Dialogue System

How to use Dialogue System 2D for Unity

Dialogue System

The best way to get started with the Dialogue System is to look at the demo scenes. The following document helps to explain more in details how to create dialogues and quests. There are two demos included: a platformer, and a top-down game.

Getting Started

There are 3 things you will need to start creating dialogues:

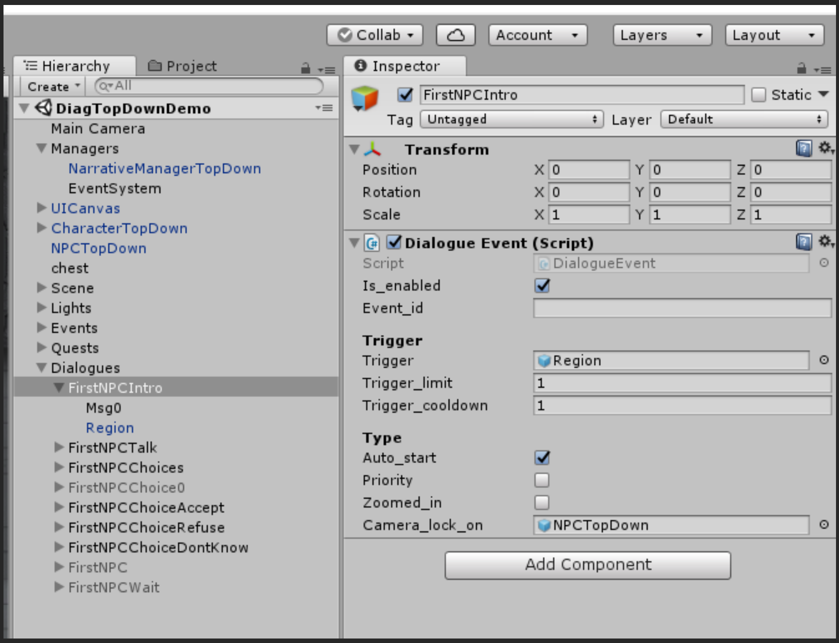
- The Narrative Manager
- A dialogue event with dialogues messages
- And a dialogue actor.

Narrative Manager

Must be included once in your scene when you want to use the dialogue system. It can be found in the prefabs folder. There's a different prefab for top down game, the only difference is that it uses a different talk bubble.

Dialogue Events

A dialogue event is a group of messages that is triggered to start a dialogue. It can contain one or more dialogue messages. You can define dialogues in your scene by creating an empty object and adding the DialogueEvent.cs script to it. That object should also have child objects with the DialogueMessage.cs script.



```
DiagTopDownDemo
  Main Camera
  Managers
    NarrativeManagerTopDown
    EventSystem
  UICanvas
  CharacterTopDown
  NPCTopDown
  chest
  Scene
  Lights
  Events
  Quests
  Dialogues
    FirstNPCIntro
      Msg0
      Region
    FirstNPCTalk
    FirstNPCChoices
    FirstNPCChoice0
    FirstNPCChoiceAccept
    FirstNPCChoiceRefuse
    FirstNPCChoiceDontKnow
    FirstNPC
    FirstNPCWait
```

```
FirstNPCIntro                         Static
Tag Untagged          Layer Default

Transform
Position     X 0   Y 0   Z 0
Rotation     X 0   Y 0   Z 0
Scale        X 1   Y 1   Z 1

Dialogue Event (Script)
Script             DialogueEvent
Is_enabled         ✓
Event_id

Trigger
Trigger            Region
Trigger_limit      1
Trigger_cooldown   1

Type
Auto_start         ✓
Priority           ☐
Zoomed_in          ☐
Camera_lock_on     NPCTopDown
```

Event_id is an optional ID reference to the event when saved to file.

Trigger Defines what will start the event (Region or Actor)

Auto_start will start the dialogue without pressing the "Talk" button.

Priority will prioritize this dialogue over other dialogues.

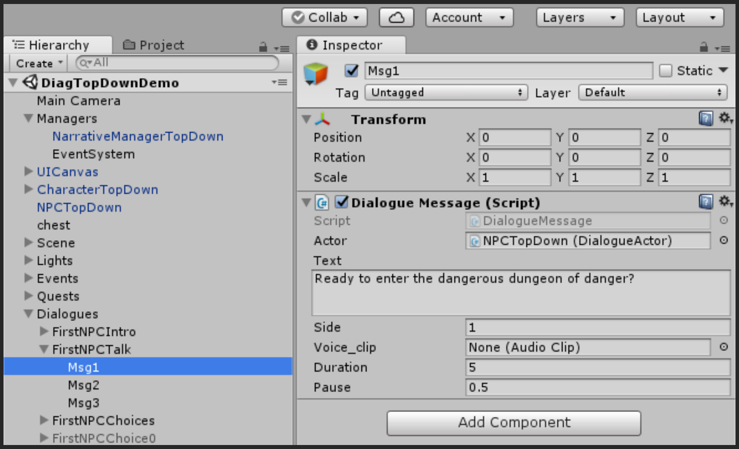
Zoomed_in will affect which UI is used to display the dialogue. It will also stop the gameplay and show a zoomed in version of the characters.

When “Trigger” is set to a region, the dialogue will start when the player enter that region. When “Trigger” is set to an actor, it will start when the character goes near that actor. If auto_start is off, the character will also need to press “Enter” when near the actor.

Trigger limit will determine how many times that event can be triggered, with **0 meaning infinite**. If the event has an event_id the trigger count will be saved into the data that can be accessed with NarrativeData.Get() and serialized or added to the save file of your choice. (optional if you want to create a save system on top of this dialogue system).

Dialogue Messages

Each dialogue event must contain one or more dialogue messages as child objects.



```
▼ DiagTopDownDemo
    Main Camera
  ▼ Managers
      NarrativeManagerTopDown
      EventSystem
  ▶ UICanvas
  ▶ CharacterTopDown
    NPCTopDown
    chest
  ▶ Scene
  ▶ Lights
  ▶ Events
  ▶ Quests
  ▼ Dialogues
    ▶ FirstNPCIntro
    ▼ FirstNPCTalk
        Msg1
        Msg2
        Msg3
    ▶ FirstNPCChoices
    ▶ FirstNPCChoice0
```

```
Msg1                     Static
Tag  Untagged            Layer  Default

Transform
Position   X 0   Y 0   Z 0
Rotation   X 0   Y 0   Z 0
Scale      X 1   Y 1   Z 1

Dialogue Message (Script)
Script       DialogueMessage
Actor        NPCTopDown (DialogueActor)
Text
Ready to enter the dangerous dungeon of danger?
Side         1
Voice_clip   None (Audio Clip)
Duration     5
Pause        0.5
```

Actor is the character that will tell that message. The character must have the DialogueActor script.

Side (1 or -1) is used to tell which side of the screen the character will appear when using the Zoomed In dialogue mode.

Voice clip is a voice over audio clip.

Duration: how long this dialogue will be shown (user can skip before that).

Pause: How long between this message and the next one.

Dialogue Actors

A character that can talk and that the main player can interact with. The player character must also have a DialogueActor.cs attached. But for the player, make sure “active” is off so it can’t talk to itself. It will just be used for the portrait and name.

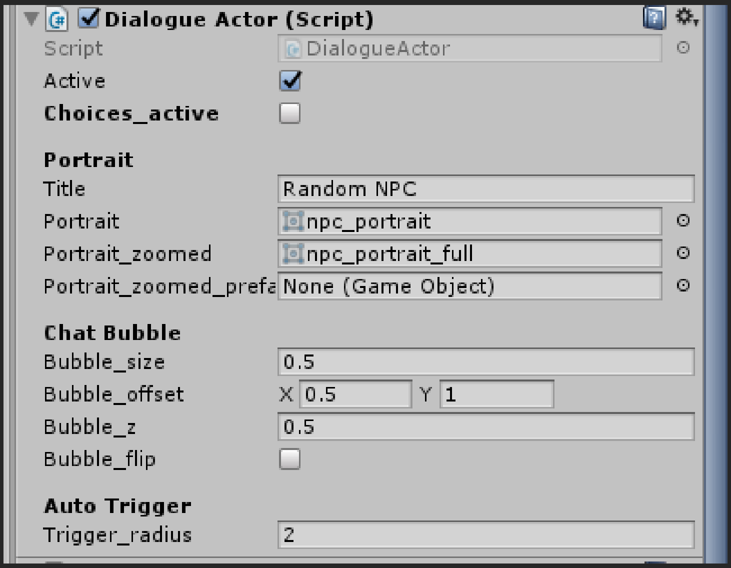
Active: means the player can interact with it.

Choices_active: that actor can also suggest dialogues choices.

Portrait: The way that actor is displayed during dialogues (title and image).

Chat bubble: how the chat bubble is displayed.

Trigger radius: The player needs to be in that range to interact with this actor.



```
Dialogue Actor (Script)
Script                   DialogueActor
Active                   ✓
Choices_active           ☐

Portrait
Title                    Random NPC
Portrait                 npc_portrait
Portrait_zoomed          npc_portrait_full
Portrait_zoomed_prefa    None (Game Object)

Chat Bubble
Bubble_size              0.5
Bubble_offset            X 0.5    Y 1
Bubble_z                 0.5
Bubble_flip              ☐

Auto Trigger
Trigger_radius           2
```

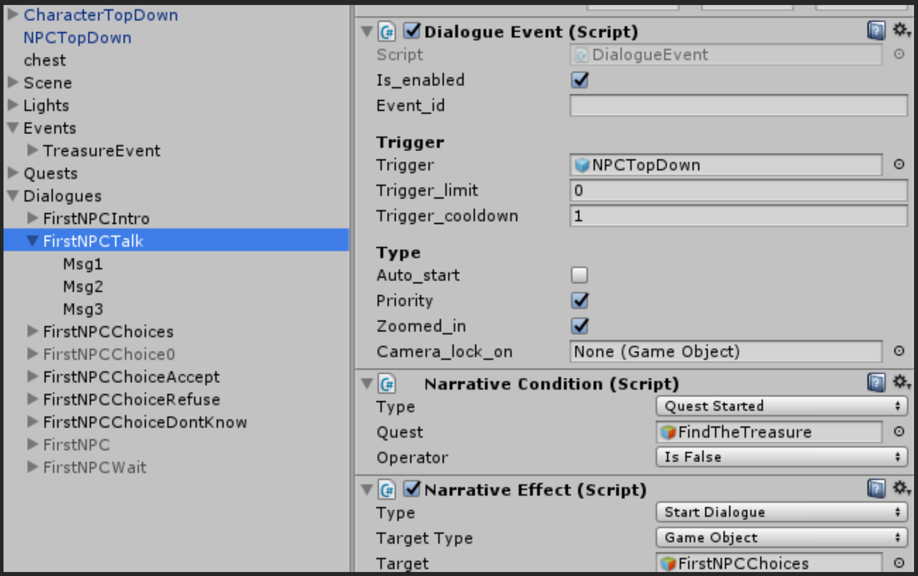
Narrative Events

A narrative event act similar to a dialogue event, but without showing any dialogues. The use of it is that you can trigger some custom effects and functions. For example you could use it to complete a quest when a character takes an item. That event would not trigger any dialogue but it would trigger custom effects. See the next section about conditions and effects for more info.

Narrative Conditions and Effects

Optionally, conditions and effects can be added to DialogueEvents or NarrativeEvents. They must be added to the same GameObject than the event (DialogueEvent.cs or NarrativeEvent.cs). More than one conditions or effects can be added to the same event.

NarrativeConditions will prevent the event from triggering unless the conditions are true. NarrativeEffect will call custom functions after the event is triggered (so you could start a quest as the result of a dialogue ending). Effects on a DialogueEvent will only trigger after all the dialogues messages are finished. Effects can be added directly to the message if you want the effect to trigger at that message (but not conditions).



```
▶ CharacterTopDown
  NPCTopDown
  chest
▶ Scene
▶ Lights
▼ Events
  ▶ TreasureEvent
▶ Quests
▼ Dialogues
  ▶ FirstNPCIntro
  ▼ FirstNPCTalk
      Msg1
      Msg2
      Msg3
  ▶ FirstNPCChoices
  ▶ FirstNPCChoice0
  ▶ FirstNPCChoiceAccept
  ▶ FirstNPCChoiceRefuse
  ▶ FirstNPCChoiceDontKnow
  ▶ FirstNPC
  ▶ FirstNPCWait
```

```
▼ ✓ Dialogue Event (Script)
  Script              DialogueEvent
  Is_enabled          ✓
  Event_id
  Trigger
  Trigger             NPCTopDown
  Trigger_limit       0
  Trigger_cooldown    1
  Type
  Auto_start          ☐
  Priority            ✓
  Zoomed_in           ✓
  Camera_lock_on      None (Game Object)
▼ Narrative Condition (Script)
  Type                Quest Started
  Quest               FindTheTreasure
  Operator            Is False
▼ ✓ Narrative Effect (Script)
  Type                Start Dialogue
  Target Type         Game Object
  Target              FirstNPCChoices
```

In this example, the event will only trigger if the quest “FindTheTreasure” is not started (condition). And will start another dialogue after that one ends (effect).

Effects

The most useful effect type that you will want to use it the “Call Function” effect, which allow you to call any function from your script. But this dialogue system also comes with some “pre-built” effects, especially related to the Dialogue System:

Custom Int/Float/String allow you to save custom data for future conditions.
Show/Hide will SetActive or not an object
Spawn/Destroy will Instantiate or Destroy an object.
Start/Cancel/Complete Quest allow you to manage quests.
Start Dialogue can start another dialogue or a dialogue choice.
RunEvent or **RunEventIfMet** can start a narrative event. (first one ignoring conditions).
Wait can add a delay before the next effect.
CallFunction allow you to call any function of your choice.

Conditions

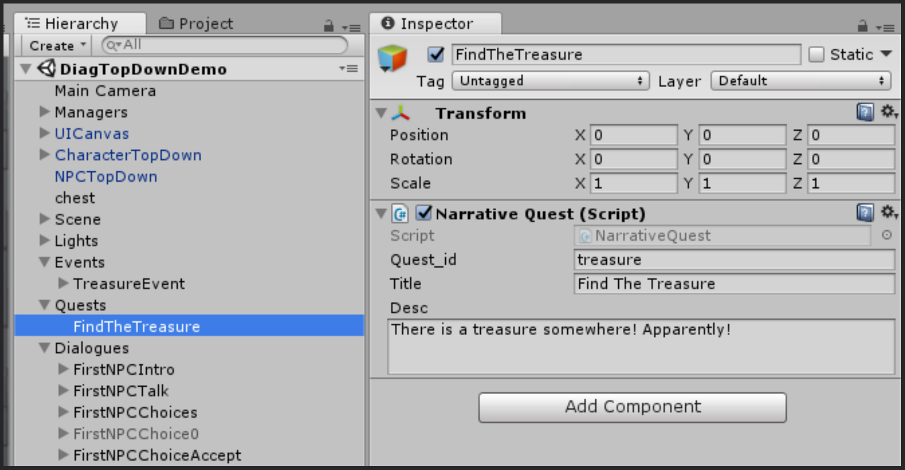
If more than one condition on the same event, all of them must be true for the event to trigger (AND). If you want different conditions to trigger the same dialogue (OR), its better to create more than one event.

Custom Int/Float/String will check the custom data set by other effects.
IsVisible will check if a gameObject exists and if its active.
Quest Started/Active will check quests status.
DialogueTriggered will check if another dialogue was triggered before.
CustomCondition Allow you to call any condition from your script. You will need to define the CustomCondition interface on your script for it to work.
Example:

```
using IndieMarc.DialogueSystem;
public class MyCondition : MonoBehavior, CustomCondition{
        public bool IsMet(){ /* --- Your Condition here --- */ }
 }
```

Quests

This dialogue system includes a simple quest system that can start a complete quests, and check the status of each quest. But it does not include a complex UI to manage quests. Its up to you how you want to access and display the quests saved in the system.



FindTheTreasure
Tag: Untagged Layer: Default
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Script: NarrativeQuest
Quest_id: treasure
Title: Find The Treasure
Desc: There is a treasure somewhere! Apparently!

Quest_id make sure all quests have a unique ID.

Effects can change the status of a quest from Inactive to Active to Completed or Failed.

Conditions can check the status of a quest.

Improving the System

If you notice a missing feature, and you think that it would be really helpful for you. Please let me know about it. I REALLY WANT to improve this system and make it great! And since I can’t predict all the use cases, your feedback would really help me know what I should include in the future versions.

If you have any questions or suggestions send me an email:
contact@indiemarc.com

Thank you!

Credits

Indie Marc (Marc-Antoine Desbiens)
Freelance Game Developer (Programmer & Game Designer)
indiemarc.com