# TABLE OF CONTENT

1. **INTRODUCTION** .................................................................................................................. 1

1.1. COMPATIBLE VR HEADSET .................................................................................................. 1

1.2. COMPATIBLE UNITY VERSIONS ........................................................................................... 1

1.3. HOW TOBI PRO VR ANALYTICS IS STRUCTURED ............................................................. 1

1.4. HOW TO READ THIS MANUAL ............................................................................................ 1

1.4.1. Reference Unity scene ...................................................................................................... 2

1.5. SETUP YOUR VR HEADSET .................................................................................................. 2

1.5.1. *Setup HTC Vive Pro Eye* ................................................................................................. 2

1.5.2. *Setup Pro VR Integration – Based on HTC Vive* ............................................................ 2

1.5.3. *Virtual Reality and your hardware* ................................................................................ 2

1.6. SCENE PERFORMANCE ....................................................................................................... 2

1.7. **UNITY PLUG-IN DESCRIPTION** ..................................................................................... 3

1.7.1. *Interaction system* ......................................................................................................... 3

1.7.2. *Movement system* ........................................................................................................ 3

1.7.3. *Multiple scenes* ............................................................................................................ 3

1.7.4. *Dynamic scene data* ..................................................................................................... 3

1.7.5. *Object spawning* ........................................................................................................... 4

1.7.6. *Level of Detail* ............................................................................................................... 4

1.8. **UNITY DEFAULT LAYOUT** .......................................................................................... 4

2. **INITIAL SETUP** .................................................................................................................... 5

3. **SCENE INTEGRATION** ........................................................................................................ 8

3.1. **ADDING REQUIRED PREFABS** .................................................................................. 8

3.1.1. *Adding the Analytics Manager* .................................................................................... 9

3.1.2. *Adding the Scene Camera Manager* ............................................................................. 9

3.1.3. *Adding participant with Pro VR Analytics character controller* .................................. 9

3.1.4. *Adding participant with other character controller* ................................................... 10

3.2. **ENTER LICENSE KEY** ................................................................................................ 10

3.2.1. *Activate the VR Headset* ............................................................................................. 11

3.3. **PARTICIPANT MOVEMENT** ......................................................................................... 11

3.3.1. *Teleport Properties* ...................................................................................................... 12

3.3.2. *Advanced Teleport Properties* .................................................................................... 13

3.3.3. *Trackpad Properties* .................................................................................................. 13

4. **TRACKING OBJECTS** ........................................................................................................ 14

5. **AREA FILTER** ................................................................................................................... 17

6. **SCENE CAMERAS** ............................................................................................................ 20

7. **BUILDING THE PROJECT** ............................................................................................... 22

8. **UPGRADE VERSION 1.1 TO 1.2** .................................................................................... 24
1. Introduction

Tobii Pro VR Analytics is an analysis software tool that integrates into new or existing Unity environments. The software enables collection and playback of eye tracking data in Unity environments. It provides analysis on a range of human behaviors, including automated visualizations and analytics for interaction, navigation, and eye tracking data. Playback and analytics are available for both individual and multi-participant sessions. The automated statistics ensure rapid availability of results after recordings are completed.

Tobii Pro VR Analytics supports a wide range of Unity3D environments.

1.1. Compatible VR headset

Tobii Pro VR Analytics has been designed to work with genuine eye-movement data, which requires a VR headset with integrated eye tracking from Tobii to access the gaze-based capabilities.

Tobii Pro VR Analytics supports the following VR headset:

- HTC Vive Pro Eye
- Tobii Pro VR Integration – Based on the HTC Vive HMD (not available for sale anymore)

1.2. Compatible Unity versions

Tobii Pro VR Analytics is continuously updated to work with the latest Unity version. To benefit from the latest functionality of Pro VR Analytics it is required that the latest compatible Unity version is used. Pro VR Analytics have been tested with the versions listed below.

Tobii Pro VR Analytics version 1.1 supports:
- Unity version 2018.2

Tobii Pro VR Analytics version 1.2 supports:
- Unity version 2018.2.0f2
- Unity version 2018.3.11f1
- Unity version 2018.4.0f1 LTS
- Unity version 2019.1.2f1

1.3. How Tobii Pro VR Analytics is structured

Tobii Pro VR Analytics is delivered as a downloadable package containing the Unity package and documentation. The Unity package needs to be integrated into a Unity project via the Unity editor. When Tobii Pro VR Analytics has been integrated into a Unity environment, record, replay and analysis is performed within the resulting executable file.

1.4. How to read this manual

The Integration Manual assumes that you have your VR headset set up and has the necessary drivers installed (if the headset is not setup see section 1.5), received the Pro VR Analytics Unity package and have the Unity editor open with your environment.

This manual provides a step-by-step guide to the process of integrating the Unity package into an environment to enable it with the functionality of Tobii Pro VR Analytics. The manual is meant to be
read in a chronical order from start until end, since all the steps are necessary the first time when integrating into a new environment.

1.4.1. Reference Unity scene
For reference there is a simple sample Unity scene with Pro VR Analytics already integrated shipped as part of the Pro VR Analytics Unity Package. This sample scene can be used to check how to add the different features to a scene or as a base for further development.

1.5. Setup your VR Headset
Before doing a recording the VR headset that you are going to use needs to be setup.

1.5.1. Setup HTC Vive Pro Eye
1. If you have not already setup your HTC Vive Pro Eye headset follow the instructions here, https://www.vive.com/eu/setup/.

1.5.2. Setup Pro VR Integration – Based on HTC Vive
1. Go to https://www.vive.com/eu/setup/ and chose the HTC Vive headset. Follow the instructions.

1.5.3. Virtual Reality and your hardware
An immersive VR experience calls for a computer that meets certain system requirements. Check that your computer is VR-ready by using the Steam VR Performance Test at: https://store.steampowered.com/app/323910/SteamVR_Performance_Test/

Making new recordings with Pro VR Analytics calls for a computer that is VR-Ready. However, replay and analysis put less stringent demands on your computer, mainly determined by the complexity of the analyzed environment, such as its size, shading and level of photorealism.

Make sure that your computer utilizes a high-performance power setting and that Tobii Pro VR Analytics runs from the computer's dedicated graphics card and not from its regular processor card. Headset and external monitors must also be connected to the dedicated graphics card.

If your computer has trouble running an environment, check whether you can select a lower graphics setting for it at launch.

1.6. Scene Performance
To carry out an effective study in VR, the environment should be immersive, which implies that it needs to run smoothly in VR. Optimize the scene so that it can run at a high frame rate within the VR headset before integrating with Tobii Pro VR Analytics. The largest performance hit comes with enabling the Unity project for VR. The computer needs to draw one camera for each eye, so it’s basically twice as heavy to run a VR enabled scene compared to a one-camera Unity scene. Typically, fps drops by half when enabling VR. How large the performance impact from Tobii Pro VR Analytics is depends on the complexity of the scene, but it is small compared to enabling VR. We recommend that the environment should be running in at least 90 frames per second after integrating
with Tobii Pro VR Analytics when running in the VR headset. Below, some basic tips for optimizing the scene are outlined:

- Real-time lighting is costly in terms of performance. Use it only when necessary. Many times, baking the light gives a good enough result.
- Decrease the resolution of textures and meshes in the scene, especially those not so relevant for the study.
- Use primitive shape colliders as much as possible and avoid mesh colliders, since they are performance heavy. On the other hand, for some objects in the scene, it might be necessary that the eye tracking rays hit a surface that follow the shape of the object precisely as described in Section 4, point 2.
- Use a smaller number of objects in the scene.

1.7. Unity Plug-in description

Tobii Pro VR Analytics (VRA) is a plug-in solution that is imported into a Unity project to enable it with data recording, sessions replay and calculation of eye tracking and interaction metrics. In order to do this, VRA provides specific systems for parts of the functionality, including interactions and movements. This section also covers some methods not supported.

⚠️ If the provided interaction systems are not used, all the functionality of the software will not be available in the Unity scene

1.7.1. Interaction system

VRA comes with its own interaction system. The interaction system is the system that works together with the Vive hand controllers and makes it possible for the participant to pick up objects in the environment. With the VRA interaction system, the participant can pick up objects and the objects snap back to their original positions as the participant lets go of the objects. Since VRA records the interactions performed with its own interaction system, it can replay those interactions.

VRA does not record interactions performed with a different interaction system and cannot replay nor get interaction metrics data for those interactions. If a different interaction system than the VRA interaction system is used, replaying interactions and interaction metrics will not work.

1.7.2. Movement system

VRA comes with its own movement system. The movement system is the system that works together with the Vive hand controllers that lets the participant move around the scene by using the trackpad to teleport or glide. VRA still supports having customer character control systems such as Steam VR.

1.7.3. Multiple scenes

VRA is designed to record and replay one scene at a time and thus, does not support multiple scene setups. If several scenes are needed in your project, the scenes shall be integrated one by one with VRA. Alternatively, the environment can be designed so that you have one scene with different parts to move between.

1.7.4. Dynamic scene data

VRA does not record any dynamic scene data other than participant actions. In other words, objects (other than the participant) moving, objects changing shape (mesh deformation) and sound playing in the scene etc. will not be recorded together with the participant. Such dynamic scene data will not be
synced with participant recordings. Eye tracking metrics for objects that move or change shape will work.

1.7.5. Object spawning
Object spawning is not supported in VRA. Objects that are created dynamically and placed into the environment will not get eye tracking data. Objects need to be present in the scene at start of the scene.

1.7.6. Level of Detail
Level of Detail (LOD) is not supported in VRA. LOD is used to optimize rendering, so that objects that are far away from the camera are rendered with less detail than objects close to the camera. With LOD, multiple meshes can be used for an object depending on the distance to the camera. If LODs are used, heatmaps will not be rendered correctly. For multiple meshes to be treated as the same object in metrics the meshes belonging to the same object should have the same name.

1.8. Unity Default Layout
This is an overview of the different sections of the Unity editor that will be referred to in this manual.
2. Initial Setup

This section outlines how you import Tobii Pro VR Analytics into your project.

1. Make sure that you do not have any compile errors in your project. Click ‘Clear’ in the Console tab in the lower left corner in the Unity Editor. If the list below the Clear button is empty, you can continue. Otherwise, a developer needs to resolve the compile errors.

2. Import TextMesh Pro Essential Resources. Click “Import”.

---

Tobii Pro VR Analytics – Integration Manual v.1.2
© Tobii Pro AB
3. Next, you will replace the InputManager.asset file. Note that, if you have made some changes to the InputManager (like adding a controller such as an Xbox controller), these changes will be overwritten in this step, so keep a backup of the settings in that case. Replace the InputManager.asset in the ProjectSettings folder of the Unity Project with the InputManager.asset in the TobiiProVRAnalytics plugin package.

4. Open Tags and Layers by clicking on Edit tab > Project Settings > Tags and Layers. This will show the tags and layers on the Inspector Window. Add “Cursor” (case-sensitive) to the list of User Layers.

5. Open PlayerSettings by clicking on the Edit tab > ProjectSettings > Player. This will show the PlayerSettings on the Inspector Window.
6. In the Inspector Window, open the XR Settings tab and click on the check box for “Virtual Reality Supported”. Then click the “+” button and select OpenVR.

Note: We only want OpenVR. If there are any other SDKs listed (e.g. Oculus, etc.), remove them by selecting them and clicking the “-” button.

Note: In some Unity versions you need to download OpenVR using the Unity Package Manager.
7. In the **Inspector Window**, open the **Other Settings** tab and set “**Scripting Runtime Version**” to “**.NET 4.x Equivalent**” and “**Api Compatibility Level**” to “**.NET 4.x**”.

8. Import the TobiiProVRAnalytics.unitypackage file into the project by clicking on **Assets > Import Package > Custom Package** and selecting the TobiiProVRAnalytics.unitypackage.

9. Click “Import”.

### 3. **Scene Integration**

**3.1. Adding required prefabs**

This section describes how to integrate a scene with the required prefabs, and how to add a Participant to the scene.
3.1.1. Adding the Analytics Manager
On the Project Window, type in “TobiiProVRAnalytics” on the search box. Then drag and drop TobiiProVRAnalytics onto the Hierarchy Window.

3.1.2. Adding the Scene Camera Manager
On the Project Window, type in “SceneCamerasManager” on the search box. Then drag and drop the prefab onto the Hierarchy Window.

3.1.3. Adding participant with Pro VR Analytics character controller
In the Project Window, type in “Participant” on the search box. Then drag and drop the Participant prefab file onto the Hierarchy Window.

Note: Place the Participant game object where you want the test subject to start when making a recording.
3.1.4. Adding participant with other character controller
If you want to use a custom character controller instead of the one provided by VR Analytics you will instead have to follow the steps below.

1. Assign the Participant component to the camera that will be rendering to the VR headset.
2. Drag the Controllers prefab into the scene and place it as a child under the same parent as the main VR camera.
3. The Spectator Camera property in the Participant is the camera that will render to the computer monitor. If left empty it will be generated at startup.

3.2. Enter license key
To enter the license key. Go to the Tobii Pro VR Analytics Manager component in the TobiiProVRAnalytics prefab and enter the license key. The license key was sent when purchasing the product.
3.2.1. Activate the VR Headset

To be able to use the record functionality there needs to be a headset activated. VR headsets are activated by:

1. Connect the computer to the Internet
2. Connect the VR headset to the computer
3. Open the .exe file.
4. Press “Activate” in the Home tab

3.3. Participant movement

Chapter 3.3 is only valid if you are using the Pro VR Analytics character controller.

In addition to walking physically, the participant can move by using the Vive controllers to teleport (point-and-click the Vive controller trackpad), or by pressing the trackpad to glide. You can select if you want to teleport or glide by pressing the trackpad by selecting “Movement type” in the Character Controller VR script. The usage can be restricted by setting the inspector property “Move Type” of the Character Controller VR script attached to the “Participant” object in the scene.
If you select Movement type “Teleport”, you will be able to set the Teleport Properties in section 3.1.1 (and section 3.1.2 for advanced properties). For Movement type “Trackpad”, you can set the properties described in section 3.1.3.

### 3.3.1. Teleport Properties

If you have selected to use Movement type “Teleport”, you can set “Teleporting Layers” and “Max Distance”.

![Teleport Properties screenshot](image)
• **Teleporting Layers**
  The layer(s) that the participant can teleport to. Setting this will limit which objects the user can teleport to. For example, if the floor object is in the “Ground” layer and the Teleporting Layers is set to “Ground”, then users can teleport to the floor. If the scene does not contain a ground layer, it is highly recommended to create one. If you want the navigation to work as in the real world, it should only be possible to teleport to the ground and you should select “Ground” as your Teleporting Layer.
  Default value: None

• **Max Distance**
  The maximum distance the participant can teleport to.
  Default value: 50 Unity units

### 3.3.2. Advanced Teleport Properties
The properties described in this section can be changed if a different appearance of the teleport functionality is needed. In most cases, the properties below do not need any modification.

• **Raycast Origin Right Hand**
  A GameObject representing the controller on the right hand, the transform is used as the starting point for drawing the vector from the controller to the teleport point.
  Default value: Handle (right controller)

• **Raycast Origin Left Hand**
  A GameObject representing the controller on the left hand, the transform is used as the starting point for drawing the vector from the controller to the teleport point.
  Default value: Handle (left controller)

• **Line Renderer**
  A LineRenderer object that is displayed to indicate which direction and position the participant is trying to teleport to.
  Default value: LineRenderer

• **Teleport Pointer**
  A GameObject that is rendered at the end of the Line Renderer to display the exact spot the participant will be teleported to.
  Default: TeleportCirclePivot

### 3.3.3. Trackpad Properties
If you have selected to use Movement type “Trackpad”, you can set Acceleration and Max Speed.
4. Tracking Objects

This section describes how to track objects in your scene.

1. Select the layers that the eye tracker should be able to see, by setting the “Tracked Layers” property of the **TobiiProVRAnalyticsManager**. Only the layers selected here will affect the eye tracker’s raycasts. In addition, objects that you want to be interactable should belong to the tracked layers.

- **Acceleration**
  - Determines the rate of change of speed of the participant with the respect of time.
  - Default value: 2 Unity units per second

- **Max Speed**
  - Limits how fast the participant can go.
  - Default value: 1 Unity unit per second
2. Select the object(s) from the **Hierarchy Window** that you want to record metrics data for (e.g. products, banners etc.). Check the **"Is Interactable"** box to allow the user to pick up the objects.

**Note:** Objects require a MeshFilter, MeshRenderer, and Collider **on the same object** to be tracked.
Note: Make sure that the colliders you use for the tracked objects are following the surface of the object you want to record data for. That is, if the shape of the object is not a box and you use a box collider for it, the gaze rays will hit the box and not the surface of the object. The picture with the armchair below shows an example of an erroneous collider design. The gaze rays will hit the green box around the armchair, and they will not hit the surface of the armchair. A mesh collider should be used instead of a box collider.

3. Add the “TobiiProVRAnalytics_TrackedObject” component to them by clicking on the “Add Component” button at the bottom of the Inspector Window, then searching for and selecting the “TobiiProVRAnalytics_TrackedObject” component. Only objects with this component will record fixations and display heatmaps during Replay mode.

Note: Tobii Pro VR Analytics caches the mesh data of and generates GUIDs for TrackedObjects at the time the “TobiiProVRAnalytics_TrackedObject” component is added. Ensure that the mesh setup is done and that the GameObject’s name is correct before adding the component.

! If any changes are made to the object after the “TobiiProVRAnalytics_TrackedObject” component is added, the component should be Reset to avoid unexpected behavior.
Note: Objects with the TobiiProVRAnalytics_TrackedObject component but that are not in layers among those selected in the “Tracked Layers” property of the TobiiProVRAnalyticsManager will not be tracked and will have no recorded data.

5. Area Filter

This section describes how to add Area Filters to the project and how to assign TrackedObjects to them. Area Filters allow more control over which metrics are displayed in the Metrics Screen.
Metrics data, such as Fixations, will be compiled and shown on the Metrics Screen depending on what Area Filters are selected in the Area Filter menu.

**Note:** Area Filters are optional and TrackedObjects without an assigned Area Filter will appear in the Metrics Screen under the “Default” area.

1. From the **Hierarchy Window**, select the game object you want to represent the Area Filter and click “Add Component” button on the **Inspector Window** and select “Area Of Interest”. It is the children of the selected game object that will belong to the area filter. Thus we recommend that you create an empty game object that will be the parent to the game objects belonging to the area and add the “Area of Interest” component to that parent game object.

   AreaOfInterest is simply another name for Area Filter and should not be confused with AOI (Area Of Interest), a term used in eye tracking for an area where eye tracking metrics are collected.
This will designate the “LIVING ROOM” as an Area Filter. All TrackedObjects under it’s hierarchy (e.g. “Sofa”, “Pillow1”, etc.) will be assigned under the “LIVING ROOM” Area Filter by default.

**Note:** TrackedObjects cannot be assigned under more than one Area Filter.

2. (Optional) If behavior other than the default is desired, the Area assignment of TrackedObjects may be explicitly set by dragging and dropping the AreaOfInterest’s GameObject into the “Area” field of the “TobiiProVRAnalytics_TrackedObject” component.
6. Scene Cameras

This section describes how to add Scene Cameras to the project and how to optimize your project to create a good view.

Scene cameras are used for Replay. They help to view multiple Participant’s movements and interactions with different objects.

A free moving camera is automatically added to the project. The Free moving camera is controlled using ASDW and right-click the mouse.

1. You need to have at least one scene camera in your scene. To add scene cameras to your scene, click Create > Camera or right click on the game object under which you want to add the camera and select Camera.
2. For all scene cameras, set the Camera setting "Target Eye" to "None (Main Display)". The cameras in the scene should be normal cameras, not VR cameras. In other words, the Target Eye should not be both displays but the main display. The Depth should also be a higher setting than the VR camera.
3. Click the `SceneCamerasManager` object on the **Hierarchy Window** to have its components show up in the **Inspector Window** and set the Scene Cameras List size to the number of cameras you wish to add.

4. Drag and drop each scene camera object from the **Hierarchy Window** to an empty Camera Element slot on the **Inspector Window**.

7. **Building the Project**

This section describes how to build the project into a `.exe` file.

1. Open the **Build Settings** window by clicking on the File tab > Build Settings...
2. Select “Windows” for the Target Platform and “x86_64” for the Architecture.

3. Make sure the scene is selected and click Build.
8. Upgrade Version 1.1 to 1.2

Upgrading the Tobii Pro VR Analytics plugin to a later version is an easy process. It is very important to follow the steps in order. Note that both version 1.1 and 1.2 support Unity version 2018.2.

Note: It is very important that you do not close or restart the Unity Editor during the upgrade procedure.

Prerequisites:
- Unity 2018.2

Steps:
1. Open the project in Unity 2018.2
2. Make sure that you do not have any compile errors in your project.
3. Under ProjectSettings, replace the InputManager.asset file with the new one
4. Delete the TobiiProVRAnalyticsManager prefab from the scene
5. Delete TobiiProVRAnalytics folder from the assets folder
6. Import the new TobiiProVRAnalytics package
7. For those with custom participant controllers (i.e. Not using the Participant prefab), follow this step, else skip to step 8.
   i. Delete the custom scripts and GameObjects attached to your “participant”
   ii. Follow steps in section 3.1.4.
8. Add the TobiiProVRAnalytics prefab to the scene
9. Enter the QLM license key in the TobiiProVRAnalytics prefab as described in section 3.2 (if you have not received it contact our support team at connect.tobiipro.com)
10. Set the TrackedLayers property in the TobiiProVRAnalytics prefab
11. Add the SceneCamerasManager prefab to the scene
12. Set the scene cameras in the SceneCamerasManager prefab
13. Build the project
14. If you want to move recordings made with v.1.1 to v.1.2, follow section 3.3 “Exporting and importing recordings” in the User Manual.