lojelis

TECH AND HUMAN INNOVATION

XR Interaction Toolkit

30 novembre 2021

Agenda

01 Rappels Unity

02 Développement d’apps RV/RA

03 XR Interaction Toolkit

PARTIE 01

Rappels Unity

Moteur de jeu

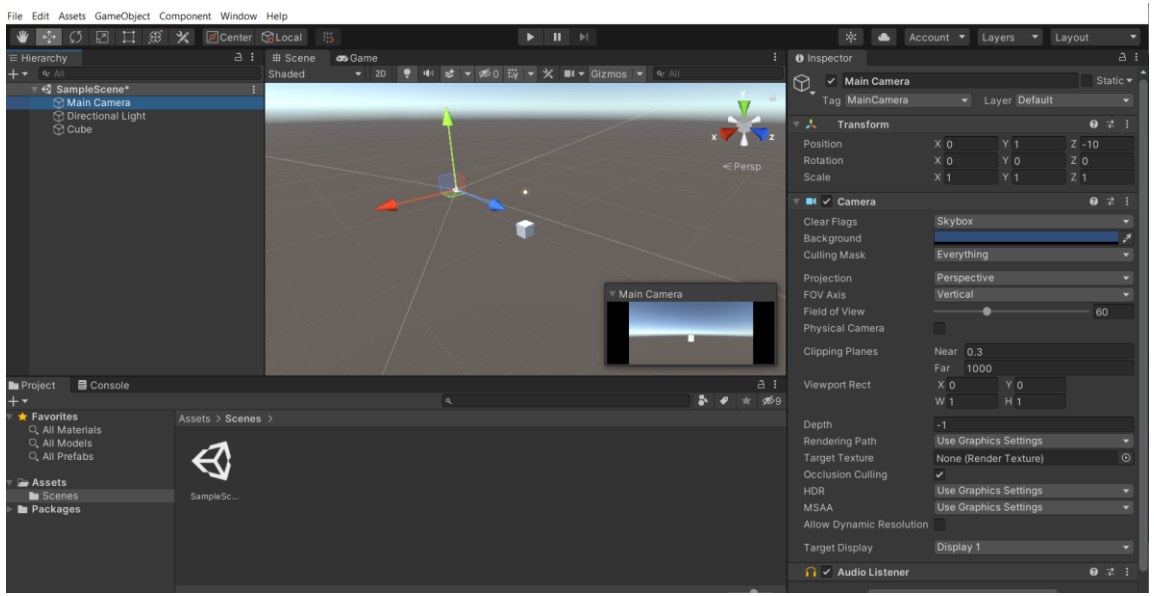
Logiciel qui regroupe et gère en temps réel les fonctionnalités principales d'un jeu vidéo liées au graphisme, son et simulation physique et qui est conçu spécifiquement pour la création et le développement de jeux vidéo

- **Gestion des E/S**
  Clavier/souris, fichiers
- **Géométrie 3D**
  Vecteurs, matrices, quaternions
- **Et plus encore**
  IA, réseau, path finding

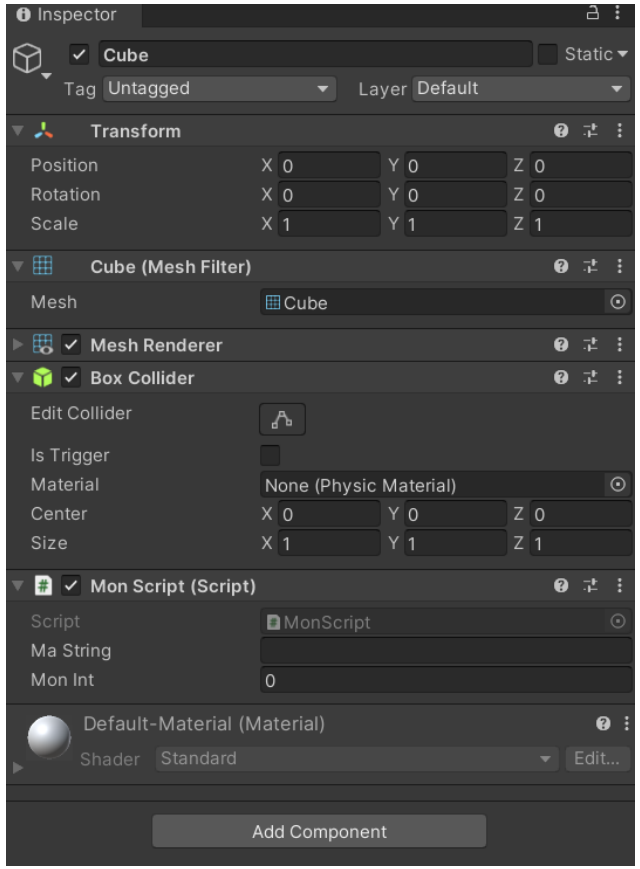
Le moteur Unity

AVANTAGES

- Multiplateforme
- « Gratuit »
- Grande communauté



VIEWPORT



ENTITÉ-COMPOSANT

lojelis
TECH AND HUMAN INNOVATION

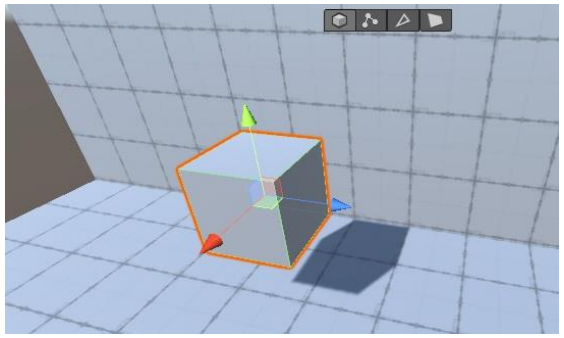
Prérequis Unity

Organisation du moteur

Viewport, Modèle Entité-Composant

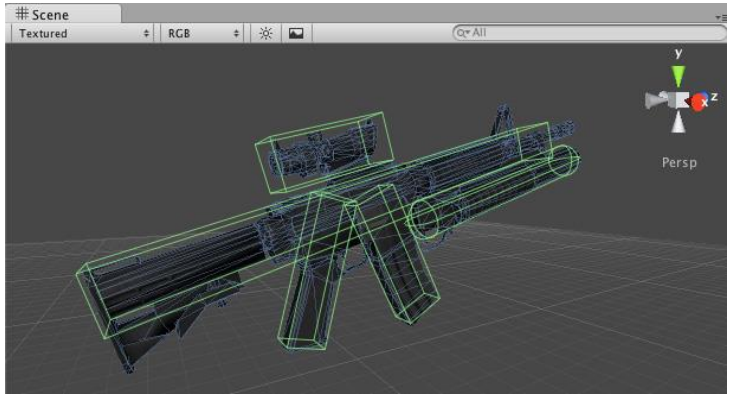
Transformations dans l'éditeur

Translation, rotation, changement d'échelle



Physique et collisions

Rigidbody, Trigger



Scripts C#

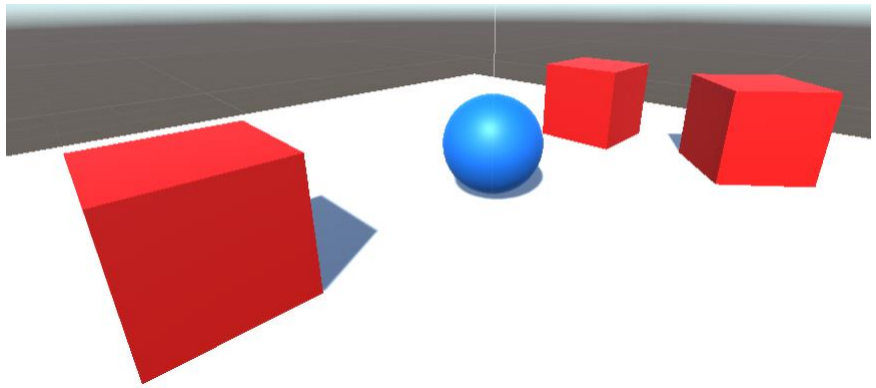
Start/Update/FixedUpdate, API Unity

lojelis
TECH AND HUMAN INNOVATION

TP : Roll and Collect

OBJECTIFS

- Composer une scène
- Physique de base
- Entrées utilisateurs
- Gestion entités par scripts



Interface matérielle

Interface permettant à un utilisateur d'interagir avec une machine.

- Interfaces d'entrée courantes : clavier/souris, écran tactile

- Interfaces de sortie courantes : écran, casque

- Interfaces d'E/S : manette de jeu

Développement d'une application

APPLICATION GÉNÉRIQUE

APPLICATION 3D (VISUEL, PHYSIQUE…)

Les entrées sous Unity – Input Manager

CLAVIER

- GetKey(KeyCode)
- GetKeyDown(KeyCode)
- GetKeyUp(KeyCode)

SOURIS

- GetMouseButton(int)
- GetAxis(string)

Entrée clavier

```
void Update()
{
    if (Input.GetKeyDown("space"))
    {
        print("space key was pressed");
    }
}
```

Entrée souris

```
void Update()
{
    float h = Input.GetAxis("Mouse X");
    float v = Input.GetAxis("Mouse Y");
}
```

Entrée tactile

```
if (Input.touchCount > 0)
{
    Touch touch = Input.GetTouch(0);
    if (touch.phase == TouchPhase.Ended)
    {

    }
}
```

Développer avec une IHM « ésotérique »

MATÉRIEL SPÉCIFIQUE

- Driver
- Contrôleur

INTERACTIONS

- Besoin de connaissances sur les E/S
- Interactions différentes

PARTIE 02

Développement d'applications de Réalité Virtuelle

Les premiers casques grand public

OCULUS DK1 (2013)

HTC VIVE (2016)

Chaque constructeur propose son propre driver et SDK

Dispositifs de RV/RA

LES PREMIERS DISPOSITIFS GRAND PUBLIC

	Virtual Reality						Augmented Reality				Console VR
	PC			AIO	Mobile		AIO		Mobile		
	Oculus Rift	SteamVR	Mixed Reality	Oculus Go	Daydream	GearVR	Hololens	ML1	ARKit	ARCore	PSVR
Company	Facebook	Valve	Microsoft	Facebook	Google	Samsung Oculus	Microsoft	Magic Leap	Apple	Google	Sony
OS support											

Reproduit d'après [Khronos Group, 2019]

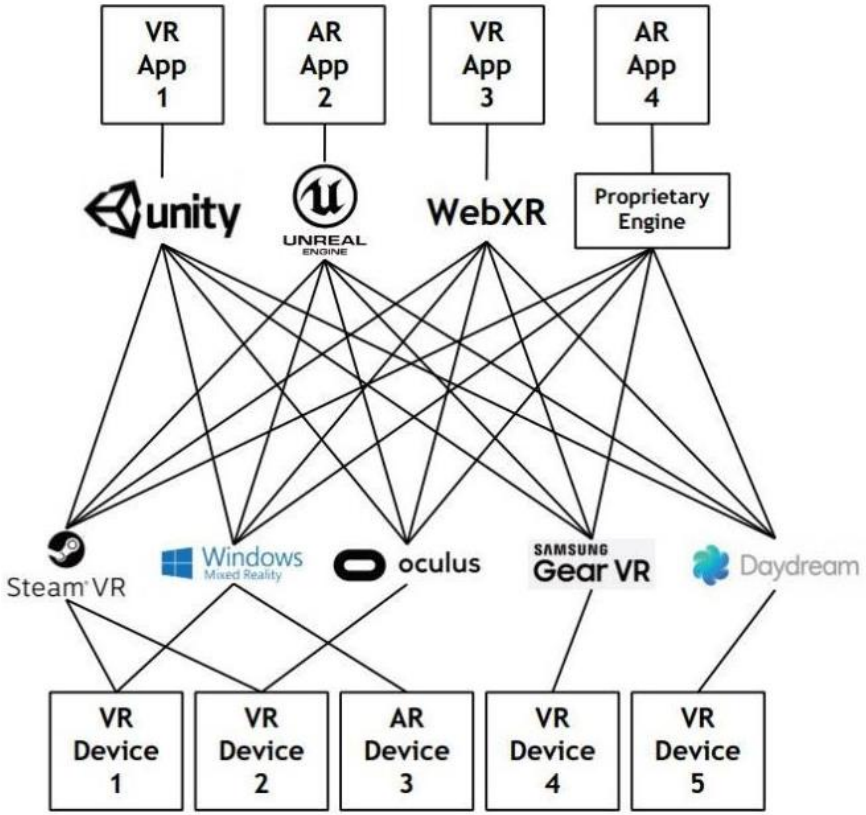
Le cauchemar du développeur RV

FRAGMENTATION

- Augmente les temps de développement
- Augmente le besoin de compétences
- Diminue l'édition / Augmente les coûts

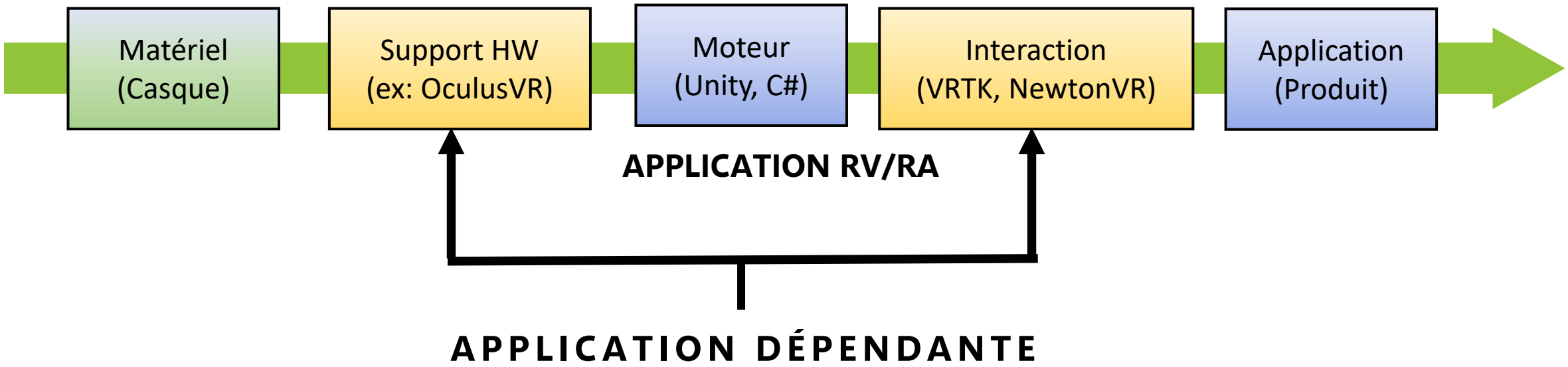
Oculus Quest

Valve Index



lojelis
TECH AND HUMAN INNOVATION

Application Unity de RV/RA



- Du matériel (ex : casque, driver)
- Du kit d'interaction (ex : NewtonVR)

Le bout du tunnel ? OpenXR

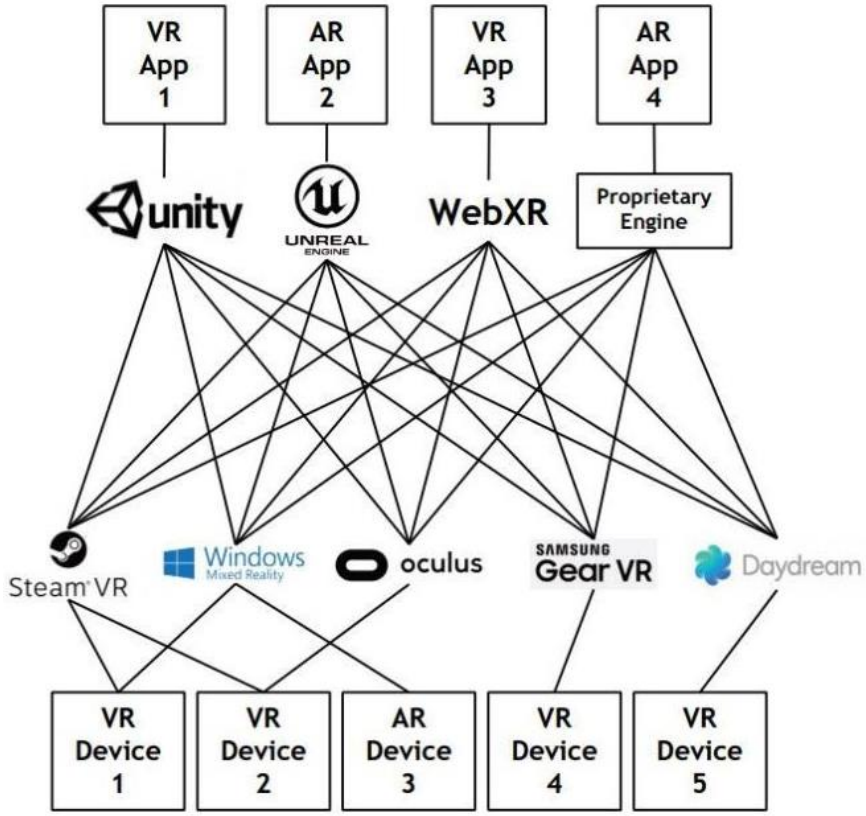
Standard Open-Source pour l'accès aux dispositifs et plateformes de réalité virtuelle et augmentée [Khronos Group, (WebGL, OpenGL)]

- *Couche d'abstraction matérielle*
- *API pour les développeurs d'applications*

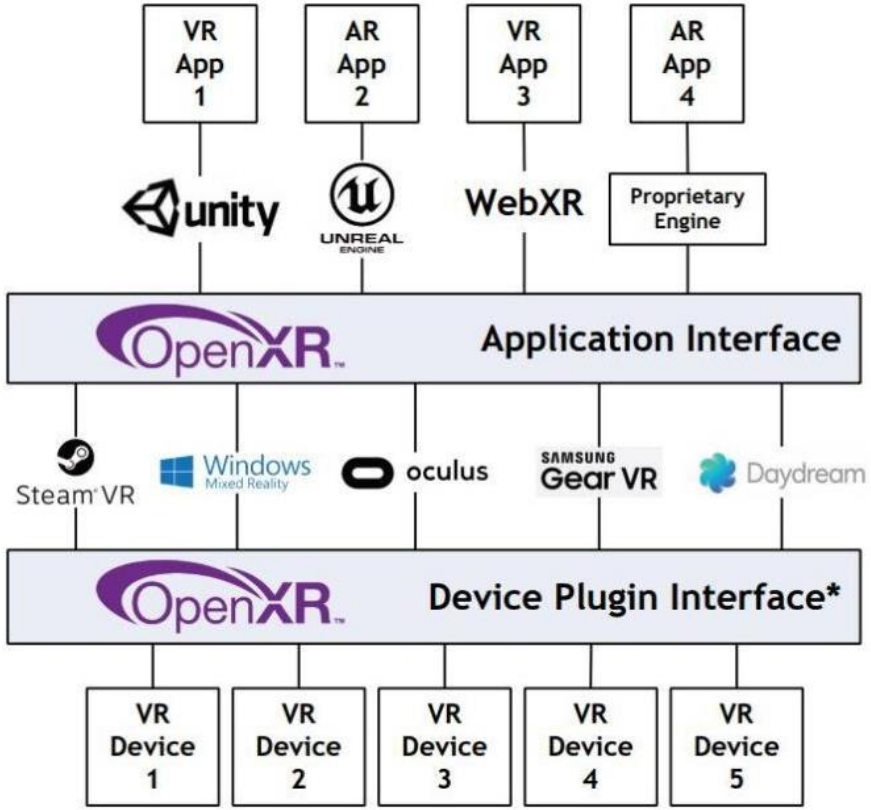
- **Constructeurs CG**
  AMD, Intel, NVidia
- **Constructeurs RV/RA**
  Google, Facebook, Microsoft
- **Téléphonie**
  Huawei, HTC, LG, Sony
- **Moteurs de jeux**
  Unity, Unreal Engine (Valve)

OpenXR

AVANT OPENXR



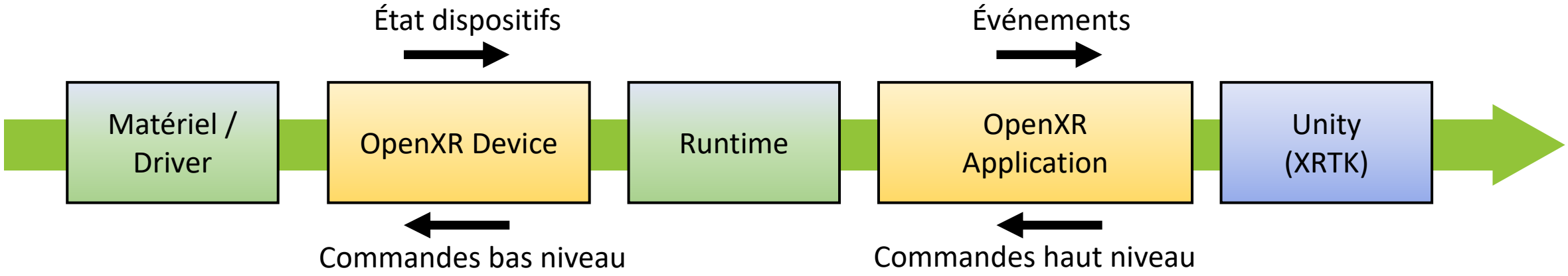
AVEC OPENXR



Couche d'abstraction matérielle + API pour les développeurs d'applications

lojelis
TECH AND HUMAN INNOVATION

OpenXR et XRTK



OPENXR + XRTK

- 2016 : Réflexion sur la création d'un standard RV/RA
- 2019 : Présentation d'OpenXR 1.0
- 2020 : XRTK sort en beta
- 2021 : Intégration du standard sous Unity XRTK

PARTIE 03

XR Interaction Toolkit

Unity : XRPM/XRTK

01 **Abstraction de la couche matérielle**

Accès uniformisé à de multiples périphériques de RA/RV

02 **Retours sensoriels**

Visuel (interactions possible), haptique (contrôleurs)

03 **Interactions pré-implémentées**

Saisie, téléportation, contrôle UI…

lojelis
TECH AND HUMAN INNOVATION

Définitions

OBJET

Quelque chose que l'utilisateur voit/interagit avec dans le monde virtuel

CONTROLLER

Composant liant une action de l'utilisateur (ex : appui gâchette manette gauche) à un événement d'interaction (ex : sélection)

GESTE

Séquence de mouvements liée à une action de manipulation d'un Interactable

HAPTIQUE

Stimulus physique donnant un retour à l'utilisateur (ex : vibration)

SURVOL/HOVER

État dans lequel un Interactor est susceptible d'interagir avec un Interactable.

SÉLECTION/SELECT

État dans lequel un Interactor est en train d'interagir avec un Interactable.

ACTIVATION/ACTIVATE

Action dépendant de et affectant l'Interactable actuellement sélectionné

XR Interaction Toolkit

*Module haut niveau orienté **composants** comprenant un **système d'interaction**.*
Gère les interactions avec objets 3D et UI.

- **Interactor**
  Composant permettant *d'interagir* avec des Interactable
- **Interactable**
  Composants définissant comment on peut *interagir* avec un objet
- **Interaction Manager**
  Composant gérant les interactions entre Interactors et Interactables
- **+ Système de hooks**

Exemple d’interaction

01

Survol de l’arc

Débloque l’action de saisir l’arc

02

Sélection de l’arc

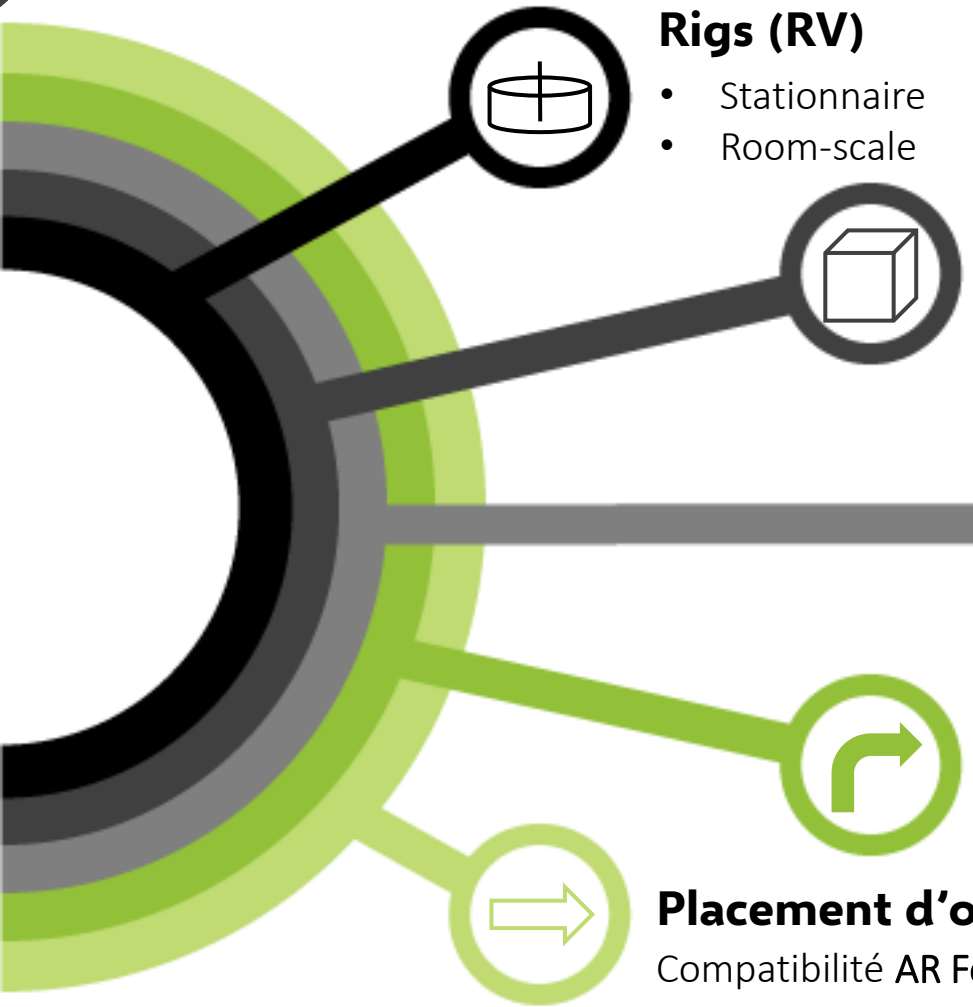
L’arc est saisie et suit la main de l’utilisateur

03

Activation de l’arc

Si une flèche est chargée, tire la flèche

Fonctionnalités XRTK



Rigs (RV)

- Stationnaire
- Room-scale

Interactions avec objets (RV/RA)

- Survoler, sélectionner
- Déplacer, manipuler

+ tapoter, glisser, pincer, zoom (RA)

Interactions UI (RV/RA)

Manipulation par pointeur

Locomotion (VR)

Snap turn, téléportation

Placement d'objets (RA)

Compatibilité AR Foundation

Architecture

```
XR Rig (XR Rig)
- Camera Offset
    - Main Camera (Tracked Pose Driver)
    - LeftHand Controller (XR Controller)
    - RightHand Controller (XR Controller)
```

- **XR Rig**
  Représente la tête / mains de l'utilisateur
- **Tracked Pose Driver**
  Gestion de la caméra
- **XR Controller**
  Gestion des manettes

Input

```
using UnityEngine.XR;

public class InputExample
{
    public XRNode controller;
    private bool primaryButton;

    void Update(){
        InputDevice device =
            InputDevices.GetDeviceAtXRNode(controller);
        device.TryGetFeatureValue(CommonUsages.primaryButton,
            out primaryButton);
    }
}
```

https://docs.unity3d.com/Manual/xr_input.html

- **XRNode**
  Ex : Main gauche
- **CommonsUsages**
  Ex : Bouton primaire
- **TryGetFeatureValue**
  Accède à la valeur d'un bouton sur un dispositif

Locomotion

La locomotion permet de se déplacer dans une scène virtuelle.

- **XR Rig**
  Représente l'utilisateur
- **Locomotion System**
  Contrôle l'accès au XR Rig
- **Teleportation System**
  Téléportation
- **+ Bonus**
  Continuous/Snap Turn Provider…

UI

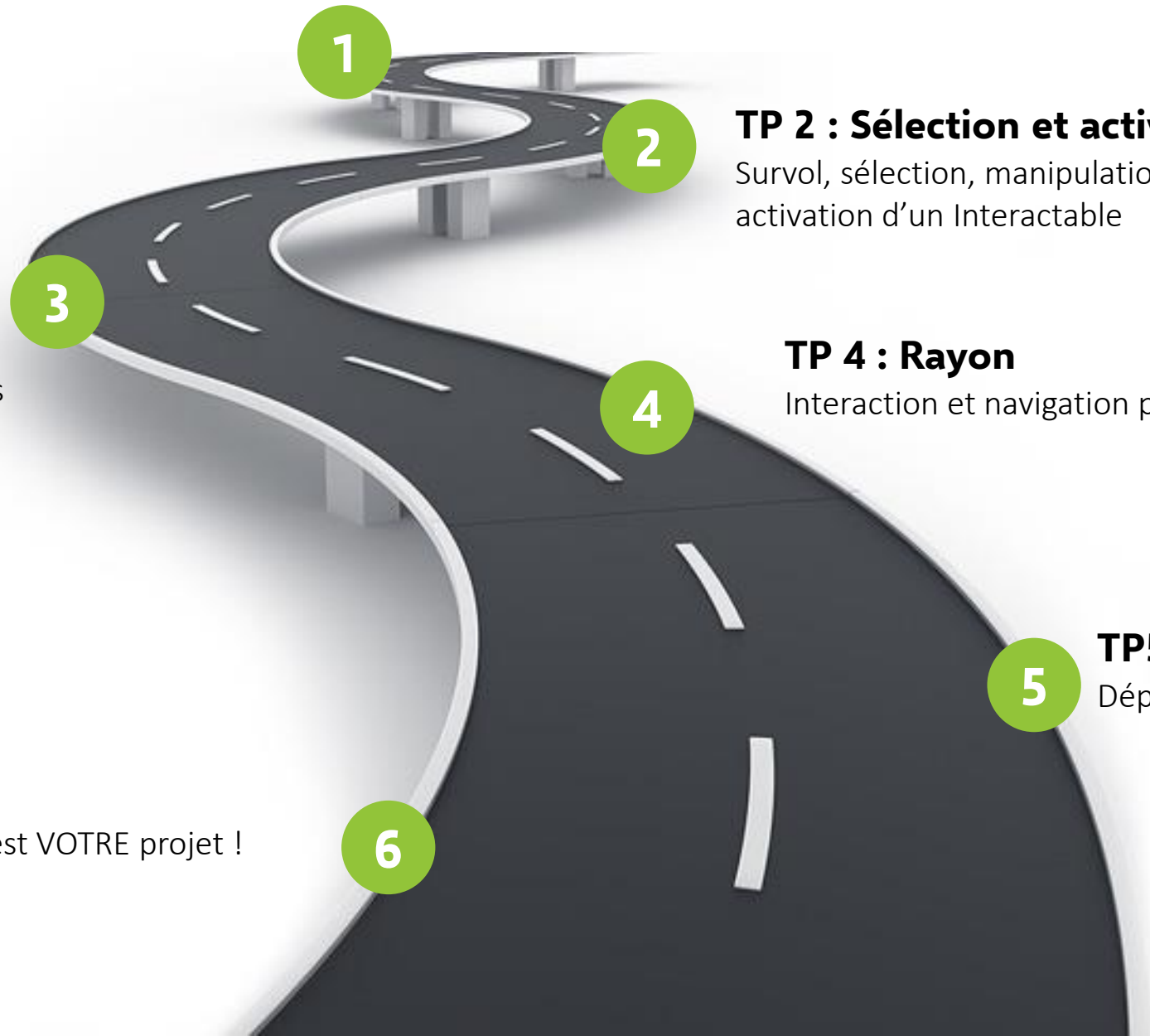
*Module haut niveau orienté **composants** comprenant un **système d'interaction**. Gère les interactions avec objets 3D et UI.*

- **UI Canvas**
  Canvas dans l'espace monde pouvant accueillir des éléments UI

- **UI EventSystem**
  Gère l'interaction et la mise à jour des UI Canvas
  Automatiquement créé lors de l'ajout d'un UI Canvas

Pratique !

TP 1 : Initialisation
Paramétrer une application pour la RV



TP 2 : Sélection et activation
Survol, sélection, manipulation et activation d'un Interactable

TP 3 : Contraintes
Manipulations contraintes sur des Interactables

TP 4 : Rayon
Interaction et navigation par rayon

TP5 : Déplacement continu
Déplacement via joystick

Projet
Parce que c'est VOTRE projet !

lojelis
TECH AND HUMAN INNOVATION

Liens utiles

RESSOURCES

- Documentation XRTK : https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@1.0/
- Documentation API : https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@1.0/api/
- XR Inputs : https://docs.unity3d.com/Manual/xr_input.html