



Neverwinter Nights 2 Legends Banter Plugin

Version 2.5.2 EE64
August 16, 2026
Marshall Vyper
© 2026 Marshall Vyper

Document Revision Information

Revision	Date	Author	Description
1.2	November 28, 2010	Marshall Vyper	Initial Release
1.21	December 14, 2010	Marshall Vyper	New Master Vars Added
1.30	December 20, 2010	Marshall Vyper	Upgraded Masters to 1.3
1.31	December 20, 2010	Marshall Vyper	Fixed Numerous Typos
1.42	September 29, 2011	Marshall Vyper	Upgrade to new Plugin System
1.84	November 1, 2012	Marshall Vyper	Updated notes and screenshots, conversations.
2.0	December 29, 2016	Marshall Vyper	Updated notes for 2.0
2.5.1	August 2026	Marshall Vyper	EE64 compatibility: x64/.NET 4.7.2 and current MySQL reliability/compatibility updates.
2.5.2	August 2026	Marshall Vyper	Corrected the optional AI built-in scripts archive lookup to Data\scripts.zip.

Plugin Revision Information

Revision	Date	Author	Description
1.2	November 28, 2010	Marshall Vyper	Initial Release
1.3	December 20, 2010	Marshall Vyper	Master Script Upgrades
1.42	September 29, 2011	Marshall Vyper	Upgrade to new Plugin System
1.5	October 5, 2011	Marshall Vyper	Updated Online Manual, standardized ini file.
1.8	October 4, 2012	Marshall Vyper	Update to v1.8
1.84	November 1, 2012	Marshall Vyper	Bug fixes.
1.88	March 1, 2013	Marshall Vyper	Bug fixes and codebase update.
1.92	December 23, 2013	Marshall Vyper	Fixed colored named bug.
1.93	July 15, 2015	Marshall Vyper	Cleanup and codebase update
2.0	December 26, 2016	Marshall Vyper	2.0 Updates, see upgrade notes doc.
2.5.1	August 2026	Marshall Vyper	EE64 compatibility: x64/.NET 4.7.2 and current MySQL reliability/compatibility updates.
2.5.2	August 2026	Marshall Vyper	Corrected the optional AI built-in scripts archive lookup to Data\scripts.zip.

Table of Contents

TABLE OF CONTENTS	3
INTRODUCTION	4
FEATURES.....	4
LEGENDS PLUGINS	5
<i>What's a Plugin</i>	<i>5</i>
<i>Important To Know.....</i>	<i>5</i>
PART 1 – GETTING STARTED.....	6
<i>What's in the File</i>	<i>6</i>
<i>YouTube Video Walkthroughs</i>	<i>6</i>
<i>Installation.....</i>	<i>7</i>
PART 2 – TRYING IT OUT	12
<i>Tutorial 1 – Your first blabbering NPC.....</i>	<i>12</i>
<i>Tutorial 2 – Greetings Adventur-ling</i>	<i>15</i>
<i>Tutorial 3 – What Time Is It?</i>	<i>18</i>
<i>Tutorial 4 – Any Last Words?.....</i>	<i>20</i>
<i>Tutorial 5 – Talkers!.....</i>	<i>23</i>
<i>Tutorial Summary.....</i>	<i>26</i>
PART 3 – ADVANCED CUSTOMIZATION.....	26
<i>Scripting</i>	<i>27</i>
FINAL SUMMARY	27

Introduction

There comes a time in every NPC's life where they simply cannot stand in silence any longer. Too many wandering commoners, monsters and other living world critters usually have more on their mind than just fighting or standing around. Sometimes they just can't wait for an adventurer to come along for them to share their thoughts, dreams or even secrets with.

This Legends Plugin adds that extra spice to the Neverwinter Nights 2 Persistent World. Giving your NPC's and monsters the ability to say random (or specific) one-line messages to nearby adventurers at various events and times. This would include during battle, when they die, when they are talked to or even when night sets in. Sometimes they can even have conversations with each other!

Allowing easy installation, an NPC or monster can speak whenever you deem desirable. Using NWNX database backend makes it simple to add, update or remove one-line messages that your living world creatures can use.

The Legends Banter plugin is designed for Persistent Worlds and requires an NWNX-compatible MySQL database backend. This guide documents builder configuration and controlled testing; complete live server/NWNX end-to-end validation remains a separate deployment gate.

Features

There are many various combinations of things you can do with bantering NPCs and monsters which can add a lot of flavors to your silent world. Some of the features this plugin can do for you are listed below:

- Randomly select messages from groups of one-liners.
- Included Day / Nighttime transition change messages.
- Random delays so not all your NPCs speak at the same time.
- Have monsters say their last dying words!
- Have monsters make comments when they hear a nearby hidden adventurer.
- Allow townsfolk to say various Hello's as adventurers pass them by.
- Easy to learn and use with step-by-step tutorials.
- You don't need to overhaul what you already have! Just build on it!

Legends Plugins

What's a Plugin

The Legends plugin system is a family of builder tools and Campaign resources used by the Legends Persistent World. Banter 2.5.2 includes the shared Master 2.5.1 component and has clearly identified runtime prerequisites. Optional integration with other Legends plugins adds features without making those plugins mandatory for core Banter operation.

Important To Know

Legends plugins can share Master resources and optional integration points. Banter ships with the shared `leg_master.dll` and `leg_master.hak`; it does not ship a Banter-specific HAK. Install only the package components documented for the current release.

- ⚠ Plugin versions are independently maintained. Banter 2.5.2 is paired with the shared Master 2.5.1 component. Back up your module before upgrading, use the supplied complete ERF, and follow a controlled compile and test process rather than relying on broad version or overwrite assumptions.

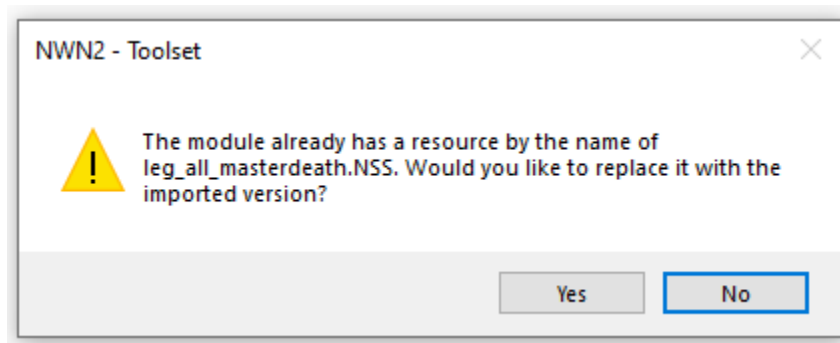


Figure 1 - Master Script Overwrite

Part 1 – Getting Started

What's in the File

By now you've already opened the archive with all the files needed for the Legends Banter Plugin to work in a test module or your persistent world. Included in the archive are several files, each of which are described below:

leg_banter.erf	The complete builder resource package: 34 resources consisting of 32 scripts, 1 trigger, and 1 conversation.
leg_banter_manual.pdf	This manual includes installation, configuration, and tutorials.
leg_banter_upgrade.pdf	Guidance for upgrading an existing Legends Banter installation to the 2.5 compatibility line.
leg_master.hak	The required shared Master HAK used by all Legends plugins.
leg_banter.dll	The x64 NWN2 Enhanced Edition Toolset plugin used to configure Banter and its chats.
leg_master.dll	The shared x64 Master Toolset component included with every Legends 2.5 plugin package.

YouTube Video Walkthroughs

Historical Legends video walkthroughs remain useful background material at <http://www.youtube.com/user/DJMarshallV>. Where an older video conflicts with this 2.5.2 EE64 manual, follow the current written instructions and <https://nwn2legends.com>.

Installation

To install this system, you must perform several tasks. You can install this into your existing persistent world or if you just wish to try it out, make a brand-new module and follow these steps.

1. Before continuing, install or confirm these prerequisites: the NWN2 Enhanced Edition 64-bit Toolset, .NET Framework 4.7.2 and MySQL Connector/NET 8.0.33.1 for the current Banter and Master plugins.
2. Close the Toolset before placing files. Copy leg_banter.dll and leg_master.dll to <NWN2 Enhanced Edition install folder>\NWN2Toolset\Plugins64. This Plugins64 folder is under the game installation, not the Neverwinter Nights 2 user Documents folders. Place leg_master.hak in the user Documents HAK folder. Place leg_banter.erf anywhere convenient as you don't need to keep it after installation, then select it to import in the toolset. No setup.exe installer is used.
3. After installing the DLLs and shared HAK, open the NWN2 Enhanced Edition 64-bit Toolset and load a backed-up test copy of the target module. From Plugins, confirm that Legends Banter Plugin and Legends # Master Configurator # are available.

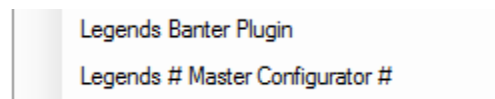


Figure 2 - Banter Plugin

- ⚠ If you do not see these entries, then select View -> Options from the top menu. On the general tab, ensure that “Allow Plugins” is set to “Load all Plugins”

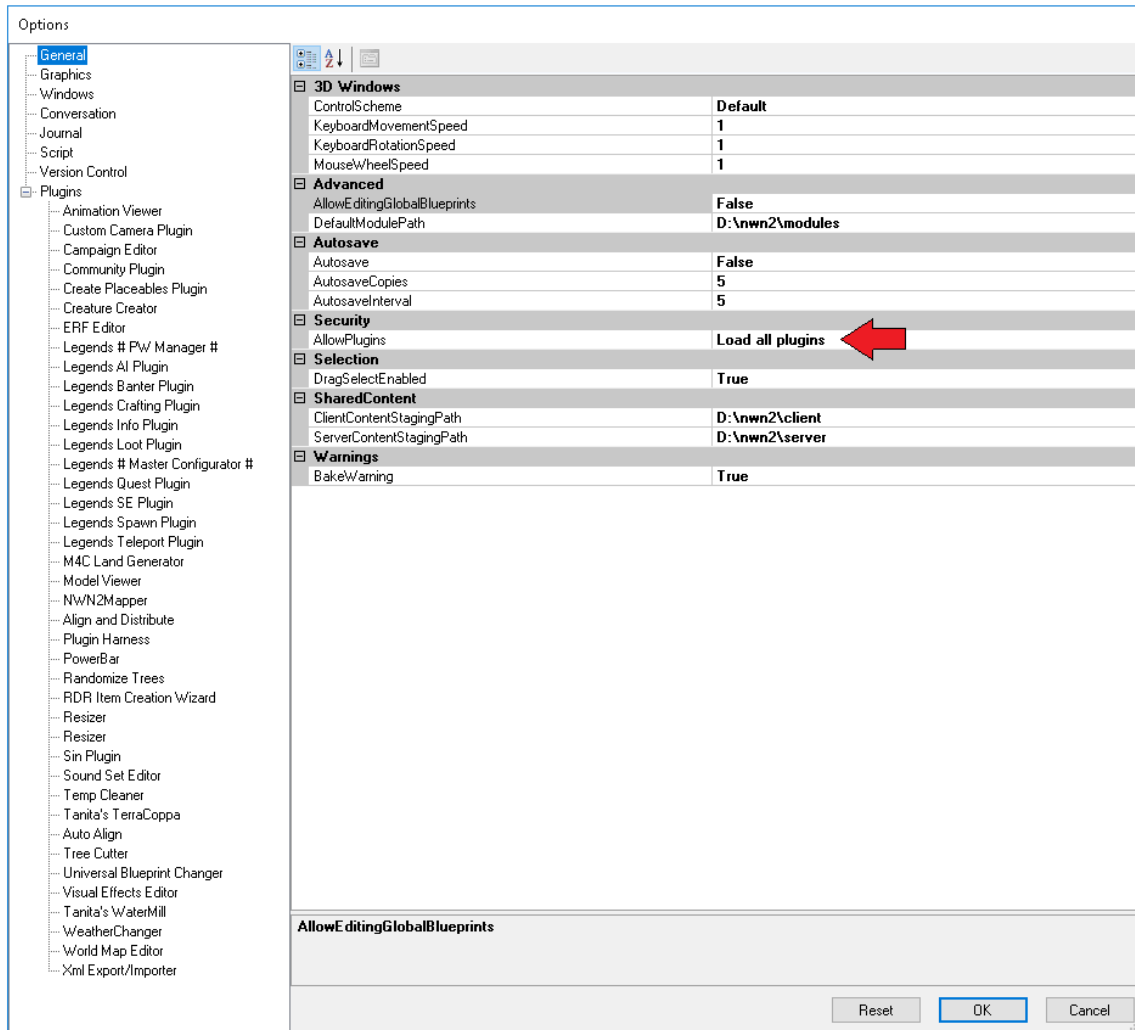


Figure 3 - Security Settings

- From the File menu, choose Import and select the current 2.5 package copy of leg_banter.erf. This is the complete current package of 32 NSS, 1 UTT, and 1 DLG resources. After importing, compile all scripts and save the module.

⚠ WARNING: Import overwrite replaces any manually customized same-name resources. Back up those custom resources before import and reapply intentional changes afterward if required.

- Open Legends # Master Configurator # and activate Legends Banter Plugin. Preserve the established database and table prefixes unless a separately planned migration is being performed.

Configuration

General Settings

☐ Loot Plugin Active ☐ Quest Plugin Active ☐ Factions Plugin Active
☐ Craft Plugin Active ☐ Time Plugin Active ☐ Bank Plugin Active
☐ Info Plugin Active ☐ Misc Plugin Active ☐ World Plugin Active
☐ AI Plugin Active ☐ Spawn Plugin Active ☐ Auctions Plugin Active
☒ Banter Plugin Active ☐ Teleport Plugin Active ☐ Housing Plugin Active

☐ Campaign Level Objects ☐ Dungeon Instance ☐ Secret Doors ☐ Pre 1.88 Plugins

Database Options

Server IP/Name:
 TCP Port:
 Database Name:
 User ID:
 Password:
 Player Table Prefix:

Figure 4 - Legends Master Configurator

- **CHECK BOXES** = Activate only the Legends plugins installed for this module. Optional integrations such as AI, Spawn, Quest, and SE are not required for core Banter operation.
- **DATABASE OPTIONS** = Banter requires an NWNX-compatible MySQL database backend. Configure only the database identity intended for the Toolset.
- **CAMPAIGN LEVEL OBJECTS** = Select this only when the module is configured to use a campaign and its scripts are intentionally maintained there. ERF import places scripts in the module. Choose one authoritative script location and avoid retaining duplicate or stale copies in both the module and campaign.

- **PLAYER TABLE PREFIX** = Preserve the established player-table prefix during a normal upgrade. Changing prefixes is a migration operation and is outside normal unchanged-prefix setup.
6. Once you've completed the configuration, click Finish and then note there are two other buttons available to use.

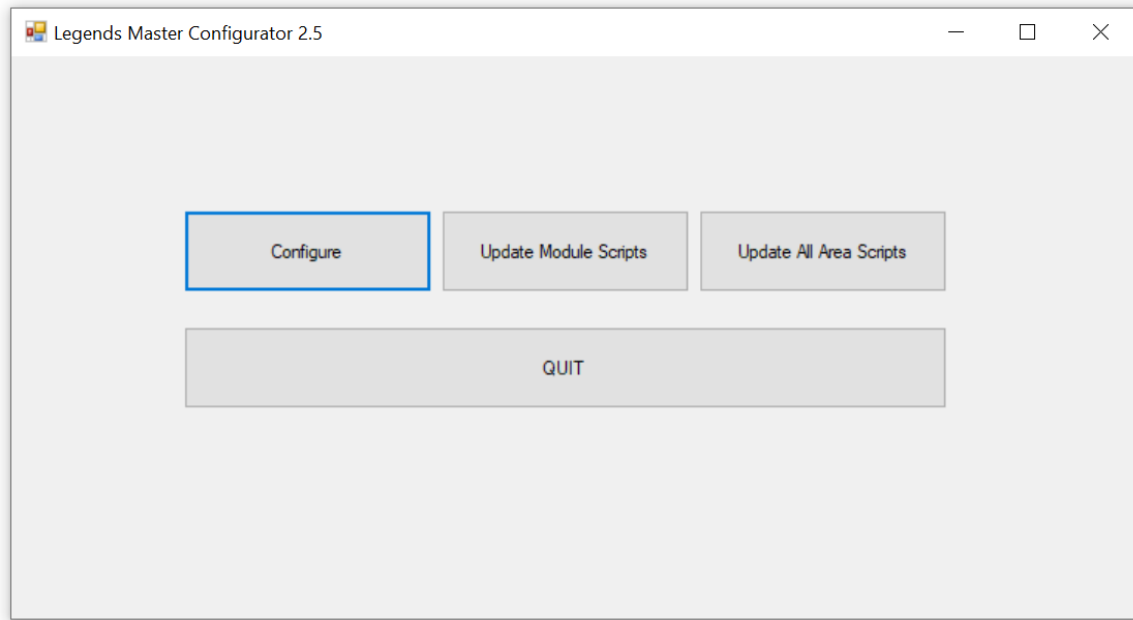


Figure 5 - Master Configurator

- **UPDATE MODULE SCRIPTS** = Use the Master Configurator only on a backed-up module and review the module-event changes it applies. If integrating manually, ensure the required supplied Master event scripts are called without discarding existing module logic.
 - i. `leg_all_mastermod_itemacquired` – On Item Acquired Event
 - ii. `leg_all_mastermod_itemactivated` – On Item Activated Event
 - iii. `leg_all_mastermod_itemunacquire` – On Item UnAcquire Event
 - iv. `leg_all_mastermod_ondeth` – On Player Death Event
 - v. `leg_all_mastermod_onheartbeat` – On Module Heartbeat Event
 - vi. `leg_all_mastermod_onload` – On Module Load Event
 - vii. `leg_all_mastermod_onrest` – On Player Rest Event

- UPDATE ALL AREA SCRIPTS = Use this only when the active plugin set requires the area OnClientEnter/OnExit integration. Back up the module first and review all generated or modified event-script calls.
7. Once you have finished configuring the Master Options, choose Plugins from the top menu and select “Legends Banter Plugin”, then choose File -> Configuration.

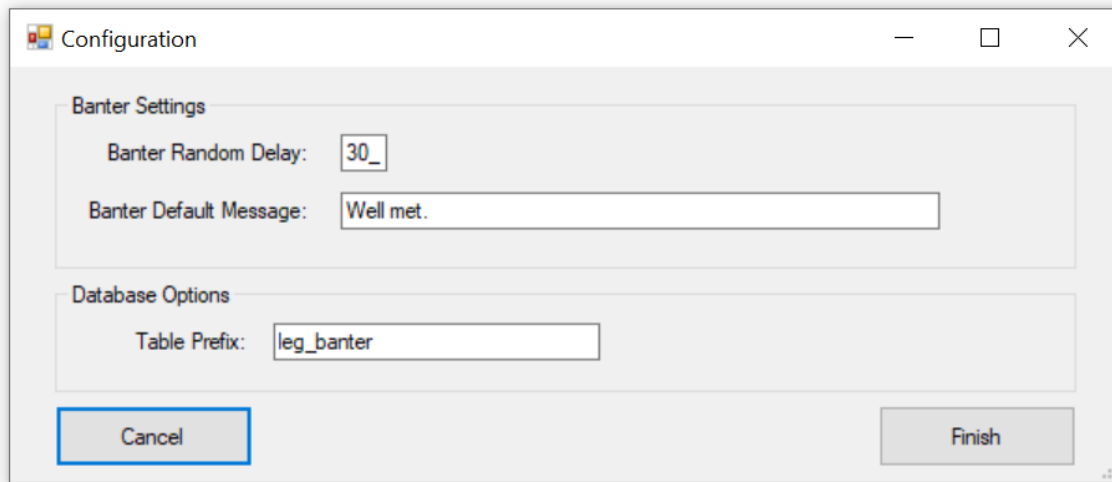


Figure 6 - Legends Banter Plugin Configuration

- BANTER RANDOM DELAY = The number of seconds to randomly choose between 0 and this number to delay when speaking a day to night or night to day message. Handy if you have a town filled with NPC's and the time changes to night. We don't want all our NPC's speaking at the same moment.
- BANTER DEFAULT MESSAGE = This is the default message an NPC will speak if they are configured to perform speaking actions but have no matching entry in the above database table.
- TABLE PREFIX = This is the name of the table to use in NWNX and your database for organization.

Part 2 – Trying it Out

Tutorial 1 – Your first blabbering NPC

Now that you've completed the installation process, you're probably itching to get started with the plugin. There are several configuration options for bantering, so it is best to start small and work your way up.

Each NPC can speak their own single or set of messages that you choose, however it can be enjoyable and more efficient to group your NPCs by type. Having a town filled with NPC's randomly choosing a message from a group of 10 or 20 one-line messages is easier to use and manage, especially if they are handing out clues to quests! Let's create an NPC that will say a random message when they are spoken to by an adventurer.

In this tutorial, we want our NPC to say something during the conversation only and choose one from three possible options.

The first thing we need to do is enter the banter messages into our database. To do this, simply run the Legends Banter Plugin from the plugins menu and then choose File -> Banter.

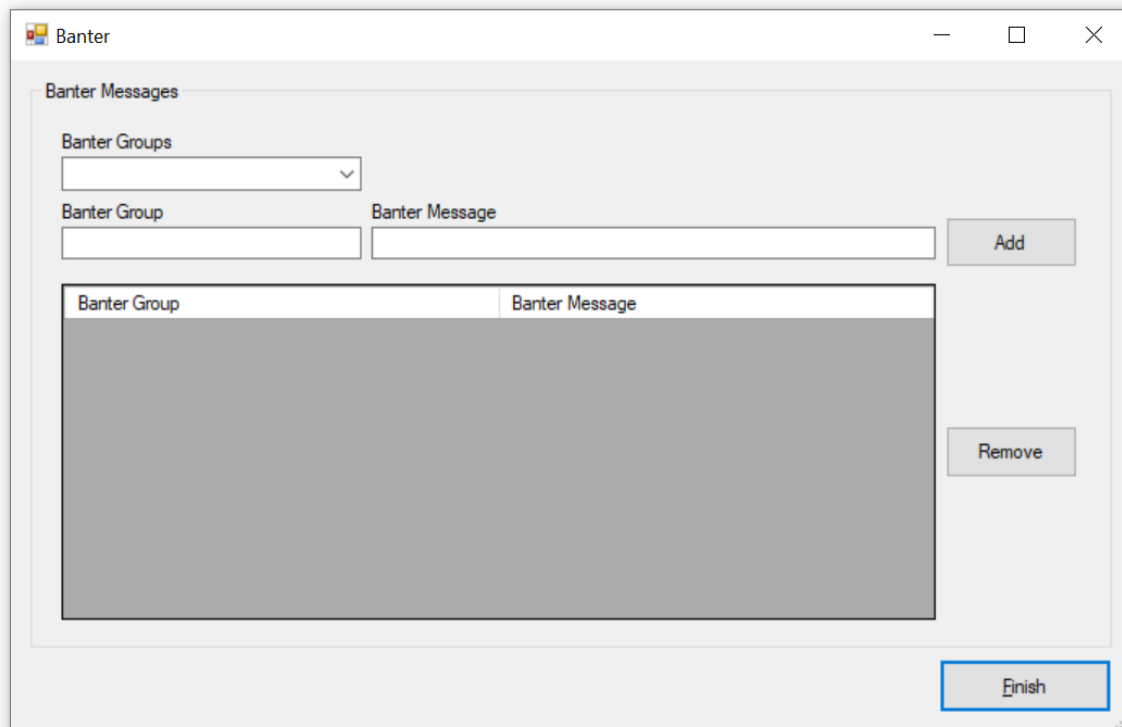


Figure 7 - The Banter Message Screen

This screen is where you will manage all the possible banter messages that NPC's and creatures can say. Banter messages are divided into groups and then those groups are assigned to NPC events. There can be one message in a group or there can be many. The choice is yours. For this tutorial, we want our NPC to say one of 3 possible messages when we click on them to start a conversation.

In the Banter Group textbox, enter "villager" and in Banter Message enter "Hello there adventurer." Then click add. Repeat this process and using the word villager in both instances, add the following messages:

- It's a fine time to be alive!
- Hello there adventurer.
- Have you met the crazy wizard in this village?

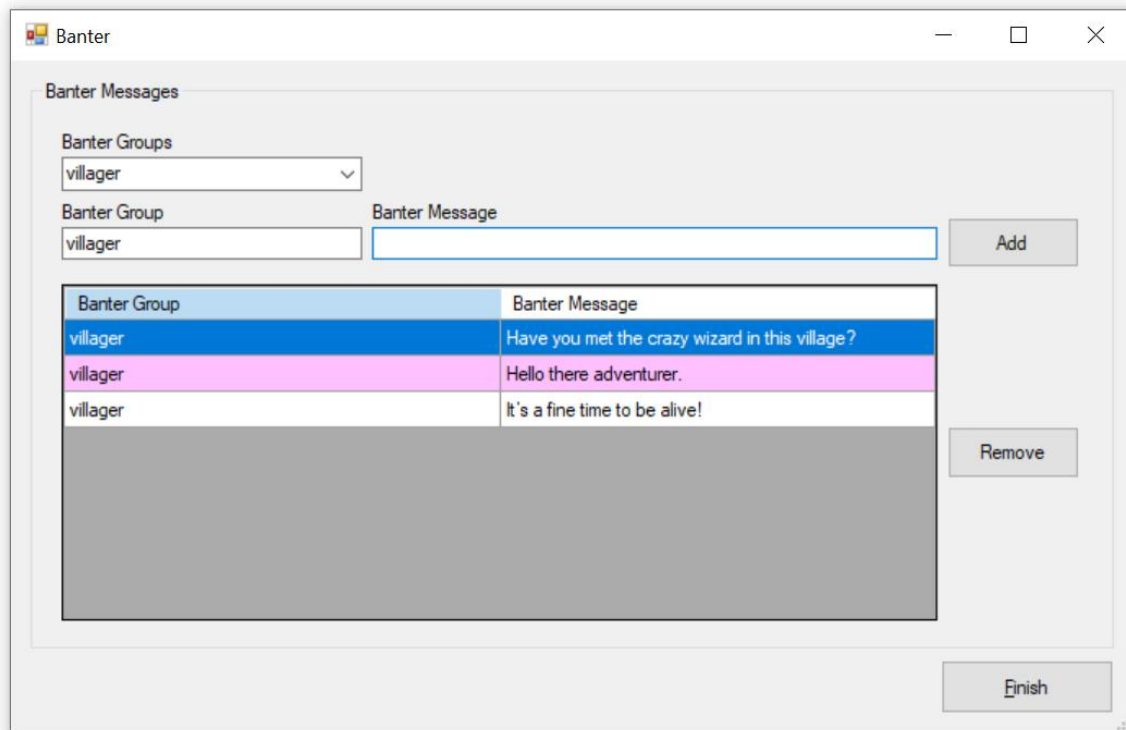


Figure 8 - Tutorial #1 Messages

Your banter message screen should look something like the above. Click Finish. Next, create an NPC or copy one from the existing stock blueprints and ensure the blueprint is selected. Load up the Banter plugin again and select File -> Edit -> Creature Blueprint.

The following screen shows the available Banter events. Additional choices may appear when optional Legends Spawn and/or Legends AI integrations are installed. Core Banter configuration does not require either plugin.

The screenshot shows the 'Edit Creature' window with the following sections and controls:

- Banter Events**
 - Standard Conversation**
 - On Conversation: A dropdown menu with 'villager' selected.
 - Perceptions**
 - On Perception See: A dropdown menu.
 - On Perception Hear Only: A dropdown menu. Chance:
 - On Perception Leave: A dropdown menu.
 - Time Based**
 - On Day Arrival: A dropdown menu. Chance:
 - On Night Arrival: A dropdown menu. Chance:
 - AI**
 - On Death: A dropdown menu. Chance:
 - On AI Range: A dropdown menu. Chance:
 - On Sleep: A dropdown menu. Chance:
 - On Wake: A dropdown menu. Chance:
 - Despawn**
 - On Despawn: A dropdown menu. Chance:
 - On Forced Despawn: A dropdown menu.
 - Combat**
 - End of Combat Round: A dropdown menu. Chance:

At the bottom are 'Cancel' and 'Finish' buttons.

Figure 9 - Edit Creature Events

For this tutorial, pull down the On Conversation combo box and select villager, then click Finish. This will assign any messages in the “villager” group that we created in the previous step to this NPC’s On Conversation event. NOTE that if this creature blueprint has an existing conversation, it will be replaced with the Legends Banter conversation. The original conversation will still exist, just not used here.

Finally, place the NPC in a test area and compile the imported scripts in the Toolset.

That's it! Don't forget to bake your area. Spawn the NPC using your normal method (or use the Legends Spawn Plugin), then log in and try talking to your new villager.

Seems simple, doesn't it? Let's move things along to more interesting messages the NPC can say.

Tutorial 2 – Greetings Adventur-ling

So, you've got an NPC that speaks when spoken to, but how about having the NPC say something when they "see" the adventurer or perhaps you would like to sneak up on them but they happen to "hear" you coming? This is very easy to do by just adding some more entries in your database and assigning the groups to your NPC.

Using the Banter plugin, choose File -> Banter once again and enter the following:

Banter Group: villagerpercep
Message: Well Met!

Banter Group: villagerpercep
Message: Another adventurer blesses us with their presence.

Banter Group: villagerpercepheard
Message: I must be hearing things.

Banter Group: villagerpercepheard
Message: Did you hear something?

When done, you can delete any text in the “Banter Groups” combo box to show all the banter entries in the database. It should look something like this:

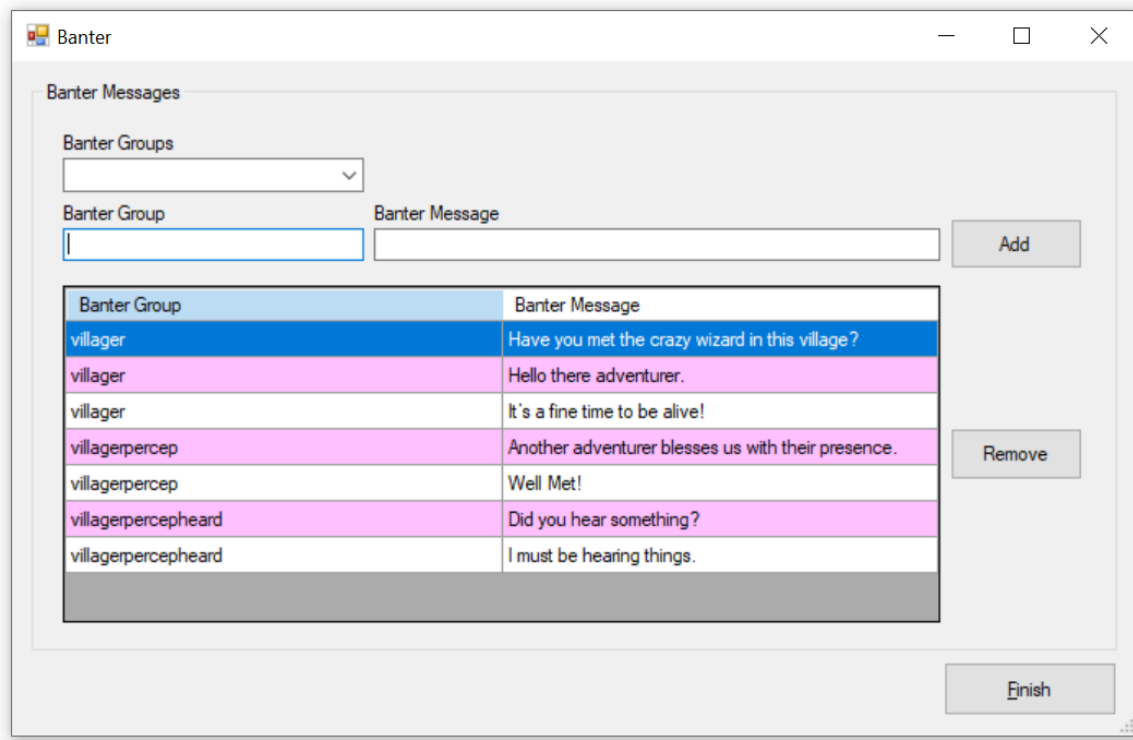


Figure 10 - Banter One Liners

Click Finish and quit the plugin. Select your NPC blueprint again and fire up the Banter plugin. Choose File -> Edit -> Creature Blueprint. In the “On Perception See” event choose the villagerpercep group, and in the “On Perception Hear Only” event, choose the villagerpercepheard group. Set the percentage “Chance” value to 100 for this demonstration.

⚠ NOTE: Normally this chance value is used during all perception events and allows us to configure a chance that the NPC may not say anything at all when they perceive an adventurer. Setting this to a low number when an adventurer wanders into a camp of kobolds, hidden, provides a nice effect should all of them hear the PC. We don't want 5 kobolds riddling off “Did you hear something?”

Edit Creature

Banter Events

Standard Conversation

On Conversation: villager

Perceptions

On Perception See: villagerpercep

On Perception Hear Only: villagerpercepheard Chance: 100

On Perception Leave:

Time Based

On Day Arrival: Chance:

On Night Arrival: Chance:

AI

On Death: Chance:

On AI Range: Chance:

On Sleep: Chance:

On Wake: Chance:

Despawn

On Despawn: Chance:

On Forced Despawn: Chance:

Combat

End of Combat Round: Chance:

Cancel Finish

Figure 11 - Perception Events

There is one other option we can apply to our NPC for perception-related events and that is when the adventurer disappears or when they leave the NPC's perception. Not used as often as the others but still can offer a nifty effect for townsfolk to say "Goodbye" or monsters to say, "Where did they go?" This is the On Perception Leave event, but we'll not use this one for this tutorial.

Finally, click Finish and quit the plugin. Re-place your NPC on the game map and if you've placed them far enough away and have your hide skill at a level where your NPC has a chance to use their Listen against you, there may be a chance you will see them speak their "hear" message. Fire up your tutorial server using NWNX and give it a try.

Now we have come to the completion of Tutorial #2. We have an NPC that will speak when spoken to, say random messages when they see or hear an adventurer and even have a chance to just say nothing at all. Fabulous – but not quite complete. Sometimes the NPC may just want to throw a comment out there when it's time for bed, or when it's time to milk the cows in the morning. This can be done as well with some extra table entries and a couple of new variables.

Tutorial 3 – What Time Is It?

Sometimes it can be fun to have our NPC's say things when night arrives or when the day begins. This can be easily accomplished with some more table entries and events set on them. Also, leveraging the delay time from the installation portion of this document, we can ensure that all the NPCs that are configured to do this and don't speak at the same moment. Some of them can have daytime messages and others have only nighttime messages or both.

Let's configure our NPC to have both a daytime message and a nighttime message. Add the following banter entries:

Banter Group: villagerday

Message: I guess it is time to milk the cows.

Banter Group: villagerday

Message: It is a lovely morning for a walk.

Banter Group: villagernight

Message: I'm afraid of the dark.

Banter Group: villagernight

Message: Time to hit the old pub!

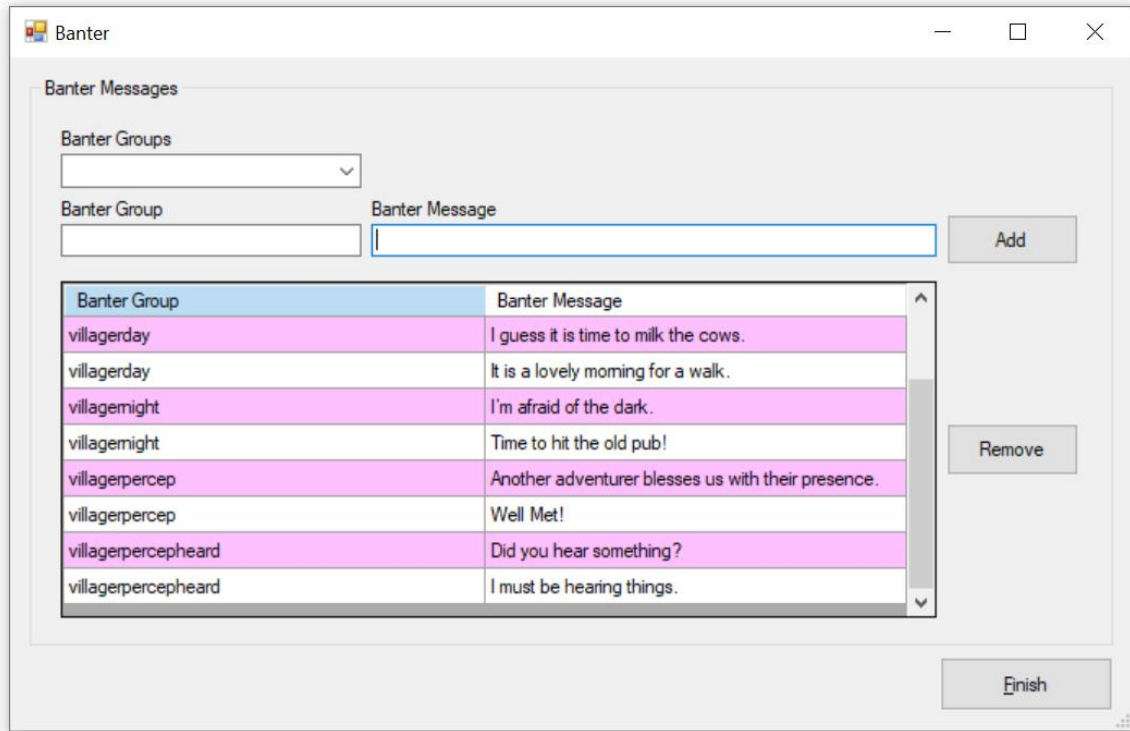


Figure 12 - Day / Night Banter

Assign these two groups to our NPC's Day arrival and Night arrival events. Don't forget that because we are editing creature blueprints, we'll have to re-spawn the NPC on our map with these changes.

- ⚠ Remember, there is a random delay of up to the number of seconds specified during the initial configuration before the NPC will speak once the appropriate time arrives so that a group of them will not attempt to speak at the same moment.

Fire up your tutorial server once again with NWNX and give it a try. You'll have to wait until the appropriate time passes before they will say anything of course.

Tutorial 4 – Any Last Words?

Of course, our banter plugin not only works on NPCs but also on monsters too. There's nothing more fun than hearing the final words of the dying kobold or the curses of the Chief of the Orcs as he falls. Adding the ability for our monsters to do this is simple just as in the other tutorials.

Create a monster of your choosing in the same method by creating something new or copying a stock creature. Add these new entries to your banter database:

Banter Group: monstersee
Message: Time to die!

Banter Group: monstersee
Message: You will not defeat me!

Banter Group: monsterhear
Message: Who goes there?

Banter Group: monsterhear
Message: What was that?

Banter Group: monsterdeath
Message: You have not heard the last from me!

Banter Group: monsterdeath
Message: The end is near, I can see a light.

Banter Group: monsterdeath
Message: I've fallen and I can't get up!

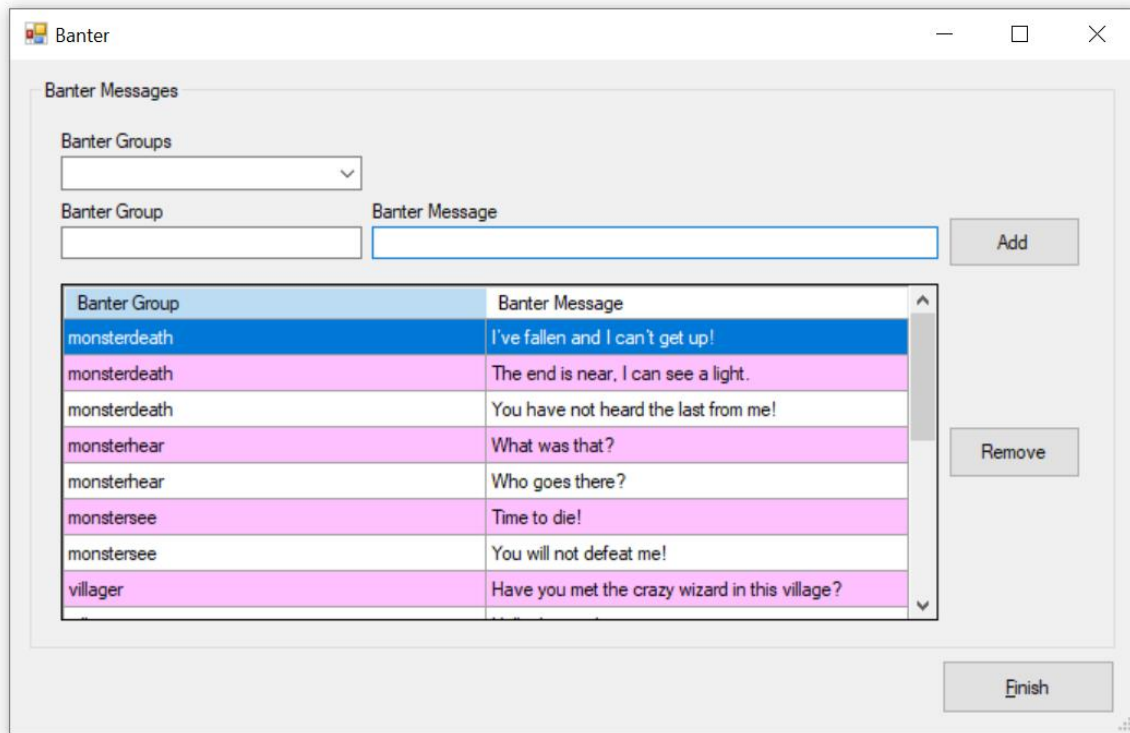


Figure 13 - Monster Banter

Assign these groups to the perception see, hear and the On Death event. For the Perception related events, enter 50 in the Chance field and 100 in the Death Chance field.

The perception-related entries which we have discussed previously allow our monster to say interesting things when they discover the adventurer, but they may not always say anything based upon the chance variable we have placed.

Edit Creature

Banter Events

Standard Conversation

On Conversation:

Perceptions

On Perception See: Chance:

On Perception Hear Only: Chance:

On Perception Leave:

Time Based

On Day Arrival: Chance:

On Night Arrival: Chance:

AI

On Death: Chance:

On AI Range: Chance:

On Sleep: Chance:

On Wake: Chance:

Despawn

On Despawn: Chance:

On Forced Despawn: Chance:

Combat

End of Combat Round: Chance:

Figure 14 - Monster Configuration

Now we have an interesting monster that's a little more fun than just grumbling, biting, nasty. Fire up the tutorial module with NWNX once again and run into battle!

Tutorial 5 – Talkers!

A new feature that was introduced in the Legends Banter plugin as of version 1.8 is conversations. With this feature, NPCs (even placeable objects) can now have text-based conversations with one another, or even a group.

It works in a similar fashion to Banter where you first create a conversation with timers and then set the NPC's tags to match the actors in the conversation. Place them in your game world and surround them with a special Conversation trigger found in the Triggers blueprints under Legends Banter. Once placed, select it and run the Banter Plugin on it and edit the trigger.

To demonstrate, let's assume we are going to have 2 NPC's having a chat about undercutting the competition. First, we'll open the Banter plugin and select File -> Conversations and add some lines that they will speak.

The conversation ID is the unique identifier for this conversation. The Message is what is spoken, the Object Tag is the NPC or placeable object that the message appears to come from, the Delay Timer is the number of seconds from the START of the conversation till when the message is spoken. It is important to note that the Delay Timer is NOT the number of seconds after the previous message is spoken, but the number since the start. Enter in the following:

Conversation ID: beachmerchants1
Message: Ssh, did you hear something?
Object Tag: merchant1
Delay Timer: 1

Conversation ID: beachmerchants1
Message: No, I didn't hear a thing.
Object Tag: merchant2
Delay Timer: 5

Conversation ID: beachmerchants1
Message: Never mind then, what's the plan again?
Object Tag: merchant1
Delay Timer: 8

Conversation ID: beachmerchants1
Message: We're going to collect ALL the seashells and sell them for cheap!
Object Tag: merchant2
Delay Timer: 13

Conversation ID: beachmerchants1
Message: Yes! Now I remember, we'll put everyone else out of business!
Object Tag: merchant1
Delay Timer: 17

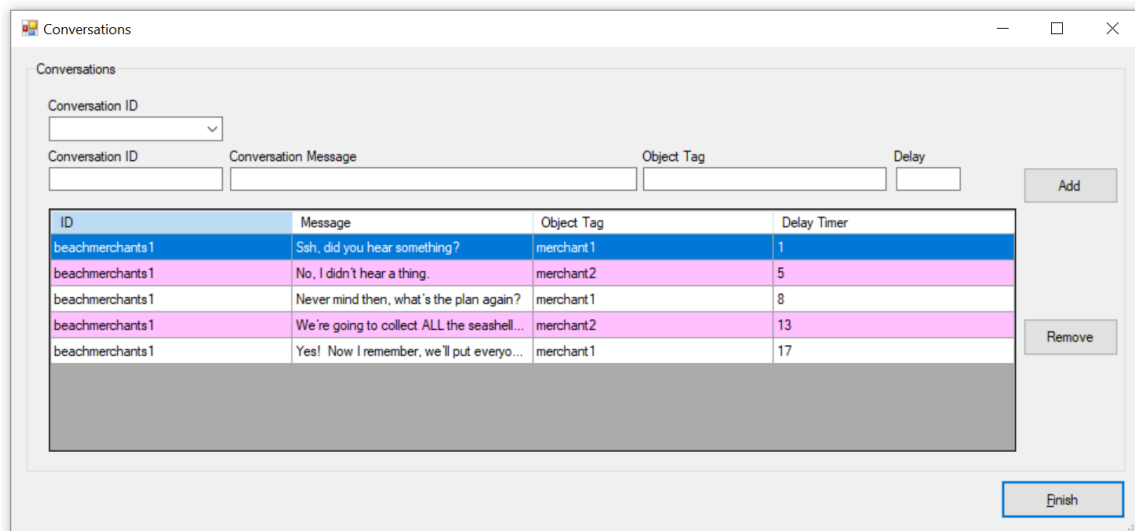


Figure 15 - Conversation Lines

Conversation ID

This is a unique ID for the conversation being edited. You can use the pulldown to select an existing one or just enter a new one in the field provided to create a new conversation.

Conversation Message

This is the line that each actor will speak in the conversation.

Object Tag

This is the Tag of the associated NPC or placeable object that will be speaking this line of the conversation.

Delay

This is the value in seconds that will elapse from the START of the conversation. This number is simply how many seconds must pass from when the adventurer entered the trigger should this line be spoken.

Once the conversation is complete, place the NPC/Object actors defined in your conversation in your game world. Ensure each one has the correct and matching TAG for the conversation line they must speak. Then select the Conversation Trigger from your blueprints in the Legends Banter category where the conversation should begin when an adventurer enters it.

Once the trigger has been placed, select it in the game area and run the Banter plugin again. Select File -> Edit -> Trigger Instance

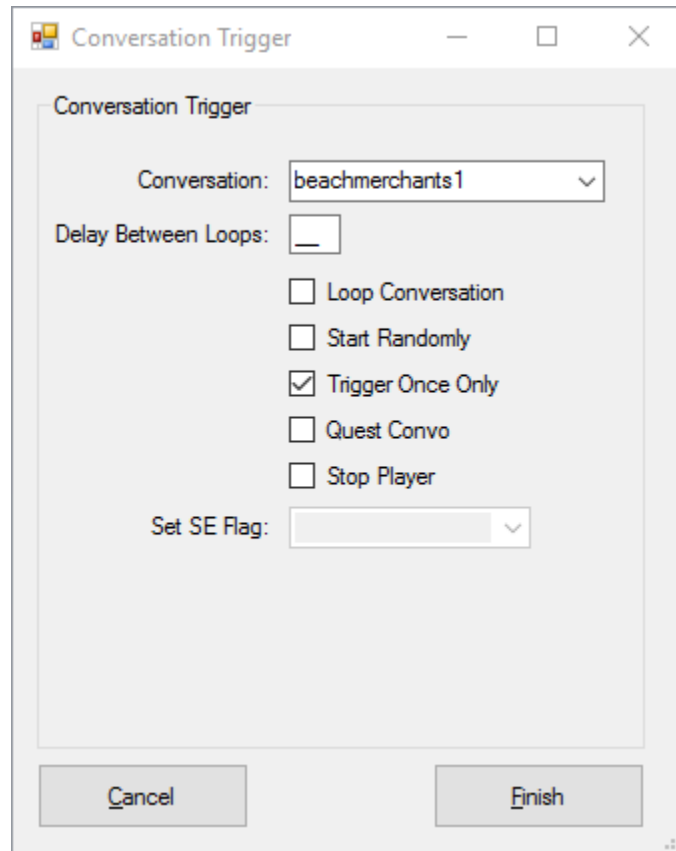


Figure 16 - Conversation Trigger

Use the pull-down box to select the conversation you've created to assign to this trigger and then enter some extra options as described below:

Delay Between Loops

This is the number of seconds to pause the conversation when it reaches the end before it should start again. This only applies if the Loop Conversation checkbox is selected.

Loop Conversation

When selected, the conversation will begin and then continue to loop until the actors are despawned or killed.

Start Randomly

When this is selected, the conversation will not begin at the first line. Instead, it will randomly select a location in the conversation and start it there. This is useful to simulate an adventurer coming along and hearing a conversation already in progress. It is usually valuable, however, to have these types of conversations loop so the adventurer can hear the entire context.

Trigger Once Only

This checkbox causes the player to be able to trigger the conversation once and once only. Re-Entering the conversation trigger does not re-fire the conversation.

Quest Convo

If this is selected then the conversation will only fire if a player is on a quest from the Legends Quest plugin. This is controlled by the Quest Plugin and causes the Banter and Quest plugin to link.

Stop Player

If this checkbox is selected, it will cause the player to freeze until the conversation is over. This is useful if you want to force a player to not move and listen to the full conversation, otherwise the player is free to move about.

Set SE Flag

Once this conversation has triggered and completed, an optional Special Effects Flag can be activated. The Special Effects plugin is a function of the Legends Quest plugin to perform actions based on events. See Legends Quest Plugin for further details. This option will not be available if the quest plugin is not detected.

With the trigger configured, NPCs placed and conversation created, it is simply a matter of entering the trigger in game and voila, the conversation will begin.

NOTE: Placeable objects must NOT be set to static otherwise their conversations lines will not appear.

Tutorial Summary

As you can see, the Banter Plugin can add substantial flavor to the world. The examples above cover the normal builder workflow; advanced custom scripting follows in Part 3.

Part 3 – Advanced Customization

For advanced users looking to get a little more out of the Banter system, it is quite easy to do in your existing scripts or new scripts that you may be interested in writing. Perhaps you'd like a placeable object to say something when a lever is thrown or a trigger is crossed? This is simple with a little bit of scripting knowledge.

Scripting

When your trigger is crossed or your lever is thrown, all you need to do is identify what object is doing the talking and what it should say. This can be done in a variety of ways, but a simple way is to have your lever or triggers have the TAG of the object as a variable, then script like so:

```
void main()
{
    object oObject = GetObjectByTag(GetLocalString(OBJECT_SELF, "Speaker"));
    string sBanterID = GetLocalString(oObject, "LEG_BANTER_ID");
    int iBanterChance = GetLocalInt(oObject, "LEG_BANTER_Chance");

    AddScriptParameterString(sBanterID);
    AddScriptParameterInt(iBanterChance);
    AddScriptParameterString("Custom");
    ExecuteScriptEnhanced("leg_banter_speak", oObject);
}
```

Then put a string variable on your trigger or lever called “Speaker” with the TAG of the placeable object you would like to have say something. A note regarding your placeable object – it should NOT be static, otherwise the system may not find it.

Final Summary

This system was originally designed for the NWN 2 Persistent World known as Legends. This quest-based world has so many different quests in it that sometimes a player needs extra help finding something or a clue to help move them on their way. I thought of the idea of having various wandering villagers make random comments that if an adventurer paid attention to, they just might learn where the kidnappers took the man you’re looking for, or where that hidden Zaranthi camp might be located.

For current support and project information, visit <https://nwn2legends.com>. Video walkthroughs may also provide useful background, but the current written instructions govern Banter 2.5.2 EE64.

I hope you enjoy this system and feel free to season it to your individual needs and tastes! Have Fun and Happy Bantering!

© 2026 Marshall Vyper. Legends Banter Plugin and documentation.
Neverwinter Nights 2 and related names and trademarks belong to their respective owners. Legends Banter is an independent community project and is not affiliated with or endorsed by those rights holders.