Overview

StudioV is working on a mixed reality production platform. Taking advantage of real-time rendering ability of game engine Unity, we produce CGI content that syncing environment, character skeleton tracking animation and facial expression animation in real-time. During production, actors will also be immersed in the virtual environment, which can help actors set up the mood.

The end product of production can distribute to multiple platforms, including VR, AR, 360 video.

Requirement

- Unity 2017.2
- Motion capture system: e.g. Optitrack motion capture system + software Motive 2.0
- VR system: e.g. HTC Vive + steamVR

Before configuration of Unity project*

1. Prepare motion capture system

Motion capture system is required to track skeleton and rigidbody movement, and stream tracking data to Unity. Follow the instruction in following link to set up motion capture system
https://wiki.optitrack.com/index.php?title=OptiTrack_Documentation_Wiki

2. Prepare VR system

VR system allows actors see the scene, other actors and themselves immersively. You need a VR compatible computer to run VR application. Follow the link to set up VR system
https://support.steampowered.com/kb_article.php?ref=2001-UXCM-4439

* In all instructions, we will only provide explanations of the components that we are using, if you want to use other hardware or software, you can try to work on your own or contact studio.

Before Unity project configuration*

3. Prepare face tracking
   - Eye tracking: hardware required. We are using Tobii VR for eye tracking. Tobii Pro SDK Unity is used for development.

     https://www.tobii.com/tech/products/vr/

     https://www.tobiipro.com/product-listing/tobii-pro-sdk/
   - Facial expression tracking: hardware required. We are using BinaryVR for facial expression tracking.

     http://www.binaryvr.com/

Configure project step by step

1. Download this project and open it in Unity 2017.2
2. Import the unity assets/plugins you need, please check Appendix A for a list of assets we use
   - **Import Photon Unity Networking(PUN), version 1.87** https://www.assetstore.unity3d.com/en/#!/content/1786

     1) To use PUN, you need an App ID, follow the instruction to get an App ID for your application https://doc.photonengine.com/en-us/realtime/current/getting-started/obtain-your-app-id

     2) Initial setup of PUN: https://doc.photonengine.com/en-us/pun/current/getting-started/initial-setup

     3) Go to PhotonServerSetting, make sure the Hosting protocal is set to Tcp.

     4) In PhotonServerSetting find Rpc list and press ”Clear RPCs” and then ”Refresh RPCs”.
   - **Import MicroLibrary**: Download the source code from link https://www.codeproject.com/Articles/98346/Microsecond-and-Millisecond-NET-Timer, copy the file MicroLibrary.cs to Assets/Scripts/Tool/ in the project

   Note: After clicking Clear in Console, there shouldn’t be any complie errors after you import these 2 assets.

Configure project step by step

- **Import SteamVR SDK** https://www.assetstore.unity3d.com/en/#!/content/32647

   1) Open Launcher scene, go to Assets/Prefabs folder, and drag [CameraRig] object to Launcher scene

   2) Find TobiiPro_Host object in the scene, assign the EyetrackerCam field with Camera(eye) object under [CameraRig]/Camera(head)

- **Import BinaryVR SDK (Recommended)**: you can get the SDK after purchase

   1) Go to path binarysdk/examples/Unity/Assets/ExampleScene/Scripts/, copy and replace FaceExpressionController.cs into Assets/Scripts/AvatarSpecific/ in the project.

   2) Go to path binarysdk/examples/Unity/Assets/BinaryFaceHMD/, copy this folder and place it under Assets folder in the project.

   3) Go to path binarysdk/examples/Unity/Assets/StreamingAssets/, copy the file model.bfh in this folder to Assets/StreamingAssets/ in the project

   4) Refer to example scene in binarysdk to set it up

Run sample

1. You need at least 2 computers configured to run a complete sample, then you will be able to use 1 computer as a master client, and other as different types of client, e.g. avatar, camera client, spectator etc.
2. After importing all necessary assets, the Unity Console should be error free. Launcher scene is open by default. Play this scene.
3. Click on “Play!” button, it will start the program as a Photon master client, and load the selected scene.

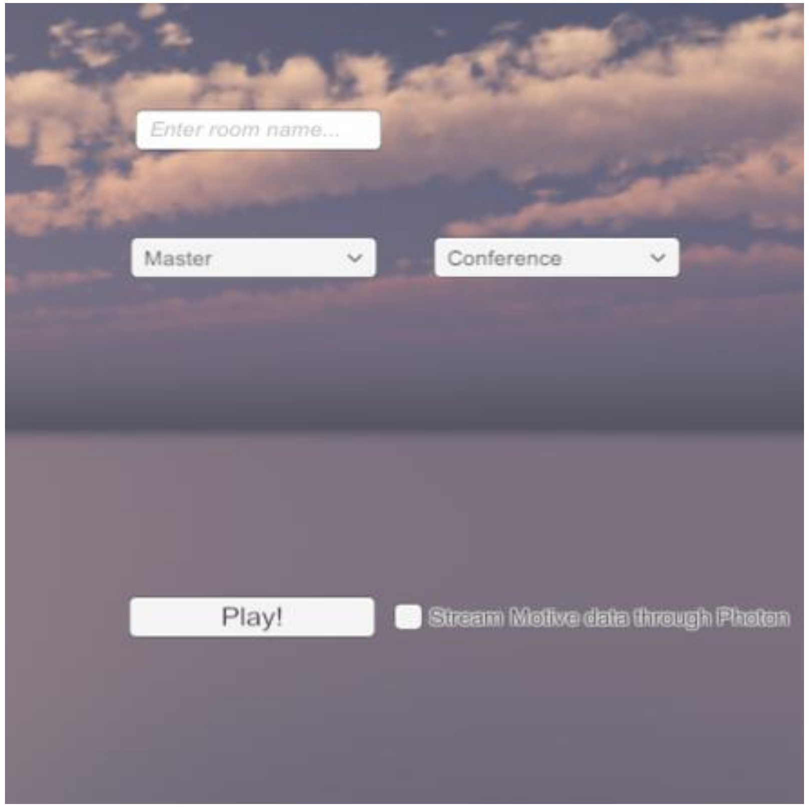

Figure1. Laucher UI

Run sample

4. On another computer, play the same scene and bring up the dropdown menu.
5. Click on the client type you would like to join as.
6. Click on “Play!” button, it will start the game as a Photon client, and load the scene that master client has selected.

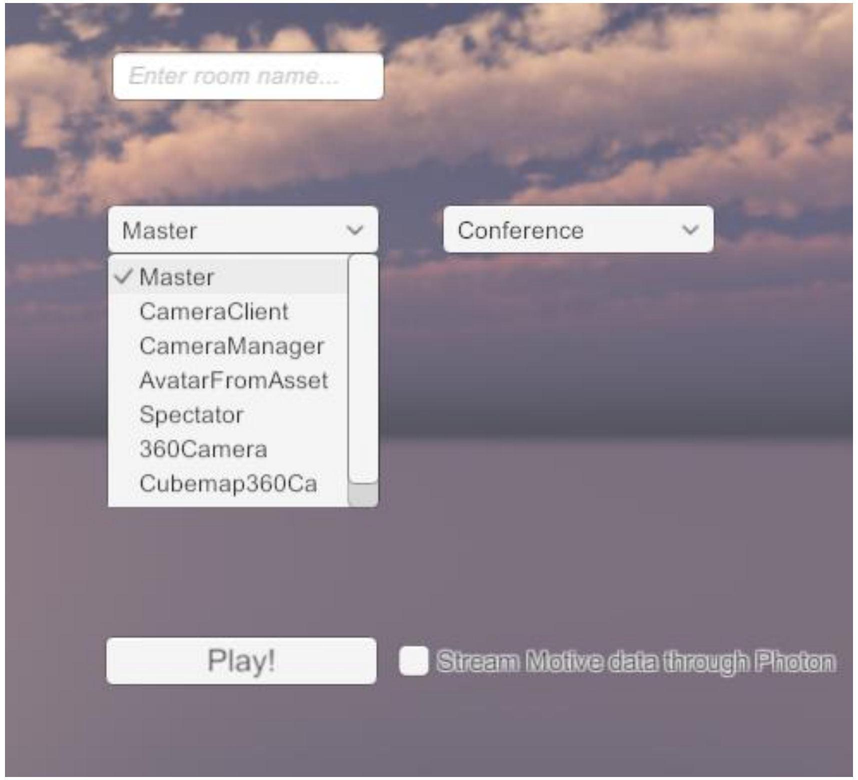

Figure 2. Dropdown Menu

Run sample – Join as CameraClient

1. Click on CameraClient in the dropdown menu, and click "Play!" button
2. Control the movement of CameraClient with keyboard or gamepad. Please check the mapping at InputManager (Top menu bar Edit/Project Settings/Input).

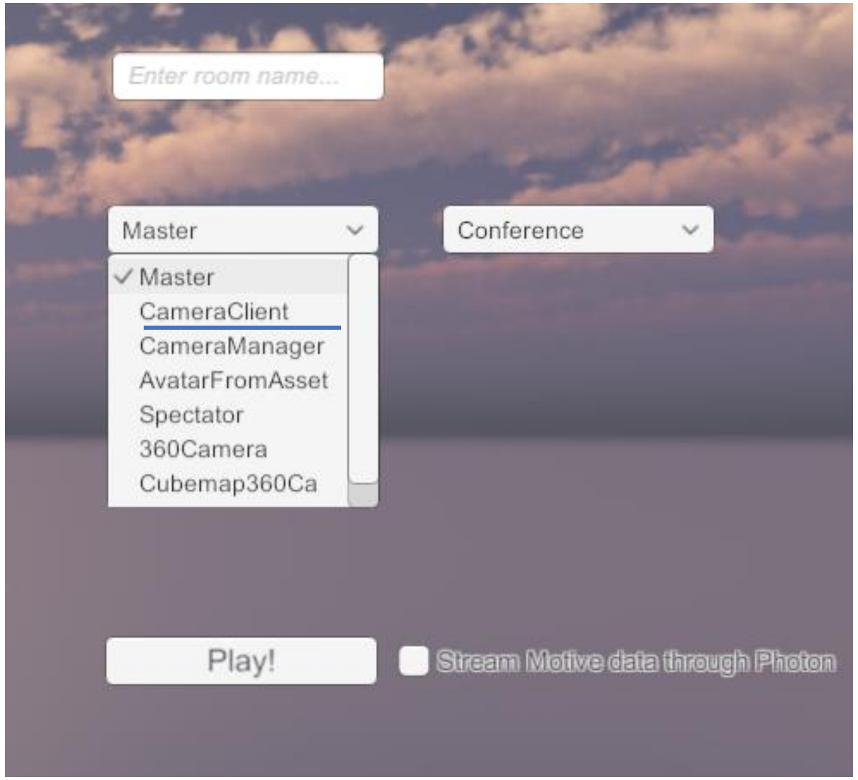

Figure 3. CamerClient

Run sample – Join as CameraManager

1. Click on CameraManager in the dropdown menu, and click “Play!” button
2. Use numeric keys on the top of the alphanumeric keyboard to switch between cameras.

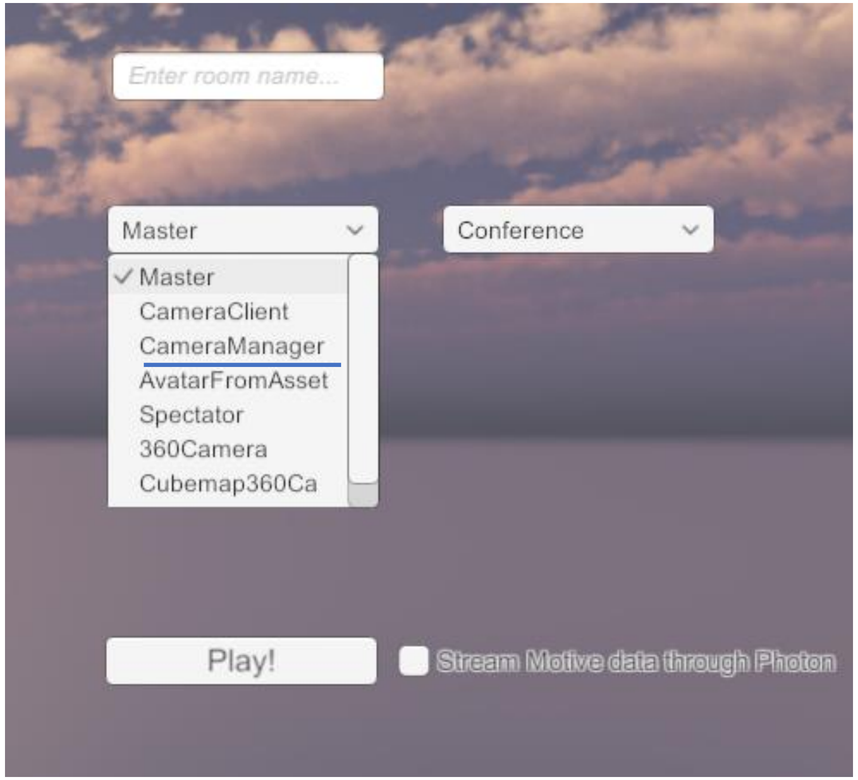

Figure 4. CameraManager

Run sample – Join as Avatar

1. Click on AvatarFromAsset in the dropdown menu, and click “Play!” button.
2. You will see a menu show in Figure 5, select the avatar, fill in the Skeleton name to match the skeleton name in Motive, fill in the height that match the real height of actor.
3. Click “Spawn Avatar”.

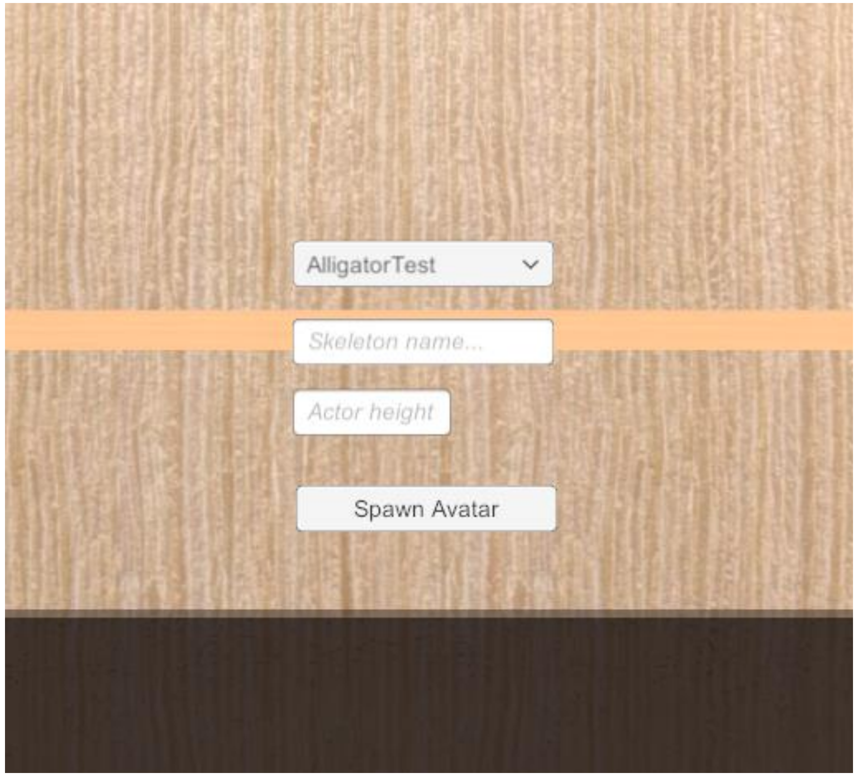

Figure 5: Avatar setting

Run sample – Join as Avatar

4. Avatar UI will be enabled after you join as an avatar. Left lower corner is a control panel and right lower corner is a display panel for Binary camera.
5. To start facial expression tracking, click “Open Device” for start Binary camera, then click “Calibration and Start Tracking”.
6. “Start/Stop Recording” buttons are for start/stop data recording of blendshapes and eye movement into text files.

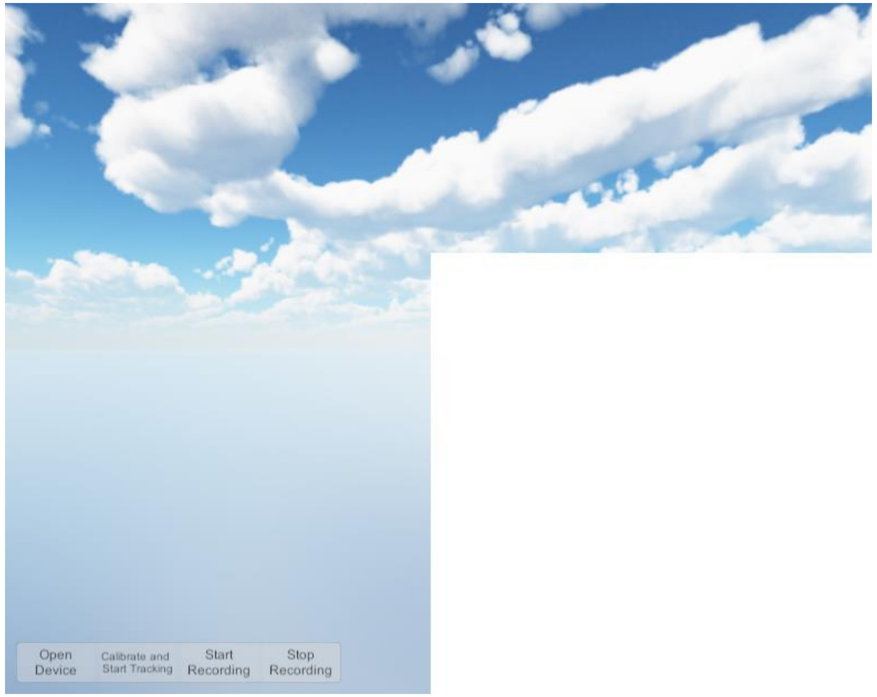

Figure 5: Avatar UI

Run sample – Ingame Menu

1. To enable ingame menu, press “Esc“ key. You can switch between Avatars, Scenes and Rigidbodies menu on top right.
2. Spawn HMDLess avatar: click on the avatar you want to spawn at lower panel, then click on the blank card at upper panel.

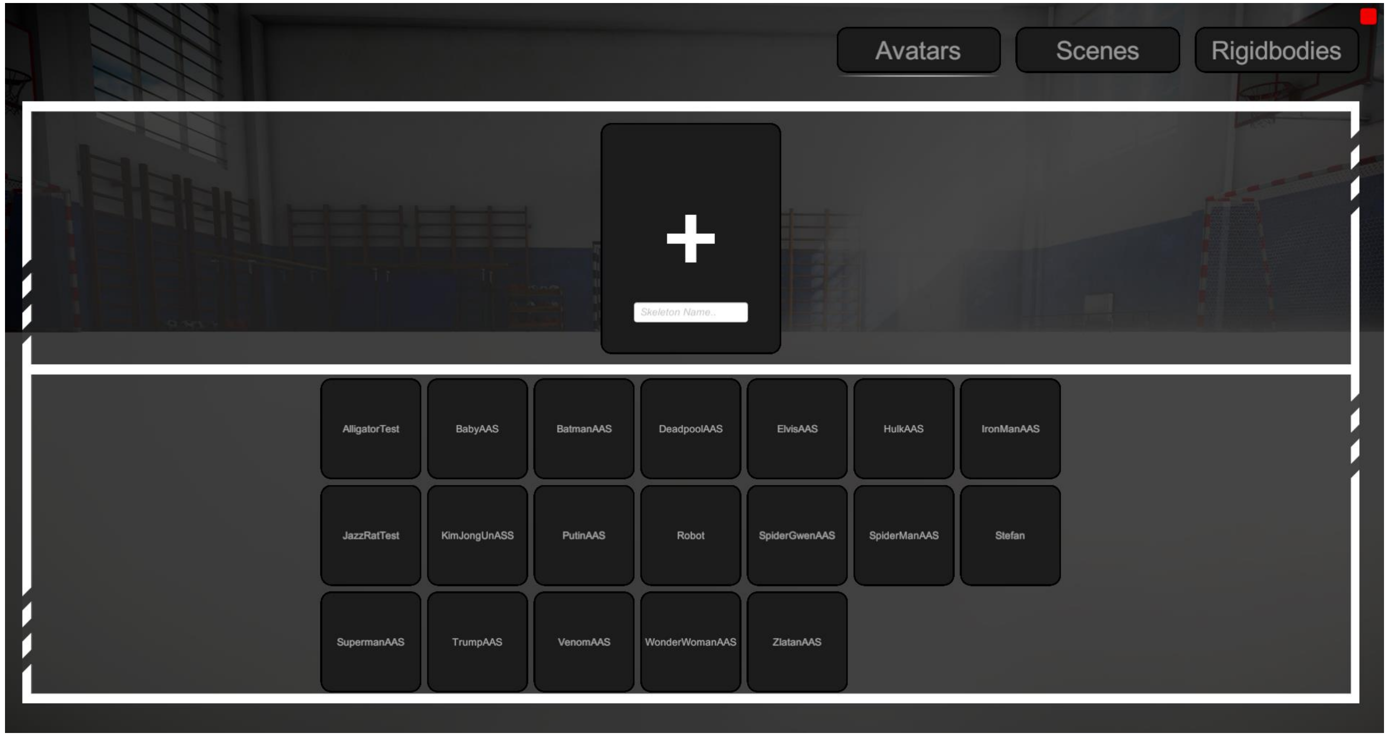

Figure 6: Ingame menu

Run sample – Ingame Menu

3. Remove HMDLess avatar: click on “Remove” button on the avatar card. (You can’t remove HMD avatars, since they join as a client, they will be removed when they leave the room)
4. Replace HMD/HMDLess avatar: select one avatar at lower panel, and click on “Replace” button on the avatar card.
5. Switch scenes: go to Scenes menu, choose from dropdown menu and click “Load Scene” button.
6. You can also add rigidbody with the same streaming ID in motive, and delete instantiated rigidbodies.

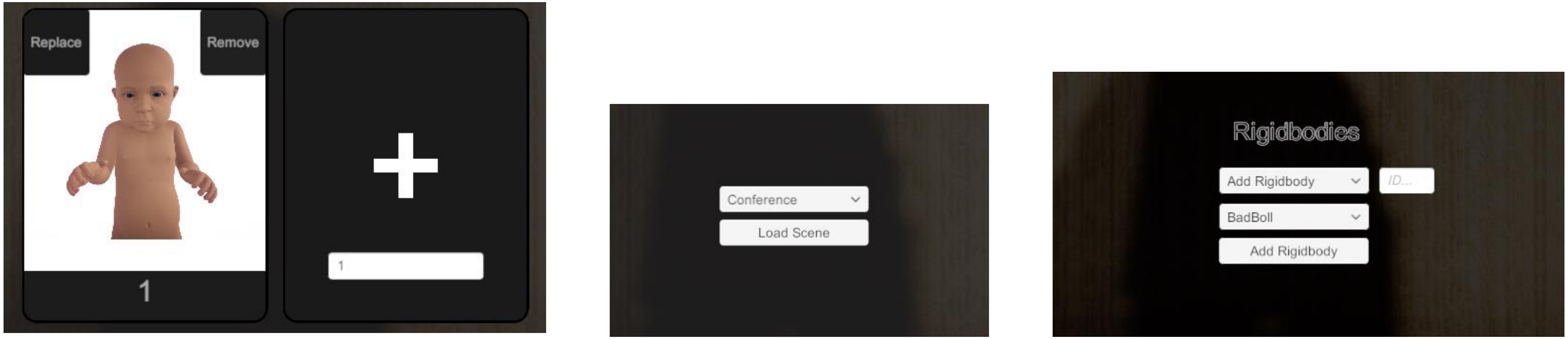

Figure : Ingame menu

Note: we are still improving UI system.

Editor Tools

Our editor tools are located at menu bar on the top of Unity Editor, they are:

1. **OpenScene**: this tool is for quick access to all the scenes you have in the project. Before using OpenScene tool, you need to use **Lukas scene handler**.
2. **Assets/Lukas scene handler/Generate Scene Menu Items**: this tool is for looking for all the scenes you have in project and load it to OpenScene tool.
3. **Assets/Reload resources into dropdowns**: this tool is used in play mode for reloading resources into dropdowns in Ingame Menu.
4. **Assign Avatar Tags Window (experimental)**: this tool is used when you set up avatars as assetbundles.

How to add custom assets step by step

After running the sample successfully, you should be able to add your own avatars and scenes.

- Add avatars

Assetbundles are used to load avatar assets into scenes. Please read the manual for how to set up your own assetbundle.

https://docs.unity3d.com/Manual/AssetBundles-Browser.html

- Add scenes

Create a scene as usual, the only requirement is an Overview Camera object included in out example scene. You can copy this object across scenes. You can also add a customized Post Processing Behaviour profile.

Please take our example scene as a reference.

Appendix A

Assets checklist

Avatar

- Body animation: Optitrack Unity plugin*(included) http://optitrack.com/downloads/plugins.html
- Facial expression
  - BinaryVR plugin: can be downloaded after purchased http://www.binaryvr.com/vr
  - SALSA with random eyes (Optional): can be downloaded after purchased https://www.assetstore.unity3d.com/en/#!/content/16944
- Eye movement: Tobii unity plugin (included) https://www.tobiipro.com/product-listing/tobii-pro-sdk/#Download

* Optitrack plugin also support rigidbody tracking, hmd tracking(if you are using Oculus or GearVR)

Camera

- Movement (Included):
  - Virtual camera: Optitrack Unity plugin
  - Camera client: Unity Input class
- Image effects (Optional):
  - Post processing (Included) https://www.assetstore.unity3d.com/en/#!/content/83912
  - Volumetric light https://www.assetstore.unity3d.com/en/#!/content/67665
  - SSAO Pro https://www.assetstore.unity3d.com/en/#!/content/22369
- Camera logic: Cinemachine (Included) https://www.assetstore.unity3d.com/en/#!/content/79898

System

- Networking: Photon Unity Networking 1.87 (required)
  https://www.assetstore.unity3d.com/en/#!/content/1786

- Virtual Reality support: SteamVR plugin (required)
  https://www.assetstore.unity3d.com/en/#!/content/32647

Other

- Mirror: Stereo Rendering
  https://www.assetstore.unity3d.com/en/#!/content/71255

- Shaders: Uber shader
  https://www.assetstore.unity3d.com/en/#!/content/39959

- Record data in high precision: MicroLibrary (required)
  https://www.codeproject.com/Articles/98346/Microsecond-and-Millisecond-NET-Timer

Appendix B

Terms definitions

Cameras

- **Camera Client** – Renders the 3D environment during realtime production within the scene. The Camera Client is controlled by a user with a keyboard/controller, who can see and move around within the 3D environment. Another program (any screen capture software, for example OBS) is then used to record the same view that the camera client has.

- **Virtual Camera** – Motion Capture based camera, that moves around in the studio. Renders data in the same way as the Camera Client and is recorded by the same external screen capture software.

Avatars

- **HMD Avatar**– this type of avatar is joined through Launcher by selecting AvatarFromAssetBundle.
- **HMDLess Avatar**– this type of avatar is spawned by any type of client via the ingame menu.

Rigidbodies

- Rigidbodies are objects which are not animated. It’s able to be tracked by the motion tracking system.

https://v20.wiki.optitrack.com/index.php?title=Rigid_Body_Tracking

Appendix C

Hotkeys

Hotkeys

F1	Edit Clapper
F2	Clap Clapper
F3	Reset camera rotation
F4	Refresh camera manager
F5	Toggle on/off HMD positioning
F6	Eye calibration
F8	Freeze camera movement
Esc.	Enable ingame menu