

Dialogue system – Documentation technique

Le 05-06-2021 par Alexandre Venet <https://twitter.com/alhomepage>

1. Introduction

Ce fichier présente **objets** utilisés pour le système de dialogue de *Reliquia: The Immortality Stone*, ainsi que les éléments **exposés** (publics, privés sérialisés), ou utiles à la compréhension, des **scripts** utiles pour une intervention par programmation.

2. Scripts

Les scripts sont distribués dans des **dossiers** du projet à des fins de clarté :

- **Game** : ce qui relève des objets de la scène, les composants,
- **Interfaces** : les interfaces C# à implémenter,
- **ScriptableObjects** : les classes définissant les fichiers instances SO,
- **Types** : autres types utiles de structure de données.

Si certains types ne peuvent pas apparaître en *Inspector*, cela est mentionné au cas par cas.

Tous les scripts, hors *packages*, appartiennent à l'**espace de nom** « **AlexandreDialogues** ».

3. Editor Scripting

Les fichiers de mise en forme pour l'éditeur (*Editor Scripting*) se trouvent et doivent rester dans le dossier **Assets/Editor**. Ils concernent :

1. **DialogueManager**,
2. **DialogueRef**,
3. **Dialogue**,
4. **HelpFile**.

Le **menu** d'*Unity* a une entrée : **Tools/Reliquia**. Les sous-entrées sélectionnent les fichiers d'aide :

- **HelpFile** (fichier intégré à l'interface),
- **notice d'utilisation** (pdf),
- **documentation technique** (pdf).

4. Prefabs

Deux *prefabs* sont fournis :

1. **CharacterUIObject** : ensemble d'éléments d'UI pour un personnage,

2. **DialogueManager** : l'objet comprenant le gestionnaire de dialogue et tous les éléments d'UI relatifs au dialogue.

5. Game

5.1. Dialogue

5.1.1. DialogueManager

Classe
Accès : public
Héritage : MonoBehaviour
Singleton
Mis en forme par Editor Scripting : - EDialogueManager .

Propriété : Instance	
Accès : public	Type : DialogueManager
Modificateur : static	Accesseur (get), pas de mutateur (set)
Propriété permettant d'accéder aux autres membres de la classe, en tant que la classe est définie comme Singleton.	

Champ : _inventory	
Accès : private	Type : Inventory
Référence au fichier ScriptableObject d'inventaire.	

Champ : _defaultCamera	
Accès : private	Type : GameObject
Référence à une caméra de dialogue par défaut, caméra virtuelle <i>Cinemachine</i> .	

Champ : _canvas	
Accès : private	Type : GameObject
Référence au Canvas d'UI enfant.	

Champ : _textLastCharIdentifier	
Accès : private	Type : char
Caractère unique identifiant la fin du texte de réplique, styles inclus.	

Classe : UIObjects	
Accès : private	
Classe interne à DialogueManager pour définir l'ensemble des références suivantes.	
Champ : m_character	
Accès : public	Type : GameObject
Un objet d'UI de personnage.	
Champ : m_image	
Accès : public	Type : Image (UnityEngine.UI)
Le composant image de l'objet précédent et qui reçoit un <i>sprite</i> (voir Character).	
Champ : m_name	
Accès : public	Type : TextMeshProUGUI
Composant accueillant le texte du nom du personnage.	

Champ : _UICharacterLeft	
Accès : private	Type : UIObjects
Voir UIObjects .	

Champ : _UICharacterRight	
Accès : private	Type : UIObjects
Voir UIObjects .	

Champ : _replyText	
Accès : private	Type : TextMeshProUGUI
Le composant accueillant le texte de réplique.	

Champ : m_controlledScripts	
Accès : public	Type : List<MonoBehaviour>
Collection de scripts destinés à implémenter l'interface IEnableForDialogue . Son contenu est testé dans Awake() pour initialiser m_IEnableForDialogue .	

Champ : m_EnabledForDialogue	
Accès : public	Type : List<IEnabledForDialogue>
Collection de scripts implémentant l'interface IEnabledForDialogue . Elle est initialisée dans Awake() avec le contenu de m_controlledScripts qui implémentent effectivement l'interface. Cette collection est utilisée par DialogueEntry pour déclencher ou non EnableMe(bool value) .	

Champ : m_conditionsBools	
Accès : public	Type : List<bool>
Collection de booléens issus des méthodes personnelles utilisées comme condition de démarrage de dialogue dans une entrée de dialogue (type StartConditionsTestType.Method).	

Méthode : DialogueCheckStart(DialogueRef dialogueRef)
Type retourné : void
Méthode de vérification d'entrée de dialogue. Requiert une référence à un composant DialogueRef . Exemple d'appel : <i>DialogueManager.Instance.DialogueCheckStart(_dialogueRef);</i>

Méthode : StartDialogueFromFile(Dialogue dialogue, GameObject virtualCamera, bool useControlledScriptsAtStart, bool useControlledScriptsAtEnd)
Type retourné : void
Méthode de lancement d'un fichier de dialogue, indépendamment de la situation 3D ou de l'étape dans la narration. Requiert : <ul style="list-style-type: none"> - une référence de dialogue, - une référence de caméra virtuelle, - premier booléen : lance on non la fonction EnableMe(bool value) de la collection d'objets implémentant l'interface IEnabledForDialogue au démarrage du dialogue, - second booléen : idem pour la fin du dialogue. Exemple d'appel : <i>DialogueManager.Instance.StartDialogueFromFile(_dialogue, _virtualCamera, true, true);</i>

5.1.2.DialogueRef

Classe
Accès : public
Héritage : MonoBehaviour
Mise en forme par Editor Scripting : <ul style="list-style-type: none"> - EDialogueRef : paramètres généraux et ouverture de fenêtre, - EDialogueRefWindow : édition des entrées de dialogue, - EBaseWindow : classe de base de EDialogueRefWindow.

Champ : m_dialogueRefType

Accès : public	Type : DialogueRefType
Le type d'interaction choisi (voir DialogueRefType). Exemple d'utilisation : <i>if (_dialogueRef.m_dialogueRefType == DialogueRefType.InteractiveZone){//...}</i>	

Champ : m_levelProgress	
Accès : public	Type : LevelProgress
Référence au fichier ScriptableObject de progression. Voir LevelProgress .	

Champ : m_virtualCamera	
Accès : public	Type : GameObject
Référence à la caméra virtuelle pour cette entrée de dialogue. Peut rester null si aucune caméra n'est prévue.	

Champ : m_myDialogs	
Accès : public	Type : DialogueEntry[]
Collection d'entrées de dialogue. Voir DialogueEntry .	

6. Interfaces

6.1. IEnableForDialogue

Interface
Accès : public
Implémenter cette interface pour préparer un script de façon à l'inclure dans la collection DialogueManager.m_controlesScripts (et par suite DialogueManager.m_IEnableForDialogue) afin que sa méthode EnableMe(bool value) soit déclenchable au début et à la fin d'un dialogue.

Méthode : EnableMe(bool value)
Type retourné : void
Méthode appelée par DialogueManager avec : <ul style="list-style-type: none"> - <i>false</i> si DialogueEntry.m_useControlledScriptsAtStart <i>true</i>, - <i>true</i> si DialogueEntry.m_useControlledScriptsAtEnd <i>true</i>. Exemple d'utilisation :

```
public void EnableMe(bool value){ _isActive = value; }
```

7. ScriptableObjects

7.1. Character

Classe
Accès : public
Héritage : ScriptableObject
Entrée dans le menu de création d'asset : ScriptableObjects/Dialogue/Character.

Champ : m_spriteLeft	
Accès : public	Type : Sprite
Référence au <i>sprite</i> du personnage qui s’affiche à gauche de l’écran.	

Champ : m_spriteRight	
Accès : public	Type : Sprite
Référence au <i>sprite</i> du personnage qui s'affiche à droite de l'écran.	

7.2. Dialogue

Classe
Accès : public
Héritage : ScriptableObject
Entrée dans le menu de création d'asset : ScriptableObjects/Dialogue/Dialogue.
Mise en forme par Editor Scripting : <ul style="list-style-type: none"> - EDialogue : ouverture de fenêtre, - EdialogueWindow : édition du dialogue, - EBaseWindow : classe de base de EdialogueWindow .

Champ : m_test	
Accès : public	Type : string
Si ce champ est renseigné, alors le texte est affiché en console lors du déclenchement du dialogue.	

Champ : m_casting

Accès : public	Type : Role[]
Collection décrivant la distribution des rôles pour ce dialogue.	

Champ : m_replies	
Accès : public	Type : Reply[]
Collection de répliques pour ce dialogue.	

7.3. Inventory

Classe
Accès : public
Héritage : ScriptableObject
Entrée dans le menu de création d'asset : ScriptableObjects/Dialogue/Inventory

Champ : m_inventoryStrings	
Accès : public	Type : string
Collection de textes décrivant l'inventaire du personnage jouable.	

7.4. LevelProgress

Classe
Accès : public
Héritage : ScriptableObject
Entrée dans le menu de création d'asset : ScriptableObjects/Dialogue/LevelProgress

Champ : m_levelStep	
Accès : public	Type : int
Le niveau actuel de progression. Lors du chargement du fichier, cette variable prend la valeur de m_initValue .	

Champ : m_initValue	
Accès : public	Type : int

La valeur initiale, par défaut 0. Lors du chargement du fichier, cette valeur est affectée à **m_levelStep**.

8. Types

8.1. Dialogue

8.1.1. *DialogueColorsRef*

Classe

Accès : public

Champ : **ColorsHash**

Accès : public

Type : Dictionary<string, int>

Non affichable en *Inspector*.

Collection des champs **Name** et **HashCode** des styles de **TMPRO Reliquia StyleSheet**

8.1.2. *Reply*

Classe

Accès : public

Champ : **m_role**

Accès : public

Type : Role

Rôle utilisé pour la réplique courante.

Champ : **m_stageDirections**

Accès : public

Type : string

Chaîne de caractères des didascalies de réplique.

Champ : **m_styleColorHash**

Accès : public

Type : int

Valeur du **HashCode** d'un style de **TMPRO Reliquia StyleSheet**.

Champ : m_charSpeed	
Accès : public	Type : float
Temps d’affichage d’une lettre de dialogue. En éditeur, valeur limitée entre 0 et 1.	

Champ : m_spaceSpeed	
Accès : public	Type : float
Temps d’affichage d’un espace entre lettres de dialogue. En éditeur, valeur limitée entre 0 et 1.	

Champ : m_text	
Accès : public	Type : string
Le texte de la réplique.	

Champ : m_replyEvent	
Accès : public	Type : UnityEvent
L’événement de la réplique. N’est déclenché que s’il a des méthodes abonnées.	

8.1.3. Role

Classe
Accès : public

Champ : m_character	
Accès : public	Type : Character
Un personnage pour ce rôle.	

Champ : m_characterView	
Accès : public	Type : RoleCharacterView
Une valeur de point de vue. Voir RoleCharacterView .	

8.1.4. RoleCharacterView

Type : enum
Accès : public
Valeurs : <ul style="list-style-type: none"> - Left - Right

Décrit la position d'un personnage à l'écran de dialogue : à gauche ou à droite. Décrit également l'angle de vue pour sélectionner le *sprite* à afficher (par exemple : image du personnage vu de $\frac{3}{4}$ droite à gauche, image du personnage vu de $\frac{3}{4}$ gauche à droite).

8.2. DialogueEntry

8.2.1. DialogueEntry

Classe
Accès : public

Champ : m_name	
Accès : public	Type : string
Nom du dialogue.	

Champ : m_useControlledScriptsAtStart	
Accès : public	Type : bool
Si <i>true</i> , alors au début du dialogue DialogueManager lance la méthode EnableMe(false) de l'interface IEnableForDialogue .	

Champ : m_eventAtStart	
Accès : public	Type : UnityEvent
Un événement au démarrage du dialogue, qui ne se déclenche que s'il a des méthodes abonnées.	

Champ : m_dialogueSO	
Accès : public	Type : Dialogue
Référence à un fichier de dialogue pour cette entrée de dialogue.	

Champ : m_startconditions	
Accès : public	Type : StartConditionsTest[]
Collection des conditions de démarrage.	

Champ : m_dialogueAltSO	
Accès : public	Type : Dialogue
Dialogue alternatif dans le cas des conditions de démarrage Object ou Method .	

Champ : m_levelChangesAtEnd	
Accès : public	Type : bool
Si <i>true</i> , alors à la fin du dialogue, le niveau de LevelProgress change.	

Champ : m_nextLevel	
Accès : public	Type : int
Si m_levelChangesAtEnd est à <i>true</i> , alors LevelProgress prend cette valeur.	

Champ : m_useControlledScriptsAtEnd	
Accès : public	Type : bool
Si <i>true</i> , alors à la fin du dialogue DialogueManager lance la méthode EnableMe(true) de l'interface IEnableForDialogue .	

Champ : m_eventAtend	
Accès : public	Type : UnityEvent
Un événement à la fin du dialogue, qui ne se déclenche que s'il a des méthodes abonnées.	

8.3. DialogueRef

8.3.1. DialogueRefType

Type : enum
Accès : public
Valeurs : <ul style="list-style-type: none"> - InteractiveZone - TriggerZone
Le type d'interaction souhaitée avec un objet présentant le composant DialogueRef : <ul style="list-style-type: none"> - InteractiveZone : par entrée utilisateur, - TriggerZone : lancement automatique.

8.4. StartConditions

8.4.1. StartConditionsOperator

Type : enum
Accès : public
Valeurs :

<ul style="list-style-type: none"> - EqualTo - NotEqualTo - LessThan - LessThanOrEqualTo - GreaterThan - GreaterThanOrEqualTo
<p>Opérateurs de comparaison utilisés pour établir une condition de démarrage de dialogue de type Level. Correspondance syntaxique C# :</p> <ul style="list-style-type: none"> - == - != - < - <= - > - >=

8.4.2. StartConditionsTest

Classe
Accès : public

Champ : m_type	
Accès : public	Type : StartConditionsTestType
Le type de condition de démarrage utilisée pour ce dialogue.	

Champ : m_operator	
Accès : public	Type : StartConditionsOperator
L'opérateur de comparaison choisi pour un type de comparaison Level .	

Champ : m_level	
Accès : public	Type : int
La valeur sur laquelle effectuer la comparaison.	

Champ : m_object	
Accès : public	Type : string
Si la condition de démarrage est un objet , alors c'est le nom de cet objet.	

Champ : m_eventCondition	
Accès : public	Type : UnityEvent

Si la condition de démarrage est de type **Method**, ceci est l'événement qui se déclenche s'il a des méthodes abonnées.

8.4.3. *StartConditionsTestType*

Type : enum
Accès : public
Valeurs : <ul style="list-style-type: none"> - Level - Object - Method
Le type de condition de démarrage d'un dialogue. <ul style="list-style-type: none"> - Level : par niveau d'un LevelProgress, - Object : par présence d'un objet dans l'inventaire, - Method : par script dans une méthode personnalisée.