый диалог с id = "". Где содержимое кавычек есть идентификатор диалога. Идентификатор диалога задается в файлах типа character\_description\_\*\*\*.xml ;

3. Этот тэг говорит XML-парсеру игры о том, что далее следуют фразы диалога. Иными словами, между <phrase\_list> и </phrase\_list> содержатся фразы диалога;

4. Этот тэг определяет фразу с id = "". Внутри кавычек содержится идентификатор фразы (ВНИМАНИЕ: идентификатором в данном случае могут быть ТОЛЬКО цифры);

5. Внутри этих двух тэгов содержится ссылка на соответствующий данной фразе текст (пример, esc\_trader\_talk\_info\_123). Сам текст берется из файлов типа stable\_dialogs\_\*\*\*\*\*.xml, в котором содержатся все фразы в текстовом виде. О создании текста фразы поговорим позже;

6. Внутри этого тэга содержится ссылка (в числовом формате) на следующую (или следующие) фразы.

Теперь приступим к тэгам, позволяющим производить какие-то действия во время диалога.

<precondition> </precondition>. Представляет собой предусловие, в случае выполнения которого (то есть предусловие вернуло ИСТИНУ) данный диалог, или фраза будут появляться в списке доступных во время игры у NPC.

<action> </action>. Данный тэг позволяет производить какое-либо действие во время диалога (например, передачу предметов между общающимися... так же позволяет производить более сложные действия). Внутри тэга располагается ссылка на какую-либо функцию (например, dialogs. actor\_set\_dolg сделает игрока членом группировки Долг).

<give\_info> </give\_info>. Позволяет выдавать игроку во время диалога так называемые info\_portion'ы (кстати, инфопоршны можно выдавать так же и через <action>, только это потребует создания специальной функции). info\_portion – является основой для квестов, info\_portion’ы определяются в файлах типа info\*\*\*\*.xml. Инфопоршны могут, как начинать квест, так и заканчивать (завершать) его. Так же, с помощью инфопоршнов можно отслеживать действия игрока, для того, чтобы, например, вывести в нужное время диалог. <has\_info> </has\_info> и <dont\_has\_info> </dont\_has\_info> являются предусловиями для текущей фразы/диалога. Первый тэг проверяет наличие данного инфопоршна у игрока, второй – отсутствие.

Ну что, теперь мы можем создать диалог, в котором нам будет выдаваться квест. Но мы не создали самих info\_portion'оф, которые позволяет активировать квест.

[править]

Создание info\_portion'ов

Итак, открываем файл gamedata\config\gameplay\ info\_l01escape.xml. Что мы там видим? Правильно, все ту же структуру XML. В данном файле нас интересуют только два тэга:

<info\_portion id = ""> </info\_portion>

<task> </task>

Первый объявляет сам info\_portion. Второй – объявляет задание (идентификатор задания id содержится внутри тэга <task>)

Для примера, создадим 3 info\_portion, которые понадобятся нам для будущего квеста:

<info\_portion id = "new\_task\_started">

<task>new\_task</task>

</info\_portion>

Это инфопоршн с именем new\_task\_started. Он активирует квест с именем new\_task.

<info\_portion id = "player\_talked\_with\_wolf">

</info\_portion>

<info\_portion id = "player\_complete\_new\_task">

</info\_portion>

[править]

Создание нового квеста

Сами квесты содержаться в файлах типа tasks\_\*\*\*\*\*.xml

Откроем файл gamedata\config\gameplay\tasks\_escape.xml. Структура – все тот же XML.

Для примера исследуем уже существующий квест:

<game\_task id="esc\_help\_wounded\_from\_raid" prio="485">

<title>esc\_help\_wounded\_from\_raid</title>

<objective>

<text>esc\_help\_wounded\_from\_raid\_0</text>

<icon>ui\_iconsTotal\_esc\_help\_wounded\_from\_raid</icon>

<function\_complete>escape\_tasks.task\_fox\_complete</function\_complete>

<infoportion\_set\_complete>garbage\_meetstalker\_start</infoportion\_set\_complete>

<article>esc\_fox\_help</article>

</objective>

<objective>

<text>esc\_help\_wounded\_from\_raid\_1</text>

<map\_location\_type hint="esc\_fox">green\_location</map\_location\_type>

<object\_story\_id>Escape\_stalker\_from\_raid</object\_story\_id>

<infoportion\_complete>escape\_fox\_heal</infoportion\_complete>

<infoportion\_fail>esc\_dogs\_return</infoportion\_fail>

</objective>

<objective>

<text>esc\_help\_wounded\_from\_raid\_2</text>

<map\_location\_type hint="esc\_fox">green\_location</map\_location\_type>

<object\_story\_id>Escape\_stalker\_from\_raid</object\_story\_id>

<infoportion\_complete>escape\_stalker\_done</infoportion\_complete>

</objective>

</game\_task>

По кусочкам:

<game\_task id="esc\_help\_wounded\_from\_raid" prio="485">

</game\_task>

Задает имя (id) нового квеста. В данном случае, esc\_help\_wounded\_from\_raid. prio = "" задает приоритет задачи. Чем выше приоритет, тем больше вероятность того, что текущий маркер квеста будет переключен на ваше задание.

<title>esc\_help\_wounded\_from\_raid</title>

Задает заголовок квеста. Т. е. его название. Название можно прописать в виде текста, либо в виде ссылки. В данном случае используется ссылка на текст, который хранится в файле string\_table\_tasks\_escape.xml.

<function\_complete>escape\_tasks.task\_fox\_complete</function\_complete>

Вызывает функцию из файла escape\_tasks.script под именем task\_fox\_complete. При выполнении данной функции дается info\_portion, который завершает данную подзадачу, в случае, если все условия функции соблюдаются.

<infoportion\_complete>escape\_fox\_heal</infoportion\_complete>

Текущее подзадание будет завершено в случае, если игроку будет дан этот info\_portion.

<infoportion\_fail>esc\_dogs\_return</infoportion\_fail>

Текущее подзадание будет провалено в случае, если игроку будет дан этот info\_portion.

<map\_location\_type hint="esc\_fox">green\_location</map\_location\_type>

Создает указатель на карте с подсказкой esc\_fox и типом green\_location.

<object\_story\_id>Escape\_stalker\_from\_raid</object\_story\_id>

Указывает на sid из файла gamedata\config\game\_story\_ids.ltx

<infoportion\_set\_complete>garbage\_meetstalker\_start</infoportion\_set\_complete>

Автоматически устанавливает данную подзадачу в положение «Выполнено», если у игрока есть данный info\_portion. //!!!!!!Не уверен – не пользовался не разу.

<text>esc\_help\_wounded\_from\_raid\_0</text>

Содержит описание подзадачи. Может содержать внутри как текст, так и ссылку на текст.

Создадим задание, которое потребуется нам для диалога:

<game\_task id="new\_task">

<title>Поговорить с Волком</title>

<objective>

<text>Вернуться к Сидорычу</text>

<map\_location\_type hint="escape\_trader">blue\_location</map\_location\_type>

<object\_story\_id>Escape\_Trader</object\_story\_id>

<infoportion\_complete>player\_complete\_new\_task</ infoportion\_complete>

</objective>

<objective>

<text>Поговорить с Волком</text>

<icon>ui\_iconsTotal\_find\_item</icon>

<map\_location\_type hint="volk">green\_location</map\_location\_type>

<object\_story\_id>Escape\_novice\_lager\_volk</object\_story\_id>

<infoportion\_complete>player\_talked\_with\_wolf</ infoportion\_complete>

</objective>

<objective>

<text>Вернуться к Сидорычу</text>

<map\_location\_type hint="escape\_trader">blue\_location</map\_location\_type>

<object\_story\_id>Escape\_Trader</object\_story\_id>

<infoportion\_complete>player\_complete\_new\_task</ infoportion\_complete>

</objective>

</game\_task>

Здесь же необходимо обязательно указать ссылку на иконку из файла ui\_iconstotal, иначе при попытке заглянуть в раздел "Задания" в ПДА при взятии квеста будет происходить вылет. Для примера я проставил иконку поиска предмета.

Теперь создадим скелеты диалогов для нашего квеста. Потребуется три: один для Волка и два для Сидорыча.

<dialog id = "volk\_new\_quest">

<has\_info>new\_task\_started</has\_info>

<dont\_has\_info>player\_talked\_with\_wolf</dont\_has\_info>

<dont\_has\_info> player\_complete\_new\_task </dont\_has\_info>

<phrase\_list>

<phrase id = "0">

<text>esc\_volk\_new\_quest\_0</text>

<next>1</next>

</phrase>

<phrase id = "1">

<text>esc\_volk\_new\_quest\_1</text>

<give\_info>player\_talked\_with\_wolf</give\_info>

<next>2</next>

</phrase>

<phrase id = "2">

<text>esc\_volk\_new\_quest\_2</text>

<action>dialogs.break\_dialog</action>

</phrase>

</phrase\_list>

</dialog>

<dialog id = "esc\_trader\_new\_quest">

<dont\_has\_info>player\_complete\_new\_task</dont\_has\_info>

<dont\_has\_info>player\_talked\_with\_wolf</dont\_has\_info>

<phrase\_list>

<phrase id = "0">

<text>esc\_trader\_new\_quest\_0</text>

<next>1</next>

</phrase>

<phrase id = "1">

<text>esc\_trader\_new\_quest\_1</text>

<give\_info>new\_task\_started</give\_info>

<next>2</next>

</phrase>

<phrase id = "2">

<text>esc\_trader\_new\_quest\_2</text>

<action>dialogs.break\_dialog</action>

</phrase>

</phrase\_list>

</dialog>

<dialog id = "esc\_trader\_new\_quest\_complete">

<dont\_has\_info>player\_complete\_new\_task</dont\_has\_info>

<phrase\_list>

<phrase id = "0">

<text>esc\_trader\_new\_quest\_complete\_0</text>

<next>1</next>

</phrase>

<phrase id = "1">

<text>esc\_trader\_new\_quest\_complete\_1</text>

<give\_info>player\_complete\_new\_task</give\_info>

<next>2</next>

</phrase>

<phrase id = "2">

<text>esc\_trader\_new\_quest\_complete\_2</text>

<action>dialogs.break\_dialog</action>

</phrase>

</phrase\_list>

</dialog>

<code>

Все «скелеты» добавляем в файл dialogs\_escape.xml. Между <game\_dialogs> и <dialog id=...>!

''Внимательно следите за правильностью написания тэгов и ОДИНАКОВЫМ написанием идентификаций - как в файле с вашим новым диалогом, так и во всех других файлах, где используются тэги или ваши идентификации(например, идентификации фраз в файле с диалогом должны быть такими же, как и идентификации этих же фраз в файле с текстом). Иначе кроме серии вылетов с не всем понятными логами вы ничего не получите. У меня самого были проблемы при попытке активировать квест, выложеный в этой статье, так что пришлось кое-что поправить - и в тэгах, и в идентификаторах.''

Примечание: посмотрите на уже готовые диалоги, которые создали разработчики игры – это облегчит вам жизнь.

Теперь бежим в файл character\_desc\_escape.xml

Ищем там NPC с <tt>id = "escape\_trader"</tt> и добавляем два новых <tt><actor\_dialog></actor\_dialog></tt>:

<code>

<actor\_dialog>esc\_trader\_new\_quest\_complete </actor\_dialog>

<actor\_dialog>esc\_trader\_new\_quest </actor\_dialog>

Там же ищем NPC с id = "esc\_wolf" и добавляем один актор\_диалог:

<actor\_dialog> volk\_new\_quest </actor\_dialog>

Примечание: <actor\_dialog> является диалогом, который активируется самим игроком по желанию. А <start\_dialog> активируется только при выполнении определенных условий, заданных в нем, и обычно недоступен.

Осталось только добавить текст трех диалогов, чтобы наш квест стал полноценным. Открываем файл stable\_dialogs\_escape.xml и добавляем в него новый текст в таком виде:

<string id="esc\_trader\_new\_quest\_complete\_0">

<text>Все, я выполнил поручение Волка!</text>

</string>

<string id="esc\_trader\_new\_quest\_complete\_1">

<text>Молодец, Меченый. На тебя можно положиться.</text>

</string>

<string id="esc\_trader\_new\_quest\_complete\_2">

<text>Рад был помочь, Сидорыч.</text>

</string>

<string id="esc\_trader\_new\_quest\_0">

<text> Здравствуй, Сидорыч! Есть работенка?</text>

</string>

<string id="esc\_trader\_new\_quest\_1">

<text>О, неужели заглянул к нам на огонек, Меченый? Ладно… ближе к телу… эээ, к делу. У меня к тебе следующая

просьба: сбегай к Волку, у него что-то намечалось.</text>

</string>

<string id=" esc\_trader\_new\_quest\_2">

<text>Без проблем! Уже бегу.</text>

</string>

<string id="esc\_volk\_new\_quest\_0">

<text>Здорово, Волк. Меня тут Сидорыч послал, говорит у тебя работа есть.</text>

</string>

<string id="esc\_volk\_new\_quest\_1">

<text>Привет, Меченый. Верь ты ему больше, этому старому маразматику! Все спокойно сейчас. Так что иди к нему и

скажи, что все хорошо.</text>

</string>

<string id="esc\_volk\_new\_quest\_2">

<text>Все, ухожу.</text>

</string>

Все! Поздравляю! Вы только что написали новый квест.

Вступление

Перед изучением этого урока советую освоить находящийся несколько выше урок "Создание диалогов".

Небольшие правила, следуя которым, можно сберечь себе много нервов:

- всегда сохраняйте резервные копии модифицируемых файлов - старайтесь почаще тестировать внесенные изменения - так проще понять, в чем вы ошиблись (если активно работать пару часов без остановки, не разу не протестировав внесенные изменения, то потом будет очень сложно понять - почему же, черт возьми, игра вылетает при попытке поговорить с Сидоровичем?) - перед началом работы создайте новую папку gamedata, а свои моды спрячьте подальше: работать с "чистой" версией игры не в пример удобнее и проще; особенно противопоказан Repair Mod, который с новыми диалогами часто конфликтует. О том, как создать диалоги и квесты, исключающие возможность конфликта с другими подобными модами, см. часть 4.

[править]

Создание базового диалога

Начнем создание квеста с написания небольшого линейного диалога.

Открываем файл gamedata\config\gameplay\character\_desc\_escape.xml и проводим такие манипуляции:

- находим строку "<actor\_dialog>escape\_trader\_done\_blockpost\_box</actor\_dialog>" - добавляем после нее новую - "<actor\_dialog>escape\_trader\_oops</actor\_dialog>" - сохраняем изменения

Мы добавили к Диалогу с Сидоровичем новую ветку, вернее, только ссылку на неё. Сама внутренняя структура ветки хранится в файле gamedata\config\gameplay\dialogs\_escape.xml, который мы открываем и производим:

- находим в любом месте файла промежуток между </dialog> и <dialog id="..."> - засовываем туда такую конструкцию:

<dialog id="escape\_trader\_oops">

<precondition>escape\_dialog.trader\_alredy\_give\_job</precondition>

<has\_info>tutorial\_end</has\_info>

<phrase\_list>

<phrase id="0">

<text>escape\_trader\_oops\_0</text>

<next>1</next>

</phrase>

<phrase id="1">

<text>escape\_trader\_oops\_1</text>

<next>2</next>

</phrase>

<phrase id="2">

<text>escape\_trader\_oops\_2</text>

<next>1</next>

</phrase>

<phrase id="3">

<text>escape\_trader\_oops\_3</text>

</phrase>

</phrase\_list>

</dialog>

- сохраняем изменения

Теперь у нас прописана структура новой ветки диалога. Осталось сделать еще одну вещь - прописать строки, которые будут отображаться вместо "escape\_trader\_oops\_...".

Для этого откроем файл gamedata\config\text\rus\stable\_dialogs\_escape.xml и сделаем следующее:

- находим любой удобный промежуток между </string> и <string id="..."> - прописываем туда следующие строки:

<string id="escape\_trader\_oops\_0">

<text>Сидрыч, давай поговорим о... водке.</text>

</string>

<string id="escape\_trader\_oops\_1">

<text> водке? Да какая водка? Нет её у меня, кончилась. Вроде у Волка на чердаке была, что-то он

однажды об этом упоминал... Принеси, а?</text>

</string>

<string id="escape\_trader\_oops\_2">

<text>Да сбегаю я сейчас, сбегаю... алкоголик, блин.</text>

</string>

<string id="escape\_trader\_oops\_3">

<text>Беги, беги... сил уж нет ждать.</text>

</string>

- сохраняем изменения

Всё. Диалог готов. Можно грузить игру и смотреть, работает ли новая ветка. А теперь будем прикреплять к этому безобразию квест ""Принеси водки Сидоровичу".

[править]

Создание базового квеста

Активацию квеста в диалог прикрепить достаточно просто. Проводим с файлом gamedata\config\gameplay\ dialogs\_escape.xml следующие манипуляции:

- добавляем содержимое фразы <phrase id="2"> следующим:

<phrase id="2">

<text>escape\_trader\_oops\_2</text>

<give\_info>kvest\_vodka\_started</give\_info>

<next>3</next>

</phrase>

- сохраняем изменения

Теперь после фразы Меченого "Да сбегаю я сейчас, сбегаю..." Сидорович таки всучивает нам квест "Принеси водку". Но не спешите тестировать изменения. Мы добавили в диалог упоминание о квесте, хотя он еще не создан.

Исправим это недоразумение. Откройте файл gamedata\config\gameplay\tasks\_escape.xml и в начале файла создайте такую конструкцию:

<game\_task id="esc\_test\_vodka\_task">

<title>Принести водки</title> <!-- заголовок квеста -->

<objective>

<text>Принести водки Сидоровичу</text>

<icon>ui\_iconsTotal\_find\_item</icon>

<infoportion\_complete>test\_quest\_vodka\_otdana</infoportion\_complete> </objective>

<objective>

<text>Вернуться с водкой обратно</text>

<map\_location\_type hint="esc\_dinamit\_to\_volk">green\_location</map\_location\_type>

<object\_story\_id>Escape\_novice\_lager\_volk</object\_story\_id>

<infoportion\_complete>test\_quest\_vodka\_otdana</infoportion\_complete>

</objective>

<objective>

<text>Найти водку на чердаке</text>

<map\_location\_type hint="escape\_trader">blue\_location</map\_location\_type>

<object\_story\_id>Escape\_Trader</object\_story\_id>

<infoportion\_complete>test\_quest\_vodka\_otdana</infoportion\_complete>

</objective>

</game\_task>

Мы прописали данные о квесте: название, иконку, строки заданий, расположение маркеров и т.д.

К этому файлу мы еще вернемся, т.к. в этом варианте многое проставлялось не иначе как от балды. Я еще подумаю, как заставить задание "Вернуться с водкой обратно" появиться в PDA, а то оно что-то не желает явить себя народу... впрочем, ладно, главное, что все работает и не вылетает. А исправление столь мелких багов - дело десятое. Продолжим.

Откроем файл gamedata\config\gameplay\info\_l01escape.xml и зарегистрируем до того отсутствовавшие в игре типы условий (они нам еще пригодятся для завершения работы над квестом):

- после строки <game\_information\_portions> прописываем:

<info\_portion id="test\_quest\_vodka\_otdana"></info\_portion>

<info\_portion id="kvest\_vodka\_started">

<task>esc\_test\_vodka\_task</task>

</info\_portion>

<info\_portion id="otkazalsya\_prinesty\_vodku"></info\_portion>

С прописыванием условий - всё.

Теперь займемся одной небольшой, но нужной вещью - прописыванием экранных подсказок, которые высвечиваются при получении/выполнении этапа/завершении задания.

Для этого откроем файл gamedata\config\text\rus\string\_table\_tasks\_escape.xml и пропишем там:

<string id="esc\_test\_prinesti\_vodku">

<text>Принести бутылку водки</text>

</string>

<string id="esc\_test\_prinesti\_vodku\_1">

<text>Найти водку на чердаке у Волка</text>

</string>

<string id="esc\_test\_prinesti\_vodku\_2">

<text>Принести водку Сидоровичу</text>

</string>

Теперь можно тестировать, что получилось. Сверяем результаты:

- по окончанию разговора про водку у вас появляется новое задание - в задании маркером отмечен Волк (ближайший к чердаку маркер, потому и отмечен) - при получении задания на экране высвечивается уведомление

Можно приступать к третьей стадии. Что мы забыли? Правильно - возможность выполнить квест.

[править]

Реализация возможности выполнить квест

На самом деле - все просто. Все условия и т.д. уже нами созданы, осталось только применить их в диалоге. Давайте несколько изменим его структуру, сделаем его ветвистым и функциональным:

0. Сидрыч, давай поговорим о... водке. ( переход к 1)

1. О водке? Да какая водка? Нет её у меня, кончилась. Вроде у Волка на чердаке была, что-то он однажды об этом упоминал... Принеси, а? ( выбор - переход к 2, 4, 5; изначально доступен только 2 вариант )

2. Да сбегаю я сейчас, сбегаю... алкоголик, блин. ( получение квеста, переход к 3, после этого перестает быть доступен )

3. Беги, беги... сил уж нет ждать. ( конец, переход к списку веток Сидоровича )

4. Вот она, твоя бутылка водки. Ну, вернее, не твоя, а Волка... ( появляется только при наличии водки в инвентаре, при выборе водка передается Сидоровичу, квест выполняется, идет переход к 7 )

5. Извини, Сидрыч, я пока еще не раздобыл, терпи... ( появляется только если взять квест, переход к 6 )

6. А ну марш отсюда! Без бутылки пришел... ууу, вражина... ( переход к 8 )

7. Теперь моя будет, Меченый! Ты так Волку и передай. Спасибо, выручил... держи награду. ( выдача награды, переход к 8 )

8. Ну тебя в пень... ( конец, переход к списку веток Сидоровича )

А вот так это будет выглядеть в файле gamedata\config\gameplay\dialogs\_escape.xml:

<dialog id="escape\_trader\_oops">

<precondition>escape\_dialog.trader\_alredy\_give\_job</precondition>

<has\_info>tutorial\_end</has\_info>

<phrase\_list>

<phrase id="0">

<text>escape\_trader\_oops\_0</text>

<next>1</next>

</phrase>

<phrase id="1">

<text>escape\_trader\_oops\_1</text>

<next>2</next>

<next>4</next>

<next>5</next>

</phrase>

<phrase id="2">

<dont\_has\_info>kvest\_vodka\_started</dont\_has\_info>

<text>escape\_trader\_oops\_2</text>

<give\_info>kvest\_vodka\_started</give\_info>

<next>3</next>

</phrase>

<phrase id="3">

<text>escape\_trader\_oops\_3</text>

<action>dialogs.break\_dialog</action>

</phrase>

<phrase id="4">

<text>escape\_trader\_oops\_4</text>

<precondition>escape\_dialog.have\_a\_vodka</precondition>

<action>escape\_dialog.otday\_vodku</action>

<next>7</next>

</phrase>

<phrase id="5">

<has\_info>kvest\_vodka\_started</has\_info>

<text>escape\_trader\_oops\_5</text>

<next>6</next>

</phrase>

<phrase id="6">

<text>escape\_trader\_oops\_6</text>

<next>8</next>

</phrase>

<phrase id="7">

<text>escape\_trader\_oops\_7</text>

<give\_info>test\_quest\_vodka\_otdana</give\_info>

<action>escape\_dialog.transfer\_deneg</action>

<next>8</next>

</phrase>

<phrase id="8">

<text>escape\_trader\_oops\_8</text>

</phrase>

</phrase\_list>

</dialog>

Теперь добавим в игру скрипты "передача денег" и "передача водки". Откройте файл gamedata\scripts\escape\_dialog.script и перед разделителем "Trader" добавьте туда следующие строки:

function transfer\_deneg(first\_speaker, second\_speaker)

dialogs.relocate\_money(second\_speaker, 4000, "in")

end

function otday\_vodku (npc, actor)

dialogs.relocate\_item\_section(npc, "vodka", "out")

end

function have\_a\_vodka (stalker, player)

return stalker:object ("vodka") ~= nil

end

Готово. Запускайте игру и любуйтесь.

Для тех, кто еще с нами - сейчас мы модифицируем квест так, чтобы исключить конфликты с другими подобными модами.

[править]

Реализация совместимости

Из-за чего наш мод не может работать вместе с аналогичными ему модами? Да потому, что файлы dialogs\_escape.xml и некоторые другие у них пересекаются. Поэтому выделим их отдельно от основных диалогов:

Вместо dialogs\_escape.xml мы будем испольховать новый - dialogs\_new.xml Вместо stable\_dialogs\_escape.xml мы будем использовать файл stable\_dialogs\_new.xml (важно! не знаю почему, но он должен лежать в text\..., а не в text\rus\...)

Соответственно, весь реализованный нами новый контент перекидываем в пустые файлы (соблюдая форматирование), подчистив за собой на старом месте.

Единственное, что останется сделать после установки того же Repair Mod'а - заново добавить ветку диалога Сидоровичу в файле character\_desc\_escape.xml. Остальное в новом варианте нашего мода переписывать не придется - для диалогов отведено три файла.

Остальные файлы относятся уже к квестовой части и ни с чем пересекаться не могут - где вы видели другие моды с новыми квестами? Я - нет.

Откройте файл gamedata\config\localization.ltx и зарегистрируйте там stable\_dialogs\_escape.ltx. Потом откройте файл gamedata\config\system.ltx и в разделе "Dialogs" в конце строки (после dialogs\_aes) пропишите dialogs\_new (без ".ltx"!).

Все. Теперь ваши диалоги будут подгружаться из сторонних файлов.

Отдельное спасибо:

XiaNi - за страшный и непричесанный, но работающий пример квеста, на основе которого я и освоил принципы квестостроения =)

Keha – за урок по созданию новых диалогов

[править]

Дополнения

1. Если изменить шапку нашего диалога в файле dialogs\_escape.xml со старого варианта на такой, то ветка исчезнет после выполнения задания:

<dialog id="escape\_trader\_oops">

<precondition>escape\_dialog.trader\_alredy\_give\_job</precondition>

<has\_info>tutorial\_end</has\_info>

<dont\_has\_info>test\_quest\_vodka\_otdana</dont\_has\_info>

<phrase\_list>

...

Не тестировал. Дело за вами =)

Фух, вроде всё... Всем спасибо, занавес =)

Вступление

Перед изучением этого урока советую освоить находящийся несколько выше урок "Создание новых веток диалогов".

Небольшие правила, следуя которым, можно сберечь себе много нервов:

- всегда сохраняйте резервные копии модифицируемых файлов - старайтесь почаще тестировать внесенные изменения - так проще понять, в чем вы ошиблись (если активно работать пару часов без остановки, не разу не протетировав внесенные изменения, то потом будет очень сложно понять - почему же, черт возьми, игра вылетает при попытке поговорить с Сидоровичем?) - перед началом работы создайте новую папку gamedata, а свои моды спрячьте подальше: работать с "чистой" версией игры не в пример удобнее и проще; особенно противопоказан Repair Mod, который с новыми диалогами часто конфликтует\*

\* о том, как создать диалоги и квесты, исключающие возможность конфликта с другими подобными модами, см. часть 4.

[править]

Создание базового диалога

Начнем создание квеста с написания небольшого линейного диалога. Это будет выглядеть примерно так:

Меченый: Сидрыч, давай поговорим о... водке. Сидорович: О водке? Да о какой водке сейчас можно говорить? Вокруг оглянись. Видишь хоть одну бутылку?! Вот. Именно. Кончилась она у меня, родимая, кончилась... помоги, принесешь, а? Вроде у Волка на чердаке была, что-то он однажды об этом упоминал... Меченый: Да сбегаю я сейчас, сбегаю... алкоголик, блин. Сидорович: Беги, беги... сил уж нет ждать, давай, родимый.

После третьей фразы у нас будет активироваться квест "Принеси водки Сидоровичу".

Открываем файл gamedata\config\gameplay\character\_desc\_escape.xml и проводим такие манипуляции:

- находим строку "<actor\_dialog>escape\_trader\_done\_blockpost\_box</actor\_dialog>" - добавляем после нее новую - "<actor\_dialog>escape\_trader\_oops</actor\_dialog>" - сохраняем изменения

Мы добавили к Диалогу с Сидоровичем новую ветку, вернее, только ссылку на неё. Сама внутренняя структура ветки хранится в файле gamedata\config\gameplay\dialogs\_escape.xml, который мы открываем и производим:

- находим в любом месте файла промежуток между </dialog> и <dialog id="..."> - засовываем туда такую конструкцию:

<dialog id="escape\_trader\_oops">

<precondition>escape\_dialog.trader\_alredy\_give\_job</precondition>

<has\_info>tutorial\_end</has\_info>

<phrase\_list>

<phrase id="0">

<text>escape\_trader\_oops\_0</text>

<next>1</next>

</phrase>

<phrase id="1">

<text>escape\_trader\_oops\_1</text>

<next>2</next>

</phrase>

<phrase id="2">

<text>escape\_trader\_oops\_2</text>

<next>1</next>

</phrase>

<phrase id="3">

<text>escape\_trader\_oops\_3</text>

</phrase>

</phrase\_list>

</dialog>

- сохраняем изменения

Теперь у нас прописана структура новой ветки диалога. Осталось сделать еще одну вещь - прописать строки, которые будут отображаться вместо "escape\_trader\_oops\_...".

Для этого откроем файл gamedata\config\text\rus\stable\_dialogs\_escape.xml и сделаем следующее:

- находим любой удобный промежуток между </string> и <string id="..."> - прописываем туда следующие строки:

<string id="escape\_trader\_oops\_0">

<text>Сидрыч, давай поговорим о... водке.</text>

</string>

<string id="escape\_trader\_oops\_1">

<text>О водке? Да о какой водке сейчас можно говорить? Вокруг оглянись. Видишь хоть одну бутылку?!

Вот. Именно. Кончилась она у меня, родимая, кончилась... помоги, принесешь, а?

Вроде у Волка на чердаке была, что-то он однажды об этом упоминал...</text>

</string>

<string id="escape\_trader\_oops\_2">

<text>Да сбегаю я сейчас, сбегаю... алкоголик, блин.</text>

</string>

<string id="escape\_trader\_oops\_3">

<text>Беги, беги... сил уж нет ждать, давай, родимый.</text>

</string>

- сохраняем изменения

Всё. Диалог готов. Можно грузить игру и смотреть, работает ли новая ветка. А теперь будем прикреплять к этому безобразию квест ""Принеси водки Сидоровичу".

[править]

Создание базового квеста

Активацию квеста в диалог прикрепить достаточно просто. Проводим с файлом gamedata\config\gameplay\ dialogs\_escape.xml следующие манипуляции:

- добавляем содержимое фразы <phrase id="2"> следующим:

<phrase id="2">

<text>escape\_trader\_oops\_2</text>

<give\_info>kvest\_vodka\_started</give\_info>

<next>3</next>

</phrase>

- сохраняем изменения

Теперь после фразы Меченого "Да сбегаю я сейчас, сбегаю..." Сидорович таки всучивает нам квест "Принеси водку". Но не спешите тестировать изменения. Мы добавили в диалог упоминание о квесте, хотя он еще не создан.

Исправим это недоразумение. Откройте файл gamedata\config\gameplay\tasks\_escape.xml и в начале файла создайте такую конструкцию:

<game\_task id="esc\_test\_vodka\_task">

<title>Принести водки</title> <!-- заголовок квеста -->

<objective>

<text>Принести водки Сидоровичу</text>

<icon>ui\_iconsTotal\_find\_item</icon>

<infoportion\_complete>test\_quest\_vodka\_otdana</infoportion\_complete> </objective>

<objective>

<text>Вернуться с водкой обратно</text>

<map\_location\_type hint="esc\_dinamit\_to\_volk">green\_location</map\_location\_type>

<object\_story\_id>Escape\_novice\_lager\_volk</object\_story\_id>

<infoportion\_complete>test\_quest\_vodka\_otdana</infoportion\_complete>

</objective>

<objective>

<text>Найти водку на чердаке</text>

<map\_location\_type hint="escape\_trader">blue\_location</map\_location\_type>

<object\_story\_id>Escape\_Trader</object\_story\_id>

<infoportion\_complete>test\_quest\_vodka\_otdana</infoportion\_complete>

</objective>

</game\_task>

Мы прописали данные о квесте: название, иконку, строки заданий, расположение маркеров и т.д.

К этому файлу мы еще вернемся, т.к. в этом варианте многое проставлялось не иначе как от балды. Я еще подумаю, как заставить задание "Вернуться с водкой обратно" появиться в PDA, а то оно что-то не желает явить себя народу... впрочем, ладно, главное, что все работает и не вылетает. А исправление столь мелких багов - дело десятое. Продолжим.

Откроем файл gamedata\config\gameplay\info\_l01escape.xml и зарегистрируем до того отсутствовавшие в игре типы условий (они нам еще пригодятся для завершения работы над квестом):

- после строки <game\_information\_portions> прописываем:

<info\_portion id="test\_quest\_vodka\_otdana"></info\_portion>

<info\_portion id="kvest\_vodka\_started">

<task>esc\_test\_vodka\_task</task>

</info\_portion>

<info\_portion id="otkazalsya\_prinesty\_vodku"></info\_portion>

С прописыванием условий - всё.

Теперь займемся одной небольшой, но нужной вещью - прописыванием экранных подсказок, которые высвечиваются при получении/выполнении этапа/завершении задания.

Для этого откроем файл gamedata\config\text\rus\string\_table\_tasks\_escape.xml и пропишем там:

<string id="esc\_test\_prinesti\_vodku">

<text>Принести бутылку водки</text>

</string>

<string id="esc\_test\_prinesti\_vodku\_1">

<text>Найти водку на чердаке у Волка</text>

</string>

<string id="esc\_test\_prinesti\_vodku\_2">

<text>Принести бутылку водки Сидоровичу</text>

</string>

Теперь можно тестировать, что получилось. Сверяем результаты:

- по окончанию разговора про водку у вас появляется новое задание - в задании маркером отмечен Волк (ближайший к чердаку маркер, потому и отмечен) - при получении задания на экране высвечивается уведомление

Можно приступать к третьей стадии. Что мы забыли? Правильно - возможность выполнить квест.

[править]

Реализация возможности выполнить квест

На самом деле - все просто. Все условия и т.д. уже нами созданы, осталось только применить их в диалоге. Давайте несколько изменим его структуру, сделаем его ветвистым и функциональным:

0. Сидрыч, давай поговорим о... водке. ( переход к 1) 1. О водке? Да о какой водке сейчас можно говорить? Вокруг оглянись. Видишь хоть одну бутылку?! Вот. Именно. Кончилась она у меня, родимая, кончилась... помоги, принесешь, а? Вроде у Волка на чердаке была, что-то он однажды об этом упоминал... ( выбор - переход к 2, 4, 5; изначально доступен только 2 вариант ) 2. Да сбегаю я сейчас, сбегаю... алкоголик, блин. ( получение квеста, переход к 3, после этого перестает быть доступен ) 3. Беги, беги... сил уж нет ждать, давай, родимый. ( конец, переход к списку веток Сидоровича ) 4. Вот она, твоя бутылка водки. Ну, вернее, не твоя, а Волка... ( появляется только при наличии водки в инвентаре, при выборе водка передается Сидоровичу, квест выполняется, идет переход к 7 ) 5. Извини, Сидрыч, я пока еще не раздобыл, терпи... ( появляется только если взять квест, переход к 6 ) 6. А ну марш отсюда! Без бутылки пришел... ууу, вражина... ( переход к 8 ) 7. Теперь моя будет, Меченый! Ты так Волку и передай. Спасибо, выручил... держи награду. Нееет, это не водка, обойдешься... денег дам. ( выдача награды, переход к 8 ) 8. Ну тебя в пень... ( конец, переход к списку веток Сидоровича )

А вот так это будет выглядеть в файле gamedata\config\gameplay\dialogs\_escape.xml:

<dialog id="escape\_trader\_oops">

<precondition>escape\_dialog.trader\_alredy\_give\_job</precondition>

<has\_info>tutorial\_end</has\_info>

<phrase\_list>

<phrase id="0">

<text>escape\_trader\_oops\_0</text>

<next>1</next>

</phrase>

<phrase id="1">

<text>escape\_trader\_oops\_1</text>

<next>2</next>

<next>4</next>

<next>5</next>

</phrase>

<phrase id="2">

<dont\_has\_info>kvest\_vodka\_started</dont\_has\_info>

<text>escape\_trader\_oops\_2</text>

<give\_info>kvest\_vodka\_started</give\_info>

<next>3</next>

</phrase>

<phrase id="3">

<text>escape\_trader\_oops\_3</text>

<action>dialogs.break\_dialog</action>

</phrase>

<phrase id="4">

<text>escape\_trader\_oops\_4</text>

<precondition>escape\_dialog.have\_a\_vodka</precondition>

<action>escape\_dialog.otday\_vodku</action>

<next>7</next>

</phrase>

<phrase id="5">

<has\_info>kvest\_vodka\_started</has\_info>

<text>escape\_trader\_oops\_5</text>

<next>6</next>

</phrase>

<phrase id="6">

<text>escape\_trader\_oops\_6</text>

<next>8</next>

</phrase>

<phrase id="7">

<text>escape\_trader\_oops\_7</text>

<give\_info>test\_quest\_vodka\_otdana</give\_info>

<action>escape\_dialog.transfer\_deneg</action>

<next>8</next>

</phrase>

<phrase id="8">

<text>escape\_trader\_oops\_8</text>

</phrase>

</phrase\_list>

</dialog>

Теперь добавим в игру скрипты "передача денег" и "передача водки". Откройте файл gamedata\scripts\escape\_dialog.script и перед разделителем "Trader" добавьте туда следующие строки:

function transfer\_deneg (npc, actor)

dialogs.relocate\_money(stalker, 8000, "in")

end

function otday\_vodku (npc, actor)

dialogs.relocate\_item\_section(npc, "vodka", "out")

end

function have\_a\_vodka (stalker, player)

return stalker:object ("vodka") ~= nil

end

Готово. Запускайте игру и любуйтесь.

Есть несколько багов, кто исправит, тому пирожок =) Итак:

- второй пункт задания в PDA не отображается - задание не обновляется при подборе бутылки водки (как это реализовать, в принципе понятно, но это как-нибудь в другой раз, сейчас лень) - премиальные не перечисляются Меченому (уу, Сидорович, жид такой)

Для тех, кто еще с нами - сейчас мы модифицируем квест так, чтобы исключить конфликты с другими подобными модами.

[править]

Реализация совместимости

Из-за чего наш мод не может работать вместе с аналогичными ему модами? Да потому, что файлы dialogs\_escape.xml и некоторые другие у них пересекаются. Поэтому выделим их отдельно от основных диалогов:

Вместо dialogs\_escape.xml мы будем испольховать новый - dialogs\_new.xml Вместо stable\_dialogs\_escape.xml мы будем использовать файл stable\_dialogs\_new.xml (важно! не знаю почему, но он должен лежать в text\..., а не в text\rus\...)

Соответственно, весь реализованный нами новый контент перекидываем в пустые файлы (соблюдая форматирование), подчистив за собой на старом месте.

Единственное, что останется сделать после установки того же Repair Mod'а - заново добавить ветку диалога Сидоровичу в файле character\_desc\_escape.xml. Остальное в новом варианте нашего мода переписывать не придется - для диалогов отведено три файла.

Остальные файлы относятся уже к квестовой части и ни с чем пересекаться не могут - где вы видели другие моды с новыми квестами? Я - нет.

Откройте файл gamedata\config\localization.ltx и зарегистрируйте там stable\_dialogs\_escape.ltx. Потом откройте файл gamedata\config\system.ltx и в разделе "Dialogs" в конце строки (после dialogs\_aes) пропишите dialogs\_new (без ".ltx"!).

Все. Теперь ваши диалоги будут подгружаться из сторонних файлов.

Отдельное спасибо:

XiaNi - за страшный и непричесанный, но работающий пример квеста, на основе которого я и освоил принципы квестостроения =) Keha – за урок по созданию новых диалогов Dйja Vu – за создание wiki-ресурса

[править]

Дополнения

1. Если изменить шапку нашего диалога в файле dialogs\_escape.xml со старого варианта на такой, то ветка исчезнет после выполнения задания:

<dialog id="escape\_trader\_oops">

<precondition>escape\_dialog.trader\_alredy\_give\_job</precondition>

<has\_info>tutorial\_end</has\_info>

<dont\_has\_info>test\_quest\_vodka\_otdana</dont\_has\_info>

<phrase\_list>

...

Не тестировал. Дело за вами =)

2. Разнообразить речь обычных сталкеров не так просто, как кажется – они не выделены в отдельных людей, а используют один и тот же набор реплик, зависящий от локации/мастерства сталкера. Но при этом в пункте "Что можешь интересного рассказать" у каждого сталкера история разная.

Кто сможет разузнать, откуда берутся тексты для этих реплик, тому буду очень благодарен...

3. Неплохо бы добавить реплики гопникам, но это уже на стадии предположений. На данный момент реплик у них нет, так что если вы, например, играете с модом "Улучшенные взаимоотношения" (или как там его) или какими-либо иными способами добились нейтрального/дружеского отношения с их стороны, то при потытке заговорить игра вылетит.

В принципе, можно зарегистрировать новые базы диалогов для бандитов (как было описано выше), созданные по образцу таковых у сталкеров. Но как эти наборы диалогов связать с гуляющими по карте гопниками - вопрос.(ответ от ЖуКа надо редактировать файл character\_desc\_simulation.xml он используется при респавне неписей)

У меня есть подозрения, что набор диалогов прописывается персонажу в его свойствах в файле all.spawn. Прошу специалистов по spawn-ресурсам посмотреть, имеются у NPC такие свойства или нет.

Фух, вроде всё...

Всем спасибо, занавес =)

[править]

Авторы

Статья создана: BAC9-FLCL

-------------------------------------------------как создать квест с рандомными наградами--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Вообщем метод не мой, я его взял у banderosa(а он в свою очередь взял Linspiro) как создать квест с рандомными наградами. Беру любого торгоша или любого другого уникального непися и добавляяю ему диалог. Например Волк. Создадим квест что если мы принесем 3 хвоста или 3 копыта Волку он нам выдаст рандомно(т.е. случайно) награду.

1)В character\_desc\_escape к уже прописанным диалогам Волка добавляю свой

<actor\_dialog>prinesi\_hvost</actor\_dialog> - вот мой диалог.

<start\_dialog>escape\_lager\_volk\_talk</start\_dialog>

<start\_dialog>dm\_hello\_dialog</start\_dialog>

<actor\_dialog>dm\_cool\_info\_dialog</actor\_dialog>

<actor\_dialog>dm\_help\_wounded\_medkit\_dialog</actor\_dialog>

<actor\_dialog>tm\_wolf\_dialog</actor\_dialog>

<actor\_dialog>tm\_wolf\_reward</actor\_dialog>

<actor\_dialog>escape\_lager\_volk\_story</actor\_dialog>

<actor\_dialog>escape\_lager\_volk\_story2</actor\_dialog>

2) Откроем файл stable\_dialogs\_escape и в конце напишем следующий диалог:

<string id="prinesi\_hvost\_0">

<text>Здорово! Что делаешь?</text>

</string>

<string id="prinesi\_hvost\_1">

<text>Привет меченый да вот бизнес налаживаю. Послушай ты же по зоне мотаешся, живность всякую отстреливаешь принеси мне 3 хвоста слепых собак либо 3 копыта кабанов местных.А то у меня заказчик появился на живность всякую,говорят он из них лекарства делает. А на местных новичков разве можно расчитывать? они ж при виде кабана сами своих копыта готовы отдать. Вообщем принесешь, отблагодарю нормально, не обижу.</text>

</string>

<string id="prinesi\_hvost\_2">

<text>Ну что Волк принес я копыта принимай.</text>

</string>

<string id="prinesi\_hvost\_3">

<text>Ну что Волк принес я хвосты принимай.</text>

</string>

<string id="prinesi\_hvost\_4">

<text>Копыта принес, это хорошо,получи расчет.</text>

</string>

<string id="prinesi\_hvost\_5">

<text>Хвосты принес ну ладно тоже ничего, может и они на что-нибудь сгодятся.</text>

</string>

<string id="prinesi\_hvost\_6">

<text>Сегодня пустой, так поболтать заскочил.</text>

</string>

3)Дальше открываем файл dialogs\_escape( в принципе лучше создать свой диалоговый файл типа"dialogs\_new" потом легче если что совмещать с другими модами главное не забудьте прописать свой файл в файле systems)

и в конце пишу диалог пример:

<game\_dialogs>

<dialog id="prinesi\_hvost"> это id диалога то что мы добавили волку в его секцию

<phrase\_list>

<phrase id="0">

<text>prinesi\_hvost\_0</text>

<next>1</next>

</phrase>

<phrase id="1">

<text>prinesi\_hvost\_1</text>

<next>2</next>

<next>3</next>

<next>6</next>

</phrase>

<phrase id="2">

<precondition>new\_dialog.prines\_kopyto</precondition> это условие при котором открывается данная

<text>prinesi\_hvost\_2</text> фраза в нашем диалоге(должна выполниться

<next>4</next> функция prines\_kopyto из файла new\_dialog

</phrase>

<phrase id="3">

<precondition>new\_dialog.prines\_hvost</precondition> аналогично 1 условию тоолько там копыта,

<text>prinesi\_hvost\_3</text> а здесь хвосты

<next>5</next>

</phrase>

<phrase id="4">

<action>new\_dialog.otdal\_kopyto</action> здесь срабатывает функция передачи копыта и получение

<text>prinesi\_hvost\_4</text> рандомной награды.

</phrase>

<phrase id="5">

<action>new\_dialog.otdal\_hvost</action> аналогично, только отдаются хвосты.

<text>prinesi\_hvost\_5</text>

</phrase>

<phrase id="6">

<text>prinesi\_hvost\_6</text>

<action>dialogs.break\_dialog</action>

</phrase>

</phrase\_list>

</dialog>

4)Теперь надо создать наш скриптовый файл и прописать наши функции. Создаем файл new\_dialog а в нем пишем следующие

local nagrad = {

level\_1 = {

"ammo\_9x39\_sp5",

"ammo\_7.62x54\_7h14",

"ammo\_5.45x39\_ap", это список того из чего будет выбираться награда 1 и2 уровня.Для хвостов 1 уровень

"ammo\_5.56x45\_ap", Для копыт второй. Специально добавил колбасу и хлеб, ну чтобы разочаровываться

"ammo\_11.43x23\_hydro", иногда(рулетка).

"bread",

"kolbasa",

"ammo\_5.56x45\_ss190"

},

level\_2 = {

"af\_armor\_1",

"af\_babka\_1",

"af\_cry\_1",

"af\_dik\_1",

"af\_kol\_1",

"af\_pudd\_1",

"af\_spirit\_1",

"bread",

"kolbasa",

"af\_night\_star",

"af\_gold\_fish",

"af\_electra\_moonlight",

"af\_rusty\_sea-urchin",

"af\_dummy\_glassbeads"

}

}

function prines\_kopyto(first\_speaker, second\_speaker)

local actor, trader

if db.actor and db.actor:id() == first\_speaker:id() then

actor = first\_speaker

trader = second\_speaker

else

actor = second\_speaker

trader = first\_speaker

end

local mutant\_boar\_leg = 0

actor:iterate\_inventory(function (dummy, item) if item:section()=="mutant\_boar\_leg" then

mutant\_boar\_leg = mutant\_boar\_leg+1 end end, nil)

if mutant\_boar\_leg>=3 then

return true

else

return false

end

end

function otdal\_kopyto(first\_speaker, second\_speaker)

local item = ""

local actor, trader

if db.actor and db.actor:id() == first\_speaker:id() then

actor = first\_speaker

trader = second\_speaker

else

actor = second\_speaker

trader = first\_speaker

end

item = nagrad.level\_2[math.random(table.getn(nagrad.level\_2))]

flamethrower.relocate\_n\_items\_section(trader, "mutant\_boar\_leg", "out", 3)

dialogs.relocate\_item\_section(second\_speaker, item , "in")

end

function prines\_hvost(first\_speaker, second\_speaker)

local actor, trader

if db.actor and db.actor:id() == first\_speaker:id() then

actor = first\_speaker

trader = second\_speaker

else

actor = second\_speaker

trader = first\_speaker

end

local mutant\_dog\_tail = 0

actor:iterate\_inventory(function (dummy, item) if item:section()=="mutant\_dog\_tail" then mutant\_dog\_tail=mutant\_dog\_tail+1 end end, nil)

if mutant\_dog\_tail>=3 then

return true

else

return false

end

end

function otdal\_hvost(first\_speaker, second\_speaker)

local item = ""

local actor, trader

if db.actor and db.actor:id() == first\_speaker:id() then

actor = first\_speaker

trader = second\_speaker

else

actor = second\_speaker

trader = first\_speaker

end

item = nagrad.level\_1[math.random(table.getn(nagrad.level\_1))]

flamethrower.relocate\_n\_items\_section(trader, "mutant\_dog\_tail", "out", 3)

flamethrower.relocate\_n\_items\_section(actor, item , "in", count)

end

Функции prines\_kopyto и prines\_hvost проверяют наличие у нас в инвентаре 3 хвостов либо 3 копыт и соответственно

допускают нас к фразам(<precondition>new\_dialog.prines\_kopyto</precondition>) в которых у нас происходит обмен копыт и хвостов на рандомную награду. Функции otdal\_hvost и otdal\_kopyto забирают у нас копыта и хвосты и выдают награды из соответствующего списка. Все.