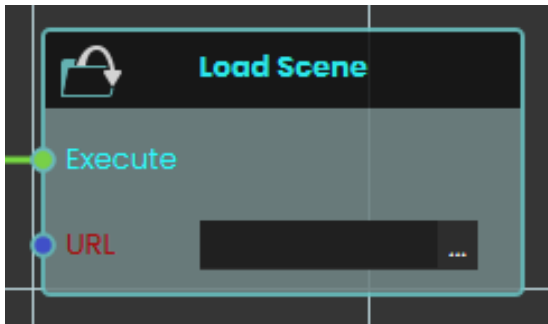


# Scene

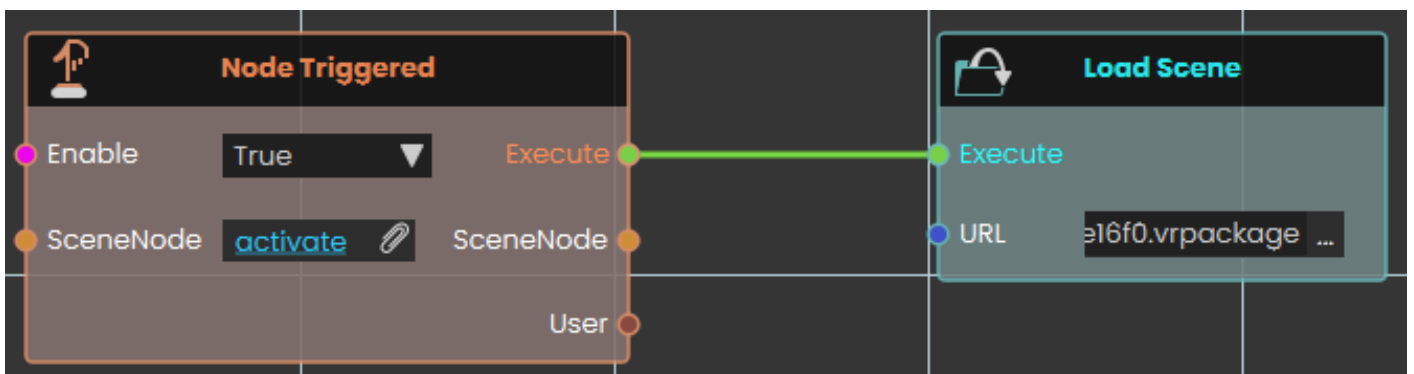
## Load Scene



The **Load Scene node** is used to teleport between

different scenes within your VR Experience. Enhancing the overall immersion and interactivity of the experience.

## Example

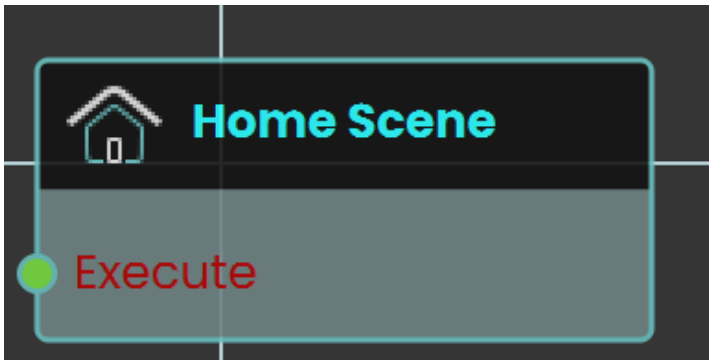


In this example, the **Load Scene Node** is used to transition the user from one scene to another. This node facilitates changing scenes, allowing for a dynamic experience as users navigate through different environments within the VR application.



Tutorial is available on SimLab VR Discord server

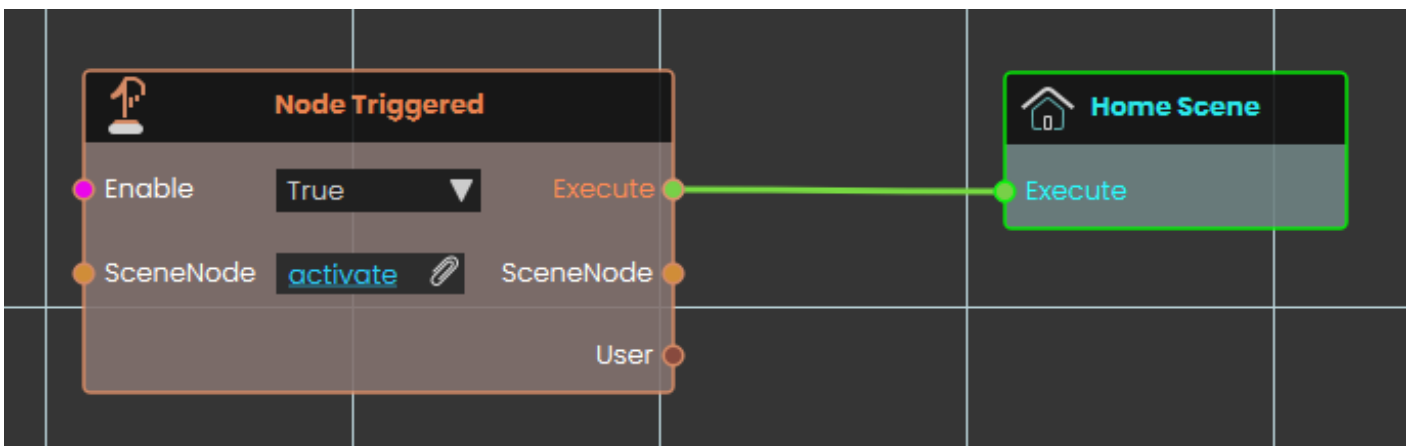
## Home Scene



The **Home Scene node** enables users to

return to the initial scene of the VR Experience. This node provides a straightforward way for users to reset the VR Experience to the starting point, ensuring a user-friendly navigation experience.

## Example



In this example, the **Home Scene Node** is used to return the user to the main scene when the trigger event occurs. This allows you to teleport back to the primary environment, enhancing the user experience by providing a clear and intuitive way to return to the starting point.



Tutorial is available on SimLab VR Discord server

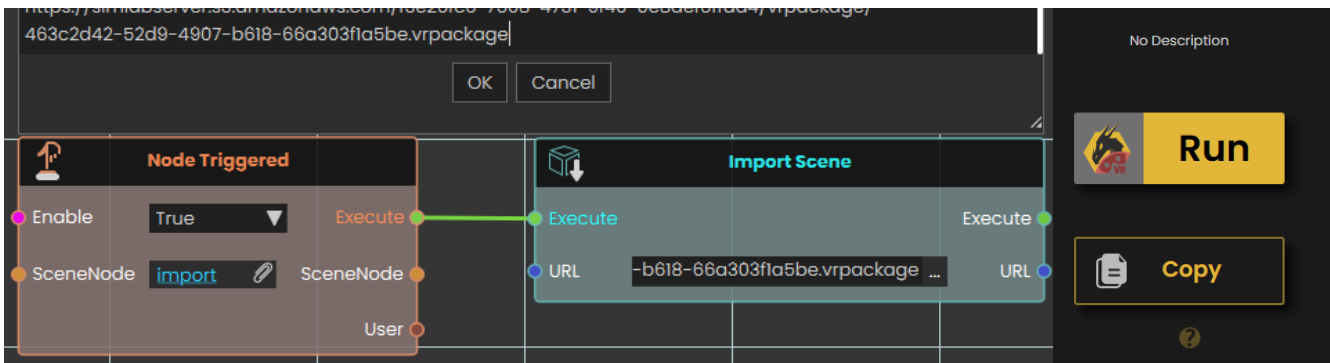
## Import Scene



The **Import Scene** node is used to

load an external scene into the current VR Experience. This allows users to dynamically bring in additional environments, objects, or elements as needed during runtime.

## Example



In this example, the **Import Scene** node is used to load a car model into the scene when the "Import" object is triggered. The Node Triggered event detects user interaction and executes the Import Scene node, which then loads the specified car model from an external VR package file. This allows the car to appear dynamically within the VR environment.

## Enterprise VR Packages

An **Enterprise VR Package** is a complete VR experience saved as a single, **protected** `.evrpackage` file. These packages are **encrypted**, so only a viewer that has the matching enterprise key can unlock and open them, and creating them is part of a SimLab **Enterprise** licence. The node below brings one of these packages into the running Viewer and adds it to the **main menu**, ready to be opened.

## Add Enterprise VR Package To Viewer

Fetches an Enterprise VR Package from a web address, unlocks it, and adds it to the Viewer's main menu.

### What it does

Give this node the address of an Enterprise VR Package (an `.evrpackage` file) and, while your experience is running, it downloads the package, unlocks it with its enterprise key, and adds it to the Viewer's **main menu** — the list of experiences someone can open. The package is stored **locally** on the device, so from then on it can be opened from the menu

like any other entry.

**It does not put anything into the scene that's currently playing.** This node makes a whole package available to open from the menu; it does not drop models or other content into the running scene. If what you want is to bring content *into* the current scene, use the **Import** node instead — that's what it's for.

Because these packages are protected, this only works with genuine Enterprise VR Packages, and the Viewer has to be able to unlock the file. The **Success** output tells you whether the package was added.

## It waits until the package is ready

Downloading a package and unlocking it isn't instant — it takes a moment, and it happens quietly in the background so the rest of your experience keeps running. This node handles that waiting for you. Its **Execute** output does *not* fire the moment the node runs; it fires only once the package has been added to the menu *or* the attempt has failed. So **the nodes you wire after this one run when the package is ready**, not the instant it's requested, and you don't need to add any delay of your own. Because **Execute** fires whether it worked or not, always check **Success** first.

## Inputs

Port	Type	What to connect
<b>Execute</b>	Trigger	Wire this from the previous node's <b>Execute</b> output.
<b>URL</b>	Text	The web address of the Enterprise VR Package (an <code>.evrpackage</code> file) to add. A local file path on the device also works.

## Outputs

Port	Type	What you get
<b>Execute</b>	Trigger	Fires once the node has finished — whether the package was added or not.

Port	Type	What you get
Success	True / false	<b>Yes</b> if the package was downloaded, unlocked, and added to the main menu; <b>No</b> otherwise — for example the address was wrong, the download couldn't be completed, or the package couldn't be unlocked.

## Example

URL input	<code>https://enterprise.example.com/packages/assembly-training.evrpackage</code>
Success output	<b>Yes</b> — the package now appears in the Viewer's main menu

## Tips

- **It adds to the menu, not the scene.** The package becomes available to open from the Viewer's main menu. To bring content *into* the scene that's currently playing, use the **Import** node instead.
- **Enterprise VR Packages only.** This works only with protected `.evrpackage` files. Ordinary files won't be added — use **Import** for those.
- **The Viewer must be able to unlock it.** These packages are encrypted. If the file is damaged, its key isn't available, or there's no connection to download it, **Success** comes back **No** and nothing is added.
- **Check Success first.** Wire the **Success** output into a **Branch** node so the rest of your experience only relies on the package on the “Yes” side.
- **No need to add a delay.** The node already waits for the download and unlock to finish before its **Execute** output fires, so the nodes after it run at the right time on their own.

---

Revision #14

Created 4 November 2024 13:04:05

Updated 1 June 2026 12:49:32 by Rafat