

Documentation utilisateur

IIViMaT

Créez de manière simple des interactions scénaristiques sur Unity à destination des casques de Réalité Virtuelle



1. INTRODUCTION

IIViMaT est un projet ayant pour but de faciliter l'accès à la technologie de la réalité virtuelle aux artistes et autres conteurs d'histoires.

À travers une interface sous forme de graphe, IIViMaT propose la création d'interactions simples entre l'utilisateur et son environnement virtuel.

Bien que IIViMaT soit un plugin principalement conçu pour la VR, une partie des interactions ne demandent pas nécessairement l'utilisation d'un casque car ces dernières sont relatives à la caméra. Il est donc tout à fait possible d'utiliser le plugin dans une application autre que VR.

INTRODUCTION	2
Glossaire	5
Installation du package IIViMaT	6
Via l'Asset Store	6
Via le fichier .package	6
Présentation de l'interface IIViMaT	7
L'arbre de projet	7
La hiérarchie des scènes IIViMaT	8
La scène IIViMaT	8
La scène Oculus IIViMaT	8
Création d'une nouvelle scène IIViMaT & le graphe IIViMaT	9
Les graphes IIViMaT	11
Les acteurs	12
Les actions	13
Les actions relatives à la tête	13
les actions relatives au corps	13
Les actions générales	13
Les actions de séquence	13
Les actions relatives aux mains [Expérimental]	14
Détail des actions	15
Les actions relatives à la tête	16
Les actions relatives au corps	18
Les actions générales	20
Les actions de séquence	21
Les actions relatives aux mains [Expérimental]	22

Les réactions	24
Les réaction d'apparences	24
Les réactions sur les lecteurs multimédias	24
Les réactions sur le Transform	24
Les réactions générales	25
Détail des réactions	26
Les réactions d'apparences	27
Les réactions sur les lecteurs multimédias	30
Les réactions sur le Transform	33
Les réactions générales	36
Exporter le projet vers un exécutable Oculus Quest 1 / 2	38
FAQ	41
Récapitulatif des objets IIViMaT	43
Récapitulatif des postures de main	45
Choix de la première version	46

2. Glossaire

- **Acteur** : Objet de l'environnement pouvant être relié à des actions
- **Action** : Acte de l'utilisateur pouvant être lié aux acteurs et réactions
- **Cible** : Éléments sur lesquelles agit la réaction
- **Réaction** : Effet pouvant être lié à une action
- **Signal** : Déclencheur lancée par une action aux réactions afin de les lancer
- **Telepad** : Marqueur visuel de téléportation avec les mains
- **Tête** : Caméra

Les termes liés à Unity sont présents en *Italique* dans ce document.

3. Installation du package IIViMaT

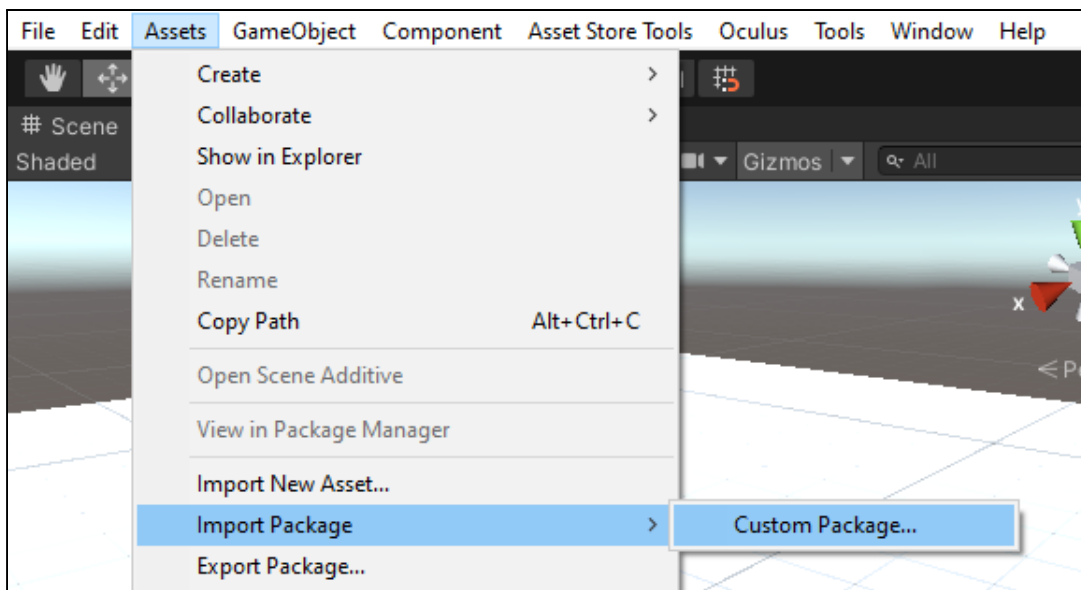
3.1. Via l'Asset Store

Le package IIViMaT peut être récupéré sur l'Asset Store en suivant ce lien ([LIEN ICI VERS L'ASSET STORE](#)). Cliquez sur “Télécharger” puis sur “Importer”.


3.2. Via le fichier .package

**** Sous Windows ****

- Cliquez sur “Assets” → “Import Package” → “Custom Package...”



- Sélectionnez le fichier .package

Nom	Modifié le	Type	Taille
 IIVIMAT	27/03/2021 22:07	Unity package file	584 583 Ko

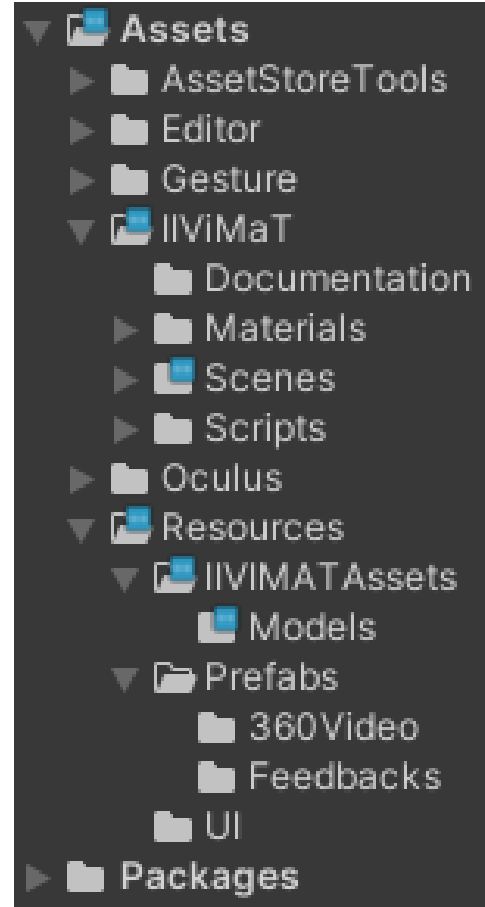
- Vérifiez que l'ensemble des fichiers est coché en sélectionnant “All” et importez en cliquant sur “Import”

4. Présentation de l'interface IIViMaT

4.1. L'arbre de projet

Le package IIViMaT est composé de plusieurs dossiers, nous avons :

- **Editor** : Contient les fichiers permettant de modifier l'éditeur (ajout automatique des scripts et création du graphe)
- **Gesture** : Contient les fichiers permettant de détecter les postures des mains
- **IIViMaT** : fichiers principaux
 - **Documentation** : documentation utilisateur
 - **Materials** : Matériaux et shaders utilisés
 - **Scenes** : Contient les scènes de démo et les Templates
 - **Script** : Scripts permettant le fonctionnement d'IIViMaT
- **Oculus** : Fichiers permettant la prise en charge du casque Oculus et plus particulièrement des mains
- **Resources** : Ressources
 - **IIViMaT Assets** :
 - **Models** : Graphe sous forme de fichier
 - **Prefabs** : Préfabriqués
 - **360Video** : Préfab pour les sphères vidéo 360
 - **Feedbacks** : Préfab pour les zones de proximité



4.2. La hiérarchie des scènes IIViMaT

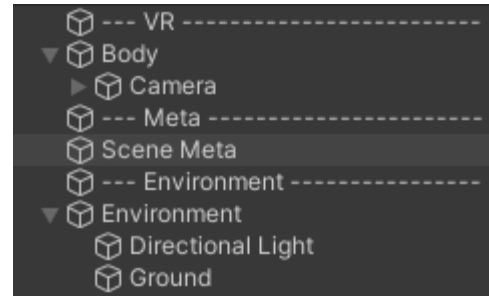
Nous allons présenter ici les objets présents dans les scènes par défauts d'IIViMaT. Il est nécessaire d'utiliser l'une de ces deux scènes afin de pouvoir utiliser les objets de la hiérarchie et les rendre interactifs. Nous fournissons deux scènes Template permettant de créer une scène IIViMaT :

4.2.1. La scène IIViMaT

La scène IIViMaT est la scène Template par défaut. Cette dernière permet d'utiliser l'ensemble des actions / réactions d'IIViMaT à l'exception de celles nécessitant les mains.

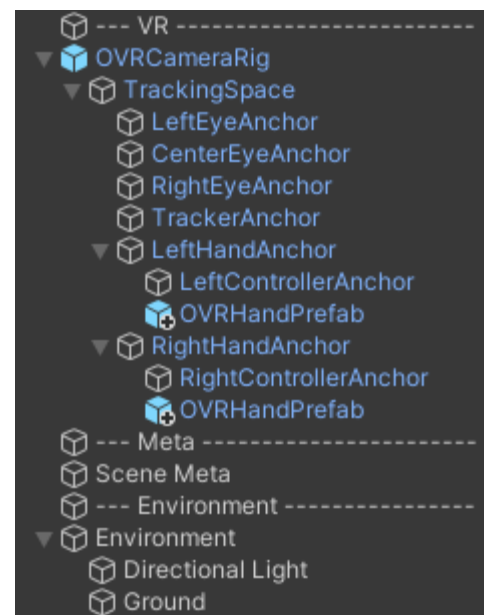
On retrouve dans la hiérarchie :

- **Body** : Représente le corps dans l'espace
 - Caméra : Représente la tête de l'utilisateur
- **Scene Meta** : Contient les scripts permettant le bon fonctionnement d'IIViMaT
- **Environment** : Contient l'ensemble des objets présents dans la scène ; les objets doivent être enfant de cet objet pour être utilisés en tant qu'acteur.



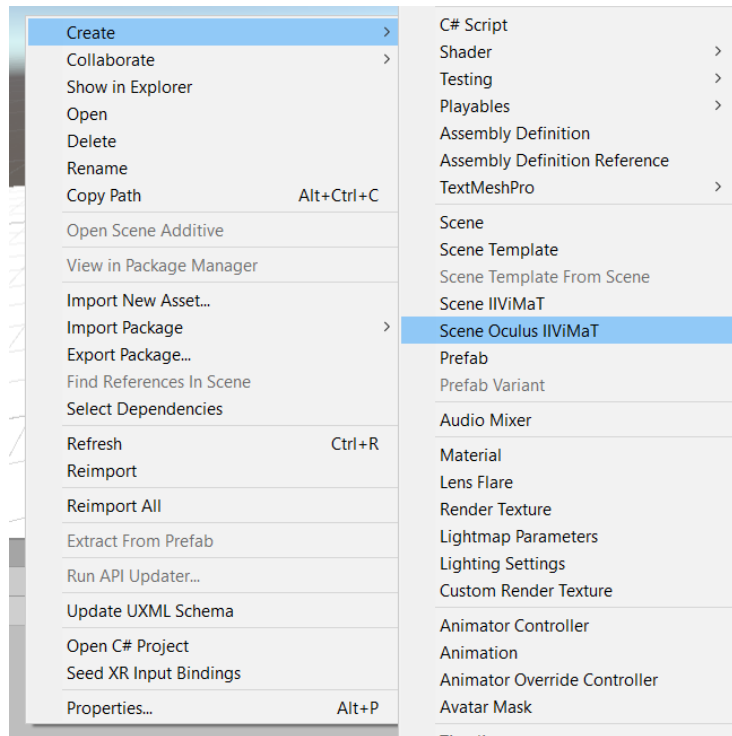
4.2.2. La scène Oculus IIViMaT

La scène IIViMaT est la seconde scène Template. Elle permet d'utiliser l'ensemble des actions et réactions d'IIViMaT avec la reconnaissance des postures des mains.

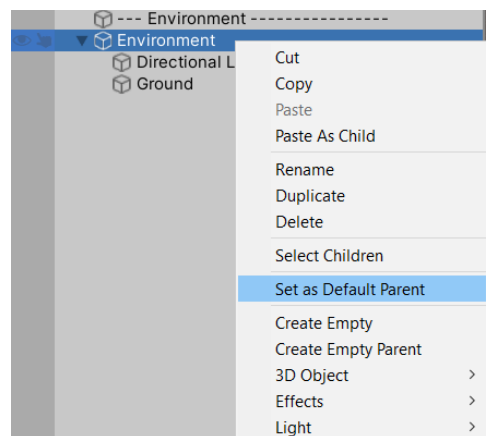


5. Création d'une nouvelle scène IIViMaT & le graphe IIViMaT

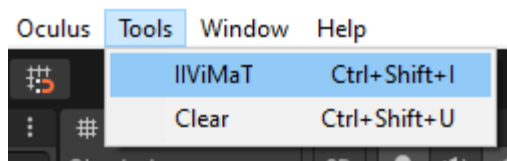
Pour créer une nouvelle scène, Cliquez droit dans votre dossier > Create > Scene IIViMaT” ou “Cliquez droit dans votre dossier > Create > Scene Oculus IIViMaT”



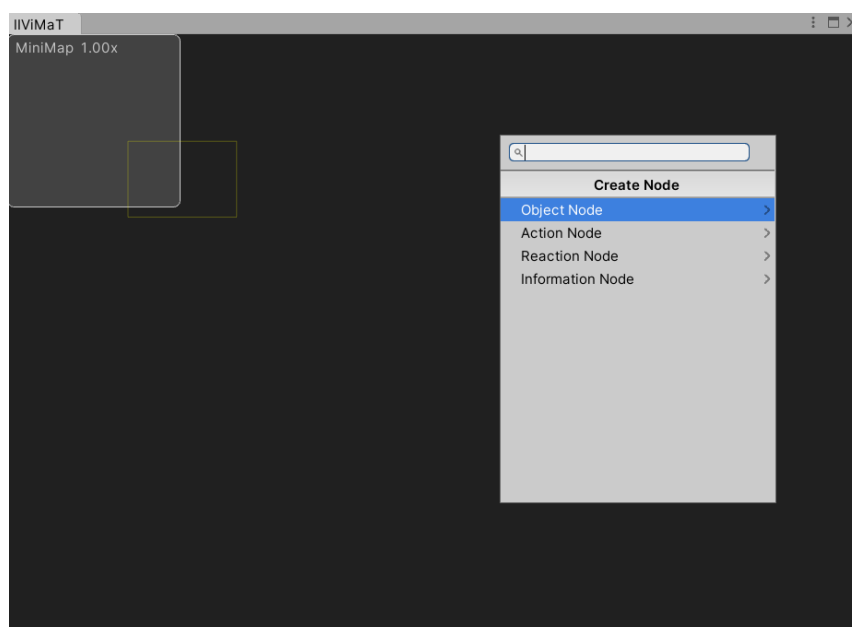
Afin que les différents objets puissent être reconnus pour être utilisés dans IIViMaT, ils doivent **obligatoirement** être enfant de Environment. Pour être sûr de ne pas vous tromper, Unity propose la fonctionnalité de définir un Parent par défaut lorsqu'un objet est créé dans la hiérarchie. Tout objet créé sera alors enfant de l'environnement. Environment apparaît alors en gras.



Pour ouvrir le graphe, cliquez sur “Tools > IIViMaT” ou faites le raccourci Ctrl + Shift + I. Le bouton “Clear” permet de supprimer l’ensemble des objets IIViMaT et de repartir d’un nouveau graphe.

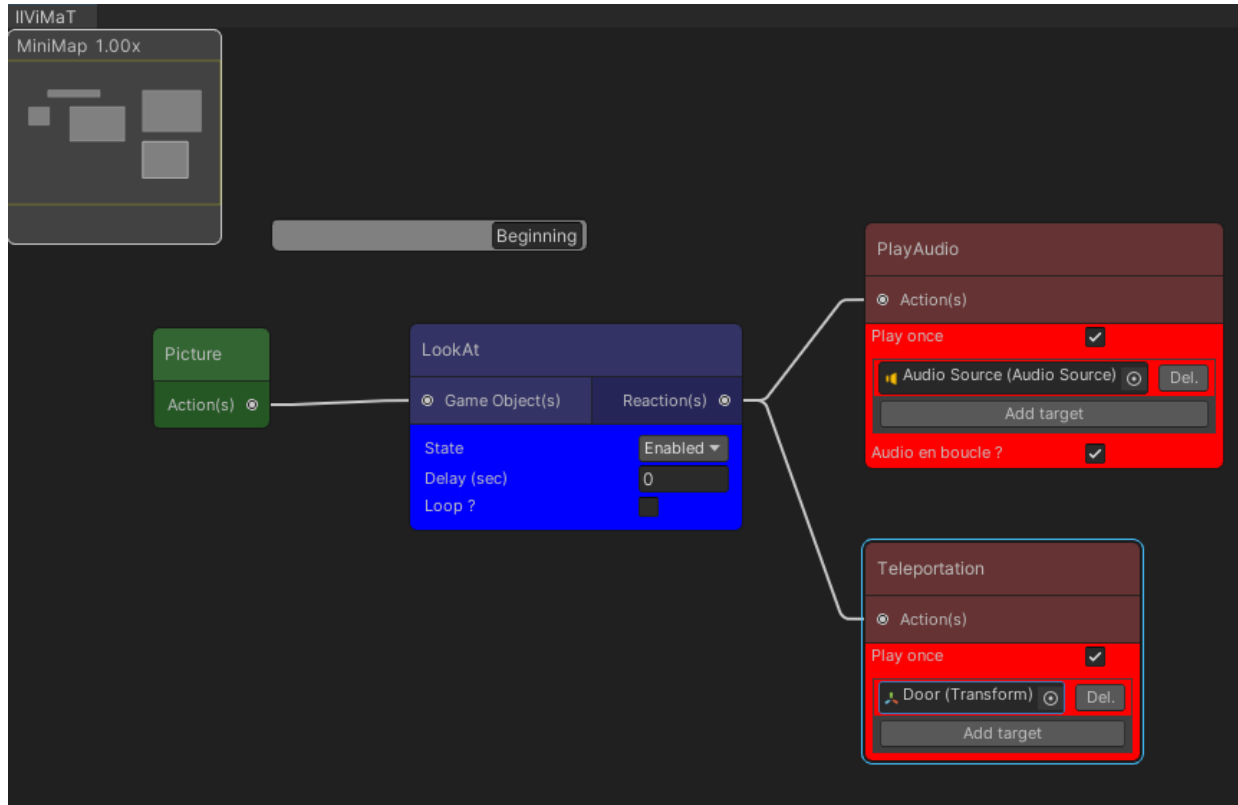


Si vous êtes bien dans une scène IIViMaT, la fenêtre de graphe s’ouvre. Afin d’ajouter des boîtes au graphe, faites un clic droit dans la fenêtre et choisissez “Create Node”.



6. Les graphes IIViMaT

IIViMaT tourne autour de trois types d'objets : les **acteurs**, les **actions** et les **réactions**.



Les **acteurs** vont déclencher les **actions** leur étant associées. Ces dernières vont elles-mêmes déclencher les **réactions** liées. Sur l'exemple précédent, nous pouvons lire ce graphe de la manière suivante :

“Lorsque je regarde le tableau, je joue la piste audio en boucle et je téléporte l'utilisateur à la position de l'objet porte”

Une **réaction** ne se déclenche que si une **action** vient l'activer. Pour ce qui est des **actions**, il en existe deux types : les **actions** classiques et les **actions** globales. Une **action** classique nécessite au minimum un **acteur** en entrée qui la déclenche. Les **actions** globales ne nécessitent pas d'**acteurs** en entrée et sont déclenchées d'une manière différente.

7. Les acteurs

Les acteurs sont les déclencheurs des actions. Pour l'heure, les seuls acteurs sont les *GameObject*. Pour certaines actions, il est nécessaire que ces acteurs possèdent soit un Video Player soit une Audio Source en tant que Component.

Pour qu'un *GameObject* soit éligible comme acteur, il doit **obligatoirement** se trouver dans l'Environment dans la hiérarchie de la scène. L'ensemble des objets présents dans Environment se voient attribuer automatiquement des scripts leur permettant de devenir acteur. Un acteur peut aussi être la cible d'une réaction.



8. Les actions

8.1. Les actions relatives à la tête

Les actions relatives à la tête sont liées à l'orientation et la position de la tête dans la scène. On y retrouve :

- **Proximity** : Action lancée lorsque la tête se trouve à proximité de l'acteur
- **Exit** : Action lancée lorsque la tête quitte la proximité de l'acteur
- **Look At** : Action lancée lorsque le regarde passe sur l'acteur
- **Look Away** : Action lancée lorsque le regarde quitte l'acteur

8.2. les actions relatives au corps

Les actions relatives au corps correspondent à la posture d'un utilisateur par rapport à un acteur. Il existe quatre actions pour la posture de l'utilisateur :

- **Stand Up** : Action lancée lorsque l'utilisateur est debout sur un acteur
- **Crouch** : Action lancée lorsque l'utilisateur est accroupi sur un acteur
- **Sit** : Action lancée lorsque l'utilisateur est assis sur un acteur
- **Lie Down** : Action lancée lorsque l'utilisateur est couché sur un acteur

La reconnaissance de la posture est basée sur la taille de l'utilisateur au moment du lancement de l'application. Il doit donc être debout lorsque l'application se lance.

8.3. Les actions générales

Les actions générales sont des actions qui ne nécessitent pas d'avoir des acteurs en entrée. Les actions des mains sont aussi des actions globales. On a :

- **Project Clock** : Action lancée après un temps donné depuis le début de la scène

8.4. Les actions de séquence

Les autres actions sont celles qui ne rentrent dans aucune des catégories précédentes. Nous avons :

- **On Video End** : Action lancée lorsque la vidéo présente sur l'acteur se termine
- **On Audio End** : Action lancée lorsque l'audio présent sur l'acteur se termine

8.5. Les actions relatives aux mains [Expérimental]

Les actions relatives aux mains sont liées aux mains de l'utilisateur. On retrouve ici la posture des mains ou leur mouvement dans l'espace. Nous avons :

- **HandGestureRight** : Action lancée lorsque la posture sélectionnée dans l'action est reconnue sur la main droite.
- **HandGestureLeft** : Action lancée lorsque la posture sélectionnée dans l'action est reconnue sur la main gauche.
- **ProximityHand** : Action lancée lorsque l'une des mains est à proximité de l'acteur
- **ExitHand** : Action lancée lorsque l'une des mains quitte la proximité de l'acteur

9. Fonctionnement des actions

Nous allons détailler ici le fonctionnement de chaque action en définissant les paramètres ou les conditions d'activation.

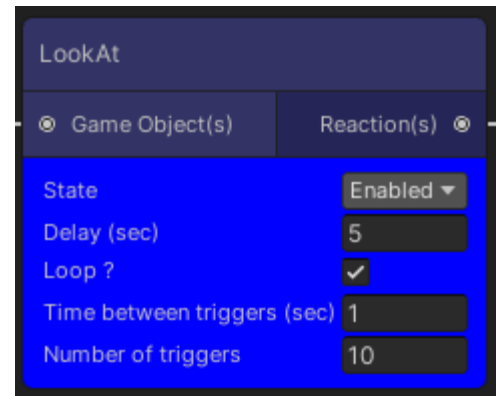
Les actions possèdent un état : *Enabled* ou *Disabled*. Si l'action est *Enabled*, celle-ci enverra son signal aux réactions en sortie. Dans le cas où l'action est *Disabled*, même si la condition de l'action est déclenchée, le signal ne sera pas transmis. L'état des actions peut être changé via la réaction "ActivationAction". La couleur de la boîte s'assombrit lorsque l'action est désactivée.

Dès que la condition de l'action est résolue, le signal est directement transmis aux réactions. Il est possible de retarder l'envoi de ce signal en modifiant le *Delay* de l'action. L'action transmettra son signal après X secondes de délai.

Elles ont aussi la possibilité d'être jouées en boucle. L'action déclenchera les réactions tant que la condition de l'action est vraie. Lorsque la case *Loop* est cochée, deux champs apparaissent : le premier est le temps entre chaque envoi du signal aux sorties de l'action, la seconde valeur représente le nombre de fois où cet envoi sera fait. Si ce second champ est à 0, l'action se fera indéfiniment.

L'action ci-dessus se résume par :

“L'action est active. L'action attendra 5 secondes avant d'être lancée la première fois. Toutes les 1 seconde, 10 fois, l'action enverra le signal aux réactions ”



9.1. Les actions relatives à la tête

Proximity	
Description	Action lancée lorsque la tête du spectateur entre dans l'une des zones autour de l'un des acteurs en entrée.
Paramètre(s) supplémentaire(s)	-----
Configuration	Lorsqu'un acteur est relié à cette action, il obtient un ProximityCollider enfant. Lorsque la tête entre dans la sphère représentée par ce Collider (sphère grise), l'action est déclenchée. Il est possible de modifier la position et l'échelle de la sphère en modifiant le Component <i>Transform</i> .
Autre	Il est possible de modifier la forme du Collider. Dans le ProximityCollider, supprimer le Component <i>SphereCollider</i> et ajouter un nouveau Component Collider avec la forme souhaitée. Attention , il faut que la case Is Trigger <input type="checkbox"/> soit cochée. La zone ne sera plus représentée par la sphère grise mais par la sphère en fil vert.

Exit	
Description	Action lancée lorsque la tête du spectateur sort de l'une des zones autour de l'un des acteurs en entrée.
Paramètre(s) supplémentaire(s)	-----
Configuration	Lorsqu'un acteur est relié à cette action, il obtient un ProximityCollider enfant. Lorsque la tête sort de la sphère représentée par ce Collider (sphère grise), l'action est déclenchée. Il est possible de modifier la position et l'échelle de la sphère en modifiant le Component <i>Transform</i> .
Autre	Il est possible de modifier la forme du Collider. Dans le ProximityCollider, supprimer le Component <i>SphereCollider</i> et ajouter un nouveau Component Collider avec la forme souhaitée. Attention , il faut que la case Is Trigger <input type="checkbox"/> soit cochée. La zone ne sera plus représentée par la sphère grise mais par la sphère en fil vert.

Look At	
Description	Action lancée lorsque l'utilisateur regarde l'un des acteurs en entrée.
Paramètre(s) supplémentaire(s)	-----
Configuration	Un rayon est lancé en ligne droite dans la direction du regard de l'utilisateur. Si ce rayon touche un <i>GameObject</i> avec un Collider et que ce <i>GameObject</i> est acteur de l'action, celle-ci est déclenchée.
Autre	Il est possible que les rayons ignorent certains objets en les plaçant dans le Layer 2 : Ignore Raycast. Cette option est disponible dans l'inspecteur sous le nom du <i>GameObject</i> .

Look Away	
Description	Action lancée lorsque l'utilisateur quitte du regard l'un des acteurs en entrée.
Paramètre(s) supplémentaire(s)	-----
Configuration	Un rayon est lancé en ligne droite dans la direction du regard de l'utilisateur. Si ce rayon touche un <i>GameObject</i> avec un Collider et que ce <i>GameObject</i> est acteur de l'action, lorsque le rayon va quitter l'acteur et pointer le vide ou un autre <i>GameObject</i> , celle-ci est déclenchée.
Autre	Il est possible que les rayons ignorent certains objets en les plaçant dans le Layer 2 : Ignore Raycast. le rayon ne fera que traverser. Cette option est disponible dans l'inspecteur sous le nom du <i>GameObject</i> .

9.2. Les actions relatives au corps

Stand Up	
Description	Action lancée lorsque l'utilisateur est debout sur un acteur.
Paramètre(s) supplémentaire(s)	-----
Configuration	Un rayon est lancé à la verticale du casque de l'utilisateur en direction du sol. Si ce rayon touche un <i>GameObject</i> avec un Collider et que ce <i>GameObject</i> est acteur de l'action, lorsque l'utilisateur sera debout dessus, celle-ci est déclenchée.
Autre	L'action est lancée lorsque la tête de l'utilisateur est entre 90% et 115% de la taille de l'utilisateur enregistré au lancement. Lors du lancement de l'application, l'utilisateur doit rester debout durant une seconde afin de sauvegarder sa taille. Il n'est pour l'instant pas possible d'étalonner la taille de l'utilisateur d'une autre façon ou de commencer dans une autre posture une application.

Crouch	
Description	Action lancée lorsque l'utilisateur est accroupi sur un acteur.
Paramètre(s) supplémentaire(s)	-----
Configuration	Un rayon est lancé à la verticale du casque de l'utilisateur en direction du sol. Si ce rayon touche un <i>GameObject</i> avec un Collider et que ce <i>GameObject</i> est acteur de l'action, lorsque l'utilisateur sera accroupi au-dessus, celle-ci est déclenchée.
Autre	L'action est lancée lorsque la tête de l'utilisateur est entre 56% et 90% de la taille de l'utilisateur enregistré au lancement. Lors du lancement de l'application, l'utilisateur doit rester debout durant une seconde afin de sauvegarder sa taille. Il n'est pour l'instant pas possible d'étalonner la taille de l'utilisateur d'une autre façon ou de commencer dans une autre posture une application.

Sit	
Description	Action lancée lorsque l'utilisateur est assis sur un acteur.
Paramètre(s) supplémentaire(s)	-----
Configuration	Un rayon est lancé à la verticale du casque de l'utilisateur en direction du sol. Si ce rayon touche un <i>GameObject</i> avec un Collider et que ce <i>GameObject</i> est acteur de l'action, lorsque l'utilisateur sera assis dessus, celle-ci est déclenchée.
Autre	L'action est lancée lorsque la tête de l'utilisateur est entre 33% et 56% de la taille de l'utilisateur enregistré au lancement. Lors du lancement de l'application, l'utilisateur doit rester debout durant une seconde afin de sauvegarder sa taille. Il n'est pour l'instant pas possible d'étalonner la taille de l'utilisateur d'une autre façon ou de commencer dans une autre posture une application.

Lie down	
Description	Action lancée lorsque l'utilisateur est couché sur un acteur.
Paramètre(s) supplémentaire(s)	-----
Configuration	Un rayon est lancé à la verticale du casque de l'utilisateur en direction du sol. Si ce rayon touche un <i>GameObject</i> avec un Collider et que ce <i>GameObject</i> est acteur de l'action, lorsque l'utilisateur sera couché dessus, celle-ci est déclenchée.
Autre	L'action est lancée lorsque la tête de l'utilisateur est entre 0% et 33% de la taille de l'utilisateur enregistré au lancement. Lors du lancement de l'application, l'utilisateur doit rester debout durant une seconde afin de sauvegarder sa taille. Il n'est pour l'instant pas possible d'étalonner la taille de l'utilisateur d'une autre façon ou de commencer dans une autre posture une application.

9.3. Les actions générales

Project Clock	
Description	Action lancée après un certain temps depuis le début de la scène.
Paramètre(s) supplémentaire(s)	Time before start : Temps en secondes depuis le début de la scène avant de déclencher l'action
Configuration	Lors du lancement de la scène, un chronomètre est lancé. Lorsque le temps a atteint celui entré en paramètre, l'action est déclenchée.
Autre	-----

9.4. Les actions de séquence

On Video End	
Description	Action lancée lorsque la vidéo de l'un des acteurs possédant un <i>Video Player</i> Component est terminée ou qu'un Loop a eu lieu.
Paramètre(s) supplémentaire(s)	-----
Configuration	Il est nécessaire que les acteurs en entrée ne possèdent chacun qu'un <i>Video Player</i> Component.
Autre	-----

On Audio End	
Description	Action lancée lorsque l'audio de l'un des acteurs possédant une <i>Audio Source</i> Component est terminée ou qu'un Loop a eu lieu.
Paramètre(s) supplémentaire(s)	-----
Configuration	Il est nécessaire que les acteurs en entrée ne possèdent chacun qu'un <i>Audio Source</i> Component.
Autre	-----

9.5. Les actions relatives aux mains [Expérimental]

Hand Gesture Right	
Description	Action lancée lorsque l'utilisateur produit avec sa main droite l'une des postures pré-enregistrées choisi en paramètre.
Paramètre(s) supplémentaire(s)	Hand gesture : Choix de la posture à reconnaître (voir la section des postures)
Configuration	L'utilisation de cette action requiert d'être sur une scène Oculus IIViMaT.
Autre	-----

Hand Gesture Left	
Description	Action lancée lorsque l'utilisateur produit avec sa main gauche l'une des postures pré-enregistrées choisi en paramètre.
Paramètre(s) supplémentaire(s)	Hand gesture : Choix de la posture à reconnaître (voir la section des postures)
Configuration	L'utilisation de cette action requiert d'être sur une scène Oculus IIViMaT.
Autre	-----

ProximityHand	
Description	Action lancée lorsque l'une des mains du spectateur entre dans l'une des zones autour de l'un des acteurs en entrée.
Paramètre(s) supplémentaire(s)	-----
Configuration	Lorsqu'un acteur est relié à cette action, il obtient un ProximityCollider enfant. Lorsque l'une des mains entre dans la sphère représentée par ce Collider (sphère grise), l'action est déclenchée. Il est possible de modifier la position et l'échelle de la sphère en modifiant le Component <i>Transform</i> .
Autre	Il est possible de modifier la forme du Collider. Dans le ProximityCollider, supprimer le Component <i>SphereCollider</i> et ajouter un nouveau Component Collider avec la forme souhaitée. Attention , il faut que la case Is Trigger <input type="checkbox"/> soit cochée. La zone ne sera plus représentée par la sphère grise mais par la sphère en fil vert.

ExitHand	
Description	Action lancée lorsque l'une des mains du spectateur sort de l'une des zones autour de l'un des acteurs en entrée.
Paramètre(s) supplémentaire(s)	-----
Configuration	Lorsqu'un acteur est relié à cette action, il obtient un ProximityCollider enfant. Lorsque l'une des mains sort de la sphère représentée par ce Collider (sphère grise), l'action est déclenchée. Il est possible de modifier la position et l'échelle de la sphère en modifiant le Component <i>Transform</i> .
Autre	Il est possible de modifier la forme du Collider. Dans le ProximityCollider, supprimer le Component <i>SphereCollider</i> et ajouter un nouveau Component Collider avec la forme souhaitée. Attention , il faut que la case Is Trigger <input type="checkbox"/> soit cochée. La zone ne sera plus représentée par la sphère grise mais par la sphère en fil vert.

10. Les réactions

10.1. Les réaction d'apparences

Les réactions d'apparences interviennent sur l'aspect visuel d'une cible. On y retrouve :

- **Color Change** : Réaction permettant de changer la couleur d'un *Renderer*
- **Texture Change** : Réaction permettant de changer la texture
- **Shader Change** : Réaction permettant de changer le shader d'un *Matériel*
- **Transparency** : Réaction permettant de modifier la transparence
- **Visibility** : Réaction permettant de changer la visibilité d'un *Renderer*
- **Activation** : Réaction permettant d'activer ou de désactiver un *GameObject*

10.2. Les réactions sur les lecteurs multimédias

Les réactions outils permettent de jouer des vidéos ou des sons ainsi que de changer de scène. On y retrouve :

- **Play Video** : Réaction permettant de lancer un Video Player
- **Pause Video** : Réaction permettant de mettre en pause un Video Player
- **Stop Vidéo** : Réaction permettant de stopper un Video Player
- **Play Audio** : Réaction permettant de lancer une Audio Source
- **Pause Audio** : Réaction permettant de mettre en pause une Audio Source
- **Stop Audio** : Réaction permettant de stopper une Audio Source
- **Play Video 360** : Réaction permettant de lancer une une Video 360
- **Pause Video 360** : Réaction permettant de mettre en pause une Video 360
- **Stop Video 360** : Réaction permettant de stopper une une Video 360

10.3. Les réactions sur le Transform

Les réactions sur le Transform sont des réactions qui viennent modifier la position l'orientation ou l'échelle d'une cible. Nous avons :

- **Orientation Transform** : Réaction permettant de changer l'orientation
- **Position Transform** : Réaction permettant de changer la position
- **Rotation Transform** : Réaction permettant de changer l'orientation
- **Scale Transform** : Réaction permettant de changer l'échelle d'un objet
- **Teleportation** : Réaction permettant de changer la position de l'utilisateur

10.4. Les réactions générales

Les réactions générales sont des réactions qui ne rentrent dans aucune des catégories ci-dessus. On a :

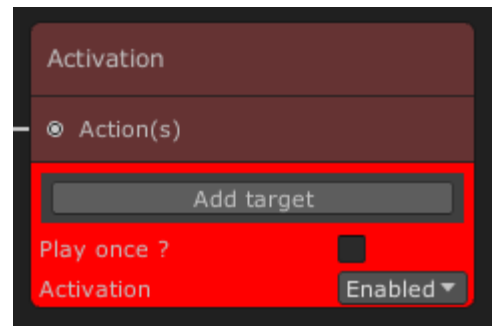
- **Launch Script** : Réaction permettant de lancer un script qui implémente une certaine interface
- **Activation Action** : Réaction permettant de changer l'état d'une action (activé ou désactivé)
- **Load Next Scene** : Réaction permettant de lancer une autre scène

11. Fonctionnement des réactions

Nous allons détailler ici le fonctionnement de chaque réaction en définissant les paramètres et leurs effets.

La grande majorité des réactions définissent les cibles sur lesquelles elles interviennent via l'ajout de cibles. Ces cibles peuvent être ajoutées avec le bouton *Add Target*. Les cibles sélectionnables ne peuvent être que du type sur lequel la réaction agit.

Les réactions ont toutes la possibilité d'être lancées qu'une seule fois. Lorsque la case *Play once* ☐ est cochée, l'action ne sera lancée qu'une seule fois même si l'une des actions en entrée lui envoie un signal.



11.1. Les réactions d'apparences

Color Change	
Description	Réaction permettant de changer la couleur d'un <i>Renderer</i> .
Paramètre(s) supplémentaire(s)	Random color : Applique une couleur aléatoire Specific color : Applique une couleur spécifique
Configuration	Si le paramètre Random color est désactivé, la couleur choisie dans le paramètre Specific color est appliquée aux cibles.
Autre	-----

Texture Change	
Description	Réaction permettant d'appliquer une texture à un <i>Renderer</i> .
Paramètre(s) supplémentaire(s)	New Texture : Nouvelle texture à appliquer au <i>Renderer</i>
Configuration	Applique la texture passé en paramètre New Texture aux cibles.
Autre	-----

Shader Change	
Description	Réaction permettant de changer le Shader d'un <i>Renderer</i> .
Paramètre(s) supplémentaire(s)	New Shader : Nouveau shader à appliquer au <i>Renderer</i>
Configuration	Applique le shader passé en paramètre New Shader aux cibles.
Autre	Les Shaders permettent d'ajouter des effets visuels sur les matériaux. Il en existe beaucoup de disponibles en ligne.

Transparency	
Description	Réaction permettant de changer le mode et la transparence d'un <i>Renderer</i> .
Paramètre(s) supplémentaire(s)	Transparency : facteur de transparence Transparency mode : Type de transparence
Configuration	Le facteur de transparence permet de faire varier le niveau de transparence (0 = opaque, 1 = transparent) Le mode de transparence permet de jouer sur les facteurs de chaque composante (Real = BlendMode.One, Virtual = BlendMode.ScrAlpha (voir documentation unity))
Autre	-----

Visibility	
Description	Réaction permettant de changer la visibilité d'une cible (sans le désactiver).
Paramètre(s) supplémentaire(s)	Visibility mode : Choisi le mode de visibilité
Configuration	Explication des modes de visibilité : <ul style="list-style-type: none"> • Enabled : Active le <i>Renderer</i> de la cible • Disabled : Désactive le <i>Renderer</i> de la cible • Toggle : Active le <i>Renderer</i> du <i>GameObject</i> cible si il est désactivé et inversement
Autre	-----

Activation	
Description	Réaction permettant d'activer ou désactiver un <i>GameObject</i> .
Paramètre(s) supplémentaire(s)	Activation : Choisi le mode d'activation
Configuration	Explication des modes d'activation : <ul style="list-style-type: none">• Enabled : Active le <i>GameObject</i> cible• Disabled : Désactive le <i>GameObject</i> cible• Toggle : Active le <i>GameObject</i> cible si il est désactivé et inversement
Autre	

11.2. Les réactions sur les lecteurs multimédias

Play Video	
Description	Réaction permettant de lancer le <i>VideoPlayer</i> présent sur une cible
Paramètre(s) supplémentaire(s)	Play in loop : Choix de jouer le <i>VideoPlayer</i> en boucle
Configuration	-----
Autre	Assurez-vous que l'option Play On Awake <input type="checkbox"/> du <i>VideoPlayer</i> n'est pas cochée.

Pause Video	
Description	Réaction permettant de mettre en pause le <i>VideoPlayer</i> présent sur une cible
Paramètre(s) supplémentaire(s)	-----
Configuration	-----
Autre	-----

Stop Video	
Description	Réaction permettant de stopper le <i>VideoPlayer</i> présent sur une cible
Paramètre(s) supplémentaire(s)	-----
Configuration	-----
Autre	-----

Play Audio	
Description	Réaction permettant de lancer l' <i>AudioSource</i> présent sur une cible.
Paramètre(s) supplémentaire(s)	Play in loop : Choix de jouer l' <i>AudioSource</i> en boucle
Configuration	-----
Autre	Assurez-vous que l'option Play On Awake <input type="checkbox"/> de l' <i>AudioSource</i> n'est pas cochée.

Pause Audio	
Description	Réaction permettant de mettre en pause l' <i>AudioSource</i> présent sur une cible.
Paramètre(s) supplémentaire(s)	-----
Configuration	-----
Autre	-----

Stop Audio	
Description	Réaction permettant de stopper l' <i>AudioSource</i> présent sur une cible.
Paramètre(s) supplémentaire(s)	-----
Configuration	-----
Autre	-----

Play Video 360	
Description	Réaction permettant de lancer une sphère de Vidéo 360
Paramètre(s) supplémentaire(s)	Play in loop : Choix de jouer le <i>VideoPlayer</i> en boucle Fixed to the head : Choix de fixer la sphère sur la tête du spectateur Back to initial position : Choix de ramener la sphère à sa position d'origine à la fin de la vidéo
Configuration	Afin d'utiliser cette réaction, il est nécessaire d'utiliser les préfabs fournis par IIViMaT. "Clique droit sur la hiérarchie de la Scene > Video > Sphere 360".
Autre	Assurez-vous que l'option Play On Awake <input type="checkbox"/> du <i>VideoPlayer</i> n'est pas cochée.

Pause Video 360	
Description	Réaction permettant de mettre en pause une sphère de Vidéo 360
Paramètre(s) supplémentaire(s)	Stop following the head : Choix de laisser la sphère où elle se trouve dans l'espace Back to initial position : Choix de ramener la sphère à sa position d'origine
Configuration	----
Autre	----

Stop Video 360	
Description	Réaction permettant de stopper une sphère de Vidéo 360
Paramètre(s) supplémentaire(s)	Stop following the head : Choix de laisser la sphère où elle se trouve dans l'espace Back to initial position : Choix de ramener la sphère à sa position d'origine
Configuration	----
Autre	----

11.3. Les réactions sur le Transform

Orientation Transform	
Description	Réaction permettant de changer l'orientation d'une cible en le faisant tourner d'une certaine valeur d'angle autour des 3 axes (d'abord l'axe z, x et y).
Paramètre(s) supplémentaire(s)	Transform Values : Valeurs (en degrés) pour chaque axe de rotation Relative To : Référence autour de laquelle faire la rotation Reference Object : Objet de référence autour duquel tourner si l'option "Relative To" est Object
Configuration	Explication des différents objets de référence : <ul style="list-style-type: none"> • World : Fait tourner l'objet en suivant les axes de la scène • Self : Fait tourner l'objet autour de son point d'origine • Object : Fait tourner l'objet en récupérant l'orientation (Rotation) de l'objet et en y ajoutant les angles • Head : Fait tourner l'objet en récupérant l'orientation (Rotation) de la tête et en y ajoutant les angles
Autre	-----

Position Transform	
Description	Réaction permettant de changer la position d'une cible.
Paramètre(s) supplémentaire(s)	Vector position : Valeurs du vecteur position Relative To : Référence à partir de laquelle faire la modification de position Reference Object : Objet de référence autour duquel tourner si l'option "Relative To" est Object
Configuration	Explication des différents objets de référence : <ul style="list-style-type: none"> • World : Place la cible dans l'espace aux coordonnées du vecteur position • Self : Place la cible à ses propres coordonnées additionné avec le vecteur position • Object : Place la cible à la position de l'objet additionné avec le vecteur position • Head : Place la cible aux coordonnées de la tête additionné avec le vecteur position
Autre	-----

Rotation Transform	
Description	Réaction permettant de faire tourner la cible autour d'un axe passant par un certain point d'un certain angle.
Paramètre(s) supplémentaire(s)	Rotate around axis : Valeur de l'axe (repère monde) autour duquel tourner Angle : Valeur de l'angle (en degrés) Relative To : Référence du point d'origine des axes Reference Object : Point de référence des axes si l'option "Relative To" est Object
Configuration	Explication des différents objets de référence : <ul style="list-style-type: none"> • World : L'origine de l'axe est le point (0,0,0) monde • Self : L'origine de l'axe est l'origine de la cible elle-même • Object : L'origine de l'axe est l'origine de l'objet passé en référence • Head : L'origine de l'axe est l'origine de la tête
Autre	-----

Scale Transform	
Description	Réaction permettant de changer l'échelle d'une cible .
Paramètre(s) supplémentaire(s)	Scale multiplier factor : Valeurs des facteurs multiplicateurs Relative To : Référence de l'échelle Reference Object : Référence de l'échelle si l'option "Relative To" est Object
Configuration	Explication des différents objets de référence : <ul style="list-style-type: none"> • World : L'échelle de la cible est définie aux valeurs des facteurs dans le repère monde • Self : L'échelle de la cible est multipliée par les facteurs multiplicateurs • Object : L'échelle de la cible est définie par la valeur de l'objet passé en paramètre multiplié par les facteurs multiplicateurs
Autre	-----

Teleportation Transform	
Description	Réaction permettant de téléporter l'utilisateur à une position dans la scène.
Paramètre(s) supplémentaire(s)	-----
Configuration	La cible de la téléportation doit être à l'endroit où les pieds de l'utilisateur seront.
Autre	Il est conseillé d'associer la téléportation avec un shader en fond au noier et au blancs sur la caméra pour éviter une téléporation brusque.

11.4. Les réactions générales

Launch Script	
Description	Réaction permettant de lancer n'importe quel code C#. Cette réaction permet l'ouverture des réactions à n'importe quel script présent sur un <i>GameObject</i> cible. Elle permet de lancer des comportements non présents sur IIViMaT.
Paramètre(s) supplémentaire(s)	-----
Configuration	Le script doit implémenter l'interface <i>IRreactionScript</i> et définir la méthode "void ReactionToScript()". Lors de l'activation de la réaction, on appelle l'ensemble des méthodes ReactionToScript() présent sur tous les scripts de toutes les cibles.
Autre	Il est possible qu'un <i>GameObject</i> est plusieurs scripts implémentant l'interface. Chacun des scripts sera alors appelé. Aucune certitude n'est donnée sur l'ordre d'appel.

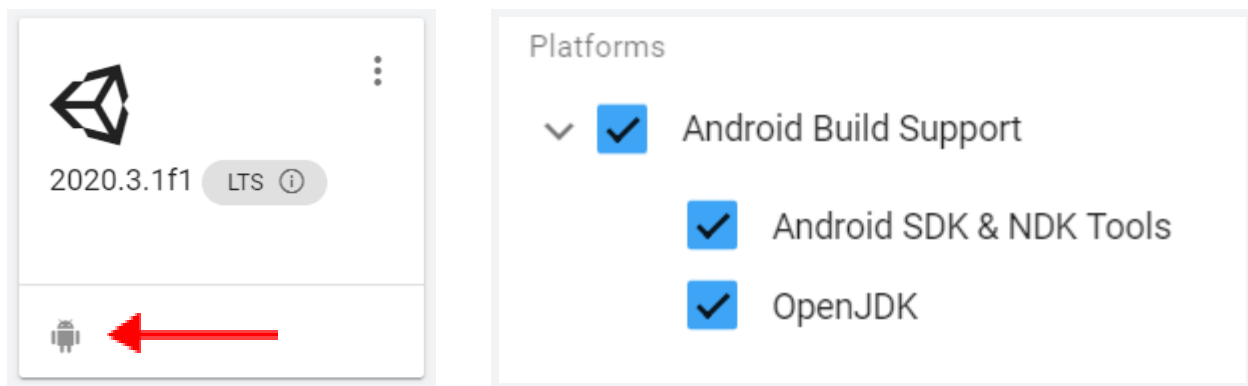
Activation Action	
Description	Réaction permettant de changer l'état d'une action.
Paramètre(s) supplémentaire(s)	State : État que prendra la cible à l'appel de la réaction
Configuration	Il suffit de rentrer le nom de l'action tel qu'elle est dans le graphe (ex : LookAt_6)
Autre	-----

Load Next Scene	
Description	Réaction permettant de lancer une scène.
Paramètre(s) supplémentaire(s)	-----
Configuration	Il suffit de rentrer le nom présent sur l'objet Scene.
Autre	Il est conseillé de ne pas mettre d'espace dans le nom de la scène et de les remplacer par des ' _ '.

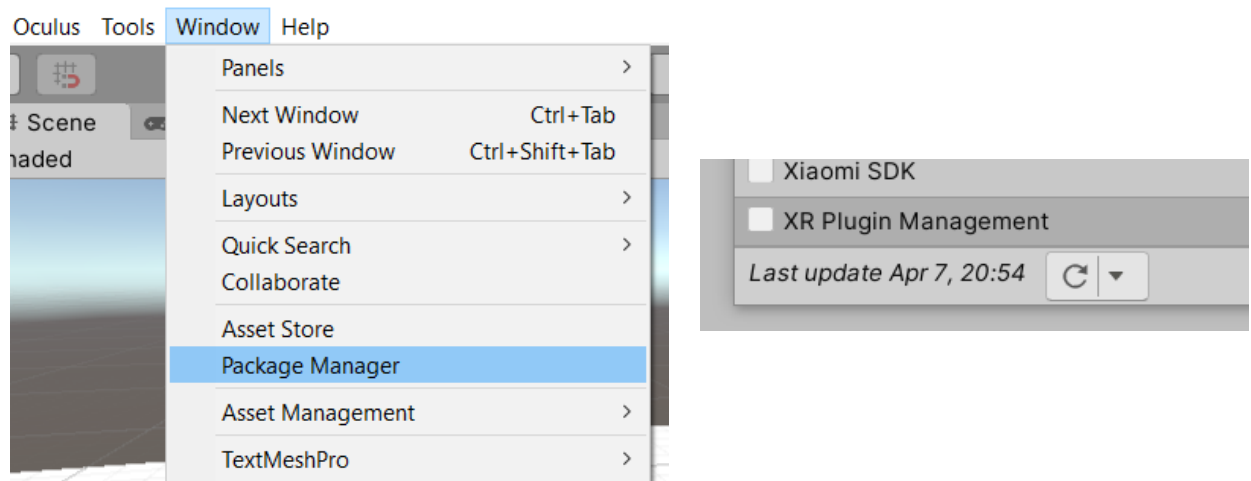
12. Exporter le projet vers un exécutable Oculus Quest 1 / 2

Afin d'exporter votre projet Unity en exécutable Oculus, il est nécessaire de suivre quelques étapes de configuration.

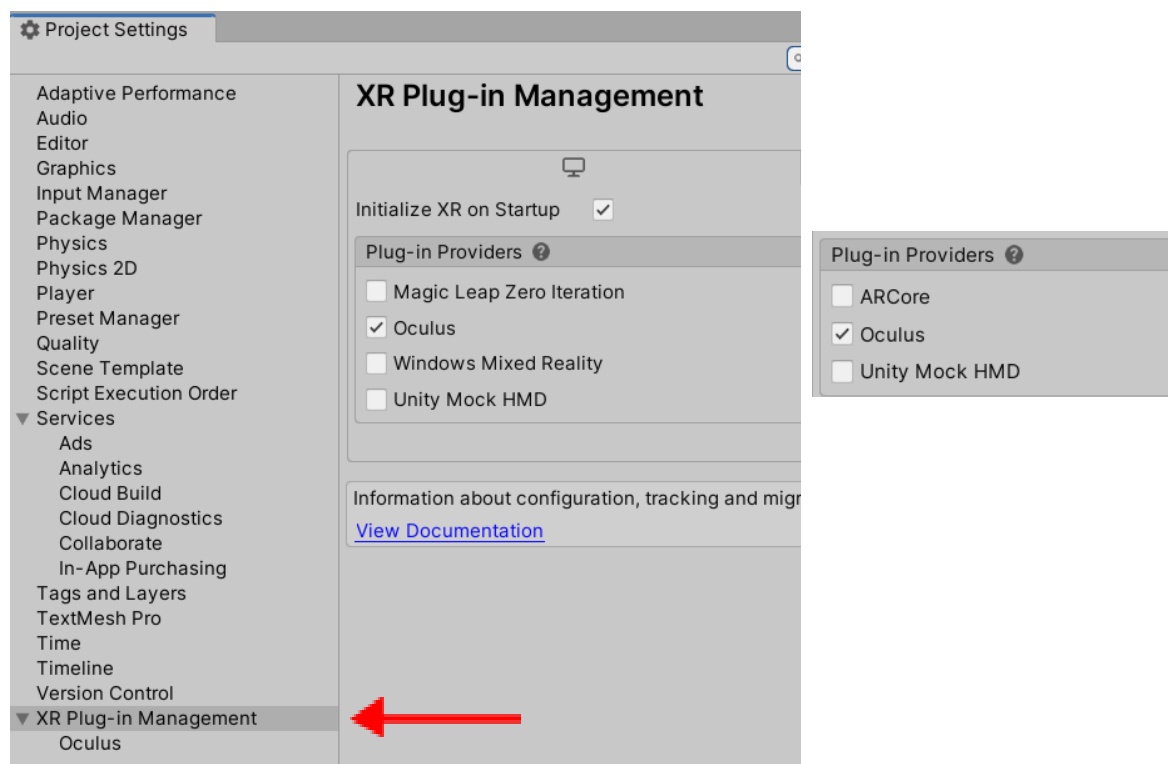
Il est d'abord nécessaire d'avoir téléchargé l'Android Build Support de votre version d'Unity. Afin de savoir si vous le possédez, il suffit de vérifier que vous avez le logo Oculus sous votre version d'Unity. Si vous ne le voyez pas, cliquez les “:” de votre version et sélectionnez “Add Modules”. Cochez l'Android Build Support en vérifiant que les sous-menus soient aussi cochés. Installez le tout.



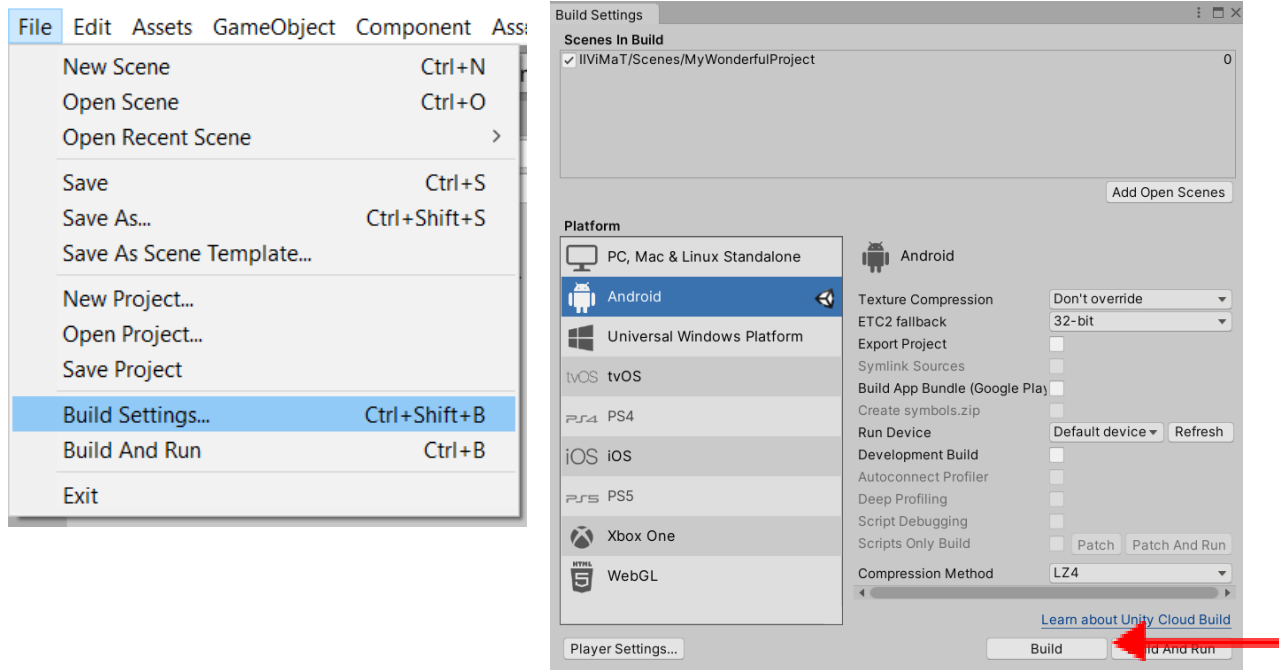
On continue en installant le XR Plugin Management. Rendez-vous dans “Windows > Package Manager”. Cherchez dans la liste le “XR Plugin Management” et installez le.



Nous allons activer les Plugins Oculus grâce au XR Plugin Management. Allez dans “Assets > Project Settings...” . En sélectionnant le XR Plugin Management, cochez les cases Oculus dans le menu Windows et Android.



Continuons en configurant le contenu de l'apk. Sélectionnez “Edit > Build Settings...”. Dans la fenêtre du haut, sélectionnez l'ensemble des scènes qui se trouveront dans votre exécutable. Dans le menu en dessous, sélectionnez le menu Android et cliquez sur “Switch Platform”. Lorsque l'export est terminé, cliquez sur Build et attendez la génération de votre APK.



Vous pouvez maintenant installer votre exécutable sur votre casque en utilisant des logiciels comme SideQuest ou adbLink.

13. FAQ

- **Q : Le message d'erreur indique : “Current scene is not an IIViMaT scene”, que faire ?**

- **A :** Afin d'utiliser le plugin IIViMaT, il est nécessaire de se trouver dans une scène IIViMaT ou Oculus IIViMaT. Pour cela, voir la section [Création d'une nouvelle scène IIViMaT](#).

- **Q : Lorsqu'une réaction se lance, les cibles ne sont pas affectées, que faire ?**

- **A :** Vérifiez que les objets qui doivent être affectés par la réaction sont bien dans les cibles. Un acteur n'est pas la cible d'une réaction. Vérifiez aussi que les cibles possèdent les éléments sur lesquelles les réactions interviennent (*AudioSource, VideoPlayer, Renderer*, etc.).

- **Q : Que dois-je faire lorsque l'action “Look At” ne se déclenche pas quand je regarde mon Game Object ?**

- **A :** Afin que l'action “Look At” puisse détecter un objet, il est nécessaire que celui-ci possède un Collider. Ajoutez-en un cliquant sur le “GameObject > Add Component > Collider (de votre préférence)”.

- **Q : Lorsque je fais une posture sur un acteur, la posture n'est pas reconnue. Que dois-je faire ?**

- **A :** La posture utilise le même principe que pour le “Look At”, il faut donc que l'acteur possède un *Collider* afin de le détecter. De plus, les postures sont calculées en fonction de la taille de l'utilisateur au lancement de l'application. Il est nécessaire de commencer l'application debout, le plus droit possible.

- **Q : Lorsque je fais une posture avec une action “HandGesture”, la posture n'est pas reconnue. Que dois-je faire ?**

- **A :** Pour reconnaître une posture de main, vérifiez que votre projet est

bien une scène Oculus IIViMaT. La posture doit être identique à celles présentes dans la section [Récapitulatif des postures de mains](#).

- **Q : Je n'arrive pas à changer de scène dans la démonstration. Que dois-je faire ?**
- A : Vérifiez que les cinq scènes de démonstration soient dans le build du projet.

- Q :

- A :

- Q :

- A :

- Q :

- A :

14. Récapitulatif des objets IIViMaT

Acteurs	
GameObject	GameObject
	GameObject with VideoPlayer
	GameObject with AudioSource

Actions		
<u>General</u>	Project Clock	
<u>Sequence</u>	On Video End	
	On Audio End	
Spectator	<u>Head Relative</u>	Proximity
		Exit
		Look At
		Look Away
	<u>Body Posture</u>	Stand Up
		Crouch
		Sit
		Lie Down
	<u>Hand Gestures (In progress)</u>	Hand Gesture Right
		Hand Gesture Left
		ProximityHand
		ExitHand

Réactions		
<u>Appearance</u>	Color Change	
	Texture Change	
	Shader Change	
	Transparency	
	Visibility	
	Activation	
<u>Media Player</u>	Video	Play Video
		Pause Video
		Stop Video
	Audio	Play Audio
		Pause Audio
		Stop Audio
	Video 360	Play Video 360
		Pause Video 360
		Stop Video 360
<u>Transform</u>	Orientation Transform	
	Position Transform	
	Rotation Transform	
	Scale Transform	
	Teleportation	
<u>General</u>	Launch Script	
	Activation Action	
	Load Next Scene	

15. Récapitulatif des postures de main

Main droite	
R Pointer	[Photo in game]
R Rockeur	[Photo in game]
R Serrer	[Photo in game]
Main gauche	
L 2doigts	[Photo in game]
L 3doigts	[Photo in game]
L 4doigts	[Photo in game]

16. Choix de la première version

Remarque :

- Changer le sphere Collider en capsule collider des feedbacks
- Ajouter une barre de scroll dans le graphe
- Ajouter le fondu au noir et blanc lors des téléportation
- En italique pour les termes Unity
- Créer prefab a l'échelle pour repere